



A Multiple Robot Map Building Strategy Using Inverse Ant System-Based Surveillance System

Janderson Rodrigo de Oliveira¹, Rodrigo Calvo^{2*} and Roseli AF Romero³

¹Federal University of Lavras, Brazil

²State University of Maringa, Brazil

³Department of Computer Science, University of Sao Paulo, Brazil

Abstract

The multiple robot map building strategies have several advantages when compared to strategies based on a single robot, in terms of flexibility, gain of information and reduction of the map building time. In this paper, a local map integration method is proposed based on the inter-robot observations, considering a new method for the environment exploration. This exploration method is based on the Inverse Ant System-Based Surveillance System strategy (IAS-SS), a bioinspired coordination strategy, which minimizes the environment exploration time due to the pheromone repulsive nature, impelling the robots for visiting unexplored regions. The main contribution of this work is the development of a distributed mapping method using IAS-SS. Simulation results show the map integration method is efficient, through several trials performed considering a variable number of robots and environments with different structures. Results obtained from experiments confirm that the integration process is effective and suitable to execute mapping during the exploration task.

Index Terms

Multi-robot systems, Local map integration, Ant colony optimization

Introduction

The multiple robot mapping [1-4], has attracted attention because the parallel exploration effectiveness using multiple robots and the reduction of time for large areas mapping. According to [5], multiple robot systems are more flexible and efficient than one single robot systems. Moreover, multiple robot systems allow a higher gain of information and a reduction of the map building time. Different aspects are investigated in multiple agent systems, such as: Agent communication and information merging [6-8].

Another important aspect in multiple agent systems is the coordination problem [9-11], which is considered complex. Coordination strategies based only on mathematical formulation and on agent and environment

models are parameter dependent and can suffer critical degradation due to agent failures [12-14]. Bio-inspired theories provide fundamentals to design alternative strategies for overcoming the main difficulties that become traditional strategies vain [15,16].

The main challenge of multiple robot mapping tasks is the integration process of the maps produced individually by each robot, called here local maps. The most of proposed methods in literature deal with this problem taking the center of each robot and performing the integration according to relative positions of the robots which, consequently, need to be known.

The integration problem becomes challenging when the coordinate transformation between the initial poses of the robots is unknown. This transformation is called *initial correspondence*. A method for merging independently created maps with unknown initial correspondence is proposed in this paper. The integration of local maps in a global map might be obtained by two different

***Corresponding author:** Rodrigo Calvo, State University of Maringa, Maringa, PR, Brazil

Accepted: August 16, 2018; **Published:** August 18, 2018

Copyright: © 2018 de Oliveira JR, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Citation: de Oliveira JR, Calvo R, Romero RAF (2018) A Multiple Robot Map Building Strategy Using Inverse Ant System-Based Surveillance System. Int J Robot Eng 3:008

ways. The first way is the search of landmarks alignments between two maps [17-19]. In this case, the most likely transformation is that which produces the higher correspondence between the landmarks. As a fundamental presupposition, the transformation based on landmarks has the existence of overlapping areas between the local maps.

The second way is the use of inter-robot observations, referred as rendez-vous measures, for computing the coordinate transformation [8,12,20]. When two robots meet and compute their relative positions, this information can be used to define the transformation required for integrating their maps. It is worth to note that the method based on landmarks alignments can be very expensive in terms of computer resources, mainly if the integration is not successful.

In the present work, a local map integration method is proposed based on inter-robot observations. The method integrates local maps using the information sharing model proposed by Tan, et al. [12]. The local map merging is defined by transformation matrices, represented by the distance between two robots and their relative positions. The method proposed by Tan, et al. [12] is not responsible for mapping the environment. It does not build a map that represents the environment. The sensing and data fusion for each robot carry out with respect only to its local coordinate frame.

Considering only the mapping task, each robot maps the environment using the occupancy grid strategy [21-24]. This approach uses a reticulated and probabilistic representation of information for modeling unknown environments according to the laser range-finder readings. It is defined as a multidimensional random field that contains stochastic estimates of the cell states, such as, *Occupied*, *Empty* or *Unknown*, in the reticulated space. Each robot builds its own map, called local map, and it moves independently of each other. The proposed integration method merges these local maps in a global representation that contains occupation estimates of the entire environment.

The exploration and coordination problems of multiple robots are designed in a coordination strategy described in [25-27], named Inverse Ant System-Based Surveillance System (IAS-SS). The IAS-SS strategy is inspired on biological mechanisms that define the social organization of swarm systems. Specifically, it is based on a modified version of the known ant colony algorithm. Ants and other insects use chemicals called pheromones for several communication and coordination tasks. Techniques based on ants' behavior have been used as models for solving patrolling and area coverage problems [28-32].

In IAS-SS strategy, the agents are able to indirect communication as the biological agents are, but their reaction to the pheromone is distinct since the direction adjustments are defined to guide preferably the robots to areas in which there is low amount of pheromone. It is important to emphasize that in our work the pheromone is just a way of monitoring the environment. The robots do not spray chemicals on the environment, the pheromone is just a virtual characteristic and it is better explained in Section IV-B. IAS-SS strategy is primarily projected for the coordination of multiple robots applied to surveillance and exploration tasks. Some characteristics of IAS-SS are: decentralized, on-line, and parameter independent from both the number of robots and the environment model.

Moreover, the coordination strategy based on ants' behavior is just a type of navigation algorithm. The proposed integration method can be generalized to other navigation algorithms, since they use a map representation based on coordinate systems. Finally, it is worth to highlight the proposed integration method merges actually two kinds of information: the estimates of the cell states of the occupancy grid obtained for each robot and the amount of pheromone perceived by them.

We have chosen Inverse Ant System-Based Surveillance System [25-27], also for its obstacle avoidance behavior, obtained as a consequence of the pheromone repulsive nature. As long as robots are close to an obstacle, the amount of pheromone in its boundary region is increased. Then, the robots avoid the obstacles due to the high concentration of pheromone. Therefore, the trajectories generated do not guide the robots to a collision situation. Besides the exploration and surveillance tasks, the robots are able to avoid collisions, keeping a reasonable distance from obstacles and among them.

The remainder of the paper is organized such as it follows. The related works are presented in Section 2. In Section 3, it is provided a brief description of the multiple robot system for exploration and surveillance tasks and the coordination strategy IAS-SS. The map integration method based on interrobot observations is proposed in Section 4. In Section 5, it is shown simulation results obtained from a set of experiments. A brief discussion about the obtained results is presented in Section 6. Finally, in Section 7, are presented the conclusion and the future works.

Related Works

The local mapping strategy adopted in our approach is based on the occupancy grid method [21-24]. The occupancy grid is defined as a multidimensional random field containing stochastic estimate of the cell states in the reticulated space. These estimates are computed

through interpretation of distance readings utilizing sensor probabilistic models.

Procedures of Bayesian estimates incremental updating of the occupancy grid through sensor readings captured by several sensors on the multiple views. The state variable $s(C)$, associated to cell C in the occupancy grid can be defined as a discrete random variable having two possible values, occupied and empty, denoted by OCC and EMP, respectively. In this way, an occupancy grid corresponds to a random field of discrete and binary states. Since the cell states are exclusive mutually then

$$P_o[s(C) = OCC] + P_o[s(C) = EMP] = 1$$

In the occupancy grid, it is used a stochastic model of sensor defined by a probabilistic density function, $p(r | z)$, which relates the input r with the space parameter z , correspondent to a position in a map. This density function is subsequently used in a procedure based on Bayesian estimate to determine the probabilities of a cell in a two-dimensional grid. Through a recursive formulation, the previous estimate of the state of cell, $P_o[s(C_i) = OCC | \{r\}]$, serves as *priori* estimate and its value is obtained directly from the grid. The new estimate is then stored in the map. In this way, a map is obtained that represents the probabilities of occupation of each cell.

An approach to matching occupancy grid maps by means of finding correspondences between a set of sparse features detected in local maps was presented in [33]. To cope with the uncertainty and ambiguity arising from matching grid maps, a method to search for a dynamic number of internally consistent subsets of feature pairings was defined, which computes hypotheses about the translation and rotation between the maps. The transformation between a pair of grid maps was defined by registering the corresponding map images, the gray scale images resulting from interpreting grid cells as pixels and occupancy probabilities as gray levels. Instead of using grid maps alone, this approach used a dual representation of local maps where both occupancy grids and point maps are maintained, since the maps complement each other, and their maintenance only require updating both maps simultaneously with the same sensory data. A similar idea is adopted in our approach. The integration method proposed also used a dual representation of local maps. However, this representation is based on occupancy probabilities and on pheromone concentrations. Therefore, our approach updates simultaneously both maps with two kinds of sensory data and both maps are simultaneously integrated when two robots find each other.

An occupancy grid map merging method based on intensity based and area-based key-point selection and matching was proposed in [34]. The proposed method is

capable of merging local maps that have a limited degree of overlapping regions, increasing the efficiency of the system. The maps are combined without knowing the initial position and orientation of the robots.

It is worth to notice that these alignments based on overlapping regions can be very expensive in terms of computer resources, mainly if the integration is not successful. So, an approach based on inter-robot observations seems more efficient, considering a real-time scenario.

In the method elucidated in [35], topological maps were built from local metric maps based on occupancy grids. On this approach, the nodes of topological maps graph represent local metric maps and each edge describes the relative positions of adjacent local maps. The integration process is done by adding an edge that connects two topological maps built by two different robots and estimating the relative positions of robots by optimizing this edge.

Another works merge landmarks-based maps, instead occupancy grids-based maps. Although landmarks-based map merging methods for multi-robot systems have demonstrated some success, according to Topal, et al. [34], these approaches remain unfeasible for unstructured and complex work environments, since feature extraction is very hard to perform in such unstructured areas and prior information does not exist. Hence, some researchers have focused on occupancy grid-based mapping methods for modeling the environment and merging the local maps.

The technique described in [36] proposed an approach in which all robots belonging to exploration system are located in a communication network. The communication network has an unlimited range, enabling a continuous transference of mapping information done by one robot to all the others inside communication network. Since our method is based on inter-robots observations, each robot has a limited range of communication, making our approach more realistic.

The approach described in [3] applied the method Extended Kalman Filter (EKF), originally developed to a single robot, in a multiple robot system. The observations obtained for each robot are sent to all the others, so a global representation of the environment is obtained. Differently of this method, our approach is based on observations inter-robots, in which the exchange of information is only carried out between pairs of robots what decreases the involved computational cost. Moreover, the exchange of information among all robots is a factor to be avoided, due to the high computational cost necessary for accomplishing this task.

A map building approach based on Particle Swarm

Optimization (PSO) algorithm and Hilbert curve was presented in [7]. A grid method is used to model the environment, which is divided into a number of sub areas. PSO algorithm is adopted to select optimal sub area for each robot and Hilbert curve exploring method to explore in each sub area. The division of the environment into sub areas is an interesting characteristic because it allows a performance analysis based on percentage of completed exploration. Therefore, a similar idea is used in our performance analysis. Our analysis also considers the division of the environments into sub areas, called here sectors, and the percentage of completed exploration is calculated for evaluation.

Since the proposed integration method is based on the approach elucidated in [12], a brief description of this approach is presented. The approach defined by Tan, et al. [12] shares environment information through an ad hoc wireless network, considering the coverage problem of the environment. The approach consists of a variety of sensors on a collection of mobile robots, which presents the cooperation and formation control of a mobile sensor network, it is worth to notice that in this model, the information shared is only about the status of the environment, since it does not build a map that represents the environment. For instance, if a robot detects an object in its coverage area, it will share this information only with the robots belonging to its neighboring. This model defines the geographical relationship of different robots by using Voronoi diagram and Delaunay triangulation. The relationship between neighboring robots is defined by Delaunay tessellation and the coverage area of each robot is defined by Voronoi diagram. The configuration of a robot is defined in a local inertial coordinate system and each robot has capabilities of sensing, computation and communication.

The sharing of meaningful information among the robots is defined by the relation between the local coordinate systems of two neighboring robots. In order to share

this information between robots, a transformation matrix is computed considering a particular pair of robots. The information sharing process is performed continuously whereas the robots move through environment. However, during this process, the coordinate system of each robot is permanently centered in current position of each one of them. This constraint is adverse to the mapping task, since the generated maps in this scenario should be recalculated at each step of the robot. So in our proposal the robot coordinate system does not need to be centered on own robot during the entire exploration stage. Actually, the robot coordinate system is centered on its initial position, allowing us to build incrementally its local map.

In next section, the IAS-SS strategy will be described, necessary to a better understanding of the coordination strategy for environment exploration.

IAS-SS

As the robot navigate, they deposit a specific substance, named pheromone (the analogue of the pheromone in biological systems), on the environment. At every time, each robot receives pheromone stimuli from the environment and adjusts its navigation direction. The lesser is the detected amount of the substance detected in a direction, the greater is the probability that the robot takes this navigation direction. Considering each robot has same role of an ant in the traditional ant algorithm, the IAS-SS strategy is the opposite of that one adopted in the sense that the ants choose the way in which there is a greater amount of pheromone.

A description of IAS-SS strategy is given to follow. Consider a group of N robots k , $k \in \{1, \dots, N\}$. Each robot k performs two operations: Steering direction adjustment and pheromone deposition. We are supposing the sensor detects pheromone stimuli at a specific distance R_D , as shown in Figure 1, from -90 degrees to 90 degrees, corresponding to the average of the amount of phero-

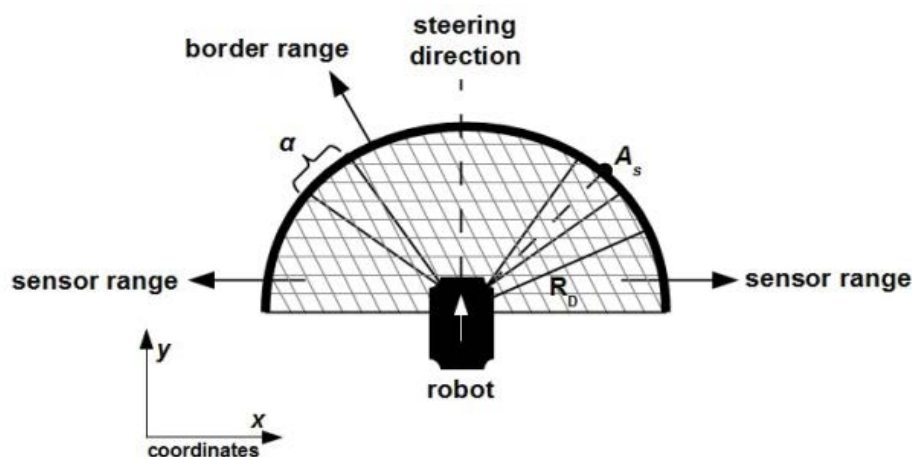


Figure 1: Robot and sensor models.

more deposited in an angle interval. The total range of 180 degrees is divided in identical angle intervals, such that the sensor detects stimuli corresponding to different angles A_s , such that: $(2S+1)\alpha = 180$ where $s \in [-S, S]$ and $s \in \mathbb{N}$.

Two subsets of angle intervals S are considered to define the steering direction. The first one, subset U , the angle intervals are those whose amount of pheromone is very low. The second, subset V , consists of elements chosen randomly, according to a uniform distribution, from the angles A_s that are not in the first subset. The subset V is built for ensuring the representativeness of the angle intervals.

The rules for building the subsets U and V are such as it follows:

- Subset U

if $A_s \in U$ and $A_z \notin U$, then $\tau_s \leq \tau_z$

- Subset V

if $A_s \in V$, then $A_s \notin U$ and A_s are chosen randomly

where $s, z \in \{-S, \dots, -1, 0, 1, \dots, S\}$; τ_s and τ_z are the amounts of the pheromone corresponding the angles A_s and A_z , respectively.

A probability value is assigned to each discrete angle in both subsets U and V . The probability assigned to the angle A_s is inversely proportional to the amount of pheromone deposited in the respective angle interval. Specifically, the probability $P(s)$ assigned to the angle A_s is given by:

$$p(s) = \frac{1 - T_s}{\sum_{i \in \{s \mid A_s \in \{U \cup V\}\}} (1 - \tau_s)} \quad (1)$$

where τ_s is the amount of the pheromone corresponding the angle A_s .

The adjusting of steering direction is determined according to a discrete random variable defined through the probability $P(s)$, assuming values in the set $A_s, s \in \{1, \dots, S\}$. The robots tend to move to the directions where there is low amount of pheromone. In this way, the robots move to unexplored areas or areas scarcely visited by robots during some period of time. The adjusting of steering direction is given by:

$$\theta_k(t) = \theta_k(t-1) + \gamma A_s^* \quad (2)$$

where $\theta_k(t)$ is the steering of movement of robot k at instant t , $\gamma \in [0, 1]$ is the constant for smoothing of the steering direction adjusting and is the selected direction by probability in (1).

The artificial agents in IAS-SS spread pheromone on a wide area in front of their respective positions, corresponding to sensor range area. The amount of the pheromone deposited on the ground decreases as the distance from the robot increases. Consider that L_t^k and $Q \in \mathbb{R}^2$

are the sensor range area at iteration t and the entire environment space, respectively, such that $L_t^k \subset Q \subset \mathbb{R}^2$. Then, the amount of pheromone $\Delta_q^k(t)$, deposited by robot k on the position q at iteration t is:

$$\Delta_q^k(t) = (1 - \tau_q(t-1)) \Gamma_q^k(t) \quad (3)$$

$$\Gamma_q^k(t) = \begin{cases} \delta e^{-\frac{(-q-q_k)^2}{\sigma^2}} \\ 0, \text{ otherwise} \end{cases}, \text{ if } X \in L_t \quad (4)$$

where q_k is the position of robot k ; σ is the dispersion and $\delta \in (0, 1)$.

Furthermore, pheromone evaporates according to a specific rate. The total amount of the pheromone that evaporates $\Phi_q(t)$ at position q and time t is given by:

$$\Phi_k(t) = \rho \tau_q(t) \quad (5)$$

where ρ is the evaporation rate and $\tau_q(t)$ is the total amount of pheromone at position q at iteration t .

Therefore, the total amount of pheromone $\tau_q(t)$ at q and time t is:

$$\tau_q(t) = (\tau_q(t-1) - \Phi_k(t-1)) + \sum_{k=1}^N \Delta_q^k(t) \quad (6)$$

Although any navigation strategy can be considered to move the robots in order to merge the local maps, the IAS-SS strategy is adopted due to its applicability in any environment configuration, including environment with narrow areas. The main proposal of the strategy is to execute the exploration and surveillance tasks. It was designed to overcome the difficulties of the surveillance task, proofed in [37] to be a NP hard problem. Several approaches present approximate algorithms in known environments and robots' pose [38-40]. However, the IAS-SS strategy is independent of environment structure and number of robots. The pheromone released by each robot is represented in a structure similar to local maps. Hence, the robots also generate local pheromone maps to merge into a global pheromone map using the integration method proposed here.

In next section, the mapping integration method is proposed.

Map Integration Method

The mapping integration method proposed here is an extension of the information sharing approach proposed by Tan, et al. [12], adapting the method for mapping tasks. The integration method proposed merges simultaneously two kinds of maps: The map of the estimates of the cell states of the occupancy grid and the pheromone map of the environment.

Differently the approach proposed by Tan, et al. [12], in our approach the robot coordinate system does not need to be centered on its position during the entire ex-

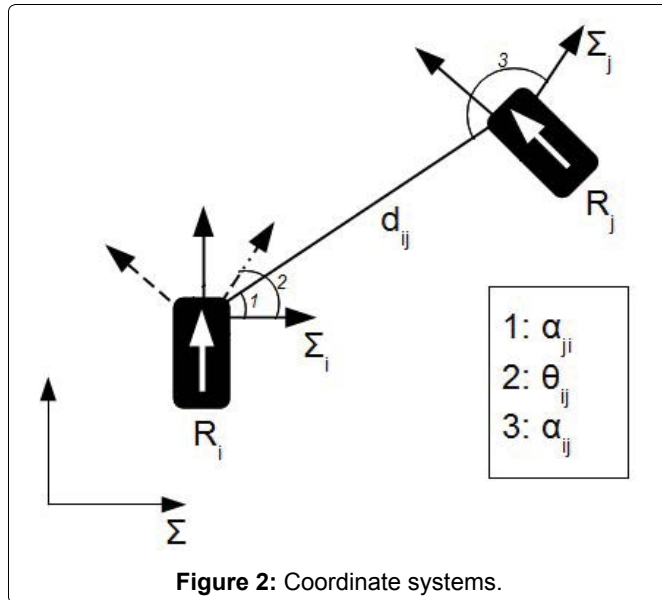


Figure 2: Coordinate systems.

ploration stage. Each robot has its own coordinate system that is centered in the initial position of each robot. It is worth to note that these initial positions do not change during the exploration process. Since the coordinate systems are independent, the initial position of each robot is established in the center of its own map. In this way, the initial position of all robots is the coordinate (0,0) of their own maps. The initial angle is also equal to 0°.

The model of the sensor used on this method is similar to model used in IAS-SS. There is only one difference between the sensor model used in our approach and the sensor model in [25-27], which is the communication radius in the integration method. Since the robots will exchange information between themselves, each robot has a communication device which allows to identify other robots and to exchange local map information. R_{C1} and R_{C2} are the communication radius of the robot R_i and R_j , respectively, and d_{12} is the distance between the robots.

For a better comprehension of the mapping integration method, consider two adjacent robots R_i and R_j and their respective coordinate systems Σ_i and Σ_j (Figure 2). Robot R_i sends to R_j its coordinate system Σ_i and its respective position (x_p, y_i) inside Σ_i . On the other hand, R_j sends to R_i its coordinate system Σ_j and its respective position (x_j, y_j) inside Σ_j . It is worth to notice that Σ_i and Σ_j are static.

Based on the perception between robots R_i and R_j , α_{ji} and α_{ji} are known for both robots, where α_{ji} is the orientation of R_i in coordinate system of R_j and α_{ji} is the orientation of R_j in coordinate system of R_i . The distance d_{ij} between robots R_i and R_j is also known.

The relative orientation between robots R_i and R_j is denoted by θ_{ij} , as illustrated in Figure 3(a). Assuming a scenario in which the robot R_i shares its local map with robot R_j , the distance d_{ij} between the robots and the ob-

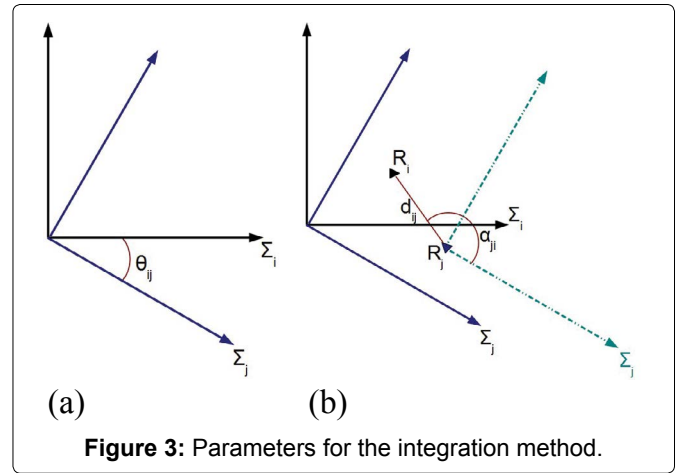


Figure 3: Parameters for the integration method.

served orientation of R_i in the coordinate system of the robot R_j are presented in Figure 3(b).

Integration process

Consider a position P_k belonging to Σ_i , $P_k = (x_{P_k}, y_{P_k})$. Initially, according to (7), the position P_k undergoes a rotation process based on the relative angle between robots in order to establish a new position $P'_k = (x'_{P_k}, y'_{P_k})$.

$$\begin{aligned} x'_{P_k} &= x_{P_k} \cos \theta_{ij} - y_{P_k} \sin \theta_{ij} \\ y'_{P_k} &= x_{P_k} \sin \theta_{ij} + y_{P_k} \cos \theta_{ij} \end{aligned} \quad (7)$$

Since the position P_k is rotated, it is required to define the position of robot R_i according to this rotation. In similar way to previous equation, the rotated position of robot R_i is defined by:

$$\begin{aligned} x'_i &= x_i \cos \theta_{ij} - y_i \sin \theta_{ij} \\ y'_i &= x_i \sin \theta_{ij} + y_i \cos \theta_{ij} \end{aligned} \quad (8)$$

Considering the rotation of coordinate system of R_i and according to relative angle between robots, it is needed to define the distance on the x -axis and the y -axis from the position P_k to position of robot R_i on the new coordinate system, that is:

$$\begin{aligned} x'_{P_k} &= x'_{P_k} - x'_i \\ y'_{P_k} &= y'_{P_k} - y'_i \end{aligned} \quad (9)$$

Since the rotation is done and the distance from the position P_k to robot R_i is established on the rotated coordinate system, the next step is to compute the translation from the position P'_k to the coordinate system Σ_j , as presented in (10). The position P_k^* obtained from this translation defines the position in coordinate system Σ_j which will be integrated with the coordinate of position P_k belonging to Σ_i .

$$\begin{aligned} x_{P_k}^* &= x_j + d_{ij} \cos \alpha_{ji} + x_{P_k}^i \\ y_{P_k}^* &= y_j + d_{ij} \sin \alpha_{ji} + y_{P_k}^i \end{aligned} \quad (10)$$

According to previous definition, the position P_k^* is established by three factors: 1) The position of robot R_j in its coordinate system; 2) The distance between two robots; and 3) The distance from the position P_k to the position of the robot R_j , considering the relative rotation of its coordinate system. It is worth to emphasize that proposed system by Tan, et al. [12] does not consider the position of the robot R_j during the translation process of coordinates of the robot R_i . Therefore it is not possible to build incrementally a map of the environment.

The exact pose of the robots is not determinant for the efficiency of the method. Pose's disturbs can occur and the method execution is not damaged. The frequent information exchanging promotes the update of maps already integrated. It can fix some incoherence of the map.

A description of the integration process between robots R_i and R_j is given in Algorithm 1. The variable *itbegin_threshold* indicates a control parameter which defines the initial amount of iterations for the integration process is initialized. This variable assures the robots obtain a minimal information about the environment. Another important control parameter is the variable *it_interval*, responsible for defining a lower interval of

Algorithm 1° Integration Method

```
integrate(itcurrent, poseRj, mapRj, idRj)
BEGIN
  IF (itcurrent >= itbegin_threshold)
    interval ← itcurrent – lastIntegrationRiRj;
    IF (interval > it_interval)
      lastIntegrationRiRj ← itcurrent;
      mapRotate(poseRj, mapRj);
      mapTranslate(poseRj, mapRj);
      integration(mapRi, mapRj);
    END-IF
  END-IF
END
```

iterations for a new integration between the same pair of robots. This parameter aims to reduce the redundant exchange of information, ensuring a multi-robot mapping method more efficient. Assuming the constraints represented by variables *itbegin_threshold* and *it_interval* are satisfied, the transformation of the coordinate system of the robot R_j can be initialized.

Functions **mapRotate** and **mapTranslate** are responsible for rotating and translating, respectively, both occupancy grid and pheromone map of the robot R_j to the coordinate system of the robot R_i . Since all maps are described in compatible coordinate systems, the occupancy

grid of the robot R_i is integrated with the occupancy grid of the robot R_j through the function **integration**. Specifically, the integration is determined by calculating the average of probabilities of occupation of each local map, considering pairs of cells with the same coordinates. Similar process is done for the pheromone maps.

In the Algorithm 2, it is presented the general operation of the exploration and mapping multi-robot system. The variable *number_iterations* indicates the total amount of iterations that the exploration and mapping method will be performed. The variable *id_{Ri}* indicates the identifier of the robot R_i and the variables *pose_{Ri}* and *map_{Ri}* are the position and the map (both occupancy and pheromone map) of robot R_i , respectively. Note that *pose_{Ri}* indicates the position of robot R_i in its map *map_{Ri}*.

The function **atualizeMap** updates the occupancy grid through sensor readings captured by the robot. Functions **sendData** and **receiveData** are responsible for transmitting and receiving, respectively, the information about the local map of a robot and its localization. If the robot R_i identifies robot R_j , considering their communication radius, the function **integrate** is called. This function realizes the integration of their local maps.

The function **detectPheromone** detects pheromone concentration at the border of the sensor range, so the adjustment of the steering direction is determined by the function **adjustMovementDirection**. Since the direction is defined, the robot deposits pheromone on the environment

Algorithm 2° Control system

```
main()
BEGIN
  FOR itcurrent ← 1 TO
    number_iterationsatualizeMap();
    sendData(poseRi, mapRi, idRi);
    IF (receiveData(poseRj, mapRj, idRj) == TRUE)
      integrate(itcurrent, poseRj, mapRj, idRj);
    END-IF
    detectPheromone();
    adjustMovementDirection();
    releasePheromone(); move();
  END-FOR END
```

(**releasePheromone**) and moves to the specified direction (**move**).

Virtual pheromone

A relevant aspect that must be analyzed is the applicability of the IAS-SS strategy in real environments us-

ing real robots. Specifically, the main question is how the pheromone, used in the exploration and surveillance strategy, will be released by real robots. A solution for this problem is to use virtual pheromone [35,36,41-43].

Information about released pheromone is allocated in a grid structure (similar to environment map). Each cell of the grid represents a portion of the environment. The pheromone information in a cell indicates the amount of pheromone in the corresponding region. The robots should check the pheromone information in the grid (virtual pheromone) to take decision. Indeed, that grid is extra memory is needed for the IAS-SS strategy, that is, a flag (the cell) for each region in order to indicate the visited areas and the period they are not be monitored. Other approach that uses a grid is the Potential Field Methods (PFM) [39,40]. In this case, the robots are guided to a goal-position and they have a global map. If the goal-position changes continuously according the areas where the amount of pheromone is low, then all cells of PFM must be updated. For the IAS-SS strategy, only the local area is modified by pheromone releasing. The virtual pheromone is needed to avoid designing complex robots able to deposit real substances and dirtying environments.

In the proposed approach, each robot is responsible for managing its pheromone concentration map, carrying out the release operation when appropriate. Since each robot keeps a local map of pheromone concentration, a process of pheromone integration is realized.

For an ideal case, where the computational cost is not important, the pheromone integration can be performed assuming both an unlimited communication range and the robots exchange information among themselves at each iteration. If all robots share the pheromone information among themselves at each iteration, their pheromone grids will correspond to the global map of the amount of pheromone.

Given the characteristics of the current communication devices, the assumption of a communication radius able to connect all robots can be implemented through Wireless devices [1,3]. In the virtual pheromone approach, the main problem is the computational time spent for the pheromone map sharing among all robots at each iteration. Since the proposed approach assumes a limited communication radius among the robots, a Wireless device can be applied for transmitting information about local maps between pairs of robots, considering indoor environments.

This problem can be solved including a constraint that limits the information exchange to minimum intervals of iterations. Besides reducing the system computational time, this constraint decreases the redundant information exchange among the robots. Another factor could be considered to reduce the computational cost is to ensure that the robots share their pheromone maps only with these ones located within a communication radius R_c .

Since the process of steering direction adjusting of a robot R_k detects the pheromone concentrations at a specific distance R_d , only the robots near R_k could interfere on this decision of movement adjusting, through pheromone releasing. Experiments using real robots and virtual pheromone are part of future works. Here, simulated experiments are realized to demonstrate the integration method for pheromone maps.

Experimental Results

Several experiments are performed to test the proposed approach, varying the number of robots and considering different types of environments.

Experiments are carried out in Player/Stage platform, that models various robots and sensors simulating simultaneously their exact dynamics. Although this platform includes navigation mechanism for obstacle

Table 1: Parameters.

Parameter	Value
Number of iterations	1000
R_d (radius of the semicircle where the pheromone is deposited and provided by laser range finder)	8.00 meters
R_{ck} (communication radius of robot R_k , $k = \{1, \dots, N\}$)	8.00 meters
robot speed	0.5 meter per second
σ (pheromone releasing rate)	$0.4R_d$
ρ (evaporation rate)	10^{-4}
$\tau_x(0)$ (amount of pheromone at iteration $t = 0$)	0.5
γ (coefficient for smoothing of steering direction adjusting)	0.5
S (number of angle intervals)	360
U (number of angle intervals)	30% of size of S set
V (number of angle intervals)	10% of size of S set
ltbegin_threshold	100
lt interval	50

avoidance, this behavior emerges in IAS-SS system only from consequence of the pheromone repulsive nature. The robot model used is the Pioneer 2DX equipped with a laser range-finder SICK LMS 200 able to scan the environment (general obstacles, e.g., walls and objects). This robot and sensor were chosen due to the future tests with the available real equipment's at our institution.

The system parameters used in the experiments are presented in Table 1 and correspond to those in which the multiple robot system reaches the best performance, considering all previous experiments executed in [25,26]. The values assigned to parameters σ and ρ were defined through analysis of the performance in [27]. The performance criteria adopted in [25-27] are: The time necessary to conclude the exploration task (SE); and maximum time interval between two consecutive sensing of any specific region (SI). The last one criteria refers to surveillance task.

Considering the integration process, the local map integrating is only started at iteration $t \geq 100$. This restriction ensures that the robots obtain the minimal information about the environment before to start the integration process. Since robot R_i joins its coordinate system with robot R_j , it is defined that robot R_i will wait 50 iterations to share again its local map with robot R_j . This strategy decreases the redundant information exchange.

All experiments were executed 10 times. Thus, the average of the performances is computed to evaluate them. The discrete time is adopted in simulation and it is equivalent to the number of iterations. The environment models adopted are illustrated in Figure 4. The environments are divided in small and connected areas, named *rooms*. The rooms are empty, however, the environments have many walls (that define the rooms) that difficulties the robots' navigation. Each room is divided in very small regions, called here *sectors*. The environments had been divided in 24 sectors.

A sector is said to be visited if it is reached by any robot. It is worth to note that if a robot is physically in a sector C_i and its sensors achieve both sector C_i and C_j , it is considered the robot visited only the sector C_i . The environment was set up for a dimension $30\text{ m} \times 20\text{ m}$. The total environment area (600 m^2) was represented by a matrix with dimension 375×375 . The maximum number of iterations (1000) spends approximately 90 minutes for each simulation. Then, each iteration spends 5.4 seconds. For the experiments follow, few robots are used. As the objective is to propose an integration method of maps, the information exchanging between only two robots is enough to validate the method (coordinate system transformation). More robots in the environment will accelerated the building of the global map. That is, a high number of robots will not affect the execution of the performance of the integration method.

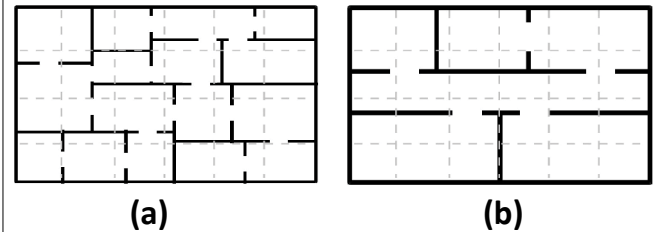


Figure 4: Models of environment: a) Environment #1; b) Environment #2.

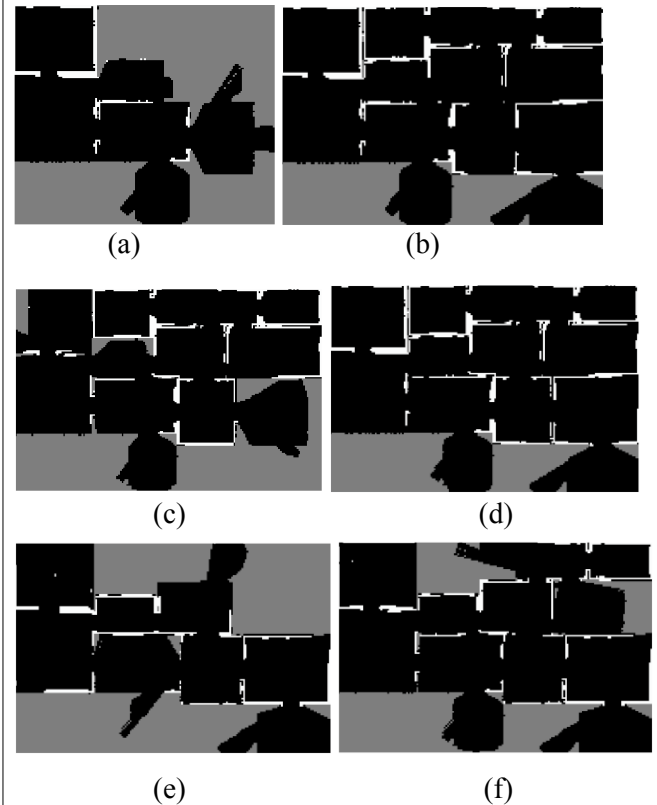


Figure 5: a), c), e) Local maps of robots R_1 , R_2 and R_3 , respectively; b), d), f) Integrated maps of R_1 , R_2 and R_3 , respectively.

In Figure 5, it is presented both local and integrated occupancy grid obtained by three different robots (R_1 , R_2 and R_3) positioned in the environment of Figure 4a. The obstacles, detected in the environment, are represented by the white color. The free space is indicated by black color. Areas not explored or unknown are represented by the gray color. The simulation was carried out for few iterations. For that reason, there are some areas not explored by any robot. That areas is not in the integrated map, obviously. The areas visited by at least one robot can be transferred to another robot at instant they are close each other. It is important to notice that the integrated map of each robot is different from the others, since the information exchange among the robots is based on rendezvous measures.

The robots start the exploration process at random positions in top left area of the environment. On the left column of Figure 5, the local maps of each robot are

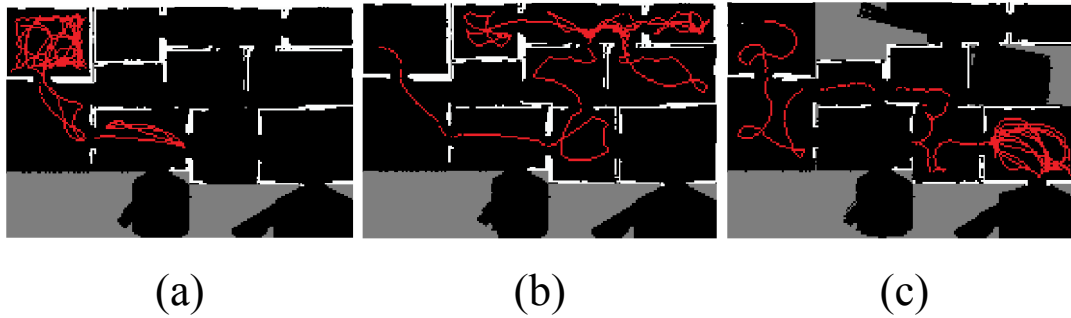


Figure 6: Trajectories of robots: a) R_1 ; b) R_2 and c) R_3 .

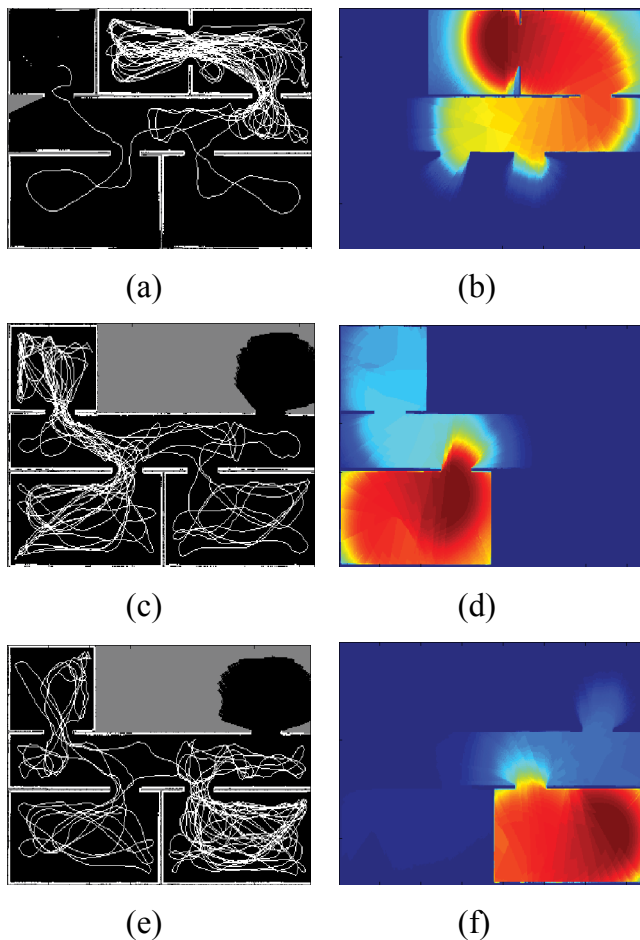


Figure 7: a), c), e) Trajectories of robots R_1 , R_2 and R_3 , respectively; b), d), f) Local pheromone maps of R_1 , R_2 and R_3 , respectively.

shown. These maps are built considering only the information acquired by robot sensors. On the right column, the maps obtained from the integration process among robots are illustrated for each one of them. These images demonstrate clearly the information gain obtained by local map integrating.

The trajectories realized by each robot during the exploration of environment in this experiment are shown in Figure 6. These trajectories are represented by red lines in the map. It is possible to identify which sectors each robot visited and which sectors were obtained by

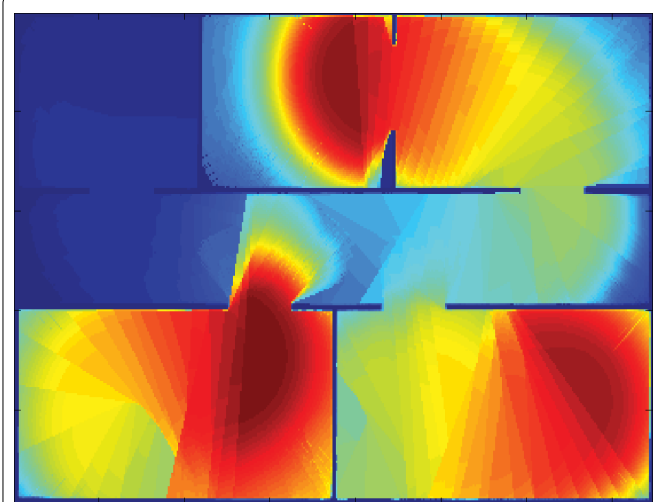


Figure 8: Integrated pheromone maps of robots R_1 , R_2 and R_3 .

the integration process among robots.

Next example shows the local and integration pheromone maps. The robots, R_1 , R_2 and R_3 start navigation at random positions in the left area of the environment Figure 4b and they navigate for 1000 iterations. Areas with high amount of pheromone are represented by dark red color. The transition of that color to dark blue indicates the reduction of the pheromone. The absence of the substance is highlighted by dark blue areas.

Pheromone is a volatile substance and the evaporation makes an area attractive again. According to the established evaporation rate, there will be not pheromone at some areas a robot visited long time ago. These areas are as attractive as those never visited. On the left column of the Figure 7, it is noticed the robots travel by several areas of the environment. However, the pheromone trail is perceived only at the recently visited areas (see right column of Figure 7).

Although the robots are at distinct areas from each other, they have information about the amount of pheromone released by another robot due to integration pheromone maps process. The approximation each other causes the maps integration. Since all of robots were close each other, the integrated pheromone maps are

identical for them (Figure 8). After the process, the robots are able to perceive the substance at locals they were not ever or they were there for long time ago (see trajectories of robots in the Figure 7). That characteristic favors the distributive accomplishment of the exploration and surveillance tasks.

The integration pheromone map process is adopted for all experiments presented in the next sections. The

emergent behavior consists of robots spreading out in the environment. Therefore, that process is considered implicitly to evaluate the performance of the exploration and surveillance tasks.

Comparing IAS-SS with uniform strategy for exploration

Considering the exploration task, the next set of ex-

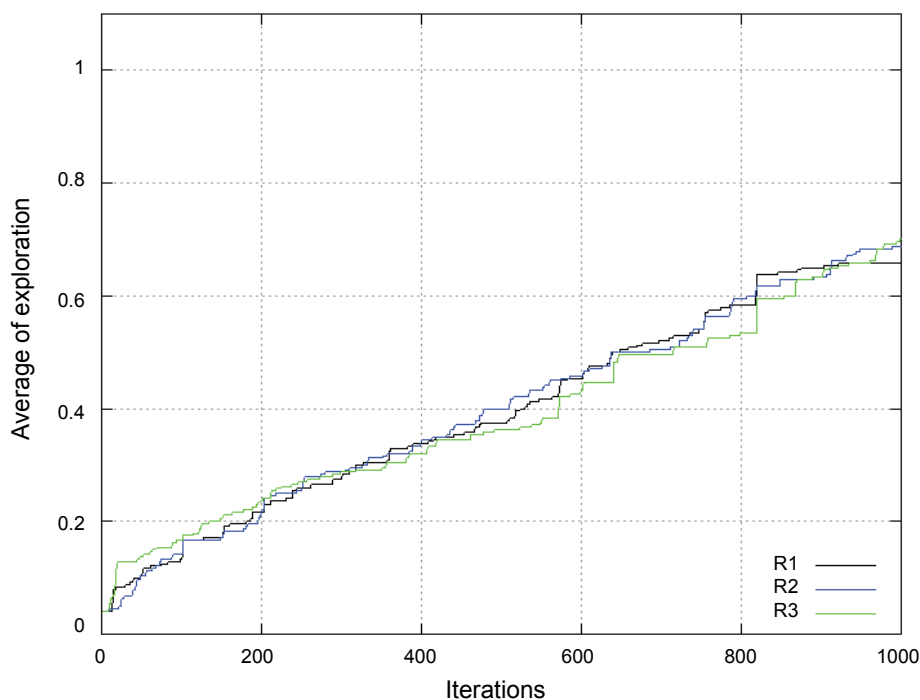


Figure 9: Uniform strategy.

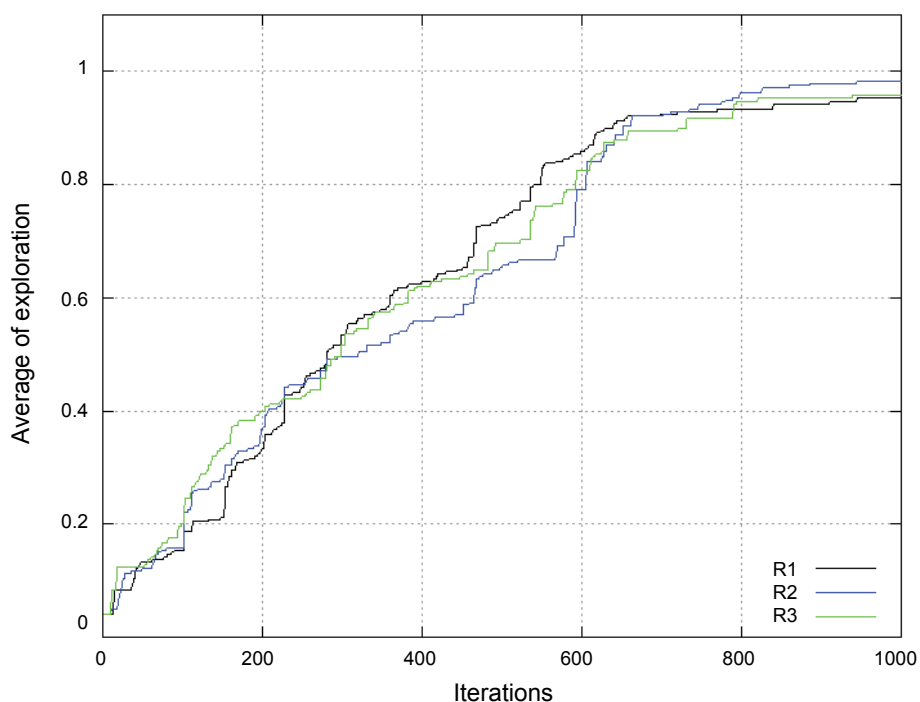


Figure 10: IAS-SS strategy.

periments was executed to evaluate the performance of exploration strategy based on IAS-SS. In order to show that performance of the mapping based on IAS-SS is not a consequence of a random behavior, the strategy is compared with a uniform strategy of exploration. According to uniform strategy, a discrete random variable, defined by an uniform distribution in the space of angles A_s , determines the steering direction adjustments. A set of 10 trials was executed for each strategy, considering three robots allocated in environment of Figure 4b.

The exploration process using the uniform strategy and the IAS-SS strategy are shown in Figure 9 and Figure 10, respectively, considering the average of performance obtained with each strategy. On the x-axis, the iterations of the exploration process are presented. On the y-axis, a variable on the interval [0,1] denotes the percentage of exploration of the environment. This variable denotes the relation between the amount of sectors represented by the map of each robot and the total amount of sectors of the environment.

The amount of sectors represented by the map of each

Table 2: Average of exploration - IAS-SS and uniform strategy.

Strategy	Average (μ)	Pattern deviation (σ)
IAS-SS	0.96	0.05
Uniform	0.66	0.19

robot indicates both the sectors visited by the robot and the sectors transferred to it through the map integration method proposed. The robots using IAS-SS strategy presents a performance substantially higher than the robots using the uniform strategy.

In Table 2, it is presented the average and the pattern deviation among the three robots, considering the endpoint of the average exploration of each strategy. The statistical difference between two strategies is evaluated through the unpaired t-test with unequal variances.

The calculated t value is $t_{cal} = 4.58$ and the degrees of freedom is $df = 9$. Entering a t table with 9 degrees of freedom, for a significance level $p = 0.05$, the tabled t value is $t_{tabled} = 1.83$. Since $t_{cal} > t_{tabled}$, the null hypothesis is rejected, indicating the average of exploration of the IASSS strategy is higher than the average of exploration of the uniform strategy.

Analyzing the integration process

In order to evaluate the integration process performance, experiments are carried out considering an increasing number of robots in environment of Figure 4a. Both the uniform and IAS-SS strategies are tested. Moreover, the mapping method was tested using two different ways. In the first case, the integration module is active and the robots are able to exchange local maps among

Table 3: Performance of mapping method using integration-uniform strategy.

Number of robots	R_1	R_2	R_3	R_4	R_5
1	0.20 ± 0.05	-	-	-	-
2	0.24 ± 0.07	0.24 ± 0.07	-	-	-
3	0.26 ± 0.05	0.26 ± 0.08	0.25 ± 0.05	-	-
4	0.29 ± 0.12	0.30 ± 0.12	0.28 ± 0.13	0.28 ± 0.13	-
5	0.39 ± 0.17	0.38 ± 0.17	0.38 ± 0.18	0.36 ± 0.16	0.36 ± 0.16

Table 4: Performance of mapping method without-uniform strategy.

Number of robots	R_1	R_2	R_3	R_4	R_5
1	0.20 ± 0.05	-	-	-	-
2	0.25 ± 0.05	0.20 ± 0.08	-	-	-
3	0.25 ± 0.05	0.25 ± 0.09	0.19 ± 0.04	-	-
4	0.24 ± 0.12	0.23 ± 0.08	0.20 ± 0.04	0.18 ± 0.02	-
5	0.30 ± 0.14	0.20 ± 0.07	0.25 ± 0.13	0.22 ± 0.10	0.20 ± 0.08

Table 5: Performance of mapping method with uniform strategy.

Number of robots	Integration	Average (m)	Pattern deviation (s)	t_{cal}
1	Yes	0.2	0.05	-
	No	0.2	0.05	
2	Yes	0.24	0.07	3.43
	No	0.2	0.06	
3	Yes	0.25	0.06	3.491
	No	0.21	0.06	
4	Yes	0.28	0.12	2.542
	No	0.21	0.06	
5	Yes	0.37	0.16	3.806
	No	0.23	0.1	

them. In the second case, the integration module is inactive and the robots do not exchange maps with the others. Thus, the robots acquire information only about the sectors they visited. Analogous to previous experiment, the robots start the navigation in top left area of the environment.

The average of exploration in the uniform strategy considering the mapping process using the integration method is presented in Table 3. Similarly, in Table 4, it is presented the average of exploration of the mapping method without the integration method. Considering the first case, the average of percentage of exploration of the environment is increased whereas the number of robots increases. Since the uniform strategy has a strongly random behavior, the increase of the average of exploration of the environment, caused by increasing the number of robots, is induced by the integration process when it is performed by a robot dispersed from the others, sharing meaningful special information to the robots in the environment.

The average of exploration and the pattern deviation for each set of experiments are presented in Table 5, considering the exploration process with and without in-

tegration in uniform strategy. The statistical difference between these two approaches is evaluated through the paired t -test. The calculated t value for each set of experiments is presented in Table 5 and the degrees of freedom is $df = 9$. Entering a t table with 9 degrees of freedom, for a significance level $p = 0.05$, the tabled t value is $t_{tabled} = 1.83$. Since $t_{cal} > t_{tabled}$ for each set of experiments, the null hypothesis is rejected in all the cases, indicating the average of exploration of the system using the integration method is higher than the average of exploration of the system without the integration in the uniform strategy. It is worth to note that using only a single robot it is impossible performing the integration process, so there is no comparison between the system with and without integration in this case.

In Table 6, it is presented the average of exploration of the mapping method using the integration method in IAS-SS strategy. Considering the mapping task using the integration process, the average of percentage of exploration of the environment is increased when the number of robots increases. In the case of using five robots, the average of environment exploration of one robot is equal to 80%. It is worth to note that the percentage of explora-

Table 6: Performance of mapping method using integration-IAS-SS strategy.

Number of robots	R ₁	R ₂	R ₃	R ₄	R ₅
1	0.44 ± 0.08	-	-	-	-
2	0.62 ± 0.09	0.63 ± 0.09	-	-	-
3	0.72 ± 0.08	0.74 ± 0.07	0.72 ± 0.07	-	-
4	0.70 ± 0.10	0.70 ± 0.10	0.70 ± 0.10	0.70 ± 0.10	-
5	0.79 ± 0.11	0.80 ± 0.11	0.80 ± 0.11	0.80 ± 0.12	0.79 ± 0.11

Table 7: Performance of mapping method without integration-IAS-SS strategy.

Number of robots	R1	R2	R3	R4	R5
1	0.44 ± 0.08	-	-	-	-
2	0.46 ± 0.13	0.52 ± 0.09	-	-	-
3	0.43 ± 0.17	0.51 ± 0.13	0.44 ± 0.12	-	-
4	0.42 ± 0.14	0.42 ± 0.16	0.38 ± 0.18	0.37 ± 0.16	-
5	0.42 ± 0.11	0.45 ± 0.15	0.44 ± 0.14	0.42 ± 0.12	0.44 ± 0.17

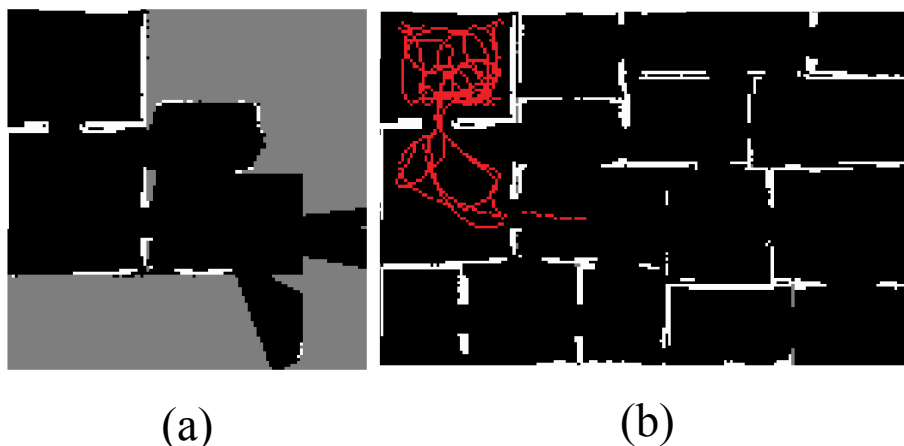


Figure 11: a) Local map; b) Trajectory map.

tion indicates the number of visited sectors and not the number of mapped sectors. On the other hand, when the exploration system is executed without the integration process in IAS-SS strategy (Table 7), the average of exploration keeps a similar value in all trials. Since the experiments are executed with a same amount of iterations, the robots tend to visit an amount of sectors with a same area. Thus the average of exploration does not depend on the number of robots, when there is no integration.

In Figure 11, it is presented the map built by one of the five robots in the experiment, using the integration process. In this trial, it is evidenced the gain of information acquired by performing the multiple robot mapping. According to robot trajectory, it is possible to see clearly that the map obtained by the robot has information about the entire environment, although it has not effectively explored the environment by itself.

The average of exploration and the pattern deviation for each set of experiments are presented in Table 8, considering the exploration process with and without integration in IAS-SS strategy. The statistical difference between these two approaches is evaluated through the paired t -test.

The calculated t value for each set of experiments is presented in Table 8 and the degrees of freedom is $df = 9$. Entering a t table with 9 degrees of freedom, for a significance level $p = 0.05$, the tabled t value is $t_{tabled} = 1.83$. Since $t_{cal} > t_{tabled}$ for each set of experiments, the null hypothesis is rejected in all the cases, indicating the average of exploration of the system using the integration method is higher than the average of exploration of the system without the integration in IAS-SS strategy as well.

Table 8: Performance of mapping method with IAS-SS strategy.

Number of robots	Integration	Average (μ)	Pattern deviation (s)	t_{cal}
1	Yes	0.44	0.08	-
	No	0.44	0.08	
2	Yes	0.62	0.09	7.939
	No	0.49	0.11	
3	Yes	0.72	0.07	16.305
	No	0.46	0.14	
4	Yes	0.7	0.1	15.478
	No	0.39	0.16	
5	Yes	0.79	0.11	9.806
	No	0.43	0.13	

Another set of experiments was executed to evaluate the performance of the integration method using the IAS-SS strategy with different communication radii. In these experiments five robots were positioned in a environment without obstacles. The tested communication radii were 4, 8, 16, 24, 32 and 40 meters. A set of 10 trials was executed for each radius. In Table 9, it is presented the average of exploration of the mapping method for each robot and for each radius.

If the radius is equal to 40 meters, each robot is able to communicate with any other robot within the environment, i.e. the distance between two any robots within an environment 30 meters \times 20 meters always is lower than 40 meters.

Initially the increment of the communication radius increases the percentage of completed exploration. However, for radius values greater than 24 meters there is a smooth decay on the percentage of the exploration. This decay is caused by the increasing of the number of integrations. Considering that the radius is equal to 32 and 40 meters, the robots always exchange information with all others in the environment, independently of the distance among them. Once there is more communication among the robots, the exploration process becomes slower.

Discussion of Results

Analyzing the mapping task, an important advantage of the method proposed here compared to model described in [12] is the capability of mapping the environment, since the model in [12] only deals with the coverage problem, it does not build a map that represents the environment. Moreover, the capability of sensing the environment in [12] depends on the number of robots and the communication radius among them. Therefore there is no assurance that the whole environment will be explored. On the other hand, the integration method uses the IAS-SS strategy as exploration strategy and, because of this, it is able to explore the whole environment in a certain number of iterations. A second difference between the two approaches is the fact that the multiple robot system proposed in [12] is applied only in environments without obstacles. In our approach each environment is divided in rooms, containing walls (obstacles) among the rooms.

Table 9: Performance of mapping method varying the radius (IAS-SS active).

Number of robots	R1	R2	R3	R4	R5
4 meters	0.92 \pm 0.11	0.91 \pm 0.11	0.95 \pm 0.06	0.92 \pm 0.11	0.94 \pm 0.09
8 meters	0.97 \pm 0.06	0.97 \pm 0.05	0.97 \pm 0.06	0.96 \pm 0.06	0.96 \pm 0.06
16 meters	0.99 \pm 0.01	0.99 \pm 0.01	0.99 \pm 0.01	0.99 \pm 0.01	0.99 \pm 0.01
24 meters	0.98 \pm 0.02	0.98 \pm 0.02	0.98 \pm 0.02	0.98 \pm 0.02	0.98 \pm 0.02
32 meters	0.96 \pm 0.04	0.94 \pm 0.04	0.97 \pm 0.03	0.94 \pm 0.05	0.95 \pm 0.05
40 meters	0.94 \pm 0.06	0.90 \pm 0.13	0.93 \pm 0.08	0.88 \pm 0.12	0.88 \pm 0.12

Another important advantage of the proposed approach is the continuous mapping. The IAS-SS strategy ensures that the environment is continuously explored based on stimuli received from the pheromone deposited. Since the environment is periodically explored, the maps obtained by the robots are constantly updated and any change in the structure of the environment can be recognized and transferred to all robots in the system.

Considering the number of robots, as bigger the number of robots is, more areas of the environment will be mapped by the proposed system. The environment does not need to be known before the exploration. Due the IAS-SS behavior, the probability of one robot to explore and monitor large areas is not null. It may just take a long time. The time to map decreases as number of robots increases, the mapping task is accomplished even with a restricted number of robots. Even considering a small number of robots, they are able to monitor and map large areas.

Some approaches were found in the literature to compare to the obtained performance here. However, the simulators, robot models, environment configuration, among others, are distinct from the resources used in this research. Besides, the information available in the approaches is not enough to reproduce and realize a comparison. This kind of problem requires detailed information to be reproduced by another researcher. For this reason, the proposed method was executed using two versions of navigation strategies to demonstrate that the integration method is efficient for strategies whose robots' coordinate favors they spread out. Thus, instants the robots meet each other, the map resulting from the integration method will cover a big area, maybe the whole environment faster.

Nevertheless, a specific approach is highlighted. The algorithm proposed by Liu, et al. [7] based on Particle Swarm Optimization and Hilbert curve, called here PSO and Hilbert curve exploration algorithm, was set up for plane coordinate 400×400 , divided into 16 sub-regions, there are three robots and the number of iterations is 3500. A method based on random S-shaped curve exploring was compared with the method based on PSO and Hilbert curve exploring. Percentages of completed exploration of random S-shaped curve exploring algorithm and PSO and Hilbert curve were approximately 30% and 70%, respectively. Considering these simulation results, the method proposed in this paper is better, since the maximum percentage of the exploration of our method was 96% (Table 2). A serious problem with this analysis is the fact the dimension of the simulated environment of Liu, et al. [7] is unknown. The fact of their plane coordinate was set up for 400×400 does not indicate the real size of the environment, so this comparison is just a superficial analysis.

However, it is worth to note that each sub-region of the method proposed by Liu, et al. [7] corresponds to 6.25% of the entire environment. On the other hand, each sector of the integration method based on IAS-SS corresponds to 4.16%. Moreover, there is a significant difference between the tested environments in each method. Environments in [7] are less complex than our simulations. Environments used by Liu, et al. [7] are formed by few obstacles allocated in a free area.

Conclusion and Future Works

In this work, a new local map integration method was proposed, which is based on inter-robot observations, considering the Inverse Ant System-Based Surveillance System strategy as method for environment exploration. In this method, the local map integrating is defined by transformation matrices, represented by the distance between two robots and their relative positions. The method is dependent neither on the knowledge of the environment structure nor initial positions of the robots.

A set of experiments were conducted for performance analyzing. Two coordination strategies are considered and compared, one of them is the IAS-SS strategy based on ants' behavior algorithm, and the other one is the uniform strategy. The IAS-SS strategy performance is significantly superior. Another evaluation is the impact of increasing the number of robots during the trials. The higher the number of robots, the higher the percentage of exploration of the environment. It was also evaluated the use of integration during the exploration process. The performance of mapping using the local map integration is higher than the performance without the integration.

As future works some aspects of the exploration system will be considered for analysis, e.g.: The communication mechanism. The multiple robot exploration with limited communication range restricts the communication abilities of the robots. Naturally, the task of mapping an environment with limited communication range is harder than without this constraint. Moreover, the virtual pheromone idea will be developed in order to evaluate the exploration and mapping system in real environments. In this case, the pheromone local maps will be integrated in a similar way to the method proposed in this paper. Actually, our research group does not have real robots in number enough to reproduce the experiments. They are in designing process. We intend to adapt the built soccer robots to execute the surveillance task. In addition, a localization method will be integrated to system in order to deploy it in real robots.

Acknowledgment

The authors would like to thank FAPESP (Grant #2010/07955-8) and CNPq for their support.

References

1. X Yun, Y Yi-Min, X Yi-Min (2005) Cooperative map building of multirobot based on grey fusion. IEEE International Conference on Industrial Technology, 353-358.
2. CM Gifford, R Webb, J Bley, D Leung, M Calnon, et al. (2008) Low-cost multi-robot exploration and mapping. IEEE International Conference on Technologies for Practical Robot Applications, 74-79.
3. T Tong, H Yalou, Y Jing, S Fengchi (2008) Multi-robot cooperative map building in unknown environment considering estimation uncertainty. Chinese Control and Decision Conference 2896-2901.
4. T Vidal-Calleja, C Berger, S Lacroix (2009) Even-driven loop closure in multi-robot mapping. IEEE International Conference on Intelligent Robots and Systems, 1535-1540.
5. P Li, X Huang, M Wang, X Zeng (2008) Multiple mobile robots map building based on DSMT. IEEE Conference on Robotics, Automation and Mechatronics, 509-514.
6. Speranzon (2006) Coordination, consensus and communication in multi-robot control systems. Royal Institute of Technology, Stockholm/Sweden.
7. Chunyang Liu, Yingwei Ma, Chang'an Liu (2009) Cooperative multi-robot map-building under unknown environment. International Conference on Artificial Intelligence and Computational Intelligence, 392-396.
8. LAA Anderson, J Nygards (2009) On Multi-robot map fusion by inter-robot observations. 12th International Conference on Information Fusion, 1712-1721.
9. A Barto, S Bradtke, S Singh (1995) Learning to act using real-time dynamic programming. Artificial Intelligence 72: 81-138.
10. F WeiXing, W KeJun, Y XiuFen, G ShuXiang (2006) Novel algorithms for coordination of underwater swarm robotics. International Conference on Mechatronics and Automation, 654-659.
11. E Ferranti, N Trigoni, M Levene (2007) Brick & Mortar: An online multiagent exploration algorithm. IEEE Int Conf Robotics and Automation.
12. J Tan, N Xi, W Sheng, J Xiao (2004) Modeling Multiple Robot Systems for Area Coverage and Cooperation. IEEE International Conference on Robotics and Automation, 2568-2573.
13. Q Jiang (2006) An improved algorithm for coordination control of multi-agent system based on r-limited voronoi partitions. IEEE International Conference on Automation Science and Engineering.
14. RA Freeman, P Yang, KM Lynch (2006) Distributed estimation and control of swarm formation statistics. American Control Conf 7: 749-755.
15. F Kobayashi, N Tomita, F Kojima (2003) Re-formation of mobile robots using genetic algorithm and reinforcement learning. SICE 2003 Annual Conf 3: 2902-2907.
16. L Barnes, W Alvis, MA Fields, K Valavanis, W Moreno (2006) Swarm formation control with potential fields formed by bivariate normal functions. 14th Mediterranean Conf on Control and Automation, 1-7.
17. F Amigoni, S Gasparini, M Gini (2006) Building segment-based maps without pose information. Proceedings of IEEE 94: 1340-1359.
18. F Amigoni, S Gasparini (2008) Analysis of methods for reducing line segments in maps: Towards a general approach. 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2896-2901.
19. Y Yu, N Wang, A Liang, H Guan, L Liu (2009) Integrating line segment based maps in multi-robots exploration. Second International Conference on Future Information Technology and Management Engineering, 230-233.
20. XS Zhou, SI Roumeliotis (2006) Multi-robot slam with unknown initial correspondence: The robot rendezvous case. IEEE/RSJ International Conference on Intelligent Robots and Systems, 1785-1792.
21. Alberto Elfes (1989) Occupancy grids: A probabilistic framework for robot perception and navigation. Carnegie Mellon University.
22. Alberto Elfes (1989) Using occupancy grid for mobile robot perception and navigation. IEEE Computer Magazine 22: 46-57.
23. Alberto Elfes (1989) A tessellated probabilistic representation for spatial robot perception. 1989 NASA Conf on Space Telerobotics, NASA/Jet Propulsion Laboratory, California Institute of Technology.
24. M Santana, KRT Aires, RMS Veras, AAD Medeiros (2011) An approach for 2d visual occupancy grid map using monocular vision. Electronic Notes in Theoretical Computer Science 281: 175-191.
25. R Calvo, JR Oliveira, M Figueiredo, RAF Romero (2011) A distributed, bio-inspired coordination strategy for multiple agent systems applied to surveillance tasks in unknown environments. International Joint Conference on Neural Networks, 3248-3255.
26. R Calvo, JR Oliveira, M Figueiredo, RAF Romero (2011) Inverse ACO applied for exploration and surveillance unknown environment exploration. The Third International Conference on Advanced Cognitive Technologies and Applications, 142-147.
27. R Calvo, JR Oliveira, M Figueiredo, RAF Romero (2011) Bioinspired coordination for multiple robots systems and stimergy mechanisms to cooperative exploration and surveillance tasks. IEEE International Conference on Cybernetics and Intelligent Systems.
28. A Wagner, M Lindenbaum, AM Bruckstein (1999) Distributed Covering by Ant-Robots Using Evaporating Traces. IEEE Transactions on Robotics and Automation 15: 918-933.
29. S Koenig, B Szymanski, Y Liu (2001) Efficient and Inefficient Ant Coverage Methods. Annals of Mathematics and Artificial Intelligence 31: 41-76.
30. V Yanovski, IA Wagner, AM Bruckstein (2003) A Distributed Ant Algorithm for Efficiently Patrolling a Network. Algorithmica 37: 165-186.
31. H Chu, A Glad, O Simonin, F Semp, A Drogoul, et al. (2007) Swarm approaches for the patrolling problem, information propagation vs. pheromone evaporation. 19th IEEE International Conference on Tools with Artificial Intelligence, 442-449.
32. Y Zhang, Y Xiao (2013) Digital pheromone based patrolling algorithm in wireless sensor and actuator networks. IEEE 10th Consumer Communications and Networking Conference, 496-501.
33. J Blanco, J Gonzalez-Jimenez, J Fernandez-Madriral (2013) A

- robust, multi-hypothesis approach to matching occupancy grid maps. *Robotica* 1-15.
34. S Topal, I Erkmen, AM Erkmen (2013) A novel multirobot map fusion strategy for occupancy grid maps. *Turkish Journal of Electrical Engineering & Computer Sciences* 21: 107-119.
 35. HJ Chang, CSG Lee, YC Hu, Y Lu (2007) Multi-robot SLAM with topological/metric maps. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1467-1472.
 36. W Burgard, M Moors, C Stachniss, F Schneider (2005) Coordinated multi-robot exploration. *IEEE Transactions on Robotics* 21: 376-386.
 37. Wallar, E Plaku, DA Sofge (2015) Reactive motion planning for unmanned aerial surveillance of risk-sensitive areas. *IEEE Transactions on Automations Science and Engineering* 12: 969-980.
 38. S Chen, F Wu, L Shen, J Chen, SD Ramchurn (2016) Decentralized Patrolling Under Constraints in Dynamic Environments. *IEEE Transactions on Cybernetics* 46: 3364-3376.
 39. Lee SK, Fekete SP, McLurkin J (2016) Structured triangulation in multi-robot systems: Coverage, patrolling, voronoi partitions, and geodesic centers. *The International Journal of Robotics Research* 35: 1234-1260.
 40. Zhang Y, Xiao Y, Wang Y, Mosca P (2017) Bio-inspired Patrolling Scheme Design in Wireless and Mobile Sensor and Robot Networks. *Wireless Pers Commun* 92: 1303-1332.
 41. D Payton, M Dally, R Estkowski, M Howard, C Lee (2001) Pheromone robotics. *Autonomous Robots* 11: 319-324.
 42. D Payton, R Estkowski, M Howard (2001) Compound behaviors in pheromone robotics. *Robotics and Autonomous Systems* 44: 229-240.
 43. HVD Parunak, M Purcell, R O'Connell (2002) Digital Pheromones for Autonomous Coordination of Swarming UAV's. *AIAA*.