Autonomous navigation strategy for robot swarms using local communication

Estrategia de navegación autónoma para un enjambre de robots utilizando comunicación local

Fredy Hernán Martínez Sarmiento
Electrical Engineer, Ph.D. Candidate in Engineering. Professor at the Universidad Distrital Francisco José de Caldas. Bogotá, Colombia.
Contact: *fhmartinezs@udistrital.edu.co*

Diego Mauricio Acero Soto
Electronic Engineer, Master Candidate in Information Sciences. Professor at the Universidad Pedagógica Nacional. Bogotá, Colombia.
Contact: *dacero@pedagogica.edu.co*

Mariela Castiblanco Ortiz
Electronic Engineer, Master Candidate in Industrial Engineer. Professor at the Universidad Distrital Francisco José de Caldas. Bogotá, Colombia.
Contact: *marielacastiblanco@gmail.com*

ABSTRACT

Our motivation focuses on answering a simple question: What is the minimum robotic structure necessary to solve a navigation problem? Our research deals with environments that are unknown, dynamic, and denied to sensors. In particular, the paper addresses problems concerning how to coordinate the navigation of multiple autonomous mobile robots without requiring system identification, geometric map building, localization or state estimation. The proposed navigation algorithm uses the gradient of the environment to set the navigation control. This gradient is continuously modified by all the robots in the form of local communication. The

design scheme, both for the algorithm and for its implementation on robots, searches for a minimal approximation, in which it minimizes the requirements of the robot (processing power, communication and kind of sensors). Besides, our research finds autonomous navigation for each robot, and also scales the system to any number of agents. The navigation algorithm is formulated for a grouping task, where the robots form autonomous groups without any external interaction or prior information of the environment or information from other robots. Finally, task performance is verified through simulation for the laboratory prototypes of the group.

RESUMEN

La motivación de la investigación se enfoca en responder una simple pregunta: ¿Cuál es la mínima estructura robótica necesaria para resolver un problema de navegación? La investigación se centra en ambientes desconocidos, dinámicos y limitantes al uso de ciertos sensores. En particular, se investiga el problema de cómo coordinar la navegación de múltiples robots autónomos sin el uso de la identificación del sistema, construcción de un mapa geométrico, localización o estimación de estados. El algoritmo de navegación propuesto utiliza el gradiente del ambiente para ajustar el control de navegación, gradiente que es continuamente modificado por todos los robots en la forma de una comunicación local. El esquema de diseño, tanto del algoritmo como de su implementación en el robot, busca una aproximación minimalista, en la que se minimizan los requerimientos del robot (capacidad de procesamiento, de comunicación y tipo de sensores). Además, la investigación busca la navegación autónoma de cada robot, y el escalado del sistema a cualquier número de agentes. El algoritmo de navegación es formulado para una tarea de agrupamiento, donde los robots forman grupos autónomamente sin interacción externa o información inicial respecto al ambiente u otros robots en él. Finalmente, el desempeño de la tarea es verificado a través de simulación para los prototipos de laboratorio del grupo.

* * *

1. INTRODUCTION

The nature has countless examples of complex systems that exhibit emergent behaviors that are able to solve difficult problems for current robotics. From the perspective of the interaction of the elements of a system with complex behavior, it is common that the observer assumes that is the result of coordination governed by a central controller. This is called by A. Clark [1] as *centralized thinking*. However, the complex behaviors can emerge from the collective activity of systems with simple behavior based on simple rules; this is what is known as emerging direct behavior [2].

We can then describe an emergent behavior as the unscheduled conduct of systems that meet basic rules, systems in which there are uncontrolled variables that lead to behavior, possibly unexpected by the observer. No centralized control systems as these allow us to expand the number of basic components of a system, and in turn the number of iterations required to achieve the desired objectives in the development of a task. These properties are well suited to solve repetitive tasks in which are required random groupings, whose priority is to form a group with similar nearby. These behaviors are especially useful in exploration tasks in unknown and dynamic environments, with the last task of grouped together to facilitate the process of physical and data collection, as well as initial solutions for data mining.

investigación

In particular, our research focuses on analyzing the collective navigation problem in environments that limit the use of sensors, such as collapsed environments in a natural disaster. Guiding questions of our approach are: A group of robots can autonomously navigate an unknown dynamic environment without collecting information about him? What is the minimum information necessary for an agent to solve the problem of navigation? Seeking answers to these questions, we are investigating different navigation strategies, in which we reduce as much as possible the collection of information by the agent, and consequently, the requirements of information processing and communication.

In this sense, we propose a collective navigation strategy for swarm robots in which robots are oriented in the environment according to the signal that they emit. The intensity of this signal allows the robot to locate their mates, and therefore grouped together.

Our ideas are also closely linked to research in behavior-based control systems for autonomous robots [3] - [7]. In this control scheme, the designer works the group of robots as a system, where each node corresponds to a robot, and to solve the task, each node has a different behavior. This structure is called a hybrid automaton, and allows use in the system design the capacity for abstraction of hybrid systems [1], [4], [6], [8] - [11]. We hope that our control ideas may be used as an alternative to the use of state-feedback control laws within continuous regions.

The paper is organized as follows. Section 2 presents preliminary concepts and problem formulation. Section 3 shows the sensor-based planning algorithm, definition, formulation and structure. Section 4 we present the prototype used in laboratory tests, and simulation results. And finally, in Section 5, we present our conclusions.

2. PROBLEM FORMULATION

Let $\mathbf{R}^2$ be the closure of a contractible open set in the plane that has a connected open interior with obstacles that represent inaccessible regions. Let $\mathbf{O}$ be a set of obstacles, in which each $\mathbf{O}$ is closed with a connected piecewise-analytic boundary that is finite in length. Furthermore, the obstacles in $\mathbf{O}$ are pairwise-disjoint and countably finite in number. Let be the free space in the environment, which is the open subset of W with the obstacles removed.

Let us assume a set of n agents in this free space. The agents know the environment E in which they move from observations, using sensors. These observations allow them to build an information space I. An information mapping is of the form:

$$q\colon E \to S \tag{1}$$

Where S denotes an *observation space*, constructed from sensor readings over time, i.e., through an observation history of the form:

$$\tilde{o}\colon [0,t] \to S \tag{2}$$

The interpretation of this information space, i.e., $I \times S \to I$, is that which allows the agent to make decisions [12].

We consider a group of n agents, differential robots, whose kinematics is defined by:

$$\dot{z}_i = u_i$$
$$z_i = (x_i, y_i, \theta_i) \tag{3}$$

with:

$$x_i = v_i \cos \theta_i \tag{4}$$

$$\theta_i$$

where $\mathbf{R}^2$ is the position of the ith robot, θ_i is its heading, and v_i and θ_i are the controlled transla-

tional and rotational velocities, respectively. $\mathbf{u}_i$ denote the agent's control input.

We assume the agents are able to sense the proximity of their teammates and/or obstacles within the environment, using minimal information. Therefore, the neighborhood of $\mathbf{z}_i$ is given by the range and field of view of the sensing hardware.

All robots broadcast a signal, which is modeled as an intensity function over $\mathbf{R}^2$. Let m denote the signal mapping m: $\mathbf{R}^2 \rightarrow [0, 1]$, in which $m(p)$ yields the intensity at $p \in E$. We want to allow intensity functions that are as complicated as those measured in practice from radio signals or other physical sources. Moreover, since all the robots broadcast their own signal while moving in the environment, the spectrum or map navigation is always dynamic.

The environment E and even the signal mapping m are unknown to the robot. Furthermore, the robot does not even know its own position and orientation.
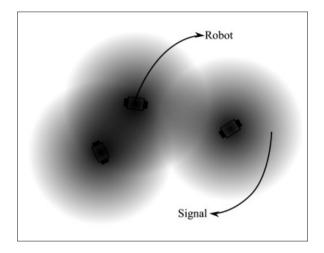
Our goal is to design the control rules for the n robots in order to independently solve navigation tasks in a dynamic and unknown environment.

3. METHODOLOGY

This section presents an algorithm for each one of the robots that guarantees that they reach the goal, i.e., they meet each other after applying a finite number of motion primitives, forming small groups of robots along the environment.

3.1 Control strategy

Our control strategy is supported by the intention to develop the simplest possible system, but able to collectively solve complex navigation tasks.

Figure 1. Robots and their signs. The dark areas represent greater intensity

Source: own work.

The main feature we want is robustness in real applications.

With this approach, we developed a strategy that not requires determining the coordinates of the robots, their mathematical models or a central control unit. Instead, we propose that the robots navigate according to the intensity of a signal in the environment, a sign that they themselves built.

In principle, all the robots in the environment transmit a given signal. These signals, which alter the environment, establish a local communication between robots. The signals are radiated, centered on each of the robots, and lose intensity with distance from the robot (figure 1). For analysis, we assume that all signals are identical, but the more general case allows some differences.

We consider two kinds of sensors. A first contact sensor, which reports when the robot is in contact with the environment boundary ∂E:

$$h_{CS_i}(x_i, y_i) = \begin{cases} 1 \text{ if } (x_i, y_i) \in \partial E \\ 0 \text{ otherwise} \end{cases} \qquad (5)$$

And the intensity sensor, which indicates the strength of the signal:

$$h_{CS_i}(x_i, y_i) = (x_i, y_i) = h(x_i, y_i, \theta_i, E, m) \quad (6)$$

The intensity values are normalized in the interval [0, 1], where 1 corresponds to the maximum intensity value. The robots must move to find these highs in the intensity map. Two intensity sensors in a robot are capable of providing information relating to the gradient of intensity map:
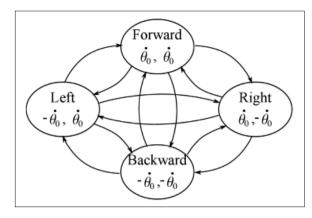
$$\nabla m_i = (p_2 - p_1) \quad (7)$$

Where p_1 and p_2 are the locations in $\mathbf{R}^2$ of the sensors on the robot i.

The robots do not require other types of sensors, such as global positioning, odometry, or a compass. Therefore, it is unable to obtain precise position or angular coordinates.

Our robots have no differential speed control on its wheels; this reduces the possible actions or motion primitives that are given to move the robot, simplifying the control and modeling. Furthermore, this allows some slack in the design of the robots; i.e., the scheme is robust regardless of variations in implementation (it is very difficult to build identical robots, and always keep them in that way). The robots are allowed only four motion primitives (figure 2):

- $u_{fwd} \rightarrow$ The robot goes straight forward in the direction it is facing. Both wheels rotate at their rated positive speed.

- $u_{bwd} \rightarrow$ The robot goes backward in the opposite direction it is facing. Both wheels rotate at their rated negative speed.

- $u_{rgt} \rightarrow$ The robot rotates clockwise, turns right. The left wheel runs at rated positive speed, and the right wheel runs at rated negative speed.

- $u_{lft} \rightarrow$ The robot rotates counterclockwise, turns left. The left wheel runs at rated nega-

Figure 2. State diagram corresponding to the robot's behavior from the point of view of movement of the motors

Source: own work.

tive speed, and the right wheel runs at rated positive speed.

3.2 Navigation plan

This section presents a navigation plan for each of the robots that guarantee that they will reach the goal, i.e., they will be grouped in small groups of robots in different points of environment after a finite number of motion primitives have been applied.

The intensity robot sensors provide information proportional to the gradient of intensity map. This information is sufficient to define the movement of the wheels of the robot, i.e., the motion primitive to be applied at any given time.

Let us consider our testing task: We want to group the robots at some arbitrary point in E. To solve this task, the robots must navigate with the steepest ascent of m. The equation 7 enables the robot to determine the intensity of m in the points p_1 and p_2 in which the sensors are located. Applying rotation primitives, the robot is able to align with the position that minimizes the difference. Then, using primitives for forward and backward, the

Table 1. Plan for the grouping task

1)	If $h_{CS} = 1$, => apply u_{bwd}, and then apply u_{lft} a random angle. Do $I_{pm} = 0$ (average intensity).	
2)	Determine $I_{p1} = h_{IS}(p_1)$ and $I_{p2} = h_{IS}(p_2)$.	
3)	Determine the orientation of m by calculating the difference of intensities $I_{p2} - I_{p1}$.	
4)	If	
	(a)	$(I_{p2} - I_{p1}) > 0$ =>Apply u_{rgt}. Go to step 2.
	(b)	$(I_{p2} - I_{p1}) < 0$ =>Apply u_{lft}. Go to step 2.
	(c)	$(I_{p2} - I_{p1}) \approx 0$ =>Go to step 5.
5)	Store the average intensity of the current position, I_{pm}.	
6)	Determine the direction of increased intensity calculating the difference of intensities stored $I_{pm}(t) - I_{pm}(t\text{-}1)$.	
7)	If	
	(a)	$(I_{pm}(t) - I_{pm}(t\text{-}1)) \geq 0$ =>Apply u_{fwd}. Go to step 1.
	(b)	$(I_{pm}(t) - I_{pm}(t\text{-}1)) < 0$ =>Apply u_{rgt} to rotate 180 degrees, then apply u_{fwd}. Go to step 1.

Source: own work.

robot can determine the direction in which the gradient of m increases. A possible navigation plan that solves this task is shown in table 1.
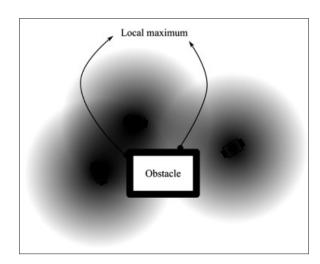
3.3 Convergence of the navigation plan

To establish the convergence of the navigation plan, we first show that following the proposed navigation plan under the described conditions, any robot can around obstacles in E following the gradient of intensity. And second, if the robot is able to surround the obstacles, then the robot will find the other robots near it in a finite number of steps.

***Lemma 1**:* For every obstacle boundary ∂O and every possible meeting point of the robots $p_j \in \mathbf{R}^2$ -**O** in E, there exist at least one intensity local maximum $p_0 \in \partial O$ in the range of the sensor of the agent for which the gradient of the signal constructed by each of the agents is disjoint from the interior of O (figure 3).

Proof: If we assume that under the conditions proposed in the navigation plan there are a finite number of local maxima along ∂O, for a meeting point $p_{j,t}$ at time t, one or more of these points

along ∂O can be a global maximum. Since the signal intensity increases with decreasing distance to $p_{j,t}$, the global maxima are points of ∂O that are close to $p_{j,t}$. If we call p to any of these points, and we call $D(p_{j,t}, p)$ the shortest distance from one of these points to $p_{j,t}$, then, by construction, no other points in O are closer to $p_{j,t}$ than p. This means that D and the interior of O are disjoint.

Figure 3. Local maxima at the obstacle boundary. The obstacle blocks the direct view of the robots to each other

Source: own work.

According to the above, if the agent follows the proposed navigation plan, the agent will always be able to find a maximum in the intensity gradient in ∂O, and therefore, he will be able to go around. This will be true all the time, regardless of the dynamics of change in the intensity gradient. Finally, if the agent is able to navigate around obstacles, the navigation plan ensures that after a finite number of motion primitives the agent will be at a point of greatest intensity, i.e., agents who are close be grouped around $p_{j,t}$.

4. EXPERIMENTAL PROTOTYPE AND SIMULATIONS

For our experiments, we use the Oomlout Company's open-source design SERB Robot. We modify the design to have a more robust bumper system using a similar geometry to the SERB robot chassis, and we added a group of microphones as audio sensor. In the spirit of minimalist design, our robot and all its peripherals, including the microphones, is handled by a single 8-bit microcontroller (Atmel ATmega328, 8-bit AVR RISC-based microcontroller).

We can control the translational and rotational speeds of the robot. However, we decided to design a software actuator on the displacement of the robot, a high-level control, which generates control states in order to facilitate the construction of state diagrams. In this way, each wheel has three choices of motion: clockwise, counterclockwise or without movement. Therefore, the robots can move forward, backward, turn on its axis in any direction or stand still. An important aspect of the design of the robot, which facilitated its modeling, is that the robot's center of rotation coincides with its geometric center.

The system is homogeneous, that is, all agents are identical in design. If a task requires more agents, we only have to place them in the environment.

Due to the impossibility of build tens of these robots, we only used the prototype robots to observe the basic behaviors of the algorithm. We've done the analysis of the dynamics of the system with a large number of robots through simulation.
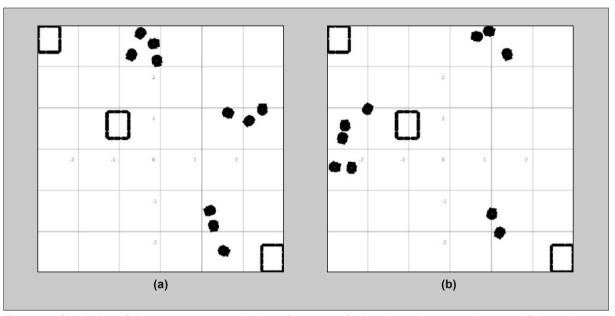
The simulations were performed on Player/Stage [13], [14], this platform allowed for the simulation run exactly the same code developed in C for the real robot. They consider not only the functionality of the algorithm, we also have them taken great care to the physical characteristics of the robot, i.e. shape, size (0.20 m x 0.22 m) and weight (0.530 kg). The microphones were installed just above its geometric center.

The simulation environment was designed in square shape with a total area of 6 m × 6 m = 36 m^2. Inside the environment, there are three holes inaccessible to the robots as an obstacle for navigation and for the sensors. In the simulations used a total of 10 agents, all completely independent of each other.

In figure 4 and figure 5 we show the results of four simulations. The four simulations were performed under the same conditions; the only difference between them was the initial position of robots (random initial location). In the first case (figure 4a), the robots formed groups in 1 min. and 48 s (time in the simulation). In the second case (figure 4b), the robots formed groups in 53 s. In the third case (figure 5a), the robots formed groups in 1 min. and 26 s, and the fourth case (figure 5b), the robots formed groups in 37 s.
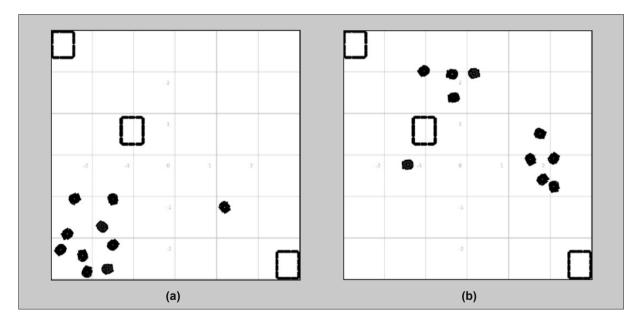
5. CONCLUSIONS

We have shown that a group of robots can be used to solve simple navigation tasks (grouping) without requiring system identification, geometric map building, localization, or state estimation, using a simple minimalist design in

Figure 4. Simulation of the autonomous navigation of a group of robots in unknown environment. Robots learn to navigate autonomously using local information, and after a while get together in small groups

Note: (a) First case, the robots are grouped after 1 min. and 48 s. (b) Second case, the robots are grouped after 53 s.

Source: own work.

Figure 5. Simulation of the autonomous navigation of a group of robots in unknown environment. Robots learn to navigate autonomously using local information, and after a while get together in small groups

Note: (a) First case, the robots are grouped after 1 min. and 48 s. (b) Second case, the robots are grouped after 53 s.

Source: own work.

hardware, software and algorithm operation. We formulate the algorithm, the navigation plan to be followed by the agent, and analyze the convergence of the strategy. The interaction between the robots responds to the simplified algorithm of local communication, which guarantees a minimum sensing (agent with limited sensing and/or environment that limits the use of sensors), what makes the strategy promissory for exploration in collapsed and unknown environments. The capabilities and system functions can be scaled without problems changing the number of robots. The grouping task is performed by robots regardless if one or more of the agents are damaged. In general, self-reconfiguration becomes very important when the system needs to be adaptive. The research raises interesting questions, such

as the time required for the development of the task. These are problems that are currently under analysis.

6. ACKNOWLEDGEMENT

REFERENCIAS

[1] Clark, A. *Being There: Putting Brain, Body, and World Together Again*. Bradford Book, first edition, January 1998.

[2] Resnick, M. *Turtles, Termites and Traffic Jams: Explorations in Massively Parallel Microworlds*. Bradford Book, January 1997.

[3] Zhang, H. and James, M.R. Bounded robust control for hybrid systems. In *Proc. 45th IEEE Conf. Decision and Control*, pp. 4801-4806, 2006.

[4] Egerstedt, M. and Hu, X. A hybrid control approach to action coordination for mobile robots. *Automatica*, vol. 38, no. 1, pp. 125-130, 2002.

[5] Chaohong, C. Rafal, G. Sanfelice, R.G. and Teel, A. R. Hybrid dynamical systems:

Robust stability and control. In *Proc. Chinese Control Conf. CCC 2007*, pp. 29-36, 2007.

[6] Gamage, C. Mann, G. and Gosine, R.G. A hybrid control strategy for multiple mobile robots with nonholonomic constraints. In *Proc. Canadian Conf. Electrical and Computer Engineering CCECE '09*, pp. 660-663, 2009.

[7] Lan, Y. Yan, G. and Lin, Z. A hybrid control approach to multi-robot coordinated path following. In *Proc. 48th IEEE Conf. Held jointly with the 2009 28th Chinese Control Conf Decision and Control CDC/CCC 2009*, pp. 3032-3037, 2009.

[8] Hernández - Martínez, E.G. and Albino, J.M.F. Hybrid architecture of multi-robot systems based on formation control and

som neural networks. In *Proc. IEEE Int Control Applications (CCA) Conf*, pp. 941-946, 2011.

[9] Lan, Y. Yan, G. and Lin, Z. A hybrid control approach to multi-robot coordinated path following. In *Proc. 48th IEEE Conf. Held jointly with the 2009 28th Chinese Control Conf Decision and Control CDC/CCC 2009*, pp. 3032-3037, 2009.

[10] Chio, T. S. and Tarn, T. J.. Rules and control strategies of multi-robot team moving in hierarchical formation. In *Proc. IEEE Int. Conf. Robotics and Automation ICRA '03*, vol. 2, pp. 2701-2706, 2003.

[11] Prieur, C. Goebel, R. and Teel, A.R. Hybrid feedback control and robust stabilization of nonlinear systems. In *IEEE Transactions on Automatic Control*, vol. 52, no. 11, pp. 2103-2117, 2007.

[12] LaValle, S.M. *Planning Algorithms*. Cambridge, U.K.: Cambridge University Press, 2006. [Online]. Available: http://planning.cs.uiuc.edu/.

[13] Vaughan, R. Massively multi-robot simulation in stage, *Swarm Intelligence*, vol. 2, no. 2, pp. 189-208, December 2008.

[14] Gerkey, B. Vaughan, R. and Howard, A. The player/stage project: Tools for multi-robot and distributed sensor systems. In *Proceedings IEEE 11th International Conference on Advanced Robotics ICAR'03*, pp. 317-323, Coimbra (Portugal), June 2003.