

# Information Aggregation Mechanisms in Social Odometry

Roman Miletitch<sup>1</sup>, Vito Trianni<sup>3</sup>, Alexandre Campo<sup>2</sup> and Marco Dorigo<sup>1</sup>

<sup>1</sup>IRIDIA, CoDE, Université Libre de Bruxelles, Belgium

<sup>2</sup>Unit of Social Ecology, Université Libre de Bruxelles, Belgium  
{rmiletitch,acampo,mdorigo}@ulb.ac.be

<sup>3</sup>ISTC, National Research Council, Rome, Italy  
vito.trianni@istc.cnr.it

## Abstract

Multi-robot exploration and navigation is a challenging task, especially within the swarm robotics domain, in which the individual robots have limited capabilities and have access to local information only. An interesting approach to exploration and navigation in swarm robotics is *social odometry*, that is, a cooperative strategy in which robots exploit odometry for individual navigation, and share their own position estimation through peer-to-peer local communication to collectively reduce the estimation error. In this paper, the robots have to localize both a home and a goal location and navigate back and forth between them. The way in which navigational information is aggregated influences both the efficiency in navigation between the two areas, and the self-organized selection of better paths. We propose three new parameter-free mechanisms for information aggregation and we provide an extensive study to ascertain their properties in terms of navigation efficiency and collective decision.

## Introduction

Navigation is a basic task for robots in most application domains, would that be for cleaning a room or demining a field. In few cases, the environment is completely known in advance, and therefore a detailed navigation plan can be produced. Most often, the environment is not completely known and exploration is required to identify and reach the desired locations. When multiple robots explore an unknown environment, cooperative strategies can be used to improve exploration and navigation efficiency. This is particularly useful in the swarm robotics domain, in which individual robots cannot rely on global information or complex algorithms (Brambilla et al., 2013). In this paper, we study a cooperative exploration and navigation strategy based on the peer-to-peer exchange of information among robots in a swarm. We propose three variants of the information aggregation mechanism, and we investigate their impact over the dynamics of navigation of the swarm as well as the resulting efficiency with respect to an exploration and exploitation task.

Exploration and navigation strategies in swarm robotics should present a low complexity to match the limited capabilities of the individual robots. The simplest way to explore

and navigate in a closed area is through random walk. While not being the most efficient way, it assures that the robots reach every part of the environment, even if this may require a long time. In order to improve over a purely random exploration, the robots can memorize and map their surroundings to avoid previously explored zones (Thrun, 2008) to reach specific areas of interest. To this purpose, the robot can position itself on the map and navigate in the environment using dead-reckoning techniques such as odometry. Odometry relies on the integration over time of the movement vector—as perceived through the robot (proprioceptive) sensors—in order to maintain an estimate of the robot position. However, this approach is quite error prone since estimation errors are cumulated over time, therefore requiring techniques for error reduction such as Kalman filters (Thrun et al., 2005).

Alternatively, the estimation error can be reduced through the shared effort of multiple robots exchanging structured information (Martinelli et al., 2005). By sharing the estimated position of a landmark, the robots can collectively reduce the overall odometric error. This is a straightforward mechanism that easily lends itself to implementation on very simple robots. Therefore, the collective reduction of odometry errors can be instantiated also in swarm robotics contexts, as it complies with the inherent limitations of the robots. This mechanism was first introduced by Gutiérrez et al. (2009) and is referred to as *social odometry*. In this approach, the robots estimate the navigation path between two target areas in the environment (*i.e.*, home and goal locations) using odometry and attach to this estimate a confidence level that decreases with the distance travelled. At the same time, the robots share their navigation information within the swarm in a local peer-to-peer manner. Thanks to this process, information about target areas spreads gradually within the swarm, contributing to reduce the error in the position estimation. Overall, this decentralized process results in an increased efficiency in the swarm navigation abilities.

An interesting aspect of social odometry is that it naturally leads to the emergence of collective decisions within the swarm (Gutiérrez et al., 2010). Indeed, when there are multiple goal areas to localize (*e.g.*, multiple resources to

exploit), by sharing the available information the robots not only improve the accuracy of their localization but can also decide which area to target. The sum of individual decisions leads to a self-organized behaviour that makes the swarm choose between focusing on a single area/resource or exploiting in parallel several ones.

The efficiency of social odometry as a navigation mechanism and the resulting collective dynamics of decision making depend heavily on the way information is shared and aggregated in the robot swarm. In particular, we found that even small variations in some parameters of the individual behaviour may lead to huge differences in the swarm dynamics. For this reason, in this paper we propose three new parameter-free mechanisms for information aggregation and processing, and study them to ascertain their properties in terms of (i) the efficiency in supporting the swarm navigation and (ii) the ability to produce a collective decision when multiple goal locations are present. The results we obtain allow us to understand the properties of the collective behaviours generated by each information processing mechanism. On the basis of such knowledge and depending on the specific needs of the application at hand, a principled choice of the most appropriate mechanism can be made.

## State of the Art

**Navigation in Swarm Robotics** There are various ways to improve navigation through information-sharing within a swarm. Ducatelle et al. (2011) model a swarm as a communication network that propagates relevant information. Each robot in the swarm maintains a table with navigation information about all known robots, similar to how nodes in a mobile ad hoc network maintain routing tables. Then, the robots propagate the available information and use the table to find the best path to reach a target robot within the swarm. Sperati et al. (2011) also study navigation in a swarm robotics context. In this case, communication is performed through visual signals only and therefore the information exchanged is much less structured. For this reason, they used artificial evolution to synthesize effective navigation strategies.

Several studies in swarm robotics implement navigation and exploration algorithms without sharing structured information, sometimes exploiting robots as physical landmarks. Rekleitis et al. (2001) divided the swarm in two teams, one moving and the other stationary, serving as a reference for navigation. The teams alternate between stationary and moving states. Nouyan et al. (2009) exploit robots to form complex structures such as chains, in which one end of the chain connects to a central place while the other end explores the environment. Once the goal location is reached, the chain can be exploited by other robots for navigation purposes, or a bucket brigade method can be used for transporting objects along the chain (Ostergaard et al., 2001).

**Collective Decisions** When there are several goal/resource locations present in the environment, the robots may make a collective decision and focus on the exploitation of a single one. This can be beneficial if it is necessary to aggregate a sufficient number of robots in support of the collective localization, or if exploitation requires several robots at the resource. However, this may lead to congestion (*i.e.*, the path to the resource is overused and robots have trouble navigating) or overexploitation of the resource. In this case, the swarm is better off exploiting several resources in parallel.

In order to agree on one option, the robots can either switch to the best option available in their neighbourhood, or average out all the available information. Social odometry allows doing both simply by tuning a single parameter (Gutiérrez et al., 2010). Olfati-Saber et al. (2007) study the swarm as a multi-agent network and present a theoretical framework for the analysis of consensus algorithms. It is possible to obtain collective decisions also through the amplification of the various opinions present in the swarm. Following this approach, the more an opinion is represented in the swarm, the higher the probability of robots switching their opinion (Garnier et al., 2007, 2009; Montes de Oca et al., 2011). This approach requires gathering the opinion of several neighbours, while social odometry works with peer-to-peer interactions, which is easier to implement.

## Social Odometry & Information Processing

In our experiments, the goal of the robots is to locate both a home area and a goal area and then to efficiently navigate back and forth between them. Once one of these two target areas is discovered, its position is kept in memory and updated using odometry. The information about target areas is shared with other robots upon encounter, following the social odometry mechanism. Within this framework, we study the navigation process, the social dynamics, and the link between the two. In the following, we first describe how robot use the available information (either from individual or social odometry) for navigation purposes. Then, we introduce the information processing mechanisms we have devised.

## The Controller

The behaviour of the robot is defined by a finite state automaton with five states: *Explore*, *Go Home*, *Go to Goal*, *Leave Home*, *Leave Goal* (Fig. 1). Robots start in the *Explore* state and return to it whenever they lack relevant information. The other four states form a loop that corresponds to the robot navigating back and forth between the target areas: go to a target area, enter and leave it, then go to the next one. On top of these control states, both short and long range collision avoidance is implemented.

The robots start without any prior knowledge about the location of the target areas. Therefore, they first have to explore the arena. When in the *Explore* state, the robots perform a random walk until they discover the position of

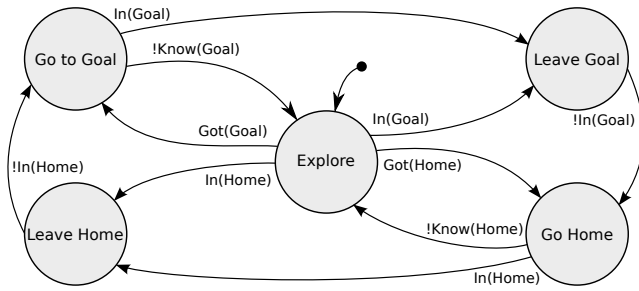


Figure 1: Robot’s finite-state automaton. The circles define the states while the arrows define the transitions.  $In(Area)$ ,  $Area \in \{Home, Goal\}$ , is true when a robot senses the grey level of the area,  $Know(Area)$  is true when the robot knows the position of the area,  $Got(Area)$  is true when it just gets this estimation. The robots start in the *Explore* state.

both target areas (home and goal). This can happen in two ways: either they receive relevant information from teammates or they stumble upon a target location ( $Got(Area)$  becomes true, with  $Area \in \{Home, Goal\}$ ). In both the *Go to Goal* and *Go Home* states, the robots move straight to the target location, possibly avoiding other robots and obstacles. Along their travel, they update the target areas location using odometry and update their confidence in the information. The confidence is defined as the inverse of the distance that the robot had travelled from the target area. Therefore, a straight path results in a higher confidence than a curved one.

Once a robot reaches an area (*i.e.*,  $In(Area)$  is true), it traverses it in a straight line (possibly dodging other robots to avoid collisions) and stores the area location. In order to get an estimated position closer to the center of the area, the robot averages its entering and its exiting positions. No matter how many goals there are in the arena, the robots always memorize only one home and one goal (the last seen or agreed upon).

### Information Sharing and Processing

While robots navigate between target areas, they share the information they have on relative locations in order to counterbalance the decrease in information confidence. Not all information is shared at the same time. When in the *Explore* state the robots share the sole information they have. In the other states, the robots share only the information of the last visited location. The information received by a robot is aggregated with the robot’s own information. The way this aggregation is performed depends on the information processing mechanism implemented. Given that robots do not share the same reference frame, a transformation is needed. This is made possible by knowing the relative position (range and bearing) of the robot that is sharing its information (for more details, see Gutiérrez et al., 2009). Once the location is obtained in a shared reference frame, the information aggrega-

tion process takes place. Here, we first describe the information aggregation mechanism used by Gutiérrez et al. (2009), and then we introduce our contributed mechanisms.

Let  $i$  and  $j$  be two robots,  $i$  receiving a message from  $j$ . Let  $\mathbf{p}_i$ ,  $\mathbf{p}_j$  be their estimated position of an area (either home or goal) and  $c_i$ ,  $c_j$  the confidence over their respective estimation. The result of any aggregation is the updated couple  $\langle \mathbf{p}_i, c_i \rangle$ . The aggregation mechanism used by Gutiérrez et al. (2009) is based on a Fermi distribution. A weight is calculated from the difference in confidence in order to make a linear combination of the positions:

$$\langle \mathbf{p}_i, c_i \rangle \leftarrow k \cdot \langle \mathbf{p}_i, c_i \rangle + (1 - k) \cdot \langle \mathbf{p}_j, c_j \rangle$$

$$k = \frac{1}{1 + e^{-\beta(c_i - c_j)}}$$

The parameter  $\beta$  measures the importance of the relative confidence levels in the information aggregation. For low values, the aggregation is close to an average, ignoring the confidence. For higher values, the aggregation is stiff: only the information with highest confidence is kept. Finding the right value of  $\beta$  is often a process of trial and error. Our contribution in this paper is the introduction of three parameter-free aggregation mechanisms: *Hard Switch (HS)*, *Random Switch (RS)* and *Weighted Average (WA)*.

**Hard Switch (HS)** In this winner-take-all mechanism, the robots keep the information with highest confidence (either the current information or the received one) and discard the other one. This mimics the Fermi mechanism with a high  $\beta$ .

$$\langle \mathbf{p}_i, c_i \rangle \leftarrow \langle \mathbf{p}_x, c_x \rangle, \quad x = \arg \max_{k \in \{i, j\}} c_k$$

**Random Switch (RS)** As in the mechanism above, here the robots keep one piece of information and discard the other. In this case, however, the switch is stochastic: the higher the confidence, the higher the probability of accepting the information. In practice, this mechanism is a stochastic version of the *HS*.

$$P(\langle \mathbf{p}_i, c_i \rangle \leftarrow \langle \mathbf{p}_j, c_j \rangle) = \frac{c_j}{c_i + c_j}$$

**Weighted Average (WA)** This mechanism consists in a linear combination of both estimated positions with their confidence as weight. On the one hand this implies no loss of information, on the other hand, when information about different goals is aggregated, the new position may not coincide with a real goal location, leading to the apparition of artefacts. While the Fermi mechanism focuses on the difference between the two confidences, here we directly use each of them as weights.

$$\langle \mathbf{p}_i, c_i \rangle \leftarrow \left\langle \frac{c_i \cdot \mathbf{p}_i + c_j \cdot \mathbf{p}_j}{c_i + c_j}, \frac{c_i + c_j}{2} \right\rangle$$

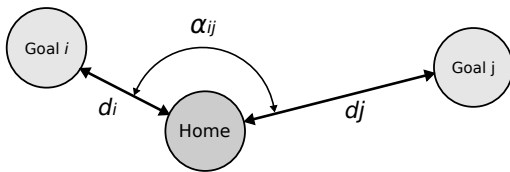


Figure 2: Setup of the experimental arena. The home area is placed in the center of a circular arena of 11 m radius surrounded by walls (not displayed). The goals are characterised by their distance to the home  $d_i, d_j$  and the angles they form with each other  $\alpha_{ij}$ .

## Experiments

We used an experimental setup with as few variables as possible: a circular arena (radius: 11 m) with the home in the center and the goals scattered around (Fig. 2, surrounding walls not shown). The goals are defined by their distance to the home ( $d_i$ ) and the angle between each other ( $\alpha_{ij} \in [\pi/3, \pi]$ ). Both goal and home are of radius 50 cm, and are differently coloured in grey levels to be distinguished by the robots.

Our experiments are performed in the ARGoS open source multi-robot simulator (Pinciroli et al., 2012). The robots we use are the marXbots (Bonani et al., 2010). To accomplish their task, the robots are equipped with several sensorimotor and communication devices. In our experiments, the robots use the infrared ground sensors to check whether they entered an area and to detect its type (home or goal) depending on the area’s grey level. They also use the infrared proximity sensors for short range collision avoidance and the range&bearing device for both communication and long range collision avoidance among robots (Bonani et al., 2010). This last device gives both angle and distance between neighbouring robots and allows them to send short messages. Wheels encoders provide the movement vector for odometry. A simulated gaussian noise with 5% standard deviation models the odometry estimation error. The control loop is executed 10 times per second. Unless stated otherwise, we used 75 robots spawned randomly.

By varying the number of goals, we study different aspects of the collective behaviour, such as the impact of the density of robots on their navigation abilities, the collective decision made by the swarm in a two goals setup, and how this generalizes in multiple goals setups. In the following, we briefly describe the experiments we present in this paper.

**Single Goal** When a single goal is present, we expect that all robots will converge on the same path. The more robots in the arena, the harder it is for them to avoid each other. As density rises, the robots have to handle more and more congestion on their path, which leads them to travel bigger distances and to accumulate more error. This also corresponds to less round trips between home and goal, hence lowering

the efficiency of the swarm. We define the density on a path as the number of robots on it divided by its length.

In order to study the impact of density on navigation, we devised an experimental setup in which we vary both the distance between the home and the goal and the number of robots. All three information processing mechanisms are tested and compared with a benchmark condition in which the robots are provided with perfect information (*PI*) about the goal and home locations. In each experiment, we measure the navigation speed, computed as the number of round trips over time and we study its evolution for values of density between 2 and 40 robots/m. For each density value, we run 100 trials in which we randomly draw the distance between home and goal in the interval [3,8] m, and we compute the corresponding number of robots to obtain the specified density value (which will be in the range [6,320]).

**Two Goals** When there is more than one goal, a decision has to be made as how to spread the robots among the available paths. In this setup, we study if and how the robots converge on one path as well as the implications of such convergence over efficiency. In order to study this decision making process, we count the number of robots committed to each goal, as well as the uncommitted ones. Given that robots do not distinguish between different goals and only store one estimated position  $\mathbf{p}_g$ , a robot is considered to be committed to a goal  $i$  among  $n$  possible if it has information about both goal ( $c_g \neq 0$ ) and home ( $c_h \neq 0$ ), and if goal  $i$  is the closest one to the robot’s estimated goal position  $\mathbf{p}_g$ .

In this setup, we have two goals which can be either at short distance (5 m) or at long distance (8 m). We run experiments with both equal and different distances for the goals: Short/Short (*SS*), Short/Long (*SL*) and Long/Long (*LL*). For each condition, we perform 1000 replications by randomly varying the angle between the sources with  $\alpha_{ij} \in [\pi/3, \pi]$  (cf. Fig. 2).

**Multiple Goals** The environment in which a swarm evolves is rarely as simple as in the two goals setup. Through a multiple goals setup, we enquire about the scalability of the results previously gathered.  $M$  goals are uniformly distributed around the home location, with an angular separation between adjacent goals of  $\pi/M$ , where  $M \in [3, 6]$ . To investigate both the navigation and the decision making abilities, we test three different conditions. Either all goals are at the same distance, short (*SSS*) or long (*LLL*), or a single goal is closer to home (*SLL*). For each condition, we performed 250 trials.

## Results

Each trial in all the previous setups lasts 20 minutes of simulated time. We use the same random initialization in all the runs for the different opinion processing. For each run



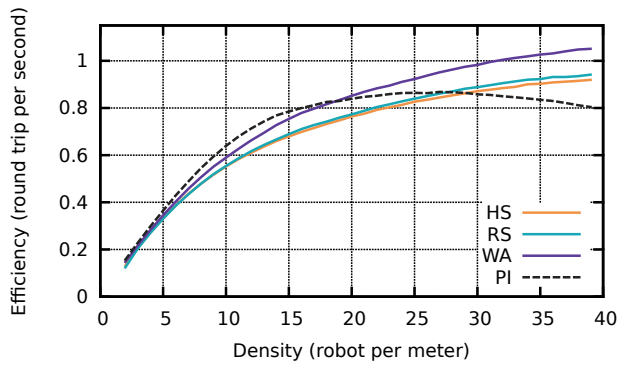


Figure 3: Impact of density on navigation efficiency for each mechanism and in the perfect information control condition. Each line is the mean over 100 trials.

we compute the number of robots on each path to study the dynamics of collective decisions, and the number of round trips to study the navigation efficiency.

### Navigation

As we can see in Fig. 3, all the proposed mechanisms and the control condition with perfect information (*PI*) follow the same tendency. For low densities, we can observe a linear increase in the number of round trips. With higher densities, the growth slows down. As expected, robots with perfect information are the most efficient at first, but their efficiency reaches a peak because of the artefacts created by perfect information. With *PI*, since all robots aim at the center of the target areas (either home or goal), as the density rises they have increased difficulties in avoiding collisions and in entering or exiting the target areas.

Congestion has a lower impact on navigation efficiency with social odometry. In this case, *WA* proves to be more resilient to congestion than *HS* and *RS*. This is due to a smoother navigation in the surrounding of the home and goals, where robots try to enter small and densely populated area. First, since the *WA* mechanism never discards information but averages it, the precision on the estimated position is better than with *HS* or *RS*. Second, the reception of even a slightly better information is smoothly integrated in the *WA* mechanisms, while in both *HS* and *RS* it may cause a large leap of the new location, which may be difficult to reach in case of high densities.

### Collective Decision

Congestion explains why sometimes it is better to spread along multiple paths when there is more than one goal/resource. This decisions impacts not only the efficiency but also the spatial arrangement of the swarm and the way it reacts to changes in the environment.

**Decision** The decision pattern of the swarm results from the sum of local decisions made by the robots. The dynam-

ics of the collective decision are shown in Fig. 4, which plots the convergence pattern generated by the *HS* mechanism when confronted with the *SL* experimental condition. Here the swarm decides to focus on the closest area/resource and most robots converge quickly on the associated path. This behaviour is typical of all three social mechanisms when there is a goal closer to home. We can observe three different phases. At first (0-120 s), most robots are uncommitted and explore for goal areas, reinforcing each as they discover them. Then (120-400 s), a competition among the two alternative paths occurs. The shorter path is reinforced more because of the better information the robots have when encountering robots coming from the other goal. Eventually, the swarm enters a maximization state in which mostly one path is exploited while uncommitted robots continue to join.

Fig. 5 left shows the percentage of robots that choose path A (*i.e.*, the shortest path in the *SL* condition). We note that in the *SL* case, all information aggregation mechanisms lead to convergence on a single path with at least 90% of the robots. Both *HS* and *RS* always lead to a convergence on the closest goal. Similarly for *WA*, which however presents also a low probability to make the robots converge on the distant goal. This happens because with *WA* no information is discarded. When a large number of robots discovers the distant goal early in the experiment, they may influence the whole swarm despite the lower confidence in their information. This cannot happen in the *HS* and *RS*, because low quality information is instantly discarded. In both the *SS* and *LL* experimental conditions, when there is no better choice, *HS* and *RS* lead to a split in the swarm, and robots spread among the two paths (Fig. 5 left). In these experimental conditions, the more robots on a path, the higher the congestion, and the larger the distance the robots travel. This causes robots to have worse confidence in their information with respect to those from a less congested path. Therefore, switches to the other path are very likely. Congestion cre-

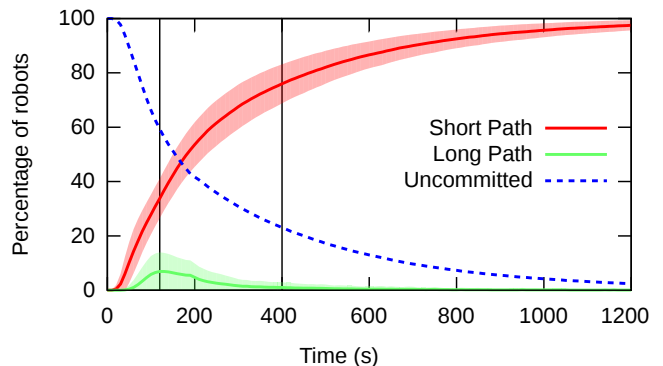


Figure 4: Evolution of the robots repartition between the two target areas using *Hard Switch* in the Short/Long condition. Bold lines indicate the mean over 1000 repetitions, and the shaded areas indicate the standard deviation.

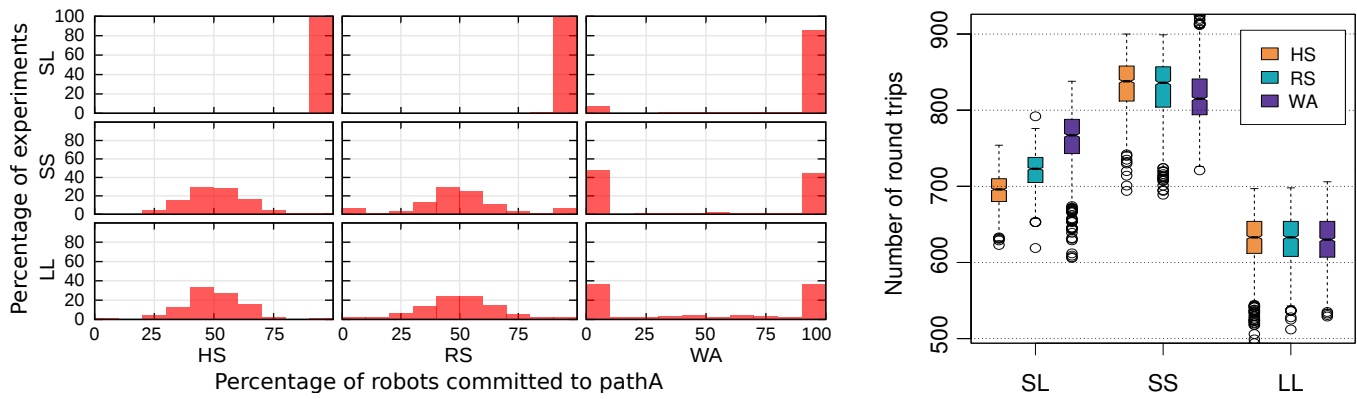


Figure 5: Left: robots repartition on path A. Each histogram shows the observed frequencies of the number of robots committed to path A (the shortest possible path). Right: efficiency of the swarm for two goals, for all mechanisms and conditions. Each box represents the inter-quartile range, whiskers extend to 1.5 times the corresponding quartiles, and dots represent outliers.

ates a sort of negative feedback that leads to an oscillating dynamics in which no decision ends up being taken. On the contrary, *WA* is not affected by such negative feedback and systematically leads to convergence (randomly on either path, the setup being symmetrical). Indeed, the worse confidence resulting from congestion is counterbalanced by the larger number of robots with which the information is shared and averaged. Therefore, the swarm converges to the more populated path.

**Efficiency** The robot behaviour does not explicitly encode the ability to make collective decisions. Instead, it is conceived to provide efficient navigation ability thanks to the information shared within the swarm. The decision process is an emergent result of this behaviour and so is the variation in efficiency depending on the setup and the mechanisms involved, as shown in Fig. 5 right. In the *SL* condition, all three mechanisms make the robots converge on the closest path, therefore resulting in density of 15 robots/m. As shown in Fig. 3, *WA* is more resilient to congestion, and this is why it is the most efficient mechanism in this setup, followed by *RS* and *HS*. In the *SS* condition, both *HS* and *RS* result in the swarm splitting between the two paths as discussed above. By exploiting two paths with a low density of 7.5 robots/m (instead of one with high density of 15 robots/m) the robots create less congestion, which explains why the performance for *HS* and *RS* is slightly better than in the *WA* case. Indeed, *WA* makes the swarm converge on a single path with high density, and navigation is slightly less efficient. Congestion has a lower impact in the *LL* conditions as both densities (9.4 robots/m on a single path, 4.7 robots/m on two paths) fall in the linear part of the congestion curve (see Fig. 3), explaining why the mechanisms result in the same efficiency.

### Generalization to Multiple Goals

The dynamics we observe with multiple goal locations are similar to the ones displayed in the two goals setup, no mat-

ter the number of added goals. Fig. 6 shows the percentage of robots that choose path A (*i.e.*, the shortest path in the *SLL* condition), when multiple goal locations are present. All mechanisms leads to convergence in the *SLL* case, even if *WA* sometimes leads to the selection of one of the distant goals, for the same reasons discussed in the two goals setup. We can observe a similar splitting behaviour in the *SSS* and *LLL* conditions for both *HS* and *RS*, while convergence is observed for *WA*. When the swarm splits, the repartition of robots is not anymore centred on 50% but closer to 33%, meaning that the repartition is not anymore among only two paths. Nonetheless, not all are exploited at the same time, as can be inferred from the existence of paths selected by no robot. This can be explained by the oscillation dynamics discussed earlier. When the amplitude of the oscillations is greater than the number of robots on a path, all the robots on this path switch to another one. This happens in the case of multiple goals because robots are spread among more paths and therefore their number on each is lower.

To better understand the exploitation of the available resources/goals, in Tab. 1 we report the average percentage of robots on the different paths, ordered from the most to the least exploited path. We note that the number of exploited goal locations is most of the time no more than 3. This explains why the efficiency of the swarm does not vary with the number of available resources, as shown in Fig. 7. The slight increase in performance can be attributed to the fact that the more goals there are, the easier it is for uncommitted robots to join a path earlier in the experiment. Overall, we note similar patterns over efficiency between the multiple goals condition and the two goals condition.

When there are multiple goals, *WA* in the *SLL* condition leads to a frequent selection of a distant goal instead of the closest one, as shown in Fig. 6. If several distant locations are present, they end up reinforcing each other as their angular distance becomes smaller. In other words, two distant goal locations that are close to each other attract more robots

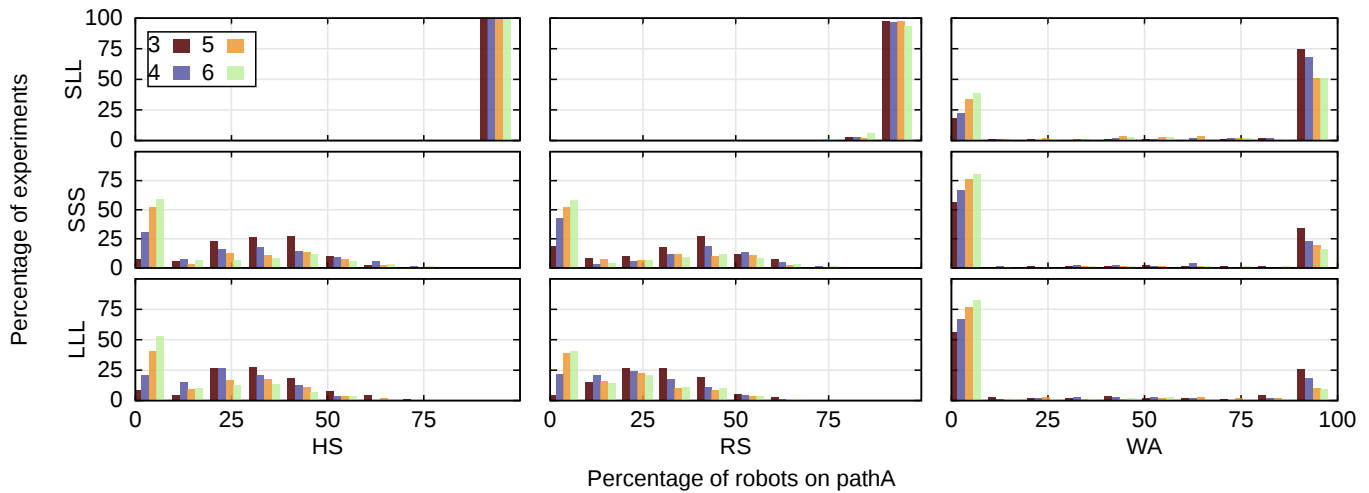


Figure 6: Robots repartition on path A for different number of goal areas (3,4, 5 and 6). Each histogram shows the observed frequencies of the number of robots committed to path A (the shortest possible path).

Table 1: Repartition in percentage of robots for 3, 4, 5 and 6 goals. The 1<sup>st</sup> goal is the one associated with the highest number of robots. The mean and maximum of the standard deviation is (4.7, 10.5) for *HS* and *RS* and (6.1, 16.9) for *WA*.

	<i>SL</i>			<i>SS</i>			<i>LL</i>		
	<i>HS</i>	<i>RS</i>	<i>WA</i>	<i>HS</i>	<i>RS</i>	<i>WA</i>	<i>HS</i>	<i>RS</i>	<i>WA</i>
1 <sup>st</sup>	98.5	98.1	96.0	48.0	52.6	93.0	48.8	47.0	90.8
2 <sup>nd</sup>	0.1	0.6	2.3	34.3	37.3	5.7	33.4	32.5	7.0
3 <sup>rd</sup>	0.0	0.0	0.0	17.2	9.7	0.0	17.4	19.7	0.1
1 <sup>st</sup>	98.4	97.7	95.2	50.6	54.1	92.3	44.8	43.8	89.5
2 <sup>nd</sup>	0.2	1.0	3.6	35.2	38.0	6.8	32.0	31.0	9.2
3 <sup>rd</sup>	0.0	0.1	0.0	12.5	6.9	0.0	17.6	17.2	0.1
4 <sup>th</sup>	0.0	0.0	0.0	2.1	0.7	0.0	5.1	7.0	0.0
1 <sup>st</sup>	98.6	97.3	92.4	51.1	51.1	94.8	44.9	42.6	89.4
2 <sup>nd</sup>	0.2	1.0	6.8	35.2	37.0	4.5	31.6	30.0	9.5
3 <sup>rd</sup>	0.0	0.1	0.0	12.5	10.4	0.2	17.7	17.7	0.5
4 <sup>th</sup>	0.0	0.0	0.0	1.1	1.1	0.0	5.1	7.3	0.0
5 <sup>th</sup>	0.0	0.0	0.0	0.0	0.0	0.0	0.3	1.4	0.0
1 <sup>st</sup>	98.6	97.3	93.6	50.1	53.0	94.7	43.7	42.4	88.5
2 <sup>nd</sup>	0.2	1.5	5.4	34.9	36.1	4.7	31.7	28.2	10.4
3 <sup>rd</sup>	0.0	0.1	0.5	13.5	9.2	0.2	17.2	17.3	0.6
4 <sup>th</sup>	0.0	0.0	0.0	1.4	1.3	0.0	6.0	8.3	0.0
5 <sup>th</sup>	0.0	0.0	0.0	0.1	0.2	0.0	0.8	2.3	0.0
6 <sup>th</sup>	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.3	0.0

than a single closer location. This explains why the chance of *WA* leading to the selection of a distant goal increases with the number of goals.

### Discussion

The experiments above reveal the specificities of the three information aggregation mechanisms. *WA* leads to convergence to a single path in all conditions, but this is slower and error-prone. In the whole, *WA* leads to better cohesion of the

swarm and deals better with congestion thanks to more accurate information about the target areas. *HS* and *RS* also lead to convergence when there is a shorter path to exploit, and handle better the presence of multiple distant goal locations. When congestion results in inefficient navigation, both mechanisms lead to the exploitation of multiple paths, spreading the load of robots in a balanced way with similar dynamics, although *HS* appears to be stiffer than *RS*.

### Conclusions

In this paper, we presented an extensive analysis of three parameter-free information processing mechanisms for social odometry. We studied the impact of these mechanisms on the navigation efficiency and on the dynamics of the swarm. In particular, we observed how the information processing mechanism can either lead to convergence on the exploitation of a single path, or to splitting over multiple comparable options. These results are meant to give future designer a guideline of which mechanism to choose depending on the situation at hand.

In future work, we plan to further investigate the dynamics of social odometry in order to provide an optimal load-balancing behaviour. This would maximize the exploitation of different resources and provide the swarm the ability to react to changes in its environment in real time. Additionally, we will experiment with more complex paths, for instance in the presence of obstacles. Also, physical objects to be retrieved may be placed within the goal areas in order to simulate a more realistic environment and making it possible to test the collective behaviour with real robots. Last, heterogeneity can be added in the swarm. On the one hand, individual robots may get committed to a goal with different individual preferences, leading to a better exploration of the environment. On the other hand, different groups of robots

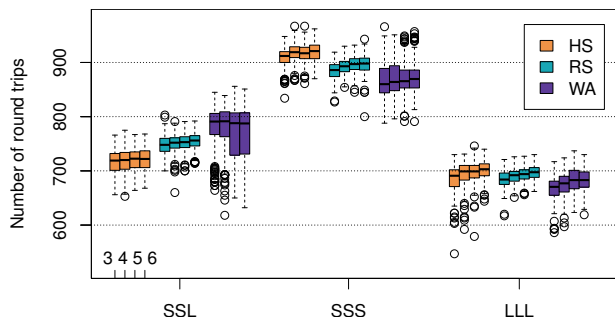


Figure 7: Efficiency of the swarm for multiple goals and all mechanisms and conditions. See Fig. 5 for more details.

could compete for the best source, each of them having different information aggregation mechanisms, leading to a different exploitation of resources among different groups.

### Acknowledgments

This work has been supported by the Italian CNR and the Belgian F.R.S.-FNRS within the EUROCORES Programme EuroBioSAS of the European Science Foundation.

### References

- Bonani, M., Longchamp, V., Magnenat, S., Rétornaz, P., Burnier, D., Roulet, G., Vaussard, F., Bleuler, H., and Mondada, F. (2010). The MarXbot, a miniature mobile robot opening new perspectives for the collective-robotic research. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 4187–4193. IEEE Press, Piscataway, NJ.
- Brambilla, M., Ferrante, E., Birattari, M., and Dorigo, M. (2013). Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7(1):1–41.
- Ducatelle, F., Di Caro, G., Pinciroli, C., Mondada, F., and Gambardella, L. (2011). Communication assisted navigation in robotic swarms: Self-organization and cooperation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4981–4988. IEEE Computer Society Press, Los Alamitos, CA.
- Garnier, S., Gautrais, J., Asadpour, M., Jost, C., and Theraulaz, G. (2009). Self-organized aggregation triggers collective decision making in a group of cockroach-like robots. *Adaptive Behavior*, 17(2):109–133.
- Garnier, S., Tâche, F., Combe, M., Grimal, A., and Theraulaz, G. (2007). Alice in pheromone land: An experimental setup for the study of ant-like robots. In *Swarm Intelligence Symposium, 2007*, pages 37–44. IEEE Press, New York City, NY.
- Gutiérrez, A., Campo, A., Monasterio-Huelin, F., Magdalena, L., and Dorigo, M. (2010). Collective decision-making based on social odometry. *Neural computing & applications*, 19(6):807–823.
- Gutiérrez, A., Campo, A., Santos, F., Monasterio-Huelin Maciá, F., and Dorigo, M. (2009). Social odometry: imitation based odometry in collective robotics. *International Journal of Advanced Robotic Systems*, 6(2):129–136.
- Martinelli, A., Pont, F., and Siegwart, R. (2005). Multi-robot localization using relative observations. In *IEEE international conference on robotics and automation (ICRA)*, pages 2797–2802. IEEE Press, New York, NY.
- Montes de Oca, M. A., Ferrante, E., Scheidler, A., Pinciroli, C., Birattari, M., and Dorigo, M. (2011). Majority-rule opinion dynamics with differential latency: a mechanism for self-organized collective decision-making. *Swarm Intelligence*, 5(3-4):305–327.
- Nouyan, S., Groß, R., Bonani, M., Mondada, F., and Dorigo, M. (2009). Teamwork in self-organized robot colonies. *IEEE Transactions on Evolutionary Computation*, 13(4):695–711.
- Olfati-Saber, R., Fax, J. A., and Murray, R. M. (2007). Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233.
- Ostergaard, E. H., Sukhatme, G. S., and Matari, M. J. (2001). Emergent bucket brigading: a simple mechanisms for improving performance in multi-robot constrained-space foraging tasks. In *Proceedings of the fifth international conference on Autonomous agents, AGENTS '01*, pages 29–30, New York, NY. ACM.
- Pinciroli, C., Trianni, V., O’Grady, R., Pini, G., Brutschy, A., Brambilla, M., Mathews, N., Ferrante, E., Di Caro, G., Ducatelle, F., Birattari, M., Gambardella, L. M., and Dorigo, M. (2012). ARGoS: a modular, parallel, multi-engine simulator for multi-robot systems. *Swarm Intelligence*, 6(4):271–295.
- Rekleitis, I., Dudek, G., and Milios, E. (2001). Multi-robot collaboration for robust exploration. *Annals of Mathematics and Artificial Intelligence*, 31(1-4):7–40.
- Sperati, V., Trianni, V., and Nolfi, S. (2011). Self-organised path formation in a swarm of robots. *Swarm Intelligence*, 5(2):97–119.
- Thrun, S. (2008). Simultaneous localization and mapping. Springer Tracts in Advanced Robotics, pages 13–41–41. Springer Berlin Heidelberg.
- Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic Robotics*. MIT Press, Cambridge, MA.