Connectivity and Stochastic Robustness of Synchronized Multi-drone Systems

S. Bereg[1], J.M. Díaz-Báñez[2], Paul Horn[3], Mario A. Lopez[4], and Jorge Urrutia[5]

[1] Department of Computer Science, University of Texas at Dallas, USA. Partially supported by NSF award CCF-1718994.
[2] Department of Applied Mathematics II, University of Seville, SPAIN. Partially supported by grants PID2020-114154RB-I00 and TED2021-129182B-I00 funded by MCIN/AEI/ 10.13039/501100011033 and the European Union NextGenerationEU/PRTR.
[3] Department of Mathematics, University of Denver, USA. Partially supported by Simons Collaboration Grant #525039.
[4] Department of Computer Science, University of Denver, USA. Partially supported by a University of Denver John Evans Award.
[5] Instituto de Matemáticas, Universidad Nacional Autónoma de México, México. Partially supported by PAPIIT Grant IN 102117, UNAM.

Abstract. A set of n drones with limited communication range is deployed to monitor a terrain partitioned into n pairwise disjoint and closed convex trajectories, one per drone. There is exactly one communication link between two trajectories if they are close enough, and drones can communicate provided they visit the link at the same time. If each robot flies around an assigned area and shares information with the neighbors periodically the system is said to be synchronized. Over time, one or more drones may fail and the ability to survey, communicate, and stay connected decreases, thus the robustness against drone failure becomes crucial. In this paper we study various problems related to the proper functioning of a synchronized system under drone failure. First, we provide efficient algorithms, both centralized and decentralized, for determining the connected components induced by the set of surviving drones. Second, we study coverage, isolation, and connectivity under a probabilistic failure model and show that, in the case of grids, the system is quite robust in the sense that it can tolerate a large probability of failure before drones fail to completely cover the terrain, become isolated, or the system loses full connectivity.

Keywords: Unmanned Aerial Vehicles · Synchronized Communication System · Communication Graph · Connectivity · Probabilistic model.

1 Introduction

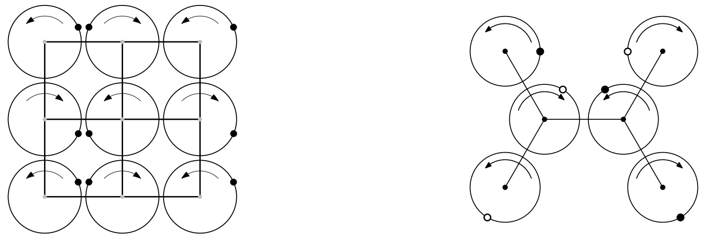
Teams of *Unmanned Aerial Vehicles* (UAVs), colloquially known as drones, are becoming a trend in the last few years for their use in a wide variety of applications such as area monitoring, precision agriculture, search and rescue, exploration and mapping, and delivery of products, to name a few; see [13,21,17] and references therein for a comprehensive survey on the topic. The coordination of a team of autonomous vehicles enables the execution of tasks that no individual autonomous vehicle can accomplish on its own, and thus there has been an increasing interest in studying teams of drones that cooperate with each other. In such multi-drone systems a desired collective outcome arises from the interaction of the drones with each other and with their environment, via a set of installed sensors and communication devices. Considering that multi-drone systems are often deployed in adverse circumstances and the network may get disconnected, robustness against drone failure becomes crucial.

The problems raised in this paper assume the framework recently proposed in [8]. A partition of a terrain to be covered is given and every drone is assigned a different section of the partition. Each drone travels on a fixed closed convex trajectory while performing a prescribed task, such as monitoring its assigned area. In order to allow cooperation, each drone needs to communicate periodically with other drones. Since the UAVs have a limited communication range, two of them need to be in close proximity of each other in order to communicate. In [8] the authors presented a framework to survey a terrain in the scenario described above. As an abstraction, they considered a model in which each drone is modeled as a point that travels on a unit circle at constant speed, and this speed is the same for all the drones. Without loss of generality, they assume that one time unit is the time required by a robot to complete a tour of a circle. These circles may intersect at a single point but do not cross. The communication between two robots can take place if their corresponding circles touch, and it is carried out at the point of intersection. They also show how to generalize the results to a more realistic model. In [8] it is assumed that the unit disk graph defined by the given set of circles (trajectories) is connected, they call it the *communication graph*.

The main problem addressed in [8] is to obtain a synchronization schedule, that is, to assign a starting position and travel direction to each trajectory so that if n drones follow this schedule, every pair of them traveling in two adjacent circles pass through the intersection point of their trajectories at the same time. A set of trajectories with a synchronization schedule conform a *synchronized communication system* (SCS) [8]. In the same paper, the authors also discuss necessary and sufficient conditions for the existence of a synchronization schedule. For an illustration see Figure 1 and related video[6]. Note that although not every pair of robots can communicate directly, a robot may relay a message to another robot through a sequence of intermediate message exchanges.

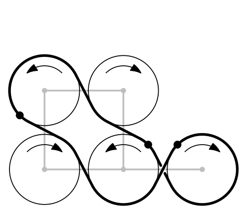
If the system is synchronized, as described above, a robot can easily detect the failure of a neighboring robot. If a robot d_i in trajectory C_i arrives at the communication point between C_i and another trajectory C_j, and it fails to meet another robot, it will assume that the robot in C_j is no longer functional. Under such circumstances, a reasonable strategy is for d_i to switch to C_j at this point and take over the task of the missing robot. In [8], this strategy is called the *shifting strategy*. Under the shifting strategy, an undesirable phenomenon known

[6] https://www.youtube.com/watch?v=T0V6tO80HOI



(a) (b)

Fig. 1: Examples of synchronized communication systems. The robots in the SCS are represented by solid black dots. (a) The communication graph is a grid. (b) The communication graph is a tree. If the white drones leave the system, the black drones become isolated.



(a) (b) (c)

Fig. 2: Examples of SCSs where some robots fail. The path traversed by the available robots is drawn in bold stroke. The robots are represented by solid black points. (a) The robots are not isolated, however there are uncovered trajectory segments. (b) The robots are isolated and everything is covered. (c) The robots are isolated and there are uncovered trajectory segments.

as *isolation*, may occur. A drone is *isolated* if it fails permanently to meet other drones. The three black drones in Figure 1(b) never meet, and thus they are isolated. Note that an isolated robot always shifts trajectories at communication links. See some examples in Figure 2(b) and Figure 2(c). Moreover, after enough failures, some portions of the trajectories may never be visited again; i.e., some parts of the terrain become uncovered. Figure 2(a) and Figure 2(c) illustrate examples in which the circles are not totally covered when two robots fail.

A *ring* is the closed path followed by an isolated drone. Each ring is composed of sections of various trajectories and has a direction of travel determined by the direction of movement in the participating trajectories. Each section of a trajectory between two consecutive link positions participates in exactly one ring, thus the rings in an SCS are pairwise disjoint. The number of rings and its length depends on the communication graph. Figure 3 illustrates some examples. See [2] for a study on rings and the isolation phenomenon.

In [8], neighboring circles are assigned opposite travel directions (clockwise and counterclockwise) so as to enable the shifting strategy. From now on we work

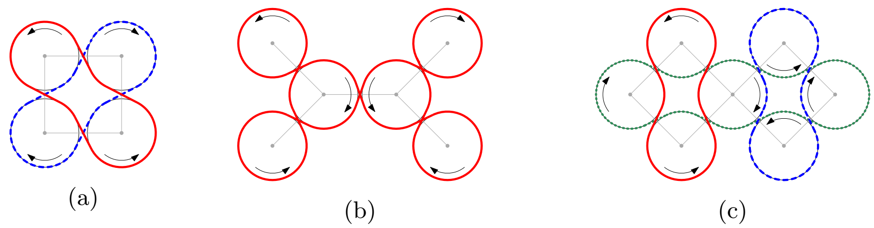


Fig. 3: SCSs with two rings (a); one ring (b) and three rings (c).

with SCSs where every pair of neighboring circles have opposite travel directions and, consequently, the communication graph is bipartite. Under this model, our main contributions are the following:

1. *Connected components.* Consider a system in which some drones may have failed. Two drones belong to the same connected component if they can exchange messages, possibly through a sequence of intermediaries. We provide efficient algorithms for computing the connected components of the system, both centralized (where a central server is privy of a snapshot of the system, see Theorem 2) and decentralized (using only the information that drones can gather while flying and meeting other drones, see Theorem 4). For the case of grids, the required flying time can be proportional to the number of trajectories, and this bound is tight (Theorem 3).
2. *Probabilistic failure model.* We address the robustness of a system in which drones survive with probability p, and study three properties: coverage, full connectivity, and drone isolation. For coverage, we establish a sharp threshold in a $t \times t$ grid (Theorem 5) and derive a general (and exact) expression for the probability that all trajectories are fully covered in a general system (Theorem 6). Also for $t \times t$ grids, we establish sharp thresholds for the existence of isolated drones (Theorem 9) and connectivity (Theorem 10). These results show that the system is extremely robust to random failure as these thresholds are $o(1)$ as $t \to \infty$. For arbitrary grids, we provide less sharp results (Theorems 11 and 12). For general systems, the problem for communication is more complicated and we show some examples that illustrate that robustness, similar to that of grids, is no longer possible.

2 Related work

There is a vast literature related to communication strategies for a team of robots monitoring a given area. Our scenario shares similarities with works on patrolling agents [10,16] where the drones patrol along predefined paths. The goal is to share data by making observations as well as synchronizing with their teammates during brief and sporadic opportunities. Typically, research has focused mostly on the construction and validation of working systems, rather than a more general and formal analysis of problems from a mathematical point of view. In this paper, we study some algorithmic and probabilistic problems related to communication and coverage in the particular framework of a synchronized communication system (SCS), as proposed in [8].

Recently, several algorithmic and combinatorial problems arising from the shifting strategy of a SCS, as outlined in the previous section, have been studied. In [2,1] the authors propose various quality measures for a synchronized system regarding the robustness of a SCS in the presence of failures. With respect to communication between robots, they consider two quality measures, k-isolation resilience and broadcasting resilience. The k-isolation resilience is the minimum number of robots whose removal may cause the isolation of at least k surviving robots. The broadcasting resilience is the minimum number of robots whose removal causes a loss of connectivity in the system. For the study of robustness with respect to area coverage, they introduce the concept of coverage resilience as the minimum number of robots whose removal may result in at least one non-covered trajectory segment. Computing these measures leads to interesting combinatorial and algorithmic problems.

On the other hand, the use of stochastic strategies instead of the deterministic one in the shifting protocol of a SCS has been studied in [4]. In this work, the authors propose the use of a *random strategy*: every time a robot arrives at a communication link between its current trajectory and a neighboring one, it chooses independently, with probability p to pass to the neighboring trajectory by means of a shifting operation. The authors showed that using a random decision in the shifting protocol generates random walks and proved theoretical bounds on metrics related to the system performance. Also, in the same paper, it has been experimentally proved that the random strategies outperform the deterministic strategy. Indeed, the study of stochastic UAV systems has attracted considerable attention in the field of mobile robots. This approach has several advantages such as shorter times to complete tasks, cost reduction, higher scalability, and more reliability, among others [3,22].

In this paper we consider a stochastic model to study the robustness of a SCS against drone failure in a synchronized multi-drone system. Connectivity and reliability are important concerns of computational networks. For example, in [11] the influence of the communication range has been examined to obtain necessary and sufficient conditions for connectivity in wireless networks. The authors derive the critical power a node in the network needs to transmit in order to ensure that the network is connected with probability one as the number of nodes in the network goes to infinity.

In the same context, the authors of [19] derived bounds for the probability that a node is isolated. Maintaining a high reliability level of a drone fleet is of great significance considering the possibility of drone failures. For instance, models for performance evaluation of multi-state degraded systems are provided in [20,15]. Theses models consider that the system degrades with use and that these degradations may affect the system efficiency. As time progresses, it can either go to the first degraded state upon degradation, or it can go to a failed state upon a random and sudden failure. When the system in its last acceptable state, a preventive maintenance is performed to restore the system to one of the previous higher performance states. Performance measures are used to evaluate the systems subjected to minimal repairs.

In addition to studying the connectivity of a SCS, the goal of this paper is to provide a probabilistic study of coverage, isolation, and connectivity of the system under drone failures. A probabilistic analysis can be used for randomly generating experimental data for the problem, with the property that the instances are asymptotically robust with probability one. Moreover, the analysis can suggest heuristics that are provably asymptotically optimal. Many problems have been probabilistically analyzed in the literature, and as a consequence of those studies, experimental data have been generated [12] and new heuristics have been proposed (see, for instance, [14] for the multi-knapsack problem and [18] for the generalized assignment problem).

3 Computing the connected components

In this section we design efficient algorithms for determining the connected components induced by a set of surviving drones in a SCS. Accordingly, assume we have a SCS where a subset of drones have left the system and the surviving ones apply the shifting protocol. It is further assumed that no more drones leave the system. While some pairs of drones may communicate directly, communication between other pairs may rely on passing information through other drones; in some cases communication between drones may be impossible. We define the *drone communication graph* G_D as the graph whose vertices are the drones, two of which are adjacent if the corresponding drones communicate directly at some point in time. The connected components of this graph identify which sets of drones can, directly or indirectly, communicate with each other. We denote by $C(d)$ the connected component of a drone d in G_D. It is easy to see that communication through other robots can sometimes be faster than direct communication, e.g. it may take a long time for two drones to meet each other and communicate directly. In this section we show how to compute the connected components in the drone communication graph under two models of computation:

1. **Centralized:** Suppose a central server contains the full information of a SCS, including the set of drone trajectories and the current locations of the drones. How can the connected components of the SCS be found efficiently?
2. **Decentralized:** Suppose the drones themselves can pass messages when they meet each other. How can they determine the other drones in their connected component, and how quickly can this be accomplished?

Note that in the second case, the drones do not know how many other drones are active or where they are; they merely learn what drones are active as they meet other drones and exchange information. For that reason, the complexity of both problems is different. The complexity of our algorithm in the first case, is the number of steps the central server needs to compute, while in the second case it is the *flying time* of the drones before each drone knows its connected component.

Nonetheless, we show that for the $s \times t$ grid both problems can be solved with highly efficient algorithms. The key notion for our results is the use of the *token graph* introduced in [1]. We assume that at time 0 each drone d_i holds a token t_i. This establishes a bijection between the drones and the tokens. When two drones meet, they exchange their tokens. The token graph G_T of a drone system is the graph whose vertices are the tokens, two of which are adjacent if at some time the corresponding tokens are exchanged. Note that each token t_i stays in the same ring of drone d_i. Thus, the token graph can also be defined using drones as vertices where two drones are adjacent if they encounter each other in a system where only two drones exist. We have the following result.

Theorem 1. *Two drones of G_D are in the same component if and only if the corresponding tokens are in the same component in G_T.*

Proof. Note that if two tokens are neighbors in G_T then the corresponding drones are connected by a path in G_D (corresponding to the drones which held the two tokens before they were exchanged). Likewise, if two drones are neighbors in G_D, it corresponds to the exchange of a pair of tokens – and those tokens are connected to the initial tokens held by the drones by a series of swaps, so that the tokens corresponding to the connected drones in G_D are connected by a path in G_T. □

In the case of an $s \times t$ grid with only two drones, the drones encounter each other if and only if they are in the same row or the same column [1]. In this case, the token graph can be viewed as the graph where the vertices are the drones and two drones are adjacent if, at any point in time, they are in the same row or column. If this happens, except when transitioning from one trajectory to another, the said drones will always be in the same row (both moving down or up) or column (both moving left or right) [1]. We call it an *RC-graph*.

Theorem 2. *The connected components in G_D can be computed in polynomial time in the centralized model. Furthermore, they can be computed in linear time in the $s \times t$ grid.*

Proof. The token graph can be computed in polynomial time and, for the $s \times t$ grid it can be computed in linear time [1]. The connected components in the token graph can be computed in linear time using breath-first search. By Theorem 1, this yields the connected components in G_D in the centralized model. For a drone system on a grid, the token graph is the RC-graph. The RC-graph and its connected components can be computed in linear time. □

3.1 Decentralized Computation

The goal in the decentralized model is for each drone d_i to compute $C(d_i)$, its connected component in G_D. We use the following algorithm. Each drone d_i

maintains a list $L(d_i)$ of some drones from $C(d_i)$. Initially, we set $L(d_i) := \{d_i\}$. When drones d_i and d_j meet at some time, they replace both $L(d_i)$ and $L(d_j)$ with the union $L(d_i) \cup L(d_j)$. It is clear that if we follow this protocol long enough, all drones will know their connected components (that is $L(d_i) = C(d_i)$ for all i.) Our goal is to give bounds for the running time of this approach.

We emphasize, again, that the time measured here is actually flying time of the drones, i.e. how long do the drones have to fly until they each, individually, know the other drones own components. We ignore, then, the computation of the set unions involved as this is negligible compared with the actual flying time from one communication point to another. We further assume that a unit time is needed to navigate a trajectory. We begin with sharp bounds for the problem on a $t \times t$ grid.

Theorem 3. *On the $t \times t$ grid, at time $t \cdot (t-1)$ we have that $L(d) = C(d)$ for all drones d. Furthermore, there are drone configurations that require $\Omega(t^2)$ time until $L(d) = C(d)$ for all drones d.*

Proof. We use the idea of tokens. At the beginning (time 0), each drone d_i holds token t_i; recall that when two drones encounter each other they exchange tokens (along with taking the union of their respective lists). Let $d(i,m)$ denote the drone holding token t_i at time m. Thus, $d(i,0) = d_i$. Note that $d(i,m)$ is always in the same component as drone d_i as it holds t_i due to a sequence of interactions with other drones, each passing t_i to the next drone of the sequence. Moreover, $L(d(i,m)) \subseteq L(d(i,m'))$ if $m \leq m'$ as whenever tokens are exchanged, the lists are passed along.

Fix an (arbitrary) drone d_0 and consider any drone d_k in $C(d_0)$. Let $d_0, d_1, \ldots, d_k$ be the shortest path between d_0 and d_k in the token graph. By the construction of the token graph as an RC-graph, it is easy to see that the diameter of the token graph is at most $t-1$, and in particular $k \leq t-1$. Note that t_i and t_{i+1} are in the same row or column, and hence drones holding them meet within time t. This implies that, for instance, $d_1 \in L(d(0,t))$ at time t when the drones with token t_0 and t_1 meet, the label d_1 is passed to the drone holding token t_0. Inductively, it follows $d_i \in L(d(0, i \cdot t))$ at time $i \cdot t$: the label d_i is given to the drone carrying token t_{i-1} in the first time t, then to the drone carrying token t_{i-2} in the next time t, until it at last is passed to the drone hoping token t_0. This shows that $L(d(0, t(t-1))$ is complete at time $k \cdot t \leq (t-1)t$ and as d_0 is arbitrary, this completes the proof.

We now provide a set of drones $\{d_1, \ldots, d_k\}$ which show this time can be quadratic. For this set of drones, $d_1 \notin L(d(k,m))$ until time $m = \Omega(t^2)$. The construction involves a set of drones $\{d_1, \ldots, d_k\}$ on the $t \times t$ grid satisfying the following conditions:

1. d_i and d_{i+1}, for $i = 1, 2, \ldots, k$ share the same row or column, and there are no rows or columns with more than two drones.
2. The distances $d_{i,i+1}$ between d_i and d_{i+1} are decreasing, for $i = 1, 2, \ldots, k$.
3. The polygonal chain formed by the union of the segments connecting d_i to d_{i+1} is a *spiral* polygonal chain; see Figure 4.

4. Drones on the same column move in opposite directions (clockwise and counterclockwise) along their ring, while drones in the same row move in the same direction.

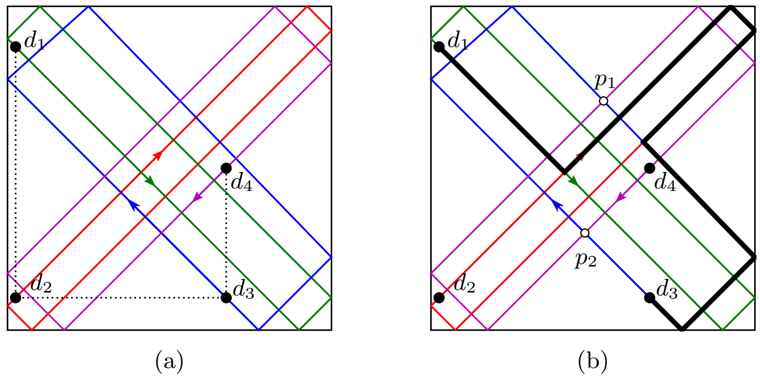


Fig. 4: (a) Drones are arranged in a spiral polygonal chain. (b) The bold line represents the propagation of the label d_1 through the system for times $\leq t$. Drones holding tokens t_3 and t_4 meet at p_1 and p_2.

The key observation is that the drone holding token t_i will only meet the drones holding token t_{i+1} and t_{i-1} and by placing the drones carefully, the intersections will be set up so that the label of d_1 will only propagate through a small number of consecutive drones in time t.

Consider four consecutive drones in $\{d_1, \ldots, d_k\}$; without loss of generality assume these are d_1, d_2, d_3 and d_4. We claim that at time t, the only elements in $\{d_1, \ldots, d_k\}$ with $d_1 \in L(d(i,t))$ are $i = 1, 2, 3$. To see this, observe that since $d_{1,2} > d_{2,3}$, t_2 meets t_3 the first time before it meets t_1. The label d_1 is thus added to the list of the drone holding token t_3 during the second time the drones holding tokens of t_2 and t_3 meet. At this point t_3 has already swapped with t_4 twice. Hence $d_1 \notin L(d(4,t))$ at time t.

Figures 3a and 3b illustrate first this setup and then the process itself. Figure 4 illustrates how the threshold of knowledge of drone d_1 moves forward through the process Note it never moves from the drone holding token t_1 directly to, say, that holding token t_3 as even though the rings of these drones intersect, drones holding these tokens never directly communicate due to the timing. It follows that to reach the drone holding token t_{2i} label d_1 will take $t(i+1)$ time. □

For a general system, a similar argument can be used to prove the following.

Theorem 4. *Consider a general system of N drones on n trajectories and ring lengths $r_1, r_2, \ldots, r_k$. Then at time $N \cdot \max\{\mathrm{lcm}(r_l, r_m) : l \neq m\}$, $L(d_i) = C(d_i)$ for all drones $d_i, i = 1, \cdots, N$.*

Proof. The proof is similar to that of Theorem 3 and we highlight the slight differences. The key observation here is that if two tokens will be exchanged, they will be exchange within any interval of time length $T = \max\{\text{lcm}(r_i, r_j) : i \neq j\}$. Thus, instead of considering the lists $L(d(i, k \cdot t))$ we consider the lists $L(d(i, k \cdot T))$. Also, the upper bound of $t-1$ on the diameter of the token graph is no longer valid; instead we replace it with an upper bound of N. The remainder of the proof, however, is identical: the argument of Theorem 3 implies that for any arbitrary drone d_0, at time $i \cdot T$ the list $L(d(0, i \cdot T))$ contains all drones at distance at most i from d_0 in the token graph. □

4 A probabilistic failure model

In this section we consider a simple stochastic model to measure the robustness of our drone systems. In [1,2], three notions of resilience of a system were introduced:

- **Coverage resilience:** The minimum number of drones that need to be destroyed so that some part of the system is no longer observed. The coverage resilience is always the length of the shortest ring.
- **Broadcasting resilience:** The minimum number of drones that need to be destroyed so that the system is no longer connected.
- **Isolation resilience**: The minimum number of drones that need to fail so that some drone is isolated[7].

Implicitly, these notions measure the robustness of the system against a malicious attacker – how many drones must an attacker destroy to ruin some desirable property of the system.

In this section, we study the robustness of solutions to *random failure*; this can be thought of, perhaps, as the robustness of the drone system to mechanical failure. Regarding coverage, we establish a sharp threshold for coverage in a $t \times t$ grid and record a general (and exact) expression for the probability that all trajectories are covered in a general system. This is fairly simple in that it only depends on the multi-set of ring lengths, and not on how the system is laid out. In comparison to the coverage resilience (which is t for the $t \times t$) grid, it turns out that nearly $t^2 - t\ln(t)$ random drones can be destroyed while the system will still be covered (with high probability).

For communication, the situation is much more complicated for general systems as the robustness depends on how the rings interact with each other. Here, we provide a number of results for a variety of systems. Our most precise result is for the $t \times t$ grid, where we establish a precise threshold for both connectivity and the existence of an isolated drone – again, there is a dramatic difference from the deterministic case; the broadcasting and isolation resiliences for the $t \times t$ grid both are $2(t-1)$, yet nearly $t^2 - \frac{t \ln t}{2}$ drones can be destroyed at random while maintaining connectivity. For general $s \times t$ grids, we establish weaker

[7] In [2], the more general problem of isolating k drones is studied under the name k-resilience.

results in some regimes. Finally, we illustrate a number of phenomena that occur in general systems to highlight the difficulties of understanding the problem in general.

4.1 Coverage under random failure

Theorem 5. *Consider a full drone system in the $t \times t$ grid, where drones survive with probability p. Let $\mathcal{V}$ denote the event that the entire system is covered. Then if $p = \frac{\ln t + c}{t}$ for some constant c*

$$\lim_{t\to\infty} \mathbb{P}(\mathcal{V}) = e^{-e^{-c}}.$$

Proof. Recall that the system is covered as long as there is at least one drone in every ring; in the full $t \times t$ grid there are t rings each with t drones. Hence, the probability some ring is uncovered is $(1-p)^t$. Since the rings partition the system, the number of empty rings hence has a binomial $Bin(t, (1-p)^t)$ distribution. Note that if $p = \frac{\ln t + c}{t}$, then

$$\lim_{t\to\infty} t(1-p)^t = \lim_{t\to\infty} t\left(1 - \frac{\ln t + c}{t}\right)^t = e^{-c}.$$

This ensures that the limiting distribution of the number of empty rings is asymptotically Poisson, and the probability that there are no empty rings is asymptotically $e^{-e^{-c}}$. □

Note that this implies that if $\omega(t) \to \infty$ arbitrarily slowly then $p = \frac{\ln t + \omega(t)}{t}$ implies $\mathbb{P}(\mathcal{V}) \to 1$ while $p = \frac{\ln(t) - \omega(t)}{t}$ implies that $\mathbb{P}(\mathcal{V}) \to 0$.

Essentially the same argument proves that ring systems whose minimum length is t are still covered (with probability tending to 1) after random drone failure so long as the probability of survival is $p = \frac{\ln t + \omega(t)}{t}$. It is rather more difficult here to give a meaningful bound in the other direction however, as this depends on the ring length distribution. None the less, the general probability is failure is given as follows:

Theorem 6. *Suppose that a general system has ring lengths $r_1, \ldots, r_t$ and drones fail with probability p. Then*

$$\mathbb{P}(\mathcal{V}) = \prod_{i=1}^{t}(1 - (1-p)^{r_i}).$$

4.2 Communication under random failures

In this section we study the connectivity of G_D under random failure. In the case of the $t \times t$ grid, we prove sharp thresholds for the properties of containing an isolated vertex and for connectivity – cf. Theorems 9 and 10 below. We remark that our results are quite similar to those for the well known Erdős-Rényi random graph [9], but our setting differs in two crucial ways. First the 'host graph' (which can be thought of as the RC-graph for the full system) is not complete; nor is the resulting random G_D a subgraph of the full G_D as which drones directly communicate within a subsystem differs from that of the full. Second, while most work generalizing results of the Erdős-Rényi graph to more general host graphs (see, eg. [6,7]) takes a random set of edges, we actually take a random set of vertices. A side effect is that the properties we study are not monotone; additional drones surviving may break these properties.

Connectivity in general systems.

In contrast to Theorem 6 above – which provides a relatively simple expression for the probability that a system is covered after random failure, based only on the ring lengths in the system – the problem of when all drones can pairwise communicate is rather more complicated. Several interesting phenomena happen which make general statements hard, as the geometry of the system is much more important. To motivate the restriction to grids, we highlight some of those phenomena which make stating general theorems difficult.

- **Bottlenecks:** Bottlenecks can occur that make the system less robust, as captured in the following proposition.

  Proposition 7 *Let S be a full drone system with broadcasting resilience t, where the failure of t drones splits the system into sets of n_1 and n_2 drones that do not communicate. Suppose that the drones survive with probability p. Let $\mathcal{C}$ denote the event that the system of drones is connected (that is, all drones can communicate with one another). Then*

  $$\mathbb{P}(\mathcal{C}) \leq 1 - (1-p)^t(1-(1-p)^{n_1})(1-(1-p)^{n_2}).$$

  Proof. This follows immediately, as if a drone system has resilience t, then there is a set of t drones whose removal disconnects the system and these drones all fail with probability $(1-p)^t$; meanwhile at least one drone survives in each side with probability $(1-(1-p)^{n_1})(1-(1-p)^{n_2})$. □

  It is easy to construct drone systems that have small (i.e. constant sized) bottlenecks. Indeed, consider a full drone system S shown in Figure 5 where S_1 and S_2 are arbitrary systems of size n_1 and n_2 respectively. This drone system has resilience at most three as if the drones in the red ring fail. Then

  $$\mathbb{P}(\mathcal{C}) \leq 1 - (1-p)^3(1-(1-p)^{n_1+2})(1-(1-p)^{n_2+2}),$$

  and hence for any $p < 1$, $\mathbb{P}(\mathcal{C})$ is bounded away from 1. This contrasts with more robust systems, like the grid, where $\mathbb{P}(\mathcal{C}) \to 1$ for even relatively small values of p (cf. Theorem 10.)
- **Ring lengths do not determine threshold:** As seen below, the $t \times t$ grid has t rings of order t, and the threshold for communcation threshold is $\ln(t)/2t$. A modified system, the staircase system, also has t rings of length t but very different connectivity properties. This figure is created by appending

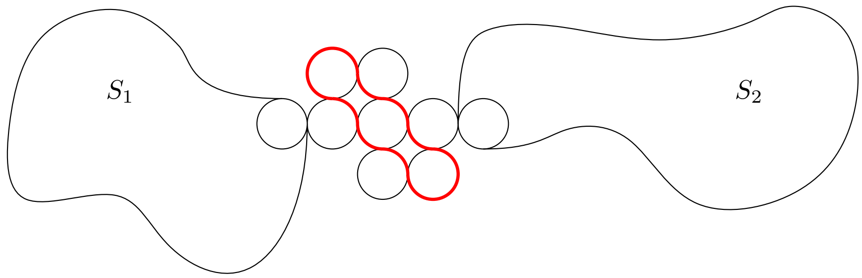


Fig. 5: An example of a non-robust system even if S_1 and S_2 are robust.

t rings in series, that each interface in only two places; see Figure 6. The following simple proposition shows that for the staircase system, already when $p = o(t^{-1/2})$ the system is already disconnected – this is far larger than the threshold in the $t \times t$.

Proposition 8 *Starting with the full t ring staircase, suppose $p = o(t^{-1/2})$ and $p = \omega(t^{-1})$ and drones survive with probability p. Then, if $\mathcal{C}$ is the event that the resulting system is connected*

$$\mathbb{P}(\mathcal{C}) = o(1).$$

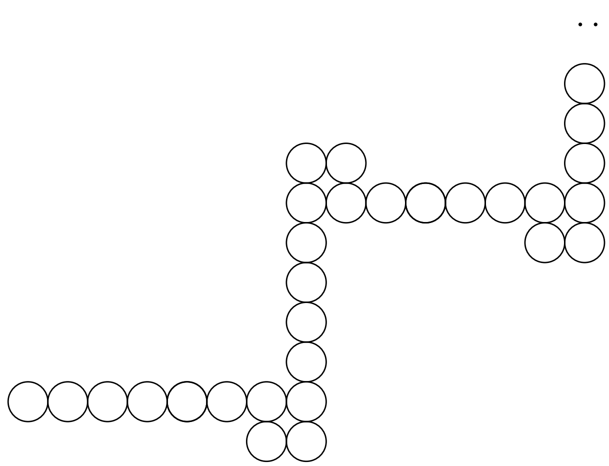


Fig. 6: A staircase example where each ring has length $t = 8$.

Proof. Take the first (lower left in Figure 6) ring. Imagine that each drone in the first ring initially holds an identifying token, that it exchanges each time it encounters another drone as in the previous section. Then, for a fixed token in the first ring, there are k ($1 \leq k \leq 4$) tokens in the second ring it might encounter. Then, the expected number of tokens in the first ring that communicate with a token in the second ring is $t \cdot p(1-(1-p)^k)) = O(p^2 t)$. If $p = o(t^{-1/2})$ this is $o(1)$. By Markov's inequality, then, the probability that there are no drones in the first ring that can communicate with a drone in the second tends to one in this regime as this would result from a token from the first ring being exchanged with a token in the second. Furthermore under the condition $p = \omega(t^{-1})$ there is (with probability tending to one) at least one drone in the first ring and one drone in the tth ring with probability tending to one. But as long as both of these events occur, which happens with probability tending to one, the system is disconnected. □

– **Communication and ring lengths:** If the ring lengths are relatively prime, then as long as there is at least one drone in each ring all drones in the system can communicate. Thus, the visibility probability and connectivity probability in this case is the same. (Not all lengths here need be relatively prime: One can construct the following auxiliary graph – the vertices are the rings of the system, and two rings are connected if their lengths are co-prime and they share a communication point. Then in these systems, the same argument applies.)

Connectivity within $t \times t$ grids

Theorem 9. *Consider a full drone system in the $t \times t$ grid, where drones survive with probability p. Let $\mathcal{I}$ denote the event that some drone is isolated. Fix an arbitrary $\varepsilon > 0$.*

(a) If $p = (1+\varepsilon)\frac{\ln t}{2t}$ then as $t \to \infty$, then $\mathbb{P}(\mathcal{I}) \to 0$.
(b) If $p = (1-\varepsilon)\frac{\ln t}{2t}$ then as $t \to \infty$, then $\mathbb{P}(\mathcal{I}) \to 1$.

Proof. For (a), note that there are t^2 drone locations and in order for a drone to be isolated it must survive while all others in its row and column must fail. Hence the expected number of isolated drones is

$$t^2 p(1-p)^{2t-1} \leq t^2 p e^{-p(2t-1)} = (1+\varepsilon)\frac{t\ln t}{2}\exp\left(-(1+\varepsilon)\frac{(2t-1)}{2t}\ln(t)\right) \to 0,$$

where we note that for t sufficiently large $(1+\varepsilon)\frac{(2t-1)}{2t} > 1$, so that the exponential term is $O(t^{-(1+\varepsilon')})$. (a) then follows by Markov's inequality.

For (b), note that the expected number of isolated drones in this situation is

$$\begin{aligned} t^2 p(1-p)^{2t-1} \geq t^2 p e^{-\frac{p}{1-p}(2t-1)} &= (1-\varepsilon)\frac{t\ln t}{2}\exp\left(-(1-\varepsilon)\frac{(2t-1)}{2t(1-p)}\ln(t)\right) \\ &= \frac{t\ln t}{2}\exp\Big(-(1-\varepsilon)(1-o(1))\ln(t)\Big) \geq t^{\varepsilon/2}, \end{aligned}$$

assuming that t is sufficiently large. Here, we make use of the fact that $\frac{2t-1}{2t(1-p)} \to 1$ as $t \to \infty$; for the claimed $\varepsilon/2$ bound, all that is needed is that this ratio is eventually strictly (because of the $\ln t$ term) bigger than $\frac{1}{2}$.

For (b), then it suffices, by Chebyschev's inequality, to show that if X is the number of isolated drones in the system, to show that $\mathrm{Var}(X) = o(\mathbb{E}[X]^2)$. Note that X can be written as $\sum_{(i,j)\in[t]^2} X_{i,j}$, where $X_{i,j}$ is the event that the drone in the $(i,j)th$ position is isolated. Then

$$\mathrm{Var}(X) \leq \mathbb{E}[X] + \sum_{(i,j)\neq(k,l)\in[t]^2} \Big(\mathbb{E}[X_{i,j}X_{k,l}] - \mathbb{E}[X_{i,j}]\mathbb{E}[X_{k,l}]\Big).$$

We bound the sum. If $i = k$ or $j = l$, then $\mathbb{E}[X_{i,j}X_{k,l}] = 0$, and $\mathbb{E}[X_{i,j}] = \mathbb{E}[X_{k,l}] = p(1-p)^{2t-1}$. As the covariance terms being sum are negative these terms can be discarded for upper bounding the variance. For the other terms, where (i,j) and (k,l) are different in both coordinates, $\mathbb{E}[X_{i,j}X_{k,l}] = p^2(1-p)^{4t-4}$ and there are at most t^4 terms of this type and these summands contribute at most

$$t^4\left(p^2(1-p)^{4t-4} - p^2(1-p)^{4t-2}\right) = \mathbb{E}[X]^2((1-p)^{-2} - 1) = o(\mathbb{E}[X]^2),$$

where the last equality follows from the form of p. Hence, by Chebyschev's inequality $X \sim \mathbb{E}[X]$ with probability tending to one, and thus are isolated drones. ◻

Remark 1. Note that for (a), if suffices that $p \geq (1+\varepsilon)\frac{\log t}{t}$ – this follows as the expected number is decreasing in as p increases (assuming that $p \geq \frac{1}{2t-1}$.) Extending the lower bound works so long as the expected number of isolated drones tends to infinity.

Remark 2. Theorem 5(a) implies that, even for fairly small p, the number of isolated drones is 0 with high probability. At the threshold, the number of surviving drones is only $O(t\log t)$, while $t^2 - O(t\log t)$ drones fail in this case. This should be compared with the 1-isolation resilience of the grid, the miniminum number of drones whose failure can result in an isolated drone, which is $O(t)$ [1].

Theorem 10. *Consider a full drone system in the $t\times t$ grid, where drones survive with probability p. Let $\mathcal{C}$ denote the event that the system of drones is connected (that is, all drones can communicate with one another). Fix an arbitrary $\varepsilon > 0$.*

(a) If $p = (1+\varepsilon)\frac{\ln t}{2t}$ then as $t\to\infty$, then $\mathbb{P}(\mathcal{C}) \to 1$.
(b) If $p = (1-\varepsilon)\frac{\ln t}{2t}$ then as $t\to\infty$, then $\mathbb{P}(\mathcal{C}) \to 0$.

Proof. Note that (b) follows directly from Theorem 9, as if there is an isolated drone (and more than one drone, as there is at such a p with high probability) then the system is not connected.

We proceed to prove (a). We have already shown that when $p = (1+\varepsilon)\frac{\ln t}{t}$ that there are no components of size 1. We still need to show there is a unique component. To do this, we study a modified breadth first search in the RC-graph, introduced in the previous section. Recall, that performing a breadth-first search in the RC-graph (where vertices are drones and they are joined if they in the same row or column) reveals the connected component of a vertex.

To show that there is precisely one component in this setting, we study a slightly modified tree finding algorithm. An *exposing tree* inside of a component is a rooted tree generated as follows: Choose an initial root vertex (drone) to explore. Add all vertices in its row and column to a queue. Now, each vertex in the queue is iteratively explored. When a vertex is explored, vertices in their row or column are added to the queue *if either their row or column is different from those already added to the queue.* Since every vertex being explored was added to the queue it shares either a row or column with one of the other vertices previously explored, and each vertex is responsible for 'exposing' a new row or new column (with the initial vertex responsible for exposing both.) The set of explored vertices forms the exposing tree.

Generating an exposure tree ends with a subset of a connected component which is both non-empty and possibly proper − but vertices in the component and not in the tree share both a row and a column with vertices in the tree. It also ends with a drone from each of some j columns and k rows (where j and k are determined by the process) and $j + k - 1$ vertices. Furthermore, the process ending means that there are no vertices in either of those j columns outside of the k rows and likewise none in the k rows outside of the j columns.

Claim: The probability that the exposing tree process ends with $2 \leq j + k \leq t + 1$ vertices from some starting point tends to zero.

Note that if there are two components, their rows and columns must be disjoint, and hence one of the non-trivial components must have $j + k \leq t$. Thus, the claim will complete the proof of the theorem.

Fix $\ell = j + k$. The number of potential exposing trees with $\ell - 1$ vertices in the $t \times t$ grid can be estimated (roughly) as follows. The degrees in the tree can be represented by a sequence of non-negative integers $(a_1, a_2, \ldots, a_\ell)$ with $\sum a_i = \ell - 2$ where a_1, a_2 are the row and column degrees of the first vertex, and a_i is the number of vertices added when the $i - 1$st vertex from the queue is explored. The number of such solutions is bounded by $\binom{2\ell-2}{\ell} \leq 4^\ell$. There are fewer than $t^2 \cdot t^{(\ell-2)} = t^\ell$ ways of choosing the vertices that are exposed. Note that this is a rather large over-count: it assumes there are t choices each time, when in reality there is a falling factorial type term and also introduces an ordering when exposing the children of a given vertex. None the less, this upper bounds the number of potential processes for a given ℓ is at most $4^\ell t^\ell$.

Now, for a given one of these potential processes, the $j + k - 1 = \ell - 1$ vertices explored must all survive, and the other vertices of their j columns outside of the k rows, and k rows outside of the j columns, must all fail. This has probability $p^{j+k-1}(1 - p)^{(t-j)k+(t-k)j} = p^{j+k-1}(1 - p)^{t(j+k)-2jk}$. Finally note that $jk \leq \frac{(j+k)^2}{4}$ so that regardless of the individual j, k – for *any* potential process with $\ell = j + k$ fixed the probability of ending is at most

$$p^{j+k-1}(1 - p)^{(t-(j+k)/2)\cdot(j+k)} = p^{\ell-1}(1 - p)^{(t-\ell/2)\cdot\ell}$$

A union bound over potential exposing trees, shows that the probability that the process ends with a given value of ℓ is at most

$$4^\ell t^\ell p^{\ell-1}(1-p)^{(t-\ell/2)\cdot\ell} = 4^\ell \cdot t((1/2+\varepsilon)\ln(t))^{\ell-1}(1-p)^{t-\ell/2\cdot\ell}$$
$$\leq \exp\left(\ln(t) + \ell\left(\ln(4(1/2+\varepsilon)) + \ln\ln(t) - (1/2+\varepsilon)\frac{\ln t}{t}(t-\ell/2)\right)\right).$$

In the last inequlality here, we used the inequality $1-x \leq e^{-x}$ along with the definition of p. Hence, per a union bound over potential ℓ it suffices to show that

$$\sum_{\ell=2}^{t+1} \exp\left(\ln(t) + \ell\left(\ln(4(1/2+\varepsilon)) + \ln\ln(t) - (1/2+\varepsilon)\frac{\ln t}{t}(t-\ell/2)\right)\right) \to 0$$

as $t \to \infty$. To do this, we note that for $2 \leq \ell \leq 10$, these terms are $o(1)$ individually as for $\ell \leq 10$ $\ell \cdot (1/2+\varepsilon) \cdot \frac{\ln(t)}{t}(t-\ell/2) > (1+\varepsilon/2)\ln(t)$ assuming t is large enough. For $t+1 \geq \ell \geq 8$, the dominant part of the terms comes from

$$\ell \cdot (1/2+\varepsilon)\frac{\ln t}{t}\left(\frac{t-1}{2}\right) > (2+\varepsilon/2)\ln(t).$$

Thus these terms are actually $o(t^{-1})$ and as there are fewer than t such terms in total the sum is $o(1)$ as desired. □

Connectivity in general grids

Theorems 9 and 10 above consider the specialized case where the initial setting is a full $t \times t$ grid. The case of general systems, even the case of general $s \times t$ grids is significantly more complicated. Indeed, in $s \times t$ grids, the asymptotic behavior of how s and t are taken to go to infinity in comparison with one another can give rise to a number of different behaviors, depending on the values of s and t.

For instance, when $s > 1$ is fixed, while t goes to infinity the isolation threshold and connectivity threshold differ from each other, and both differ greatly from the above. In this case, we have the following:

Theorem 11. *Consider a full drone system $s \times t$, where drones survive with probability p, where $s > 1$ is fixed as $t \to \infty$.*

1. *If $p = \omega(1/t)$, then $\mathbb{P}(\mathcal{I}) \to 0$.*
2. *If $p = o(1/\sqrt{t})$, then $\mathbb{P}(\mathcal{C}) \to 0$ while if $p = \omega(1/\sqrt{t})$, then $\mathbb{P}(\mathcal{C}) \to 1$.*

Proof. For (1), the probability a row contains at most one drone is $tp(1-p)^{t-1} + (1-p)^t$ and if $p = \omega(1/t)$ this tends to zero and the result follows from a union bound. For (2), if $p = o(1/\sqrt{t})$ the expected number of columns containing two drones is $\binom{s}{2} \cdot t \cdot p^2 \to 0$, which implies the resulting communication graph is disconnected as there will be no communication between rows. Once $p = \omega(1/\sqrt{t})$, each of the $\binom{s}{2}$ pairs of rows will have some column where there is a drone in that column in both rows with high probability, and this forces connectivity. □

When both s and t both tend to infinity, the situation becomes more complicated, and we do not pursue a full investigation here. We do note, however, that the following holds:

Theorem 12. *Consider a full drone system in the $s \times t$ grid, where drones survive with probability p. If $s \leq t$ and $s \to \infty$ then if $p = (1+\epsilon)\frac{\ln(s)}{s}$, then $\mathbb{P}(\mathcal{C}) \to 1$.*

Proof. Let S be a subset of drones taken initially from the $s \times s$ subgrid consisting of the s rows and the first s columns. By Theorem 10 these drones are in the same connected component with probability tending to one and each row contains at least one drone from S. Then, having additional drones in the remaining columns cannot destroy the connectivity, as they are in the same row as some drone in the already connected S. □

One final interesting aspect we highlight about general $s \times t$ grids is the relationship between the thresholds for coverage and for communication. As we showed, for the $t \times t$ grid, there is a factor of two difference between the threshold for coverage (Theorem 5) and the threshold for connectivity (Theorem 10), and the coverage threshold (about $\frac{\ln t}{t}$) is larger than the connectivty threshold (about $\frac{\ln t}{2t}$). This difference indicates that, for the $t \times t$ grid, communication is a more robust property than coverage. This is not, however, a universal property of grids. For instance, in the $t \times (t+1)$ grid, the threshold for connectivity is (asymptotically) the same as in the $t \times t$ grid, about $\frac{\ln t}{2t}$ and the proof of Theorem 10 works verbatim in this setting. But, since there is a single ring in this setting, as t and $t+1$ are relatively prime, the system is covered with probability tending to one if $p = \omega(1/t^2)$.

5 Conclusions

Communication and coverage are fundamental and important properties of a multi-drone system. This paper addresses some problems related to the connectivity and robustness of a synchronized multi-drone system, where each drone is in charge of supervising a certain area and shares information with its neighbors every time they reach a communication link. This particular system has been recently studied from a discrete and combinatorial optimization perspective [5]. First, we propose efficient algorithms both centralized and decentralized to compute the connected components of a synchronized multi-drone system when some drones are missing. Second, we consider a stochastic failure model to study coverage, isolation and connectivity under drone failure. The results show that the grid configuration is quite robust against drone failure. We also studied the problem for general configurations showing that the robustness depends on the ring lengths for the case of coverage and on the ring interactions for the communication problem.

For $t \times t$ grids, Theorem 10 provides a sharp threshold $\frac{\ln t}{2t}$ for the connectivity. We also observed that there is system, the staircase system, that has a significantly worse threshold for connectivity and the same multi-set of ring lengths.

An interesting open question is to determine whether the square grid, which has exactly t rings of length t has the best (smallest) threshold for connectivity among all systems which have the same distribution of ring lengths.

Acknowledgment. This work was initiated at the IX Spanish Workshop on Geometric Optimization, El Rocío, Huelva, Spain, June 18-22, 2018. We thank the other participants of that workshop - L.E. Caraballo, M.A. Heredia and I. Ventura - for helpful discussions and their contribution to a creative atmosphere.

References

1. Bereg, S., Brunner, A., Caraballo, L.E., Díaz-Báñez, J.M., Lopez, M.A.: On the robustness of a synchronized multi-robot system. Journal of Combinatorial Optimization **39**, 988–1016 (2020)
2. Bereg, S., Caraballo, L.E., Díaz-Báñez, J.M., Lopez, M.A.: Computing the k-resilience of a synchronized multi-robot system. Journal of Combinatorial Optimization **36**(2), 365–391 (2018)
3. Camp, T., Boleng, J., Davies, V.: A survey of mobility models for ad hoc network research. Wireless communications and mobile computing **2**(5), 483–502 (2002)
4. Caraballo, L.E., Díaz-Báñez, J.M., Fabila-Monroy, R., Hidalgo-Toscano, C.: Patrolling a terrain with cooperative uav's using random walks. In: 2019 Int'l Conference on Unmanned Aircraft Systems (ICUAS). pp. 828–837. IEEE (2019)
5. Caraballo, L.: Algorithmic and combinatorial problems on multi-UAV systems. Doctoral Thesis. University of Seville. Available at https://diazbanez.files.wordpress.com/2020/07/main_rev.pdf (2020)
6. Chung, F., Horn, P.: The spectral gap of a random subgraph of a graph. Internet Math. **4**(2-3), 225–244 (2007)
7. Chung, F., Horn, P., Lu, L.: Percolation in general graphs. Internet Math. **6**(3), 331–347 (2010) (2009)
8. Díaz-Báñez, J.M., Caraballo, L.E., López, M.A., Bereg, S., Maza, J.I., Ollero, A.: A general framework for synchronizing a team of robots under communication constraints. IEEE Trans. Robotics **33**(3), 748–755 (2017)
9. Erdős, P., Rényi, A.: On random graphs. I. Publ. Math. Debrecen **6**, 290–297 (1959)
10. Farinelli, A., Iocchi, L., Nardi, D.: Distributed on-line dynamic task assignment for multi-robot patrolling. Autonomous Robots **41**(6), 1321–1345 (2017)
11. Gupta, P., Kumar, P.R.: Critical power for asymptotic connectivity in wireless networks. In: Stochastic analysis, control, optimization and applications, pp. 547–566. Springer (1999)
12. Hall, N.G., Posner, M.E.: Generating experimental data for scheduling problems. Max M. Fisher College of Business, Ohio State University (1996)
13. Hayat, S., Yanmaz, E., Muzaffar, R.: Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint. IEEE Communications Surveys & Tutorials **18**(4), 2624–2661 (2016)
14. Kan, A.R., Stougie, L., Vercellis, C.: A class of generalized greedy algorithms for the multi-knapsack problem. Discrete Applied Mathematics **42**(2-3), 279–290 (1993)
15. Kharchenko, V., Sachenko, A., Kochan, V., Fesenko, H.: Reliability and survivability models of integrated drone-based systems for post emergency monitoring of npps. In: 2016 International Conference on Information and Digital Technologies (IDT). pp. 127–132. IEEE (2016)

16. Marchukov, Y., Montano, L.: Multi-agent coordination for on-demand data gathering with periodic information upload. In: International Conference on Practical Applications of Agents and Multi-Agent Systems. pp. 153–167. Springer (2019)
17. Otto, A., Agatz, N., Campbell, J., Golden, B., Pesch, E.: Optimization approaches for civil applications of unmanned aerial vehicles (uavs) or aerial drones: A survey. Networks **72**(4), 411–458 (2018)
18. Romeijn, H.E., Morales, D.R.: A class of greedy algorithms for the generalized assignment problem. Discrete Applied Mathematics **103**(1-3), 209–235 (2000)
19. Santi, P., Blough, D.M., Vainstein, F.: A probabilistic analysis for the range assignment problem in ad hoc networks. In: Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing. pp. 212–220 (2001)
20. Soro, I.W., Nourelfath, M., Ait-Kadi, D.: Performance evaluation of multi-state degraded systems with minimal repairs and imperfect preventive maintenance. Reliability Engineering & System Safety **95**(2), 65–69 (2010)
21. Yanmaz, E., Yahyanejad, S., Rinner, B., Hellwagner, H., Bettstetter, C.: Drone networks: Communications, coordination, and sensing. Ad Hoc Networks **68**, 1–15 (2018)
22. Yanmaz, E., Costanzo, C., Bettstetter, C., Elmenreich, W.: A discrete stochastic process for coverage analysis of autonomous uav networks. In: 2010 IEEE Globecom Workshops. pp. 1777–1782. IEEE (2010)