# Social Odometry: Imitation Based Odometry in Collective Robotics

**Álvaro Gutiérrez, Alexandre Campo, Francisco C. Santos,**
**Félix Monasterio-Huelin and Marco Dorigo**
ETSIT, Universidad Politécnica de Madrid, Madrid Spain
IRIDIA, CoDE, Université Libre de Bruxelles, Bruxelles, Belgium
aguti@etsit.upm.es

*Abstract: The improvement of odometry systems in collective robotics remains an important challenge for several applications. Social odometry is an online social dynamics which confers the robots the possibility to learn from the others. Robots neither share any movement constraint nor access to centralized information. Each robot has an estimate of its own location and an associated confidence level that decreases with distance traveled. Social odometry guides a robot to its goal by imitating estimated locations, confidence levels and actual locations of its neighbors. This simple online social form of odometry is shown to produce a self-organized collective pattern which allows a group of robots to both increase the quality of individuals' estimates and efficiently improve their collective performance.*

*Keywords: Social odometry, foraging, local communication, range and bearing.*

## 1. Introduction

The problem of exploring an unknown environment is critical to most intelligent organisms. Many robotics applications require positioning methods to carry out tasks. Several solutions to this problem have been proposed and implemented. Among these, odometry is probably the most used as it provides easy and cheap real time position information by the integration of incremental motion information over time. However, it causes an accumulation of errors during the movement of the robot which undermines the overall success of the methodology. Different approaches have been implemented to deal with odometry's errors (Chong & Klemman, 1997; Wang, 1988; Martinelli, 2002), mostly by using Kalman filters (Larsen et al., 1998; Martinelli & Siegwart, 2003). The Kalman filter estimates a vector state containing the robot position and the parameters characterizing the odometry error introduced by the designer. For instance, in (Roumeliotis & Bekey, 2002), each robot measures its relative orientation and shares the information with the rest of the group. Although the Kalman filter is an efficient recursive filter, it requires external information, introduced to the robots, that models the environment. In other implementations as in (Roumeliotis & Rekleitis, 2004), the robots are equipped with a compass which increases the efficiency of the filter, but relies on the use of an absolute orientation.

Other approaches for exploring the environment are based on map construction, which are costly in computational terms. Most of these implementations build maps incrementally by iterating localization for each new reading of each sensor on a robot. Thrun and colleagues (Thrun et al., 2000) create a map of indoor environments combining the idea of posterior estimation with incremental map construction using maximum likelihood estimators, whereas in (Sim & Dudek, 2004), the authors deal with the automatic learning of the spatial distribution given a set of images offered by all the members of the group, thanks to a centralize system which takes care of all the information.

Some applications in multirobot exploration are implemented without using odometry or dead-reckoning techniques. Some of these implementations rely on the use of static robots which are not allowed to move reducing the effectivity of the group. In (Karazume et al, 1996), a group of robots remains stationary while the other team is in motion. The moving group stops after some steps and becomes a landmark for the others that take the role of moving robots. In (Grabowski et al., 2000), only one robot is allowed to move while the others act as immobile landmarks, while in (Rekleitis et al., 2001) just one robot remains stationary when the rest of the group can navigate. These approaches either require synchronous communications between all members of the team or a central processing unit. In (Nouyan et al., 2008), a chain between two specific areas is created and the group can follow it to the goals. The robots in the chain are not able to move; consequently a direct relationship exists between the number of robots and the distance between the goals. In (Vaughan et al., 2002), each robot shares its best known path to goal based on landmarks with a central computer. The group has to be permanently in contact with the data center; therefore the environment must be properly configured before setting up the experiment.

129

While being successful in achieving their goals, these frameworks present several limitations: *i)* they are computational consuming as a result of the Kalman filters and maps, *ii)* some robots are not allowed to move while others are tracking distance between them, representing in this way a misuse of resources *iii)* robots must maintain visual contact at all times with the rest of the group, and *iv)* in some cases robots have to communicate with a central device to update or download maps, synchronize movements, or update positions.

In this work, we analyze and study a localization strategy, introduced in (Gutierrez et al, 2008a), in which robots neither share any movement constraint nor access to centralized information. This solution exploits self-organized cooperation in a group of robots to reduce each individual location error. In a nutshell, each robot's knowledge consists of an estimate of its own location and an associated confidence level that decreases with the distance traveled. This information is purely local and results from individual experience with the environment and other robots. In order to maximize its confidence level, each individual updates its estimates using the information available in its neighborhood. Hence, each individual will adopt the estimate of its immediate neighbors with a probability that increases with the confidence level difference. This adaptive dynamics, which we call social odometry, confers to each robot the possibility to *learn* from others, by *imitating* estimated locations that offer higher confidence levels. Estimated locations, confidence levels and actual locations of the robots dynamically change in order to guide each robot to its goal. This simple *online* social odometry allows the population of robots to both reduce individual's errors and efficiently reach a common objective. Moreover, to the best of our knowledge, this is the first collective navigation system in which a minimal local communication system is successfully used to promote an efficient collective performance, devoid the need of stationary robots.

The paper is organized as follows. In Section 2 we detail the dead-reckoning problem. Section 3 introduces the task used to study the social odometry, the experimental setup, and the behavior of the robots. In Section 4, we describe the experimental setup and the social odometry algorithm. Experimental results are presented in Section 5, where details of our solution and its performance are discussed. Section 6 concludes the paper and suggests future developments.

## 2. Mobile Robot Positioning

Dead reckoning is a simple mathematical procedure for determining the location of a vessel given its direction and velocity. The simplest implementation of dead reckoning in robotics is *odometry*, in which a travel path is derived from sensors computing the movement of the robot.

However, the accuracy of odometry measurements strongly depends on the kinematics of the robot (Klarer, 1988). Typical sensors for robots with a differential drive system are incremental encoders. Incremental encoders are mounted into the drive motors to count the wheel revolutions. A robot can perform odometry using simple geometry equations. Its estimated position will be related to the nature of the movement, as described by the following kinematics equations.

At each step the left and right wheel encoders show an increment of $N_l$ and $N_r$ pulses, respectively. The incremental travel distance for each wheel at each time $k$ ($\Delta U_l$ and $\Delta U_r$) may be written as:

$$\Delta U_{l,k} = c_m N_{l,k}$$

$$\Delta U_{r,k} = c_m N_{r,k} \qquad (1)$$

$$c_m = \pi D_n / n C_e$$

where $c_m$ is the conversion factor that translates encoder pulses into linear wheel displacement, $D_n$ is the nominal wheel diameter, $C_e$ is the encoder resolution and $n$ is the ratio of the reduction gear between the motor and the driver wheel.

After determining the displacement of each wheel, the movement of the centre point $c$ of the platform is obtained by computing the linear ($\Delta\rho$) and angular ($\Delta\theta$) displacement of the center as follows (see also Figure 1):

$$\Delta\rho_k = (\Delta U_{r,k} + \Delta U_{l,k})/2$$
$$\Delta\theta_k = (\Delta U_{r,k} - \Delta U_{l,k})/b \qquad (2)$$

where $b$ is the wheelbase of the robot measured as the distance between the two ideal contact points between the wheels and the floor.

Then, let the location $L$ of a robot at time *k-1* be:

$$L_{k-1} = [x_{k-1} \quad y_{k-1} \quad \theta_{k-1}]^T \qquad (3)$$

where ($x_{k-1}$ $y_{k-1}$) are the Cartesian coordinates and $\theta_{k-1}$ is the orientation with respect to a global reference frame. Thus, a rotation $\Delta\theta_k$ and a translation $\Delta\rho_k$ move the robot to a new location $L_k$:

$$L_k = L_{k-1} + \begin{bmatrix} \Delta\rho_k \cos(\theta_{k-1} + \Delta\theta_k/2) \\ \Delta\rho_k \sin(\theta_{k-1} + \Delta\theta_k/2) \\ \Delta\theta_k \end{bmatrix} \qquad (4)$$
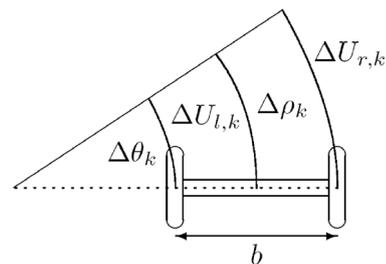


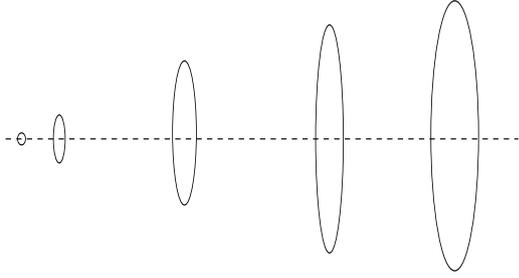Fig. 1. Path taken by the wheels of a robot during a turn.

Fig. 2. Growth of uncertainty region about localization while performing a straight line movement.

Equation 4 is not taking into account problems such as slippage, unequal wheels diameters or wheels misalignments. These errors can be classified as either *systematic* or *non-systematic errors* (Borenstein et al., 1996). *Systematic* errors can be modeled and corrected (Borenstein & Feng, 1996), while the *non-systematic* ones cannot be corrected and many classical techniques have been implemented to cope with them (Chenavier & Crowley, 1992; Chong & Kleeman, 1996).

When modeling non-systematic errors, each robot, apart from its position, also computes the distinctive *error ellipse* that represents a region of uncertainty in which the actual location lies (see Figure 2).

This region grows with the distance traveled, and it is reset to zero when the robot can localize itself exactly thanks to an environmental landmark (entering one of the two areas in our case). The error ellipse model is based on the covariance matrix ($P$) of the robot's location. We define the actual state and its covariance (Chong & Kleeman, 1996 ) as:

$$L_k = f(x_k, y_k, \theta_k, \Delta U_{r,k}, \Delta U_{l,k})$$

$$P_k = A_k P_{k-1} A_k^T + W_k Q_k W_k^T \qquad (5)$$

where $A_k$ and $W_k$ are the jacobians of $f(\cdot)$ with respect to $L_{k-1}$ and $U_k$, the latter representing the ($\Delta\rho, \Delta\theta$) vector and $Q_k$ is the covariance matrix of $U_k$. The covariance matrix $P_0$ has an initial value of 0.

## 3. The Task

We have devised a task of central place foraging (Balch, 1997) in which robots need to explore the environment to find resource sites and bring items back to a central place. Experiments take place in a bounded arena with two distinctive sites: A resource site and a central one. If one considers the typically used ants' metaphors, one can imagine the first as food and the second the nest of a colony. As a final goal, robots need to go from the central place to the resource site, back and forth.

The robots can perceive the central place and the resource site only when they are closer than a threshold distance, defined by their limited sensorial capabilities. Initially, robots are randomly scattered in the arena and they explore the environment to locate both areas. Robots rely on odometry to maintain an estimate of the location of

each area (central place and resource site). As soon as a robot comes back to an area, the corresponding location estimate is reset and odometry errors affecting this estimate are discarded.

In an ideal case, robots would make no mistake in guessing the location of the two areas. They could travel endlessly from one place to the other without drifting away. As soon as errors are introduced in the odometry system, estimated locations differ from true locations. A robot may not manage to go back to a given area, and may end up lost. In that case, the robot is doomed to explore again the environment to find the area.

To reduce the impact of odometry errors, robots keep in memory both estimated area locations and share this information among them. Being the communication range limited, a robot may update its estimates only when it encounters another robot.

## 4. Methods

### 4.1. Experimental Setup

Our experiments are carried out using simulation software based on ODE3. The arena is a bounded square area of LxL m² (3x3m², 3.5x3.5m², 4x4m² or 4.5x4.5m²) and the robots are randomly distributed in a specific area of the arena at the beginning of the experiment. The ground of the arena is white except for the two goal areas: central place is a black circle of 20 cm radius and the resource site is a grey circle of the same dimensions.

Robots are modeled as a cylindrical body of *7 cm.* diameter. Motion is achieved by a differential drive system composed of two wheels with encoder sensors. Robots can perceive obstacles thanks to 8 infrared sensors distributed around the body. Robots differentiate areas using an infrared sensor directed to the ground. A specific simulated range and bearing communication board (Gutierrez et al, 2008b) allows robots to send messages to each other when their interdistance is less than 25 cm. Additionally, a robot receiving a message has also knowledge about the relative distance of the emitter (*range*) and its relative direction (*bearing*).

We introduce errors to simulate the imperfect response of the range and bearing sensor. Noise is added to the bearing (±20º) and range (±2.5 cm) values. Moreover, each message emitted can be lost with a probability that varies linearly from 1% when the sender-receiver distance is less than 1 cm, to 50% when the two robots are at 25 cm from each other. $Q$ defines the increment of the movement as follows:

$$Q = \begin{bmatrix} \sigma_{\Delta\rho}^2 & 0 \\ 0 & \sigma_{\Delta\theta}^2 \end{bmatrix} \qquad (5)$$

where $k_l$ and $k_r$ represent the nondeterministic parameters of the motor drive and the wheel-floor interaction. These values are randomly chosen from a uniform distribution, being limited to ±20% of the maximum movement at each time step.
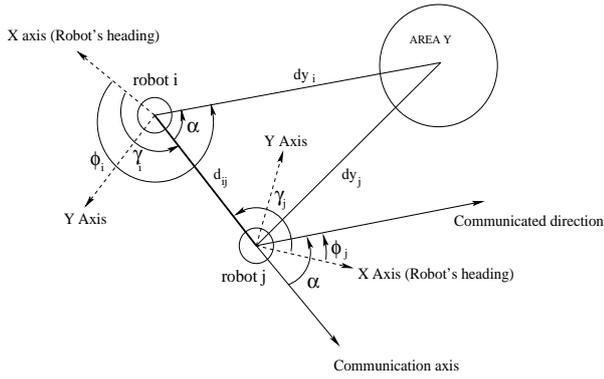
Fig. 3. Robots sharing information about the estimated location of area Y.

## 4.2. Learning from others

The estimated location of the last visited area is more accurate than the previous one since odometry errors increase with the distance traveled. While moving from one area to the other, if a robot encounters another robot, it transmits its estimated location of the last visited area. Hence, each robot has the opportunity to adopt the estimates of other robots present in their neighborhood.

Since robots do not share a global coordinates system, they rely on the range and bearing of each other to communicate locations. Figure 3 shows how information about the estimated location of area $Y$ is transmitted from robot $i$ to robot $j$.

In a first step, robot $i$ transmits its estimate of the distance $dy_i$ and direction $\phi_i$ of area $Y$ to robot $j$. For the direction, the value transmitted is the angle $\alpha$, obtained from $\phi_i$ using the communication beam as reference axis: $\alpha = \phi_i - \gamma_i$. In a second step, robot $j$ transforms the received data into its own coordinates system. First, it calculates the direction pointed by robot $i$ as $\phi_j = \gamma_j + \alpha - \pi$, followed by the calculation of the location $loc_j = (x, y)$ of area $Y$. This computation takes into account the transformation to its own reference frame and integrates robot $i$ information in the following way:

$$x = d_{ij} \cdot cos\ \gamma_j + dy_i \cdot cos \phi_j$$
$$y = d_{ij} \cdot sin\ \gamma_j + dy_i \cdot sin \phi_j \qquad (5)$$

After receiving a communication, robot $j$ has two different locations $loc_i$ and $loc_j$ at hand. At this stage, robot $j$ has the opportunity to adopt (or not) the estimate of the neighbor depending on its neighbor confidence level. Given that estimates get worse with distance traveled, the robots use the inverse of the distance traveled as a confidence level of their guess location. This confidence level, denoted by $\varepsilon_i$ for robot $i$, respectively $\varepsilon_j$ for robot $j$, is part of any communicated location and informs about the reliability and quality of the information. We consider that, in order to produce a new guess location ($loc\_up_j$) each robot takes into account all information available, but weights its sources in a different way. To calculate $loc\_up_j$, we adopt the so-called pairwise comparison rule (Traulsen et al., 2006; Traulsen et al., 2007; Santos et al.,
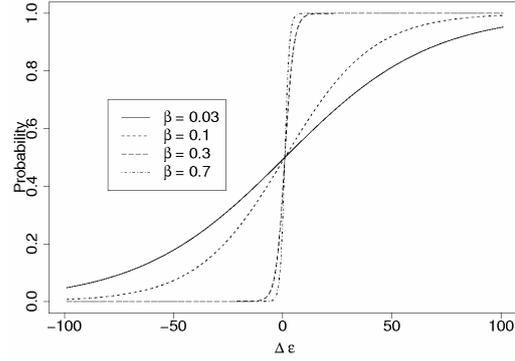


Fig. 4. The Fermi function which allows robots to decide between their own estimate and the information provided by the others.

2006) often adopted in evolutionary/social dynamics studies, to code the social learning dynamics, which makes use of the Fermi distribution (see also Figure 4):

$$c = \frac{1}{1 + e^{-\beta(\varepsilon_i - \varepsilon_j)}}, \qquad (8)$$

where $\beta$ measures the importance of the relative confidence levels in the decision making. For low values of $\beta$, the decision making proceeds by neglecting the confidence levels whereas for high value of $\beta$, we obtain a *pure* imitation dynamics commonly used in cultural evolution (Hammerstein, 2003) defined by a sharp step function. In the first case, the confidence level works as a small perturbation to a simple average between the two estimates, while in the latter, each robot is ready to completely neglect the estimate which has smaller relative confidence.

Hence, we use a weighted average to obtain the new location ($loc\_up_j$) and confidence level (e_up$_j$) using the Fermi function as shown in Equation 9.

$$loc\_up_j = c \cdot loc_i + (1 - c) \cdot loc_j$$
$$\varepsilon\_up_j = c \cdot \varepsilon_i + (1 - c) \cdot \varepsilon_j \qquad (9)$$

## 5. Results

In the following, we report results of simulated experiments with 30 robots where the outcomes with and without imitation dynamics among robots are compared. Each situation tested was repeated 30 times. The performance of the robots in the foraging task under study is measured as the number of total round trips completed from the central place to the resource site and back during one simulated hour.

We implement 4 different scenarios (see Figure 5), where robots are located in a random initial position and orientation inside an area of 50 cm radius and targets have an area of 20 cm radius. For each configuration environment we modified i) the surface of the arena and ii) the number of robots involved in the experiment from 1 to 30 robots set.
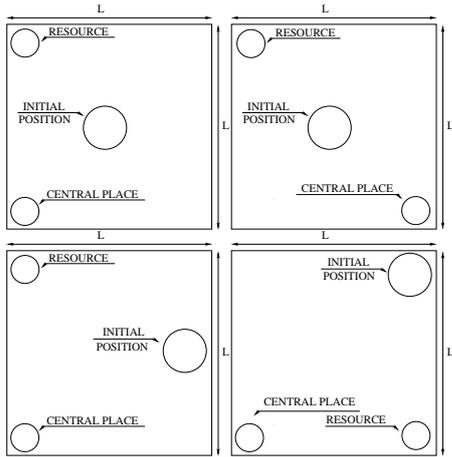
Fig. 5. Configuration of an arena of L² meters area, for the 4 different scenarios.
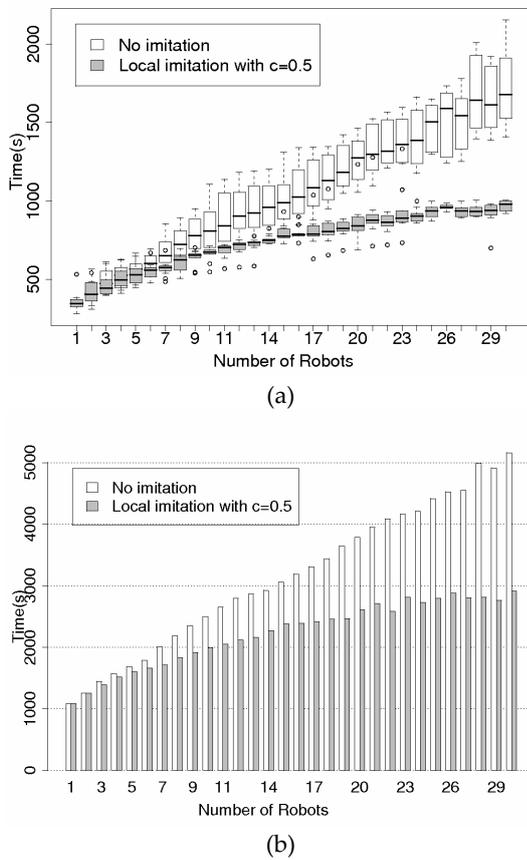


(a)



(b)

Fig. 6. (a) Comparison of the time employed for a group of 30 robots to locate both goal areas. Experiment is set up for a non-imitation dynamics set-up (blank) and imitation dynamics one with c=0.5 value (grey) algorithm in a 4x4m2 arena. Each box comprises observations ranging from the first to the third quartile. The median is indicated by a horizontal bar, dividing the box into the upper and lower part. The whiskers extend to the farthest data points that are within 1.5 times the interquartile range. Outliers are shown as dots. (b) Comparison of the average time employed for a group of 30 robots in 4 different arenas size and 4 scenarios. We compare the number of times robots make a complete travel from one goal to the other with different imitation choices.

We first study the time it takes for all the robots to locate both goals areas. Figure 6a shows this time for a set of 30 robots in a experiment of 4x4 m² arena with imitation based on c = 0.5 (see equation 8). A comparison between a non-imitation dynamics set-up and an imitation dynamics one with a fixed c = 0.5 value is shown in figure 6b as an average of experiments in the 4 different size areas and 4 different scenarios. With few robots, the two behaviors perform equally well, which is explained by the infrequent encounters of the robots and the consequent low amount of communications. As the number of robots in the experiment increases, we clearly observe that the use of imitation allows the robots to find the areas faster. With imitation dynamics, robots are intrinsically carrying out a recruitment process which speeds up the initial exploration phase, getting a 40% reduction of time for a group of 30 robots.

Later we focus on the analysis of the algorithm's performance related to the non movement-error implementation. We carry out a comparison of different behaviors:

- **No imitation**: robots do not communicate and are affected by odometry errors.
- **No odometry error**: robots imitate but are not affected by odometry errors.
- **Covariance knowledge**: robots communicate and update new location using their own covariance matrix value and the one offered by their neighbors.
- **Global imitation**: robots learn from all robots present in the arena. Each robot updates its estimate by averaging the knowledge of the whole group.
- **Local imitation**: robots communicate and learn with each other locally; that is, the social odometry behavior detailed in section 4.

Figure 7 shows the improvement of social odometry for a set of 30 robots in 4 different arenas size in one hour experiment. Without imitation, robots rely solely on error prone odometry to find the areas. Once lost, they have to explore the environment and find the areas by chance, which explains the poor performance. In the no odometry error case, robots know accurately the location of both areas. In any of the four scenarios the best performing β for local imitation is statistically improving the results compared to the global imitation. This is because by using a global knowledge, robots take at each time step the estimate of all their partners, the ones with reliable information and the ones that are already lost. Interestingly, the covariance knowledge behavior does not show statistically different performance with social odometry using the best performing β.

Nevertheless analyzing how the area of the arena and the distance between the goals decreases the efficiency of the robot, we can notice a relationship between β and *d* (distance between the centre of the two goal areas).

Worst situation for the social odometry implementation is when a robot travels from the resource site to the central place and comes back without finding any other communicative robot till the end. In this case the
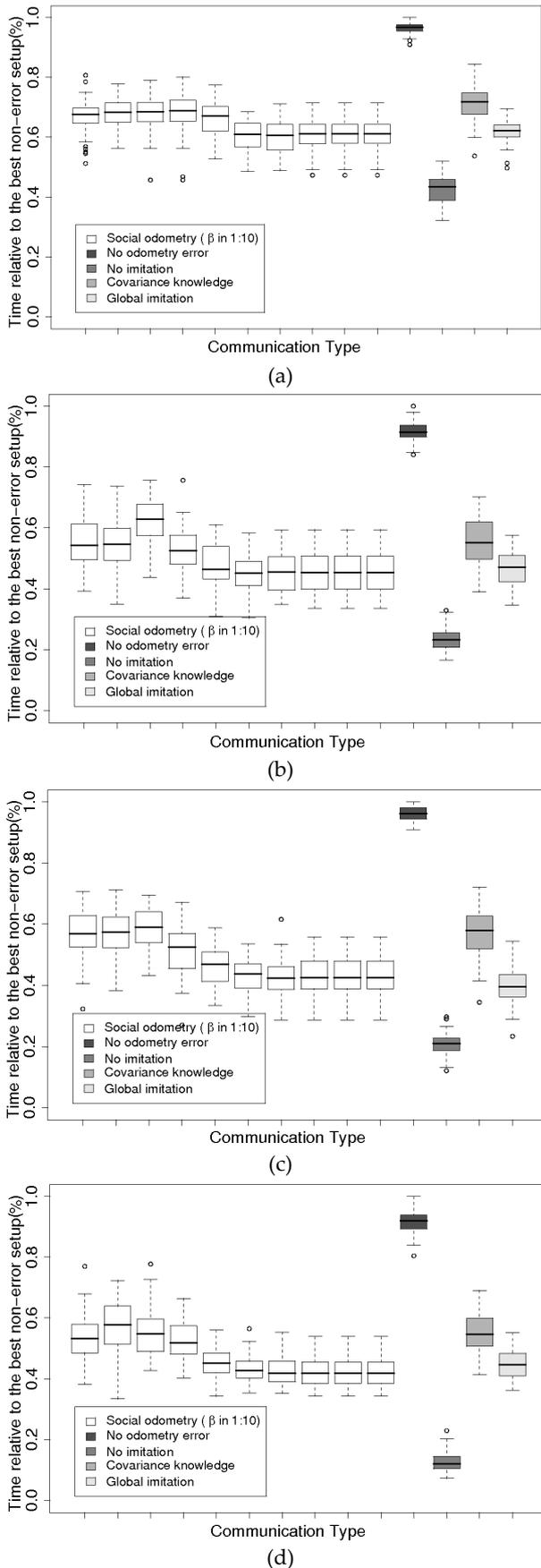
Fig. 7. Task performance for the different imitation choices with a set of 30 robots in an arena of (a) 3x3 m², (b) 3.5x3.5 m², (c) 4x4 m² and (d) 4.5x4.5 m².

confidence level difference is $|\Delta\varepsilon| = 2d$ and the best performing $\beta$ creates a situation in which $c = 0.6$. This suggests robots can set up the optimal $\beta$ knowing the distance between both goals. Robots could tune $\beta$ while they are performing the task.

Finally, we show in Figure 8 how the number of robots modifies the efficiency $\eta$ (related to the non movement-error implementation) in a 3x3 m² arena with the better $\beta$ performance. We observe a maximum $\eta$ for a set of 11 robots going slightly down for higher number of robots. This is due to the amount of agents in the same path that disrupt communication between robots and makes robot navigation slow. This disruption is strongly dependant of the arena size. Results suggest that a high density of robots disrupts performance and there is most likely an optimal density to carry out the foraging task as reported in (Goldberg & Matarić, 1997). We also see that social odometry allows the robots to cope with density to some extent and have performance that scales linearly in a wide range of situations. Nevertheless the improvement with respect to the non-imitation dynamics situation doubles its performance for all the experiments.

## 6. Conclusions

In this paper we have described a social strategy in which robots use pairwise local communication to share knowledge about specific locations to improve their performance in a foraging task. By letting the robots use the estimates of others, we engineer an efficient and decentralized knowledge sharing mechanism which allows the robots to achieve their goals, both from an individual and group perspective. This simple mechanism drives the system to a successful collective pattern that none of the individuals is able to achieve independently.

Moreover, we show that local imitation is more effective than global imitation which would additionally require either more expensive devices or a central system taking
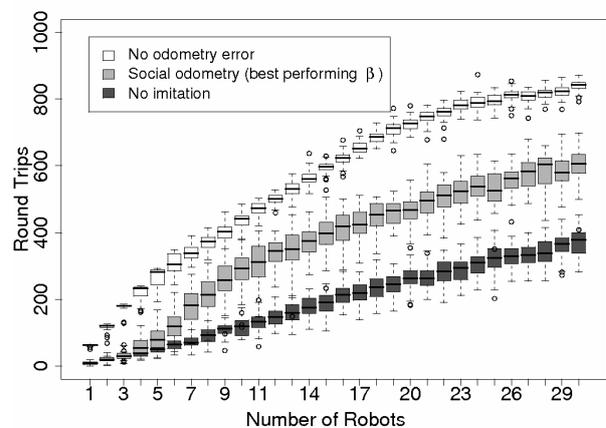


Fig. 8. Efficiency related to the non movement-error implementation (white) for a communication based algorithm with the best performing $\beta$ (grey) and a non communication implementation (black).

care of routing robots' communications. Furthermore, decentralized systems are often associated with better scalability of the results with the number of robots.

We also compare social odometry with a fusion mechanism based on the covariance matrix, in which robots have an innate knowledge of the non-systematic errors model. Social odometry does not rely on any internal model provided to each robot, but exclusively on the self-organized nature of collective dynamics (Buchanan, 2007) and knowledge sharing. This has obvious advantages, given that less effort is needed for tuning parameters that depend on the environment. Additionally, we observe that social odometry implements an implicit recruitment process that speeds up the initial exploration phase mandatory to achieve the foraging task. Lastly, a study of the impact of robots' density on the performance indicates that social odometry is not drastically affected by overcrowding and can cope with it.

Preliminary observations suggest that the β parameter depends on the distance between the areas, suggesting that the robots can tune parameter β knowing the distance between the central place and the resource site. The tuning of the parameter can be done off-line, where the designer introduces the value, or on-line, where robots update the β parameter once they have located both areas.

In the future, we intend to implement and test this strategy on real robots, emphasizing the online tuning of the parameter β.

Finally, we would like to stress that the performance of the social odometry allows an optimistic forecast concerning the use of *online* self-organized methodologies in the field of collective robotics. Online methodologies, from communication to evolution, may represent the missing aspect necessary for truly adaptive populations of robots. Work along these general lines is in progress.

## 7. Acknowledgments

## 8. References

Balch, T. (1997). Reward and Diversity in Multirobot Foraging, Proceedings of the IJCAI-99 Workshop on Agents Learning About, From and With other Agents, pp. 15-21, ISBN 1-558-60613-0, Stockholm, Sweden, August 1999, Morgann Kaufman, San Francisco, CA.

Borenstein, J.; Everett, H.R. & Feng L. (1996). *Navigating Mobile Robots: Sensors and Techniques,* A.K. Peters, Ltd., ISBN 1-568-81058-X, Wellesley, MA.

Borenstein, J. & Feng, L. (1996). Measurement and Correction of Systematic Odometry Errors in Mobile Robots. *IEEE Transactions on Robotics and Automation*, Vol. 12, No. 6, (December 1996), pp. 869-880, ISSN 1042-296X.

Buchanan, M. (2007) *The Social Atom*, Bloomsbury, ISBN 1-596-91013-5, New York, NY.

Chenavier, F. & Crowley, J. L. (1992). Position Estimation for a Mobile Robot Using Vision and Odometry, Proceedings of the 1992 IEEE International Conference on Robotics and Automation, pp. 2588-2593, ISBN 0-818-62722-0, Nice, France, May 1992, IEEE Press, Piscataway, NJ.

Chong, K.S. & Kleeman, L. (1997). Accurate Odometry and Error Modeling for a Mobile Robot, Proceedings of the 1997 IEEE International Conference on Robotics and Automation, pp. 2783 - 2788 , ISBN 0-780-33612-7, Albuquerque, New Mexico, April 1997, IEEE Press, Piscataway, NJ.

Goldberg, D. & Matarić, M.J. (1997). Interference as a Tool for Designing and Evaluating Multi-Robot Controllers, Proceedings of the Fourteenth National Conference on Artificial Intelligence, pp. 637-642, ISBN 0-262-51095-2, Providence, Rhode Island, July 1997, MIT Press, Cambridge, MA.

Grabowski, R.; Navarro-Serment, L.; Paredis, C. & Khosla, P. (2000). Heterogeneous Teams of Modular Robots for Mapping and Exploration. *Autonomous Robots*, Vol. 8, No. 2, (January 2000), pp. 293–308, ISSN 0929-5593.

Gutiérrez, A.; Campo, A.; Santos, F. C.; Pinciroli, C. & Dorigo, M. (2008a). Social Odometry in Populations of Autonomous Robots, Proceedings of Ant Colony Optimization and Swarm Intelligence, 5th International Workshop, ANTS 2008 (LNCS 5217), pp. 371-378, ISBN 3-540-87526-0, Brussels, Belgium, September 2008, Springer-Verlag, Berlin, Germany.

Gutiérrez, A.; Campo, A.; Dorigo, M.; Amor, D.; Magdalena, L. & Monasterio-Huelin, F. (2008b) An Open Localisation and Local Communication Embodied Sensor, Tech. Rep. 2008-019, IRIDIA, Institut de Recherches Interdisciplinaires et de D´eveloppements en Intelligence Artificielle, 2008.

Hammerstein, P. (2003). *Genetic and Cultural Evolution of Cooperation*, MIT Press, ISBN 0-262-08326-4, Cambridge, MA.

Karazume, R.; Hirose, S.; Nagata, S. & Sashida, N. (1996). Study on Cooperative Positioning System (Basic Principle and Measurement Experiment), Proceedings of the 1996 IEEE International Conference on Robotics and Automation, pp. 1421-1426, ISBN 0-780-32988-4, Minneapolis, Minnesota, April 1996, IEEE Press, Piscataway, NJ.

Klarer, P.R. (1988). Simple 2-d Navigation for Wheeled Vehicles. Technical Report, Sandia Report SAND88-0540, Sandia National Laboratories, Livermore, CA.

Larsen, T.D.; Bak, M.; Andersen, N.A. & Ravn, O. (1998) Location Estimation for Autonomously Guided Vehicle Using an Augmented Kalman Filter to Autocalibrate the Odometry, Proceeding of the FUSION 98 Spie Conference, pp. 138-152 , ISBN 1-892-51200-9 , Las Vegas, NV, July 1998, CSREA Press, Sunnyvale, CA.

Martinelli, A. (2002). The Odometry Error of a Mobile Robot With a Synchronous Drive System. *IEEE Transactions on Robotics and Automation,* Vol. 18, No. 3, (June 2002), pp. 399-405, ISSN 1042-296X

Martinelli, A. & Siegwart, R. (2003). Estimating the Odometry Error of a Mobile Robot During Navigation, Proceedings of the first European Conference on Mobile Robots, pp. 218-223, ISBN 8-889-17718-7, Radziejowice, Poland, September 2003, Zturek Research Scientific Inst. Warszawa, Poland.

Nouyan, S.; Campo, A.; & Dorigo, M. (2008). Path Formation in a Robot Swarm: Self-Organized Strategies to Find Your Way Home. *Swarm Intelligence,* Vol. 2, No. 1, (March 2008), pp. 1-23, ISSN 1935-3812.

Rekleitis, I.M.; Dudek, G. & Milios, E. (2001). Multi-Robot Collaboration for Robust Exploration. *Annals of Mathematics and Artificial Intelligence,* Vol. 31, No. 1, (January 2001), pp. 7-40, ISSN 1012-2443.

Roumeliotis, S.I, & Bekey, G.A. (2002). Distributed Multirobot Localization. *IEEE Transactions on Robotics and Automation,* Vol. 18, No. 5, (October 2002), pp. 781-794, ISSN 1042-296X

Roumeliotis, S.I. & Rekleitis, I.M. (2004). Propagation of Uncertainty in Cooperative Multirobot Localization: Analysis and Experimental Results. *Autonomous Robots,* Vol. 17, No.1, (July 2004), pp. 41-54, ISSN 0929-5593.

Santos, F.C.; Pacheco, J.M. & Lenaerts, T. (2006). Cooperation prevails when individuals adjust their social ties. *PLoS Computational Biology,* Vol. 2, No. 10, (October 2006), pp. e140.

Sim, R. & Dudek, G. (2004). Self-Organizing Visual Maps, Proceedings of the National Conference on Artificial Intelligence (AAAI), pp. 470-475 , ISBN 0-262-51183-5, San Jose, California, July 2004, MIT Press, Cambridge, MA.

Thrun, S.; Burgard, W. & Fox D. (2000). A Real-Time Algorithm for Mobile Robot Mapping with Applications to Multi-Robot and 3D Mapping, Proceedings of the 2000 IEEE International Conference on Robotics and Automation, pp. 321-328, ISBN 0-780-35886-4, San Francisco, California, April 2000, IEEE Press, Piscataway, NJ.

Traulsen, A.; Nowak, M.A. & Pacheco, J.M. (2006). Stochastic Dynamics of Invasion and Fixation. *Physical Review – Series E -,* Vol. 74, No. 1, (July 2006), pp. 011909.1-011909.5, ISSN 1539-3755.

Traulsen, A.; Pacheco, J.M. & Nowak, M.A. (2007). Pariwise Comparison and Selection Temperature in Evolutionary Game Dynamics. *Journal of Theoretical Biology,* Vol. 246, No. 3, (June 2007), pp. 522-529, ISSN 0022-5193.

Vaughan, R.; Stoy, K.; Sukhatme, G. & Matarić, M. (2002). LOST: Localization-Space Trails for Robot Teams. *IEEE Transactions on Robotics and Automation,* Vol. 18, Vol. 5, (October 2002), pp. 796–812, ISSN 1042-296X.

Wang, C.M. (1988), Localization Estimation and Uncertainty Analysis for Mobile, Proceeding of the International Conference on Robotics and Automation, pp 1231-1235, ISBN 0-780-34300-X, Leuven, Belgium, May 1998, IEEE Press, Piscataway, NJ.