Experiments on multi-robot routing with communication constraints

Alejandro R. Mosteo and Luis Montano

Abstract—As field robot teams mature and their missions become more ambitious, new real world challenges become more prominent. Problems that were previously not an issue in small experimental setups, or that were typically left out in simulation, are attracting increasing interest in order to expand the applicability of such robotic teams. Communication constraints in large or cluttered scenarios, where a full infrastructure network cannot be taken for granted, is such a forefront issue in current multi-robot field robotic missions like exploration and rescue. This paper highlights recent results achieved by our research group from a point of view of task allocation for multi-robot routing. Our methodologies have in common a strong bias towards practical application, aiming at easing the transition between simulation and experimental setups, while making guarantees on the completeness or competitiveness of the achieved solutions. The experiments presented in this work validate this practical approach, and also serve to highlight the kind of problems that may arise in this experimental area.

I. INTRODUCTION

Autonomous robots working in a shared environment often exploit coordination capabilities in order to achieve enhanced efficiency [1]. Explicit communication is a commonly leveraged tool used at various levels of cooperation, be it as a way of improving plans [2], be it in order to perform task allocation (e.g. the popular auction paradigm [3]). This communication is usually carried out by means of Wi-Fi transmitters, since this is a well established technology that cheaply provides good performance without, in principle, impairing robots’ mobility. However, Wi-Fi devices have in practice a limited range of operation and exhibit complex propagation characteristics in cluttered environments.

Thus, leaving aside problem domains where communications can indeed be presumed (urban setups like offices, museums or malls), many realistic applications may be forced to cope with limited or inexistent network coverage (areas too large to be economically efficient or missions in a priori unspecified scenarios). This, in turn, prompts the use of MANET techniques that manage multi-hop messaging and network route preservation.

These techniques, however, cannot guarantee connectivity amongst all the networked robots, unless specific measures are taken to organize the deployment and placement of the mobile units. While this is not a strict requisite in some cases, some particular problem domains may require continuous network integrity in order to, for example, enable human intervention (teleoperation), uninterrupted monitoring or surveillance (visual or auditive channels), tight real-time coordination (multi-robot tasks [4]) or timely reporting to a control base.

The results herein presented are framed in this latter context, in which robots have to perform a series of single-robot tasks at arbitrary environment locations (a problem also known as multi-robot routing), while always maintaining a connected network that includes a static base. In this paper, we analyze the experiments performed using two variations of our CONNECTREE approach [5], [6], in which a spanning tree of guaranteed connections among nodes is used as the basis for task allocation algorithms with interesting completeness and competitiveness properties.

Firstly, we position our work in relation to related research (Section II). The formalization of the underlying task allocation problem is given in Section III. We present next a reactive solution to the problem which is well-suited for use in obstacle-free environments (Section IV), while addressing cluttered environments in Section V. Then, experiments for both methods are detailed in Section VI, paying special attention to the implications for prospective real use of the technology. Summarizing observations and open aspects are finally given (Section VII).

II. RELATED WORK

Limited communications introduce new complexity to the already NP-hard [4] task allocation problem, although they bring the multi-robot routing problem closer to reality. For some problems, it is enough to opportunistically take advantage of network connectivity when available [7], without explicitly addressing integrity maintenance. Arbitrary mission execution not being the foremost issue, solutions can be found [8] that are more akin to sensor-network deployment. Other early solutions consider connectivity a soft constraint (i.e. temporary interruptions can be expected in e.g. [9]).

A possible reactive approach is to include link quality in the motion [10] or goal generation functions [11]. For example, in exploration, goals may be decided as the result of cost functions that depend on signal quality [12]. This is difficult to carry over to more flexible service missions, since tasks are conditioned by external requirements (in general visiting arbitrary locations), and may be totally unrelated to those generated by the team in order to preserve communication. These conflicting requirements are prone to local minima situations, leading to (possibly deliberate [13]) connectivity interruptions. As an alternative, a recent auction-based proposal appears in [14], where robots can subcontract...
the placement of relays in order to increase its network range. An explicit mechanism is given for circular deadlock prevention.

Our first solution, excerpted in Section IV and detailed in [6], is a reactive solution in the spirit of these approaches (robots move as the result of a distributedly computed force field), with a singular characteristic: explicit cluster management enables best-effort execution of arbitrarily placed tasks, while guaranteeing network integrity.

As a parallel issue, there is growing evidence [15] suggesting that simple signal models, like the common “guaranteed circular range” can badly misrepresent the behavior of real signals [16], leading to algorithm failures. Simpler approaches are thus justified, since even building a single multi-hop chain to reach a single target is challenging, involving quality and resource compromises [17].

Our second solution, summarized in Section V and detailed in [5], is grounded on the same basic ideas present in [18], i.e. deploying a series of static relays as robots go away from their starting base location. While that work is more focused on hardware issues, ours is on allocation-related performance issues.

Summarizing, both of our solutions respect two advantageous premises: reactive allocation based on observed signal conditions, considered the critical constraint; and allocation of externally given tasks, suitable for a general purpose team not tied to a particular problem.

III. MULTI-ROBOT ROUTING

A multi-robot routing problem is formally specified by a set of robots, $R = \{r_1, r_2, \ldots, r_n\}$, a set of targets, $T = \{t_1, t_2, \ldots, t_m\}$, the locations of all robots and targets on the two-dimensional plane, and a non-negative cost function $c(i, j)$, $i, j \in R \cup T$, which denotes some abstract cost of moving between locations $i$ and $j$ in either direction (e.g., distance, energy, time, etc.). For simplicity, robots are assumed to be identical, therefore the same cost function applies to all of them.

The objective of multi-robot routing is to find an allocation of targets to robots and paths for all robots, so that all targets are visited and a team objective function is minimized. In general, a team objective is expressed as

$$\min_A f(g(r_1, a_1), \ldots, g(r_n, a_n)),$$

where function $g$ measures the performance of each robot, function $f$ measures the performance of the team, and $A = \{a_1, a_2, \ldots, a_n\}$ is a partition of the targets, such that targets in $a_i$ are allocated to robot $r_i$. Three intuitive team objectives are [19]:

- **MINSUM**: Minimize the sum of the robot path costs over all robots.
- **MINMAX**: Minimize the maximum robot path cost over all robots.
- **MINAVE**: Minimize the average target path cost over all targets.

The robot path cost of a robot $r$ is the sum of the costs along its entire path, from its initial location to the last target on its path. The target path cost of a target $t$ is the total cost of the path traversed by robot $r$ (the unique robot assigned to visit $t$) from its initial location up to target $t$, including any previous targets visited along its path.

Let $RPC(r_i, a_i)$ denote the robot path cost for robot $r_i$ to visit all targets in $a_i$ from its current location. Similarly, let $TPC(t_k, a_i)$ be the target path cost of a target $t_k$ in the assignment $a_i$ of robot $r_i$. Finally, let $CTPC(r_i, a_i)$ denote the cumulative target path cost of all targets in $a_i$, again, if robot $r_i$ visits all targets in $a_i$ from its current location:

$$CTPC(r_i, a_i) = \sum_k TPC(r_i, t_k, a_i)$$

Then, the three team objectives can be expressed as

- **MINSUM**: $\min_A \sum_j RPC(r_j, a_j)$,
- **MINMAX**: $\min_j \max_A RPC(r_j, a_j)$,
- **MINAVE**: $\frac{1}{m} \sum_j CTPC(r_j, a_j)$.

Solving the multi-robot routing problem optimally under any of the above objectives is NP-hard [20]. In our experiments we have used the MINMAX objective for optimization and evaluation.

IV. OBSTACLE-FREE ENVIRONMENT

The first solution we explored in [6], [21] starts from the base assumption that maintaining network integrity is a primary constraint never to be violated. Network integrity is defined as the existence of, at least, a spanning tree of links of sufficient quality reaching over all nodes.

This approach relies on a modular solution replicated in each robot:

- A communication module (COM) which provides multi-hop, real-time communication among robots. Per its design, it is able to provide each robot with complete knowledge of the network topology in the form of a $n \times n$ matrix of link qualities.
- A cooperative navigation module (CNM) which generates velocity commands for the robots, derived from qualities and robot goals. By using the topology information, this module generates control motions that ensure that no robot becomes disconnected, based on the assumption that quality is a function of closeness.
- A task allocation module (TAM) which assigns tasks to robots in such a way that ongoing mission progress is achieved while honoring network constraints.

The information flows between modules are depicted in Fig. 1. The solution is totally decentralized, thanks to the COM which ensures timely distribution in time $O(n)$ of all necessary information. Thus, while the COM distributedly implements a real-time protocol [22], the CNM and TAM
are replicas which compute the same global solution, feeding only the local robot with its relevant speed output.

A. Cooperative navigation module

The CNM is responsible for robot motion and, principally, preventing connectivity losses. It keeps the network connected using a coordinate motion strategy for all the robots. The solution relies on a virtual mesh of spring-dampers among robots that are enabled dynamically according to the quality of the communication network links. There are three modelled force sources acting upon a robot: a pulling force towards its assigned task, pulling forces resulting from any spring-dampers necessary to maintain network integrity, and forces pushing away the robot from nearby obstacles. The resulting aggregated force is translated into velocity commands in an overdamped, smooth behavior. Full details for the navigation subsystem are described in [6]. Note that, by definition of the constants involved, the network integrity forces dominate the system if necessary, and take precedence over goal reaching. This has implications for task allocation which are discussed in Section IV-B.

The reason we assume in this method that the robot team is moving in open workspace is twofold: on the one hand, we presume that link quality is a function of node closeness, which cannot otherwise be guaranteed in cluttered or confined spaces (e.g. tunnels), where serious fading and interferences can occur. On the other hand, the resulting force field of the CNM could reach local minima configurations in presence of large or complex obstacles, which would require higher level strategies for motion planning.

Robots forming the MANET are consequently allowed to move only in ways that do not break network connectivity. The simulation of the physical spring-damper mesh model (Fig. 2) is continually computed by the CNM with a period in the order of tenths of second. Any link belonging to the network spanning tree that falls below a minimum required quality level causes the activation of its corresponding spring-damper, that in turn exerts attracting force between its endpoint robots. Note that these spring-dampers are (de)activated at rest length, so no sudden forces appear.

Finally, these forces are translated into velocities that match the robot real capabilities (that is, including nonholonomicity). This model enforces (a) smooth, jerk-free robot motion, (b) MANET connectivity maintenance, as there is always a spanning tree covering the entire network, and (c) maximal freedom of movement, as the spanning tree contains the minimum number of required links (and hence spring-dampers) to maintain a connected graph.

B. Task allocation module

The TAM module assigns tasks to the robots so that they visit all targets as efficiently as possible. Since in spite of any given robot targets (e.g. too far apart) the CNM module never allows movements that would cause a network split, traditional allocation techniques cannot be readily applied. Consequently, the TAM algorithm takes into account the CNM mesh state to ensure compatible task allocations that will eventually lead to mission completion, recomputing the current allocation whenever the CNM mesh changes (or a target is visited).

We are thus concerned about deadlock states, in which no target is ever reached due to mesh forces and a faulty task allocation. Deadlocks must be avoided because they can render the entire team useless (total deadlock involving all robots) or highly degrade its performance (partial deadlock involving some robots).

We distinguish between two classes of deadlocks. On the one hand, equilibrium (or static) deadlocks happen when a standstill is reached between mesh forces and goal attractors (Fig. 2, R1 and R2). In this case, no robots are moving and no goal can be reached. On the other hand, dynamic deadlocks (or livelocks) appear when robots are moving, yet indefinitely fail to complete any task (Fig. 2, R3).

In this work we opted for a proactive prevention approach which is, by design, deadlock-free, instead of relying on detection and mitigation of deadlocks. The intuitive idea (which details and proof are given, again, in [6]) is that any robots that share an active spring-damper also share, in turn, the same task. This precludes competing forces between the spring-damper and the robot task and consequently static deadlocks too.

Livelocks are dealt with by having robots share the same task if they have shared, at some point since the last task completion by the team, an active spring-damper. In other words, the number of robots that share a target can only monotonically increase between task completions, and thus cyclic task allocations cannot happen.

Note that this technique often allocates more than one robot to a single target, despite that only one robot is required to visit it. This necessary redundancy is intended to form chains that allow reaching targets that are distant from the base (Fig. 2, R0−8).
Within these conditions, the actual allocation of targets to groups of robots can be performed in many ways. Several algorithms were described and simulated in [21]. For the physical experiment detailed in Section VI-A, a solution relying on the Hungarian algorithm was used.

V. CLUTTERED ENVIRONMENTS

In cluttered or confined working areas, network links change its quality more erratically, due to e.g. interferences, multi-path reflections or fading. Our second solution, detailed in [5], [23], is tailored to these concerns, providing a routing solution suitable for dense environments while preventing network splits with minimal assumptions on the signal propagation. In practice, we only require that the decay function is reasonably continuous, meaning that, between two consecutive measurements of signal quality made by a moving robot, the signal will not skip a controlled zone that serves as a buffer between a safe and an insufficient quality.

This generic strategy for multi-robot routing under limited communication range in dense environments is based on the following ideas. Robots move together as a whole towards a target, monitoring the network link to the base; once this link quality is exhausted, one robot is stopped at that point to act as a relay, and the rest continue advancing while monitoring now their link to the stopped relay robot. Using this pattern repeatedly, the robots can reach far away from the base without ever breaking the chain to the base. Note that the robots do not need to know their communication range; the decision to stop a robot as a relay is made dynamically as dictated by the locally measured signal quality.

Once the target is visited, the robots are allowed to retreat from the target only along the path they used to reach it, in order to avoid potential deadlocks due to the density of obstacles and the connectivity constraint. This retreat mirrors the deployment, and a relay can only start retreating when all moving robots reach the relay location. In practice, the set of paths used by the robots is a tree rooted at the base station (Fig. 3). Note that two paths to different targets may share a portion in the upper levels of the tree. Therefore, after visiting a target, robots do not need to retreat all the way to the base in order to move on to the next target. Instead, it is sufficient to retreat up to a common ancestor point between the two paths.

There are two key factors affecting performance in this pattern. One is deciding the next target to visit, but by using this tree structure, any depth-first traversal will have minimum backtracking along the branches, and thus gives minimum \( \text{MINSUM} \) and \( \text{MINMAX} \) cost for the given tree.

The most influential factor, however, is the building of the tree. If some branch depth exceeds the maximum range robots can reach collectively, the remaining target(s) along that branch will be unreachable. One failsafe solution is constructing a star tree where each target is attached directly to the root (base) using the least-cost path between them. Despite its simplicity and the potential inefficiency of the resulting tree, this scheme guarantees that any reachable target will be eventually visited. Our research in [5], however, showed that a good compromise is a minimum-weight spanning tree linking the base and targets (although the optimal solution would be a Steiner tree, which is however NP-hard to find). Furthermore, because of known cost ratios between a minimum spanning tree and the optimal cycle of a graph, we could state that the solution is 2-competitive against the optimal single-robot tour in terms of worst case mission time, and 2\( n \)-competitive in terms of cumulative (e.g. distance or energy) resources.

Despite the fact that these results represent a worst-case, pessimistic analysis that yields loose bounds, it is striking that the additional cost introduced by the limited communication range cannot really grow arbitrarily when using our approach. Our simulation results showed that performance in practice is well below the theoretical bounds. The experiments detailed in Section VI-B, in turn, discuss physical observations and engineering issues.

VI. EXPERIMENTS


A. Obstacle-free environments

This experiment was intended to test the system outlined in Section [V] in a real scenario using the measured link quality signal to compute spring-damper forces. Twelve goals were placed to be visited by three GPS equipped Pioneer 3-AT robots supervised by a fixed base station (a laptop). Task allocation used a method derived from the Hungarian algorithm, respecting the rules previously described.

Fig. 4.a shows the paths followed by the robots, and three superimposed snapshots of the active spring-dampers at different times. \( t_1 \) shows the activation of the first one, \( t_2 \) reflects the first instant in which two spring-dampers exist, and \( t_3 \) shows the moment in which three of them are active for the first time. Fig. 4.b shows \( \gamma \), a function inversely related to the quality of the links that form the spanning tree of the communication network which is used for the spring-damper pulling force (as detailed in [6]).

The three plots of the figure correspond to the three necessary links in this case. As can be seen, the links in the spanning tree change during the experiment depending on the relative quality (note the changes in the line types in all the plots). When the quality of a link in the spanning
Fig. 4. a) Paths followed by the robots and spring-dampers (SD) at the time of their activation. b) $\gamma$ of the links composing the Minimum Spanning Tree of the network during the complete experiment. $\gamma_0$ is the SD activation threshold. c) Distances to base and between robots.

Finally, Fig. 4c shows the distances between all nodes. A relevant observation here is that we can see that distance and quality do not perfectly correlate, which suggests that using simple distance-based models in simulation might not always accurately represent the behavior of real experiments.

Studying the three graphs at once, we can dive into the details of how the mission evolved. Prior to $t_1$, there was no need for restrictions, and there were several link switches that did not affect the team movements. At $t_1$ appeared the first spring-damper, linking base to $R_2$. However, since all the robots were at a comparable distance to base, link quality changes prevented any strong preference for one of them as a relay, and thus we can observe that the base used for short periods of time any of the three robots as a relay for the other two. Eventually, $R_1$ lagged behind and became the stable relay between base and $R_2$, $R_3$. When these robots moved away from $R_1$, eventually a new relay was necessary: at $t_2$, $R_2$ was linked to $R_1$. Finally, at $t_3$ a third spring-damper appeared when $R_3$ took the lead towards the farthest goal.

The parameters chosen for the spring-damper mesh are such that the first restrictions appear at a relative short range (circa 10m), but the robots still retained a good level of mobility as the spring-dampers elongated due to target forces. It was not until shortly before $t_2$, as link quality decreased, that $R_1$ was stopped by force pulling from base. Also, $R_2$ did not stop moving (Fig. 4c) until the very end of the mission, which reveals that its connection to $R_1$ was not yet at maximum length. This highlights that using distance as a substitute would be difficult: either we could err by being too conservative, or we could try to allow more mobility and lose connectivity due to unpredictable signal oscillations. We can thus conclude that solutions that rely on measured signal quality are robust at that geometric reasoning.

B. Cluttered environments

The scenario for the experiments using the strategy described in Section V is an L-shaped corridor (Fig. 5), with a narrow long leg, at the end of which robots are deployed, and a wider and shorter leg, which has the target at its farthest end. This provides insights into the out-of-sight link behavior. The initial long leg is oriented along the X-axis, which is the reason why in some of the following figures only the X position of a robot is plotted. We consider this a first relevant environment for experimentation, since it is large enough, noisy due to alien networks, and presents out-of-sight spots. Ongoing experiments in Canfranc (Huesca) old train tunnel (with more acute fading phenomena) and future ones in a larger laboratory (more challenging from the task allocation point of view) are being prepared to ascertain the generality of the presented results.

In general, and unless noted otherwise, these experiments consisted on sending the robots away towards a goal out
of reach, in order to force the exhaustion of all the links between robots, thus forcing a total standstill when all robots became relays and no remaining robot could proceed forwards. At such point, and after a ten seconds delay, the team automatically detected that the target was out of reach and aborted the mission, beginning its scheduled retreat towards the base as if the goal had been already reached.

C. Single-robot experiments

Our initial tests were done using a single mobile robot and the static base (a laptop) communicating in the 5.2GHz band by means of the RT-WMP protocol. The threshold in this first test was quite conservatively set at 50%. In all experiments, robots move at a maximum speed of 0.5 m/s. Albeit robots follow as closely as possible the deployment path tree, they are equipped with a VFH algorithm for local obstacle avoidance; this makes the experiments robust against passers-by or unmapped small obstacles. Signal was measured at a rate of 10Hz, allowing us to fulfill the expectancy of a reasonably continuous change in signal quality between measurements.

Results from one of such tests are shown in Fig. 6. One particular circumstance of this test is that, by mistake, we used an internal antenna from a PCMCIA card, instead of the aerial antenna that has proper clearance in all directions. Nonetheless, as sometimes happens, this provided some insights into the signal behavior. The most relevant issue, caused by the internal antenna, is that robot orientation had a large influence on signal quality. In practice, when the robot faced towards the base, a sharp drop was evident (around t=80s), presumably because the robot body interfered with the Wi-Fi signal. During the moments of lowest quality (around 20%), a few packets could not be delivered by the network layer. Note that the network protocol implements retransmission as long as the packets can satisfy their real-time requirements [22]. Hence, this means that these failures could have implied a failure to meet real-time deadlines, which could in turn have implications for HRT control applications. Nonetheless, the amount of lost packets is quite small, considering that four probes per second were being sent.

Packet loss is depicted cumulatively in order to improve plot readability. For a loss to be detected, a subsequent packet carrying a sequence counter in it had to be received. In practice, this means that large jumps in packet loss are in reality a steady loss that has been just detected.

Another point that can be mentioned is that signal quality actually improved when the robot was a few meters away from the base, as evidenced by the peaks at t=25s and t=125s approximately. We did not observe this effect later with the aerial antenna (Fig. 7).

Once identified the issue with the antenna, we proceeded to repeat this experiment using the aerial unobstructed antenna. Fig. 7 shows results that are much more satisfactory. No single packet loss occurred, and the signal fell controlably under the safety threshold. One particular observation we made in this test was that, even when the robot had started to retreat, signal did not improve for some time (t=105s–125s). This cannot be attributed to the quality smoothing filters, which operate in much lower time scales (around 1s), and once again illustrates that real signals exhibit complex behaviors that simplistic simulations fail to capture.
Next tests focused on determining good threshold values for the two main classes of frequencies used by the Wi-Fi standard at the time of the experiments, 802.11b/g. In order to do so, the threshold was set at 1%, causing the robot to move forward until total loss of signal. Figs. 8, 9, 5 show the results for these runs. In addition to packet loss, these graphs show time between successful communications (silence in the plots). This value is important because while the b band shows a higher packet loss count, it is a steady loss that does not impair communications nor the distance the robot can go away from the base. Several conclusions can be extracted from these results: the b band provides more range and a steadier signal degradation. While both suffer from a noticeable drop in quality when the line-of-sight is broken (around the point at which the Y value starts to rise), the drop is much sharper in the g case, which would indicate that it is less suited for our strategy. The steady loss of packets in the b case could be caused by interferences with other networks in our laboratory.

D. Two-robot experiment

An experiment using two robots was performed using the g band in order to minimize packet loss, as previously observed. The results are shown in Fig. 10. There are no plots for lost packets or silence in communications since everything performed as expected with the chosen safety threshold of 40% signal quality. As the trajectory plot shows, the tail robot stopped just at the junction of the two perpendicular sections, most likely due to the loss of line-of-sight with the base. The head robot then continued for a third of the length of the remaining section. Quality remained acceptable during all the experiment. Note that the head robot quality in respect to the tail robot remained reasonably stable during all the time that the tail robot was close, that is, as long as it did not need to stop to act as a relay. Another noticeable circumstance is that Tail-Base quality started higher than Head-Tail quality, albeit they all were very close at the beginning of the experiment. We attribute this to different performance of the radio hardware (since it was configured with the same parameters, particularly transmission power) or influence of robot chassis (since one of the robots was an indoors two-wheeled 3-DX while the other was an outdoors 3-AT).

E. Three-robot experiments

The first experiment involving three robots was performed in the same indoor setup. Results are shown in Figs. 11 and 12. While the experiment was completed successfully, there were some puzzling results that we expect to come to understand with further experimentation. For example, the downward spikes in quality in the tail robot do not have a clear origin, since it never lost line-of-sight with the base, nor was particularly close to losing it. Since each robot has a different assortment of mounted hardware (cameras,
GPS receivers, pan-tilt units), this is a possible source of differences and interferences.

Packet loss is not represented in the figures since it was very moderate, with 0, 22 and 35 probes lost for the head, middle and tail robot respectively. These losses do not depend on the distance or quality, since they happened steadily, and as such we do not consider a failure that is preventable, but and issue to be dealt at the network protocol level.

In this particular run, the tail robot stopped before turning the corner, as evidenced by its X coordinate. The middle robot went around the corner, but after losing line-of-sight with the tail robot, the quality quickly fell under the threshold. The head robot was able to advance half-way to the corner, as evidenced by its X coordinate. The middle robot went around the corner, but after losing line-of-sight with the corner, as evidenced by its X coordinate. The middle robot was very moderate, with 0, 22 and 35 probes lost for the middle robot. The GPS receivers, pan-tilt units, this is a possible source of differences and interferences.

Nonetheless, the confined corridor area used, in a noisy environment with alien networks present, was well-suited to focused on signal behavior, since the experimental scenario was not challenging from a task allocation perspective.

The second approach involved a convoy of robots orderly visiting targets, using the trailing robots as relays as it became necessary. The use of purposely planned routes forming a tree rooted at the base permits to quantify the competitiveness of the solution, although the experiments focused on signal behavior, since the experimental scenario was not challenging from a task allocation point of view. Nonetheless, the confined corridor area used, in a noisy environment with alien networks present, was well-suited to validate the proposed strategy, and highlighted the noisy and unsteady quality degradation that can be expected in such conditions. Despite these difficulties, successful completion of the experiments showed that this solution too achieves its objective of continuous network integrity in its intended working conditions. Current work is aimed at bringing together the advantages of both approaches. We expect the combination of the spring-damper mechanism described in Section V with the path trees of Section V will enable the application of this strategy in even noisier environments, where static relays would require high safety thresholds and consequently reduced range. The risk of communication losses after sharp quality drops when turning around corners can too be averted this way. Finally, we intend to build signal quality maps as the convoy advances, in order to better place trailing relay robots in automatically detected spots with better reception.

VII. CONCLUSION

Multi-robot missions requiring uninterrupted connectivity present a tough challenge derived from the many factors affecting the quality of links. We have reviewed our physical experimental results obtained by means of two reactive but distinctly characteristic approaches.

In the first one, a dynamic mesh of spring-dampers enforces the maintenance of a minimum spanning network tree over the robotic team. A task allocation layer working on top of this mesh ensures mission progress. The experiment performed in an open-sky car park successfully demonstrated a complete architecture involving a hard real-time network and distributed navigation control visiting targets as dictated by the task allocation module, without any communication failures. Highlighted by this experiment was the importance of using real quality measurements as the input to the connectivity enforcing algorithm, instead of relying on possibly inaccurate distance models.

The second approach involved a convoy of robots orderly visiting targets, using the trailing robots as relays as it became necessary. The use of purposely planned routes forming a tree rooted at the base permits to quantify the competitiveness of the solution, although the experiments focused on signal behavior, since the experimental scenario was not challenging from a task allocation point of view. Nonetheless, the confined corridor area used, in a noisy environment with alien networks present, was well-suited to validate the proposed strategy, and highlighted the noisy and unsteady quality degradation that can be expected in such conditions. Despite these difficulties, successful completion of the experiments showed that this solution too achieves its objective of continuous network integrity in its intended working conditions.

Current work is aimed at bringing together the advantages of both approaches. We expect the combination of the spring-damper mechanism described in Section V with the path trees of Section V will enable the application of this strategy in even noisier environments, where static relays would require high safety thresholds and consequently reduced range. The risk of communication losses after sharp quality drops when turning around corners can too be averted this way. Finally, we intend to build signal quality maps as the convoy advances, in order to better place trailing relay robots in automatically detected spots with better reception.

REFERENCES