International Journal of Combinatorial Optimization Problems and Informatics, 13(3), Sep-Dec 2022, 87-97. ISSN: 2007-1558.

www.editada.org

Bio-Inspired Algorithm for Coordinated Control in Mobile Devices

Ocotlán Díaz-Parra[1], José G. Benítez-Morales[1], Julio C. Ramos-Fernández[1], Eduardo Hernández-Huerta[1], Francisco R. Trejo-Macotela[1], Juan M. Xicoténcatl Pérez[1], Miguel A. Ruiz-Jaimes[2]

[1] Universidad Politécnica de Pachuca, México.

[2] Universidad Politécnica del Estado de Morelos

Corresponding Author. ocotlan@diazparra.net

Abstract. The technological progress that has been achieved today, has a tendency in the application of automated mechanisms to improve the life of the human being. In the area of communications there are great advances in different applications such as, for example, being able to monitor from a mobile device event that are happening remotely at a remote point, or, for example, to monitor the mechanized control of taking medication to patients in a hospital. The objective of this research work is to be able to establish communication with mobile devices to execute instructions issued by an intelligent logic system. This system is built by means of a bio-inspired algorithm ACO (Ant Colony Optimization) that has the purpose of generating logical information and three differential type LBOOT mobile devices.

Keywords: **A**utomated mechanisms, bio-inspired algorithm, mobile device coordination.

Article Info
Received Dec 26, 2021
Accepted Aug 16, 2022

1 Introduction

Bio-inspired algorithms are conceptualized that way because they mimic the behavior of living beings. The ACO algorithm imitates the behavior of ants to find a food source. A scout ant goes out in search of a food source and on its way leaves a pheromone trail so that the next scout ant can follow this trail, which also leaves a pheromone trail on its way, in such a way that when several ants pass along this path the pheromone trail becomes more intense, making it easier for the other ants to follow the route. The trajectory of the food source from the exit point which is the anthill to the point where the food is located is easily found by the collecting ants because they are guided only by the pheromone trail, thus the system is completed, the ant fulfilled the objective of bringing food to the anthill. The purpose of using the ACO algorithm in this research is to generate the logical information that will be transmitted to the mobile devices. The objective is to generate an optimized logical path that can be traversed by several mobile devices coordinated with each other from an initial point to an end point.

The mobile devices used in this research, are differential type mobile robots, which are characterized because the movement is achieved on a base of two wheels, which are coupled to its own engine, to generate the forward movement both engines must be synchronized in speed. If both motors are synchronized at the same speed the motion generated by the robot will be in a straight line, if they have different speeds the motion effect achieved is in different directions and if each motor has speeds with equal magnitude, but with opposite directions then the motion achieved is to rotate on its own axis [15]. The type of mobile robot used in the experimentation is a ground type differential drive mobile robot whose motion control is characterized in four fundamental tasks: localization, path planning, path following and obstacle avoidance. To achieve the movement this type of robot is based on kinematics models which relate velocity, position and orientation. For trajectory control the equations can be represented in Cartesian coordinates and polar coordinates, based on control theories and laws.
This type of mobile devices can be applied in logistics and distribution systems.

1.1 Description of the problem

Traditional distribution systems by air, rail, sea and land are affected by weather conditions, putting at risk human lives, material losses and losses of customers due to poor service. One of the options to avoid this type of problem is the use of automated mechanisms coordinated by intelligent logic systems in logistics and product distribution systems. The use of unmanned distribution systems can reduce the risk of loss of human lives. Communication with unmanned devices basically consists of establishing wireless contact so that the control mechanism can perform remote manipulations of the device. In this research the main purpose is to use the ACO algorithm as an intelligent logic system that will allow generating the optimized route information, which through a communication system will replicate the information to three mobile devices called LBoot, to execute the coordinated movement of route tracking in simulation to the ants. The coordination of the mobile devices controlled by means of an intelligent logic system can be applied in various delivery systems with autonomous vehicles such as fast-food distribution system, perishable products distribution system, distribution system of medicines sensitive to environmental conditions, distribution system of hazardous materials among others.

1.2 State of the art

The literature search conducted for this research was on works related to the use of the ACO algorithm and differential type mobile devices. The ACO algorithm has been used in research to solve optimization problems Dorzán et al. used it to solve geometric problems [1], Barcos et al. used ACO to solve transportation problems with multiple origins and multiple destinations [2], Minelli Sierra G. in 2019, applied ACO for the problem of community detection in social networks [3], Berchtold et al. applied ACO to solve the scheduling problem in which classroom scheduling is per-formed [4], Velandia Mahecha applied ACO to solve task assignment problems in a company [5], Palacios et al. used ACO to establish speed control in a DC motor using MATLAB to combine ACO tuning with the controller structure [6], García Londoño and Ramírez Vélez used a hybrid ACO algorithm with genetic to solve the hospital waste collection problem [7], Márquez Cortes et al. used the ACO algorithm to generate solid waste collection routes in large cities [8], Espejo Gutiérrez and Villena Soto used ACO for a pneumatic pressure control system [9], Sanchez Solis et al. used ACO to solve the problem of picking order products inside a warehouse [10], Aguilar et al. used ACO for emergent system attention in video game adaptation systems applied in an educational context [11], Hidalgo Boix applied ACO for the optimization of bicycle distribution routes in BiciMAD stations [12], Mera and Cáceres applied ACO for frequency allocation in GSM mobile networks [13], Tomé Martín has used ACO to solve industrial logistics problems by optimizing the solutions [14].

The study of the art reveals that ACO is an algorithm used to generate solutions to different problems related to route optimization, to detect communities in social networks, to assign activities, for scheduling, for speed control among others. The ACO algorithm in this research is to generate the logical information that will be transmitted to LBoot differential type mobile devices.

The mobile devices used for this research are differential type mobile robot, this type of devices is generally used to perform line tracking. For example, in the work done by Hernandez Millan et al. designed a position and motion controller for a differential robot for which they used neural networks in order that the robot could track explicit paths such as rectilinear and curvilinear trajectories [16].

The main developments of differential type robots are oriented to achieve vehicle autonomy, either in an indoor or outdoor environment considering a null, partial or total knowledge of the navigation space. Therefore, a fundamental part to achieve this is the design of position control or trajectory tracking strategies. Generally, these control techniques are based on the kinematic or dynamic model, in which they consider the position of the vehicle with respect to the "x", "y" axes, together with the vehicle orientation [17-22]; in addition, some authors include the kinematics of the wheels [22]. Another variant of the kinematic model is presented by [23], which considers as variables the distance between the robot and the desired point, the orientation of the vehicle with respect to an inertial reference frame, and the angle between the robot orientation and the desired reference. As for the control techniques, researchers have designed different control strategies such as PID control [24], feedback linearization control [25], the backstepping-based control [26], the adaptive control [27], the sliding mode control [28], the neural-network-based control [29], the fuzzy PID controller [30]. Recent scientific works have sought cooperative work between several robots through synchronization techniques [31] and [32] with the aim of establishing formations during the performance of such tasks.

Silvestre and Roberto propose a dynamic trajectory tracking method for the development of the control law for a differential type of robot [33], Castellanos Perez realized the cooperation between two mobile robots with differential traction by designing two algorithms, one for open loop control and the other for closed loop control [34]. In the work Marín Paniagua proposed a navigation method with differential configuration based on sensory fusion of several inertial sensors using a mobile robot in differential configuration [35], Tibaduiza Burgos et al. proposed a path approach for a differential type of mobile robot and the generation of trajectories using two algorithms a genetic algorithm and another by potential fields [36].

From the literature review we can say that differential type mobile robots have been very attractive in the area of robotics and control due to several factors, among the most important are the simplicity in its locomotion, the versatility of its maneuverability and diversity of applications, among which stand out: warehouse automation, border line supervision, bomb disposal, cleaning of areas with hazardous waste, cleaning of homes, applications in agriculture, logistics and distribution systems, among other activities.

2 Experimental procedures

The elements of the ACO algorithm for implementation are the variables, the number of iterations, the total number of ants, the node positions, the edges, the visibility weight, the pheromone weight and the evaporation constant.

The analogy between ACO elements and mobile device elements is presented in Table 1:

Table 1. Analogy between ACO elements and mobile devices.

ACO	LBoot
Ants	Mobile device
Pheromone	Coordinates of the path
Initial pheromone	Coordinate of the initial position
Evaporation constant	Time in which the information is stored
Total number of nodes	Total positions through which it can pass in the mobile device.
Visibility	Distance between each node.

The representation of the trajectory cycle in ACO starts with the random selection of an ant that, guided by the pheromone, moves along the indicated trajectory; the exploration trajectories are selected randomly. At the end of the total trajectory traveled by each ant, the value of the local pheromone is calculated, which represents the number of coordinates visited by each ant, and the distance of the route traveled by each ant is calculated, so that at the end of the last ant's journey, a comparison is made between the different trajectories generated to detect which of the routes represents the shortest path. If it is true, the global pheromone value is calculated, which is the sum of the local pheromones. The logical representation is presented graphically in Figure 1.

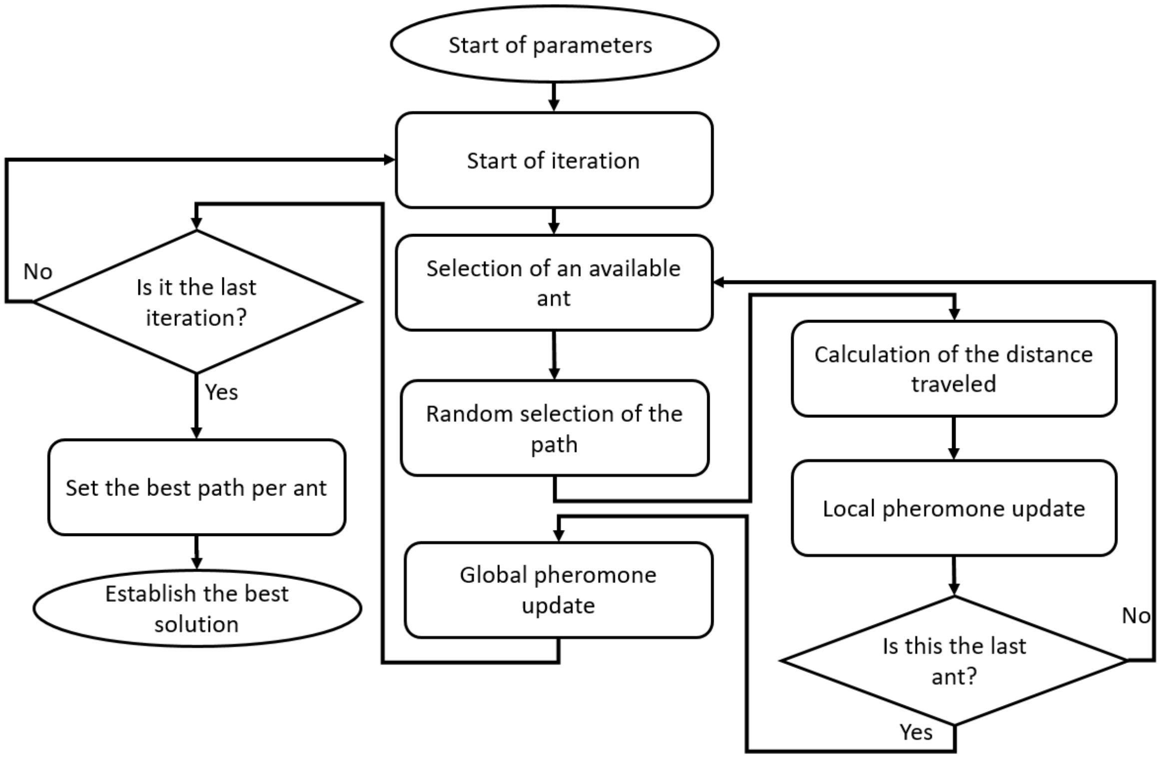

Fig. 1. ACO algorithm flow chart.

The mathematical model for the route optimization problem with mobile devices is a distance minimization model represented by:

$$Min\sum_{i=1}^{m}\sum_{j=1}^{r} x_{C_{ij}} t_{C_{ij}} \tag{1}$$

Where C represents each mobile m device, represents the total number of mobile devices i, j, represents the valid coordinates belonging to the nodes in the route. A route is represented by: $(i, j) \in V = \{(1,3),(1,1),(1,5),...(\mathrm{m},\mathrm{r})\}$.

The variable $x_{C_{ij}}$ represents the distance traveled by each mobile device and the variable $t_{C_{ij}}$ represents the time traveled by the mobile device.

Constraints: $a_{lrcC} \leq A_{tp}$ Where a_{lrcC} represents the limiting area of travel through the device C , which must be less than the area of the experimental platform A_{tp} .

$$\sum_{C=1}^{n} x_{C_{ij}} = 1 \quad \forall i, j \in \mathbb{R}\{(\mathrm{i},\mathrm{j})\} \tag{2}$$

Equation two represents another constraint that must be met: that all coordinates in the routes must be visited by all mobile devices involved in the instance.

$$\sum_{C=1}^{n} x_{C0,0} = 0 \qquad \forall i, j = 0 \tag{3}$$

Equation three represents that the devices are at the start coordinate since in order to perform the path it must be ensured that the mobile devices involved must start from the start coordinate.

The mobile devices used in the experimentation are differential type mobile robots called LBoot and the Robotic Operating System (ROS). The structure of the LBoot is presented in Figure 2.

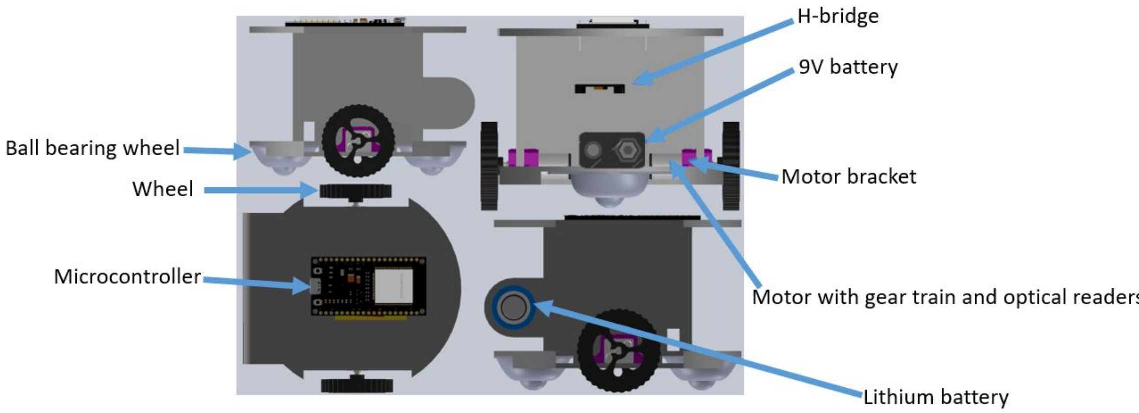

Fig. 2. LBoot Design.

Figure 3 shows the actual dimensions of the LBoot device which are 13.0cm long 10.0 cm wide with a weight of 334 grams.

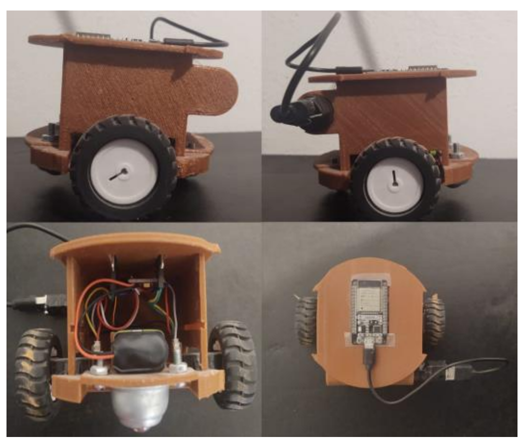

Fig. 3. Real LBoot model.

Three self-developed LBoot mobile devices were used on an experimental platform in the LANAVEX laboratory.

The communication between the devices is sent via USB serial from the computer to the "ESP32" microcontroller using heterogeneously distributed interconnection points which allows the link between several devices to the network.

The microcontroller can send or receive information via wi-fi by identifying the unique network adapter address (MAC address). The communication is of the "Esp-Now" type whose characteristic is that it does not have a direct client or server. The coordinates of the robot are obtained by means of optical encoders which have two dual-channel Hall effect sensors and two six-pole magnetic disks to adapt to the gearmotors with extended shaft, these detect the rotation of the motor shaft on the edge of both channels, which allows to know the direction of the motor.

With the information provided by the optical encoders, the rotation of the magnetic disks of the encoder and the calculation of the motor speed the position of the device can be obtained. Mathematically the position of the robot is represented by the equations (4), (5) and (6), where the angles of the tires are represented by ϕ, the radius of the tires is represented by r and l is the distance that exists between each wheel.

$$X = \frac{(\phi_r + \phi_l)\,\mathrm{r}}{2}\cos(\theta) \tag{4}$$

$$Y = \frac{(\phi_r + \phi_l)\,\mathrm{r}}{2}\sin(\theta) \tag{5}$$

$$\theta = \frac{(\phi_r + \phi_l)\,\mathrm{r}}{2d} \tag{6}$$

The test instances used in the experimentation were of size seven, twenty-five and fifty nodes, described in tables 2, 3 and 4.

The name of each instance is composed of relevant information: the first character describes the type of platform which can be real (R) or simulated (S), the next character corresponds to the total number of nodes, and the last character corresponds to the type of flota used in the experimentation which can be homogeneous (H) where all devices are the same or heterogeneous (He) where all devices are different. For example, R7H is the actual instance of seven nodes with homogeneous mobile devices. The mobile devices identified in the instance are differential type mobile robot (RMTD).

Table 2. Definition of the R7H instance.

Name:			R7H		
Number of nodes: 7	Fleet:		Type of mobile device	Number of mobile devices	
	Homogeneous	Heterogeneous			
(x, y)	yes		RMTD	3	
Coordinates					
(0,0)	(30,90)	(60,-60)	(50,0)	(100,60)	(120,-90)
(140,30)					

Table 3. Definition of the S25H instance.

Name:			S25H		
Number of nodes: 25	Fleet:		Type of mobile device	Number of mobile devices	
	Homogeneous	Heterogeneous			
(x, y)	yes		RMTD	3	
Coordinates					
(0,0)	(1,0)	(2,1.5)	(2,-2)	(2.5,-0.5)	(3.5,2)
(3.5,1)	(3.5,-1.5)	(3.5,-3)	(4.5,-0.5)	(5,-2)	(5,1)
(5,2.5)	(8,2)	(7.5,1)	(6,1.5)	(6.5,0)	(6.5,-1.5)
(5.5,-3.5)	(7,-2.5)	(8,-0.5)	(8.5,-2)	(9,1)	(9.5,-1)
(10.5,0)					

Table 4. Definition of the S50H instance.

Name:			S50H		
Number of nodes: 50	Fleet:		Type of mobile device	Number of mobile devices	
	Homogeneous	Heterogeneous			
(x, y) Coordinates	yes		RMTD	3	
(0,0)	(0.5,2.5)	(1,1)	(1,-1.5)	(0.5,-3.5)	(2,3)
(2.5,2)	(3,1)	(2.5,-1)	(2,-2)	(2.5,-4)	(4,2)
(4,-0.5)	(3,-2)	(4,-3.5)	(5,3)	(5.5,1)	(6,-1)
(5,-2.5)	(6,-3.5)	(7,3.5)	(7,2)	(8,1)	(8,-0.5)
(7.5,-1.5)	(7,-2.5)	(8.5,2.5)	(9.5,1.5)	(10,-0.5)	(9,-2)
(8.5,-3.5)	(10.5,-2)	(10.5,-3.5)	(10,3)	(11,2)	(11.5,0.5)
(12,-1)	(11.5,-2.5)	(12,3)	(12.5,2)	(13,0.5)	(13.5,-1.5)
(13,-2)	(12.5,-3.5)	(14,1.5)	(15,1)	(14.5,0)	(15,-2)
(14,-3)	(16,0)				

3 Results

Experimentation with the R7H instance was carried out in the LANAVEX laboratory with the real platform as shown in Figure 4. Obtaining a route optimized by the ACO algorithm for the R7H instance, whose total distance traveled is 178.1 centimeters, with an execution time of 30.3 seconds.

The set of nodes involved in the route is N= {(0,0), (50,0), (100,60), (140,30)}, the information generated by the intelligent logic system was transmitted to the first LBoot to execute the route, the first LBoot performed the movement and transmitted movement information to the second LBoot to calculate the position of the movement and in turn the second LBoot transmitted its movement information to the third LBoot to calculate the position to track the route of the second LBoot.

The coordination of the LBoots is by means of an ordered trajectory tracking since the LBoot that is executing the path is able to calculate the position of the following one. For the calculation of the position, the difference of paths is used, where the coordinates are subtracted a distance so that they can move properly and simulate the movement of the ants in simulation to the pheromone trail.

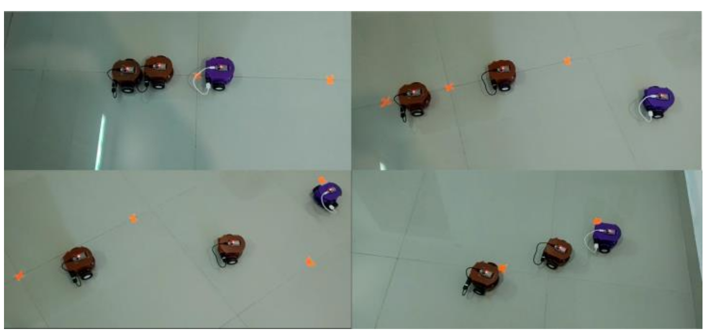

Fig. 4. Actual Instance Result RH7 LANAVEX.

Figure 5 shows an illustration of the positions in which each LBoot device calculates the position of the LBoot that precedes it to perform the movement.

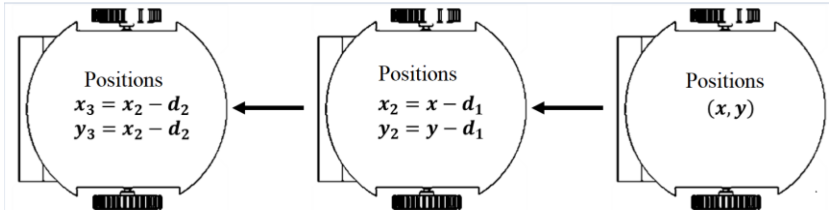

Fig. 5. LBoot positions illustration.

The coordination starts when the main LBoot receives the coordinates (x, y) of the workspace, these coordinates are transformed into coordinates of the next LBoot (x_n, y_n), where **n** represents the number of robots. The coordinates are calculated by means of the difference of distances.

The experimentation performed with the S25H instance was carried out in a simulated way with a homogeneous fleet obtaining a total path distance of 11.88 centimeters and an execution time of 12.39 seconds. The set of nodes involved in the route is N= {(0,0), (1,0), (2.5, -0.5), (3.5,1), (5,1), (6,1.5), (7.5,1), (9,1), (10.5,0)}. For an instance of 25 nodes the algorithm properly executed the solution (see figure 6).

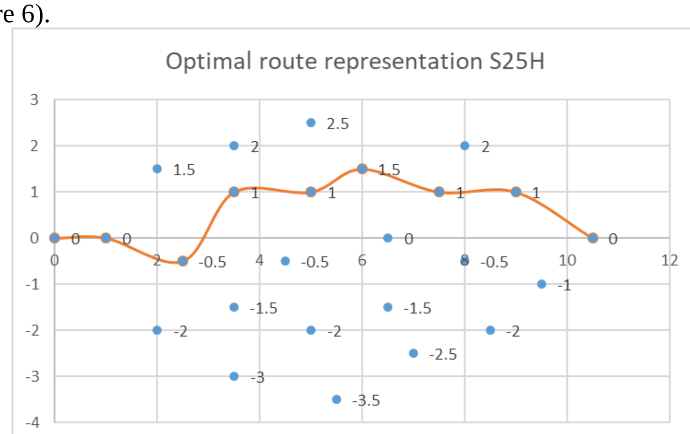

Fig. 6. Optimal route result S25H.

The experimentation using the S50H instance generated a path N= {(0,0), (0.5,2.5), (2.5,2), (4,2), (5.5,1), (8,1), (10, -0.5), (12, -1), (13.5, -1.5), (15, -2), (16,0)} with a simulation time of 17 seconds with an optimized path distance of 20.33 cm. as shown in Figure 7.

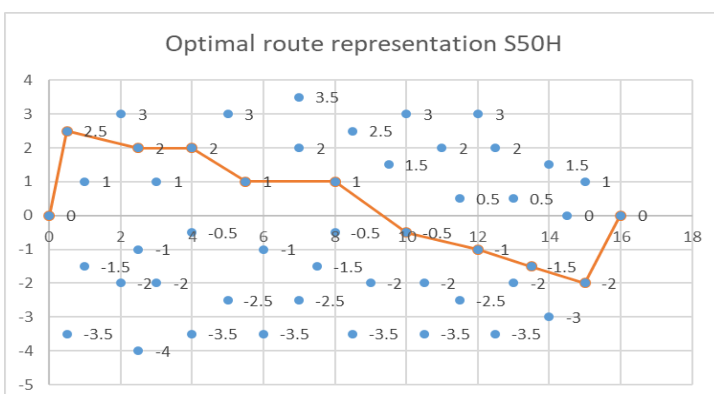

Fig. 7. Optimal route result S50H.

.

The summary of results obtained by entering as data the simulated instances and the real instance are presented in Table 5.

Table 5. Summary of the results obtained from the ACO algorithm.

Instance	Nodes	Time	Executions number	Optimal route	Distance
RH7	7	30.3	30	(0,0), (50,0), (100,60), (140,30)	178.1
S25H	25	12.39	30	(0,0), (1,0), (2.5,-0.5), (3.5,1), (5,1), (6,1.5), (7.5,1), (9,1), (10.5,0)	11.88
S50H	50	17	30	(0,0), (0.5,2.5), (2.5,2), (4,2), (5.5,1), (8,1), (10,-0.5), (12,-1), (13.5,-1.5), (15,-2), (16,0)	20.33

As can be seen in Table 5, the time to find the optimal route in the real instance R7H was higher than the times of the simulated instances, this is because in the R7H instance the times in which the Lboot devices performed the optimal route were considered as shown in Figure 4.

For the real instance R7H, ten instances were run, with thirty repetitions each. The R7H instances are described in Table 6.

Table 6. R7H instances considered in the experimentation.

Instance	Coordinates
R7H-1	(0,0), (30,90), (60,-60), (50,0), (100,60), (120,-90), (140,30)
R7H-2	(0,0), (20,70), (60,-60), (50,0), (100,60), (120-90), (140,30)
R7H-3	(0,0), (20,60), (50,-60), (50,0), (100,60), (120-90), (140,30)
R7H-4	(0,0), (20,70), (60,-60), (50,0), (90,60), (120-90), (140,30)
R7H-5	(0,0), (20,70), (60,-60), (80,0), (100,60), (120-90), (140,30)
R7H-6	(0,0), (20,70), (60,-60), (50,0), (110,60), (130-90), (140,30)
R7H-7	(0,0), (20,10), (60,-60), (80,0), (100,60), (120-90), (140,30)
R7H-8	(0,0), (20,40), (60,-60), (80,0), (100,60), (110-90), (140,30)
R7H-9	(0,0), (20,70), (60,-60), (50,0), (110,50), (120-90), (140,30)
R7H-10	(0,0), (30,90), (60,-60), (50,0), (100,60), (120,-90), (140,30)

The tools used for programming were: MatLab software, on a computer with the following characteristics: 24GB of RAM memory, 512GB solid state hard disk, 1TB hard disk, 6GB GTX1060 graphics card, a Windows 10 Home Single Language operating system, and an 8th generation Intel Core i7 microprocessor.

The mobile devices used in the experimentation were designed based on the model of a differential type robot with the elements: an ESp32 card, TB661 h-bridge, N20 motors, lithium battery, 9v battery and 3D printing for the casing, which were considered to be the necessary features for the purpose of this research, since each mobile device must be able to communicate, receive message and execute trajectory tracking autonomously.

4 Conclusions

The results obtained in the experiment show that the intelligent logic system generated the best route considering as input the three types of instances: R7H, S25H and S50H. The information obtained exclusively for the R7H instance served as input to the three mobile devices called LBoot, which executed the coordinated route following movement. This coordination mechanism between mobile devices controlled by means of an intelligent logic system can be used to coordinate delivery systems with autonomous vehicles such as fast-food distribution system, perishable goods distribution system, environmentally sensitive drug distribution system, hazardous materials distribution system, among others.

ACKNOWLEDGMENT

The authors are grateful to CONACYT (National Council of Science and Technology) for its support. We thank Laboratorio Nacional en Vehículos Autónomos y Exoesqueletos (LANAVEX).

References

1. Dorzán, M., Gagliardi, E., Leguizamón, M., Taranilla, M., & Penalver, G. H. (2009). Algoritmos aco aplicados a problemas geométricos de optimización. XIII Encuentros de Geometrıa Computacional.
2. Barcos, L., Rodríguez, V., Álvarez, M. J., & Robusté, F. (2002). Algoritmo basado en la optimización mediante colonias de hormigas para la resolución del problema del transporte de carga desde varios orígenes a varios destinos. Santander. España.
3. Minelli Sierra, G. (2019). Desarrollo de un algoritmo ACO para el problema de la detección de comunidades en redes sociales (Bachelor's thesis).
4. Berchtold, A., Ruiz, D., & Gómez, W. (2019). Optimización de asignación de salones de clase: un enfoque con el algoritmo Ant Colony Optimization (ACO).
5. Velandia Mahecha, R. C. (2019). Solución al problema de asignación y atención de requerimientos tecnológicos para la empresa VGG Aplicaciones SAS, usando un modelo de asignación de tareas.
6. Palacios, S., Chiriboga, S., & Montalvo, W. (2020). Controlador de velocidad PID-2DOF-ACO para motor DC sobre plataforma ARM. Revista Ibérica de Sistemas e Tecnologias de Informação, (E30), 217-228.
7. García Londoño, G. A., & Ramírez Vélez, A. F. (2019). Desarrollo de un modelo hibrido entre la metaheuristica “colonia de hormigas” y el algoritmo genético modificado de chu-beasley aplicado a la recolección de desechos en los hospitales de Pereira.
8. Márquez Cortés, F. J., Rodríguez Vázquez, K., & Garro Licón, B. A. (2019). Diseño de rutas de recolección utilizando el algoritmo de optimización por colonia de hormigas.
9. Espejo Gutiérrez, R. S., & Villena Soto, G. J. Diseño e implementación de un sistema de control de presión neumática basado en algoritmo de colonia de hormigas (ACO).
10. Sánchez Solís, J. P., Rivera Zarate, G., García, V., Florencia, R., & Cisneros Pérez, M. S. (2020). Algoritmo de colonia de hormigas para abordar el problema de order picking. Instituto de Ingeniería y Tecnología.
11. Aguilar, J., Altamiranda, J., & Díaz, F. (2020). Sistema adaptativo para la generación de comportamientos emergentes en juegos serios emergentes. Computación y Sistemas, 24(3), 1029-1051.
12. Hidalgo Boix, A. (2020). Optimización de rutas de distribución de bicicletas entre las estaciones de BiciMAD aplicando el algoritmo de la colonia de hormigas.
13. Mera, E. R. Z., & Cáceres, A. Y. P. (2020). Asignación de frecuencias en redes móviles gsm utilizando meta heurística aco: Frequency assignment in gsm mobile networks using aco heuristic goal. Centro Sur, 4(1), 1-17.
14. Tomé Martín, V. (2020). Estudio y aplicación de métodos de optimización por colonia de hormigas en industria.
15. Gutiérrez Martínez, S. V. (2021). Control de robots móviles para el seguimiento de trayectorias (Doctoral dissertation, Universidad Autónoma de Nuevo León).
16. Hernández Millán, G., Ríos Gonzales, L. H., & Bueno López, M. (2016). Implementación de un controlador de posición y movimiento de un robot móvil diferencial. Tecnura, 20(48), 123-136.
17. Y. Kanayama, Y Kimura, F. Miyazaki and T. Noguchi, “A stable tracking control method for an autonomous mobile robot”, in Proceedings of IEEE International Conference on Robotics and Automation, pp384-389 vol .1, 1990.
18. A. Gholipour and M. Yazdanpanah, “Dynamic tracking control of nonholonomic mobile robot with reference adaptation for uncertain parameters”, in Proc. Of the European Control Conference, 2003.
19. P. Petrov, “Modeling and adaptive path control of a differential drive mobile robot” in Proceedings of the 12 th WSEAS International conference on automatic control, modelling and simulation, pp 403-408, 2010.
20. Divya Aneesh, “Tracking Controller of Mobile Robot”, 2012 International Conference on Computing, Electronics and Electrical Technologies, pp. 343-349, 2012.
21. Farhan A. Salem, “Dynamic and kinematic model and control for differential drive mobile robots”, International Journal of current engineering and technology. 2013.
22. R. Dhaouadi, A. A. Hatab, Dynamic modelling of differential-drive mobile robots using Lagrange and Newton-Euler Methodologies: A Unified Framework, Advances in robotics & automation, 2013.
23. Siegwart, I R. Nourbakhsh, Introduction to autonomous mobile robots, Ed. MIT Press, 2011.
24. J. Meng, A. Liu, Y. Yang, Z. Wu and Q. Xu, “Two-Wheeled Robot Platform Based on PID Control,” in International Conference on Information Science and Control Engineering (ICISCE), Zhengzhou, pp. 1011-1014, 2018.
25. N. V. Tinh, N. T. Linh, P. T. Cat, P. M. Tuan, M. N. Anh and N. P.T. Anh, “Modeling and feedback linearization control of a nonholonomic wheeled mobile robot with longitudinal, lateral slips,” in IEEE International Conference on Automation Science and Engineering (CASE), Fort Worth, TX, pp. 996-1001, 2016.
26. D. Diaz and R. Kelly, “On modeling and position tracking control of the generalized differential driven wheeled mobile robot,” in IEEE International Conference on Automatica (ICA-ACCA), Curico, pp. 1-6. 2016.
27. P. Petrov and V. Georgieva, “Adaptive Velocity Control for a Differential Drive Mobile Robot,” in International Symposium on Electrical Apparatus and Technologies (SIELA), Bourgas, pp.1-4, 2018.
28. F. G. R. Contreras, A. I. C. Lopez, L. Fridman and V. J. G. Villela, “Trajectory tracking using continuous sliding mode algorithms for differential drive robots,” in IEEE Annual Conference on Decision and control (CDC), Melbourne, Australia, pp. 6027-6032, 2017.
29. M. Yue, L. Wang, and T. Ma, “Neural Network Based Terminal Sliding Mode Control for WMRs Affected by an Augmented Ground Friction With Slippage Effect,” IEEE J. of Automatica, vol. 4, pp. 498 - 506, July. 2017.
30. J. Heikkinen, T. Minav and A. D. Stotckaia, "Self-tuning parameter fuzzy PID controller for autonomous differential drive mobile robot," 2017 XX IEEE International Conference on Soft Computing and Measurements (SCM), pp. 382-385, doi: 10.1109/SCM.2017.7970592, 2017.

31. Aguilar, Jose & Castro-Linares, Rafael & Duarte-Mermoud, Manuel. Robust Formation of Mobile Robots with Synchronization Using Fractional Order Sliding Modes. 1-6. 10.1109/CCE50788.2020.9299149. 2020.
32. Arteaga-Escamilla, C.M., Castro-Linares, R. & Álvarez-Gallegos, J. Synchronization Approach to Formation Control of Mobile Robots from the Cluster Space Perspective. J Intell Robot Syst 103, 56 https://doi.org/10.1007/s10846-021-01495-y, 2021.
33. Silvestre, I. R., & Roberto, E. (2011). Point-to-point control for trajectory tracking of a differential type wheeled mobile robot (Doctoral dissertation).
34. Castellanos Pérez, M. A. (2015). Development and application of cooperative algorithms in mobile robots (Doctoral dissertation).
35. Marin Paniagua, L. J. (2012). Navigation of a differential configuration mobile robot based on sensory fusion.
36. Tibaduiza Burgos, D. A., Barrero Pérez, J. G., & Martínez Ángel, R. (2011). Path and trajectory planning using genetic algorithms and potential fields for a mobile robot. Iteckne, 8(2).