A Loose Synchronisation Protocol for Managing RF Ranging in Mobile Ad-Hoc Networks [⋆]

Luis Oliveira[1], Luis Almeida[1,3], and Frederico Santos[2,3]

[1] DEEC-FEUP/IT, Universidade do Porto, Portugal
[2] DEE-ISEC, Inst. Politecnico de Coimbra, Portugal
[3] IEETA, Universidade de Aveiro, Portugal

Abstract. Robot motion coordination and cooperative sensing are nowadays two important and inter-related components of multi-robot cooperation. Particularly, when concerning motion coordination, distance information plays a very important role in mobile robotics. In this work, we investigate a new solution based on ad-hoc communication without global knowledge, particularly clock synchronisation, to measure distance between mobile units and to share that information. In order to improve ranging, medium throughput, and application predictability, we propose using a synchronisation protocol that keeps transmissions in the team as separated as possible in time, independently of the topology. Results show around 3.3 times reduction in the number of failed ranges without external interference and an order of magnitude reduction in the asymmetries among the nodes concerning the number of failed ranges when using the proposed synchronisation protocol.

Keywords: TDMA, synchronisation, cooperation, information exchange, relative localisation.

1 Introduction

Mobile autonomous robotics are key elements to many current applications which, similarly to the present trend towards multi/many-core computing platforms, exploit the benefits of parallelism. Using multiple such units can increase the effectiveness of surveillance, improve the rate of coverage in search and rescue, enable the transport of large parts, etc. However, achieving cooperation among multiple robots requires information exchange to enable, for example, cooperative sensing [10][8] and inter-robot motion coordination [11][3]. In addition, global services such as managing formations and sensor fusion typically require localisation services.

In this paper we explore both topics, i.e., relative localisation using an RF ranging method, and information sharing by means of a novel broadcast protocol for wireless ad-hoc multi-hop networks. This protocol enforces loose synchronisation among the units transmissions so that they are periodic but as much

---

[⋆] This work was supported by project FCT PTDC/EEA-CRO/100692/2008.

separated in time as possible and we show that this feature has a strong positive impact on the effectiveness of the ranging and thus on the quality of the localisation service. The paper is organised as follows. Section 2 discusses related work. Section 3 presents the information sharing approach. Section 4 presents the proposed synchronisation protocol. Section 5 shows the worst-case topologies in terms of information dissemination latency. Section 6 presents an experimental validation of the proposed framework, and finally, Sect. 7 conlcudes the paper.

2 Related Work and Contribution

Different approaches to relative (anchor-free) localisation based on wireless communications have already been explored [6]. One typical approach is to measure the pairwise distances and then share them among the units. With such distances, each node can use an adequate algorithm, such as Multi-Dimensional Scaling (MDS), to compute positions in a coordinate system. Concerning the measurement of pairwise distances, two main possibilities have been explored. One uses the received strength of the RF signal in message exchanges as distance estimation [4][7]. However, the received signal strength, despite its simplicity of use, is affected by many phenomena that hinder its relationship with distance and thus other approaches have been emerging such as using the Time-of-Flight (ToF) [6]. This approach, which we follow in our work, is independent of signal amplitude being thus much more robust. However, it is also more complex to use and takes substantially longer to measure.

Concerning the sharing of the pairwise distances a broadcast protocol must be used. We propose using the one in [2] and extended in [4] where a so-called extended connectivity matrix, filled in with the measured distances, is disseminated across a multi-hop network. This dissemination can be done without [4][6] or with [2][9][1] synchronisation of transmissions to reduce collisions. In the latter case, the synchronisation can be based on a global clock [2] or just relative [9][1] but [9] does not work in ad-hoc networks and [1] considers the network fully linked.

In this context, our contribution is a relative (loose) synchronisation protocol, based on the Recongurable and Adaptive Time Division Multiple Access (RA-TDMA) approach proposed in [9], to support the coordination and dissemination of pairwise distance measurements obtained with ToF. The protocol is fully distributed, works in ad-hoc networks supporting dynamic topologies including the separation and joining of cliques (subnetworks formed by partitions) and does not rely on clock synchronisation.

3 Information Sharing

Sharing information throughout the network is done with a broadcast protocol that disseminates a set of shared variables, each having one single producer and multiple consumers. This protocol makes use of a set of controls that regulate the updating of those variables in order to enforce consistency between the copies at the producer and consumers. They ensure that newer produced data eventually reaches all copies of each variable at the consumers as well as that stale information is detected and removed following a unit crash, departure from the team, or simply a link rupture. These controls are the following:

1. Local time-stamps, indicating the freshness of the data
   - one time-stamp per shared variable
   - time-stamps are reset when their respective information is updated (t_u), allowing to control the age
   - information is removed if not updated after a pre-set variable-dependent validity interval (t_{val})
2. Sequence numbers indicating between copies of the same shared variable which is the one containing fresher information
   - one sequence number per shared variable
   - each sequence number is increased by the producer unit right before it is sent together with the new information
   - larger number corresponds to newer data

Finally, note that each unit cleans up its own variables, i.e., removes (deletes) stale information, every time it broadcasts them, just before transmission. This means removing all variables for which $t_{now} \geq t_u + t_{val}$ (Fig. 1).

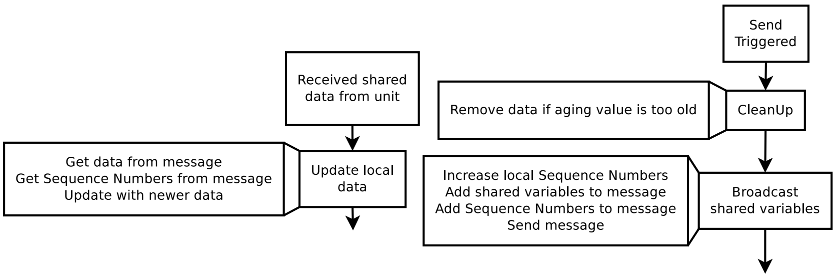


Fig. 1: Broadcast protocol - sending and receiving procedures

3.1 Sharing Distances

To share the pairwise distances obtained with the RF ranging mechanism we create in each unit k an extended connectivity matrix $M^k_{n\times n}$ similarly to [4], whose element (i, j) is the measured distance between nodes i and j and n is the number of units in the team. Each unit i writes in the i^{th} line, only, so that $M^k(i)$ contains the view unit i has of the network, stored in unit k. When two nodes are out of range, a special code Ω is written in the matrix to represent such situation.

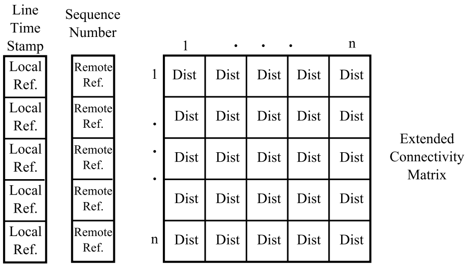


Fig. 2: Connectivity matrix and associated control variables

Each of these lines will be one shared variable, thus having an associated time-stamp and sequence number (Fig. 2).

The distances are obtained using the nanoLoc development kit [5] that measures the ToF (ranging). We configured the ranging to be done in two phases (Fig. 3). The first phase measures $r_1 = V \times (t_1 - t_2)/2$ and the second one measures $r_2 = V \times (t_3 - t_4)/2$, where V is the propagation speed of the RF signal. Finally, r_2 is sent back and the values are averaged, thus the whole ranging procedure returns $Dist = (r_1 + r_2)/2$.

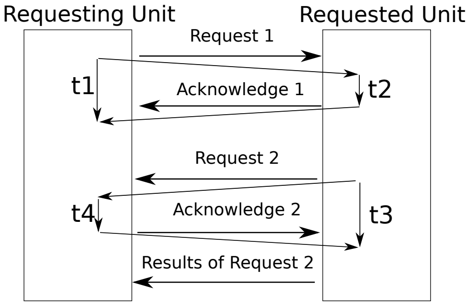


Fig. 3: ToF - Ilustration of the ranging process

One of the problems with this method is the latency resulting from each ranging ($t_{ranging}$), since each unit can only range another one at each time and such latency is variable depending on whether the ranged unit is online or not:

$$t_{ranging} = \begin{cases} 20ms, & \text{if unit is online} \\ [30, 100+]\,ms, & \text{if unit is not online} \end{cases} \tag{1}$$

Therefore, in order to avoid long latencies, we keep track in a vector of the current neighbourhood of each node, i.e., nodes from which messages were received in the previous cycle, and we range one unit from that vector per cycle.

4 Reconfigurable Ad-Hoc Synchronisation Protocol

Our approach follows the RA-TDMA protocol [9] in which the team units transmit in a round. Moreover, the round duration is predetermined, since it sets the reactivity of the communications, but it is divided in a dynamic number of slots according to the current number of units in the team. Similarly, we also use an underlying medium access protocol that provides Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) arbitration, reducing the collisions with alien traffic, i.e., transmissions of nodes external to the team, and even among team units while the slot structure of the TDMA round is being adjusted. The main purpose of RA-TDMA, which we keep in our protocol, is to separate the transmissions of the team units in time, within the round, as much as possible, without using global mechanisms, particularly clock synchronisation. This is done synchronising on the receptions of the messages sent by the other team units as shown further on.

However, unlike the original proposal, our protocol must cope with ad-hoc networks and dynamic topology, which requires new approaches to the propagation of the information in the network and to the agreement on the slots structure and assignment at each instant. These new features are supported on the extended connectivity matrix $M^k_{n\times n}$ presented before, which combines pairwise distances with topological information. The lines of the matrix present the vision each unit has of the network, and the columns present the vision the network has of each unit. In particular, a unit j is considered to be on-line if the j^{th} column contains at least one valid distance, i.e., $\exists i : M^k(i,j) \neq \Omega$. Otherwise, it is considered to have left the network and will be removed from the team.

The current number of team units n is a fundamental parameter for the proposed synchronisation protocol. Firstly, all update periods for all nodes (t_{up}) are configured to the same value, i.e., the desired TDMA round period. Then, each unit autonomously divides this period in a number of slots equal to n with the duration of $t_{slot} = t_{up}/n$. Then, each slot is uniquely assigned to one unit, which is done with a slot allocation table based on the knowledge retrieved from the connectivity matrix.

Naturally, this mechanism requires all connectivity matrices of connected units to be consistent, which is enforced within a bounded interval (see Sect. 5) by the updating rules shown in Sect. 3. This interval sets a limit on the rate of topology changes that our protocol is capable of handling. Faster rates may prevent the team to reach consistent connectivity matrices.

Given a team of units, we define that all units reached an agreement when all have the same slot allocation table. In order to simplify consistency when updating the table we use the same strategy as in RA-TDMA, based on a unique identifier per unit. Whenever the table is changed we sort the list of on-line units by growing identifiers and assign them to the n slots in order, starting from slot 0. Figure 4 shows a situation in which units 3 and 4 are connected through unit 0. The matrix transmitted from unit 0 to unit 4 carries the knowledge of a new team unit, 3 in this case, which allows unit 4 to build a consistent table.

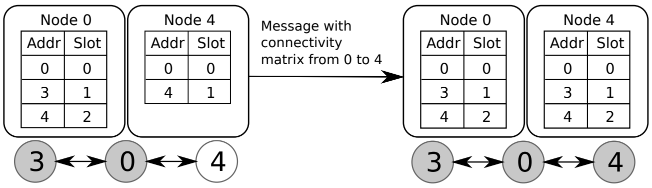


Fig. 4: Updating the slot allocation table

Beyond the consistency of the slot allocation table, the units must also agree on the start of their slots. This particular aspect is also handled similarly to RA-TDMA but in a localised fashion in which each unit synchronises in each round with the units in range, only, using the messages received from them. This synchronisation propagates to the whole network through any connection path. In the beginning of each slot, each node sets the start of the next slot as one round later ($t_{tx}^{nxt} = t_{tx}^{now} + t_{up}$). Then, upon reception of a message in slot m at t_{rx}^{m} and duration t_{len}^{m}, algorithm 1 is executed to possibly adjust the start of the next slot. This causes a phase shift of the whole TDMA round.

Algorithm 1 Re-synchronisation upon reception of message in m^{th} slot

```
$t_{tx}^{nxt'} = t_{tx}^{now} + t_{rx}^{m} - t_{len}^{m} + (n - m) \times t_{slot}$
$t_{tx}^{nxt} = max(t_{tx}^{nxt'}, t_{tx}^{nxt})$
```

Figure 5 shows the synchronisation mechanism where the initial slots are marked with dashed lines. A delay in unit 1 is noticed by unit 3 that delays its next slot setting a new timeframe, marked with full lines. Units 2 and 4 are still unaware of this delay and keep their initial slots. Once unit 3 transmits in the adjusted slot, unit 2 is made aware of this adjustment and will synchronise. Finally, unit 4 will also synchronise after receiving a message from unit 2.

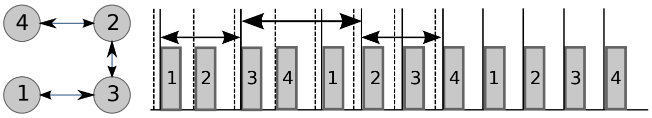


Fig. 5: Propagating slots synchronisation

Figure 6 shows the complete sending-receiving procedures of our ad-hoc broadcast and synchronisation protocol. In each round each node will receive at most once from each of its neighbours, aggregate all received matrices with its own, update its neighbourhood vector, possibly resynchronise the round timeframe, range one of its neighbours, update its matrix and transmit it.

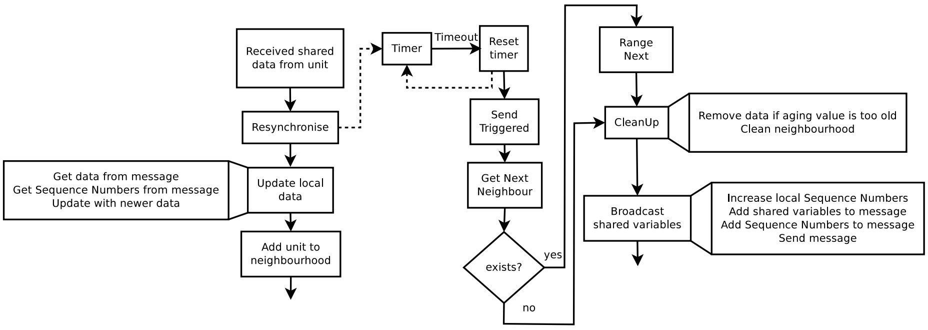


Fig. 6: Complete broadcast protocol - sending and receiving procedures

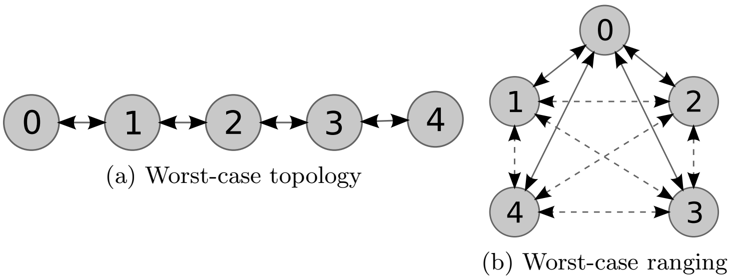
5 Upper Bounds to Information Propagation

Despite the unreliability of the wireless medium, it is reasonable to consider the medium lossless for the purpose of establishing some baseline properties that are intrinsic to the protocol. Here we will analyse the conditions that maximise the information propagation latency in the absence message losses.

First we analyse the worst case topology for information sharing. This situation is similar to the one reported in [2] and corresponds to the case in which all units form a line but sorted such that the identifiers decrease in the direction of the propagation of the information. For example, Fig. 7a shows the worst-case topology for propagating information from unit 4 to unit 0. In this case we will need one initial round for unit 4 to transmit its new information, such as a new node, which will be received by unit 3 that will transmit it it to unit 2 in the following round until the information gets to unit 1. At that point, one slot is enough to finally transfer the information to unit 0. The total worst-case latency is $(n-2) \times t_{up} + t_{slot}$. If any two units switch position, or if there is any parallelism, the latency will be lower.

However, acquiring the pairwise distances also takes time since each node will range only one of its neighbours per round. Therefore, the worst-case situation occurs when a node is connected to all the other $n-1$ units requiring $(n-1) \times t_{up}$ time (Fig. 7b). This latency is not increased by the ranging carried out by the other nodes since they occur in parallel in different slots.

Considering both information dissemination and ranging, simultaneously, we realise that for each extra ranging one unit has to perform, one less round is needed for disseminating the information. For example, in a line topology (Fig. 7a) unit n ranges unit $n-1$ and then, in the same round, transmits its information taking $n-2$ hops plus 1 slot to get to unit 0. If unit n was also connected to unit $n-2$, then it would take two rounds to range both units but it would take one less hop to get the information to unit 0, thus taking the same time as before. Consequently, the worst-case number of rounds required for all units to be in agreement after a topology change is upper bounded by $n-1$.



(a) Worst-case topology

(b) Worst-case ranging

Fig. 7: Worst-case data propagation latency scenarios

Finally, in terms of the maximum age that a given information might develop in the network, let us consider $max_{val} = t_{val}/t_{up}$ as the number of rounds in the information validity interval. Suppose now that a given information is max_{val} old when transmitted by unit n in a worst-case line topology. This information will arrive at unit 0 in less than $n-1$ rounds, thus within $max_{val}+(n-1)$ rounds after its generation. Then, unit 0 will keep it for another max_{val} rounds before removing it. Therefore, the maximum number of rounds that a piece of stale information can remain in the network before being removed is upper bounded by $2 \times max_{val} + (n-1)$.

6 Experimental Results

An experimental validation was carried out with a Nanotron's nanoLoc development kit [5]. This kit includes 5 nodes, each using an Atmega128l μC, communicating in the 2.4GHz ISM band according to IEEE 802.15.4 with a chirp modulation, which allows RF ranging using ToF.

We organise the experiments in two sections, firstly showing the synchronisation capabilities of this algorithm in a small room environment and secondly, exhibiting the improvements of using synchronisation in the ranging process. In all cases, $max_{val} = 10$ rounds, the ranges resolution is 1 byte, expressing distance in dm, and the value 255 is used to signal the out-of-range condition (Ω).

6.1 Validating the Synchronisation Protocol

We start by setting $t_{up} = 500ms$ and activating units 1, 3 and 4 which run the protocol. Unit 0 is used for monitoring purposes, only. As shown in Fig. 8, there are two disjoint subnetworks, one with unit 1 and the other with units 3 and 4. Note that the protocol allows each subnetwork to synchronise internally independently of each other (Fig. 8 left plot, up to round 26). Then, at that point, unit 2 is switched on and connects to both subnetworks, joining them, thus allowing the synchronisation to propagate across. After a short transient of

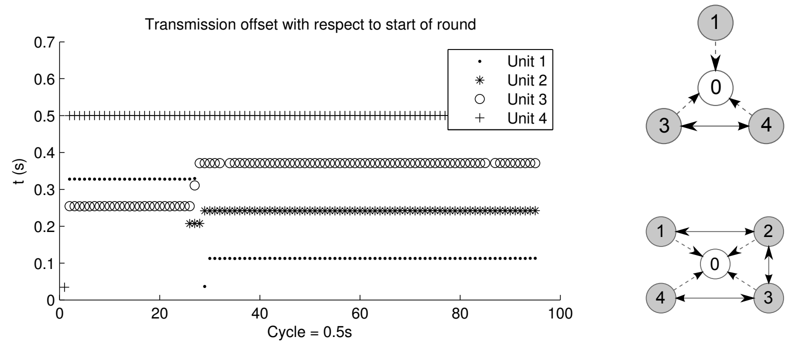


Fig. 8: Synchronising disjoint subnetworks with one unit

2 rounds, all units are synchronised with their transmissions separated as much as possible (125ms).

Figure 9 shows a case with 5 units and $t_{up} = 200ms$, where several consecutive network reconfigurations occur, with nodes joining and leaving. Units 0, 2 and 4 form a network and unit 3 joins at round 24 causing a resynchronisation from 3 to 4 slots. At round 45 unit 1 also joins. Then, at round 60, unit 4 leaves, which causes a resynchronisation later on, after $max_{val} + 1 = 11$ rounds, which is when it is removed by the nodes it was connected to. The same happens when unit 2 and later unit 3 leave the network. All protocol timings were verified.

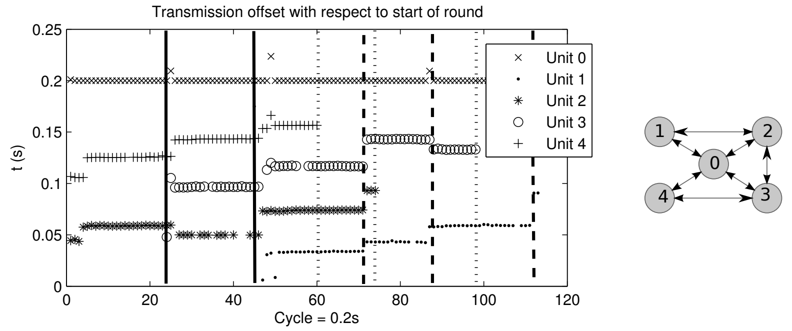
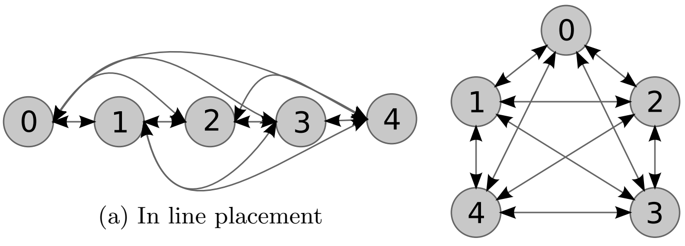


Fig. 9: Synchronisation protocol operating with joining and leaving units

6.2 Improvements in Ranging Success

In the following experiments, we set up a fully linked network with 5 nodes but two physical layouts aiming at analysing the impact the synchronisation has in the ranges performance, both on accuracy and in failure rate.

Concerning accuracy, we used the physical layout of Fig. 10a with a separation of $1m$ between two consecutive units. Unit 0 ranged every other unit and logged the results. We analysed 3500 rounds of operation with, and another 3500 without, synchronisation, with $t_{up} = 200ms$. The ranging results showed that the accuracy was similar in both cases with a negligible difference of the average errors, $abs(mean(D_{error}^{synch}) - mean(D_{error}^{\overline{synch}})) < 0.01m$. The difference in terms of standard deviation of the distance measurements was also small, despite larger, $abs(std(D^{synch}) - std(D^{\overline{synch}})) < 0.1m$.



(a) In line placement

(b) Arbitrary positions

Fig. 10: Two fully linked physical layouts

Concerning ranging failure rates, we used the layout shown in Fig. 10b but we used units 1 to 4 only. Two runs of 5000 consecutive rounds were logged using a $t_{up} = 200ms$ and for both cases, with and without synchronisation. The percentage of failed ranges for each unit is shown in Table 1a. Then we repeated these experiments after having switched on unit 0, which was programmed to send a 127B packet every $20ms$ ($64.3kbps$), without synchronisation, just to create interference. The percentages of failed ranges for each unit, are shown in Table 1b.

The results of the synchronised experiments show a residual percentage of range failures that is similar for all units, between 3% and 4.4%. The results with interference show a minor degradation, with percentages of losses from 3.5% to 4.9%. On the other hand, the results without synchronisation show substantial degradation with certain nodes, in one case going up to 25%. Nevertheless, even without synchronisation it is still possible to find units exhibiting range failures similar to the synchronised case. This is easily explained looking at Fig. 11. In fact, without synchronisation some units will end up transmitting almost at the same time, which causes a high number of failures due to collisions, and other units will transmit very far apart, thus similarly to the synchronised case. For example, in Fig. 11b unit 1 has a very high clearance from the other units while

Table 1: Experimental results using topology in Fig. 10b

(a) Baseline measurements

$t_{up} = 200ms$
5000 samples
Interference: No

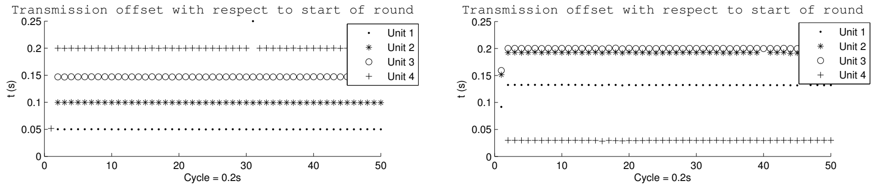
Synchronisation	Unit	Error Rates	
		Run 1	Run 2
Yes	1	3.82%	2.98%
	2	3.70%	3.53%
	3	4.36%	3.54%
	4	4.42%	3.52%
MEAN		3.73%	
STD		0.47%	
No	1	4.20%	3.92%
	2	13.64%	23.94%
	3	11.82%	24.54%
	4	3.62%	14.70%
MEAN		12.55%	
STD		8.48%	

(b) Measurements with noise

$t_{up} = 200ms$
5000 samples
Interference: $127 Bytes/20ms$

Synchronisation	Unit	Error Rates	
		Run 1	Run 2
Yes	1	4.16%	3.92%
	2	4.32%	3.96%
	3	3.54%	3.50%
	4	4.46%	4.90%
MEAN		4.09%	
STD		0.47%	
No	1	4.36%	12.00%
	2	16.14%	4.88%
	3	10.82%	5.94%
	4	4.28%	13.02%
MEAN		8.93%	
STD		4.62%	

units 2 and 3 are transmitting very close to each other. This log corresponds to the baseline Run 2 experiments without synchronisation in Table 1a. With synchronisation the team units do not practically interfere with each other. Consequently, the average range failure rate without synchronisation and without interference is 3.3 times higher than that obtained with synchronisation (12.5% compared to 3.7%), and the standard deviation of the units failures rates is one order of magnitude higher without than with synchronisation (8.48% and 4.62% compared to 0.47%). However, the actual degradation of the situation without synchronisation depends on too many factors, such as starting conditions, and is thus very difficult to characterise accurately.



(a) Synchronised (b) Non-Synchronised

Fig. 11: Periodic dissemination with and without synchronisation

7 Conclusions

Robot motion coordination and cooperative sensing are two areas that benefit from multi-robot cooperation based on wireless communication. In this paper we proposed a novel ad-hoc synchronisation / broadcast protocol that integrates the dissemination of ranging data through the network in an effective way contributing to an improved relative localisation service. This protocol extends a previous one named RA-TDMA which is meant for infrastructured scenarios.

Experimental results with IEEE 802.15.4 nanoLOC nodes validate the properties of the protocol, namely its ability to enforce synchronisation in ad-hoc scenarios even when a single path connects different nodes, its ability to acquire and efficiently disseminate ranging information thorough the network, as well as its effectiveness in reducing the failure rates of the ranging operations.

As future work we plan to experiment on the limits of units velocities that the protocol can cope with.

References

1. Crenshaw, T., Tirumala, A., Hoke, S., Caccamo, M.: A robust implicit access protocol for real-time wireless collaboration pp. 177 – 186 (2005)
2. Facchinetti, T., Buttazzo, G., Almeida, L.: Dynamic resource reservation and connectivity tracking to support real-time communication among mobile units. EURASIP J. Wirel. Commun. Netw. 2005(5), 712–730 (2005)
3. Fazenda, P., Lima, P.: Non-holonomic robot formations with obstacle compliant geometry (2007)
4. Li, H., Almeida, L., Carramate, F., Wang, Z., Sun, Y.: Connectivity-aware motion control among autonomous mobile units pp. 155 –162 (2008)
5. nanoloc development kit (2010), `http://www.nanotron.com/EN/PR_nl_dev_kit.php`
6. Neuwinger, B., Witkowski, U., Ruckert, U.: Ad-hoc communication and localization system for mobile robots. In: Advances in Robotics, Lecture Notes in Computer Science, vol. 5744, pp. 220–229. Springer Berlin / Heidelberg (2009)
7. Oliveira, L., Li, H., Almeida, L.: Experiments with navigation based on the rss of wireless communication (march 2010)
8. Ryan, A., Tisdale, J., Godwin, M., Coatta, D., Nguyen, D., Spry, S., Sengupta, R., Hedrick, J.: Decentralized control of unmanned aerial vehicle collaborative sensing missions. American Control Conference, 2007. ACC '07 pp. 4672 –4677 (jul 2007)
9. Santos, F., Almeida, L., Lopes, L.: Self-configuration of an adaptive tdma wireless communication protocol for teams of mobile robots. Emerging Technologies and Factory Automation, 2008. ETFA 2008. IEEE International Conference on pp. 1197 –1204 (sep 2008)
10. Song, K.T., Tsai, C.Y., Huang, C.H.C.: Multi-robot cooperative sensing and localization. Automation and Logistics, 2008. ICAL 2008. IEEE International Conference on pp. 431–436 (sep 2008)
11. Wang, Z., Hirata, Y., Kosuge, K.: Checking movable configuration space in c-closure object for object caging and handling. Robotics and Biomimetics (ROBIO). 2005 IEEE International Conference on pp. 336 –341 (2005)