



A new approach for dynamic fuzzy logic parameter tuning in Ant Colony Optimization and its application in fuzzy control of a mobile robot



Oscar Castillo*, Héctor Neyoy, José Soria, Patricia Melin, Fevrier Valdez

Tijuana Institute of Technology, Tijuana, Mexico

ARTICLE INFO

Article history:

Received 18 October 2012

Received in revised form

29 November 2014

Accepted 1 December 2014

Available online 10 December 2014

Keywords:

ACO

Fuzzy logic

Fuzzy control

Robotics

Fuzzy ACO

ABSTRACT

Ant Colony Optimization is a population-based meta-heuristic that exploits a form of past performance memory that is inspired by the foraging behavior of real ants. The behavior of the Ant Colony Optimization algorithm is highly dependent on the values defined for its parameters. Adaptation and parameter control are recurring themes in the field of bio-inspired optimization algorithms. The present paper explores a new fuzzy approach for diversity control in Ant Colony Optimization. The main idea is to avoid or slow down full convergence through the dynamic variation of a particular parameter. The performance of different variants of the Ant Colony Optimization algorithm is analyzed to choose one as the basis to the proposed approach. A convergence fuzzy logic controller with the objective of maintaining diversity at some level to avoid premature convergence is created. Encouraging results on several traveling salesman problem instances and its application to the design of fuzzy controllers, in particular the optimization of membership functions for a unicycle mobile robot trajectory control are presented with the proposed method.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Ant Colony Optimization (ACO) is inspired by the foraging behavior of ant colonies, and is aimed at solving discrete optimization problems [8].

The behavior of the ACO algorithm is highly dependent on the values defined for its parameters as these have an effect on its convergence. Usually these are kept static during the execution of the algorithm. Changing the parameters at runtime, at a given time or depending on the search progress may improve the performance of the algorithm [25–27].

Controlling the dynamics of convergence to maintain a balance between exploration and exploitation is critical for good performance in ACO. Early convergence leaves large sections of the search space unexplored. Slow convergence does not concentrate its attention on areas where good solutions are found.

Fuzzy control has emerged as one of the most active and fruitful areas of research in the application of fuzzy sets and fuzzy logic. The methodology of fuzzy logic controllers is useful when processes are too complex for analysis by conventional quantitative techniques

or when the available sources of information are interpreted in a qualitatively inaccurate or uncertain way [40].

Determining the correct parameters for the fuzzy logic controller is a complex problem and it is also a task that consumes considerable time. Because of their ability to solve complex NP hard problems we made use of ACO for the selection of those already mentioned parameters.

There is also some recent interest in using ACO algorithms in mobile robotics [5,28]. Nowadays robotic automation is an essential part in the manufacturing process. Autonomous navigation of mobile robots is a challenge. A mobile robot can be useful in unattainable goal situations due to geological conditions or where the human are being is endangered. So, mobile robotics is an interesting subject for science and engineering.

This paper explores a new method of diversity control in ACO. The main idea is to prevent or stop the total convergence through the dynamic adjustment of certain parameter of the algorithm applied to the design of fuzzy controllers, specifically to the optimization of membership functions of a trajectory controller for a unicycle mobile robot.

The rest of the paper is organized as follows. Section 2 presents an overview of ACO. Section 3 describes a performance analysis on several TSP instances. Section 4 presents a new method of parameter tuning using fuzzy logic, Section 5 shows some simulation

* Corresponding author. Tel.: +52 6646236318.

E-mail address: ocastillo@tectijuana.mx (O. Castillo).

results in TSP problems, Section 6 describes the optimized fuzzy controller, Section 7 presents the considerations that are used to implement the ACO algorithm in the optimization of membership functions, Section 8 describes how the proposed method is applied, Sections 9 and 10 show simulation results in the membership functions optimization problem, and finally Section 11 presents some conclusions.

2. Ant Colony Optimization

The first ACO algorithm was called Ant System (AS) and its main objective was to solve the traveling salesman problem (TSP), whose goal is to find the shortest route to link a number of cities. In each iteration each ant keeps adding components to build a complete solution, the next component to be added is chosen with respect to a probability that depends on two factors. The pheromone factor that reflects the past experience of the colony and the heuristic factor that evaluates the interest of selecting a component with respect to an objective function. Both factors weighted by the parameters α and β respectively define the probability P in (1)

$$P_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta}, \quad \text{if } j \in N_i^k \quad (1)$$

In (1) τ_{ij} represents the pheromone value between nodes i and j and η_{ij} represents the heuristic factor that evaluates the interest of selecting a component with respect to an objective function. Finally, N_i represents a neighborhood of node i .

After all ants have built their tours, the pheromone trails are updated. This is done by decreasing the pheromone value on all arcs by a constant factor (2), which prevents the unlimited accumulation of pheromone trails and allows the algorithm to forget bad decisions previously taken.

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij}, \quad \forall (i, j) \in L \quad (2)$$

And by depositing pheromone on the arcs that ants have crossed in its path (3). The better the tour, the greater the amount of pheromone that the arcs will receive. In (2) ρ represents the rate of pheromone evaporation, which is a value between 0 and 1.

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^n \Delta \tau_{ij}^k, \quad \forall (i, j) \in L$$

$$\Delta \tau_{ij}^k = \begin{cases} \frac{1}{C^k}, & \text{if arc } (i, j) \text{ belong to } T^k; \\ 0, & \text{otherwise;} \end{cases} \quad (3)$$

In (3) C represents the cost of an arc in a graph. A first improvement on the initial AS, called the elitist strategy for Ant System (EAS) is as follows. The idea is to provide strong additional reinforcement to the arcs belonging to the best tour found since the start of the algorithm (4) [8].

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^n \Delta \tau_{ij}^k + e \Delta \tau_{ij}^{bs}, \quad \forall (i, j) \in L$$

$$\Delta \tau_{ij}^{bs} = \begin{cases} \frac{1}{C^{bs}}, & \text{if arc } (i, j) \text{ belong to } T^{bs}; \\ 0, & \text{otherwise;} \end{cases} \quad (4)$$

In (4) the term $\Delta \tau$ represents the pheromone increment and the bs indication is to distinguish the best-so-far ant. Another improvement over AS is the rank-based version of AS (denoted AS_{Rank}). In AS_{Rank} each ant deposits an amount of pheromone that decreases with its rank. Additionally, as in EAS, the best-so-far ant always deposits the largest amount of pheromone in each iteration [8]. In

Table 1
TSP instances considered.

TSP	Number of cities	Best tour length
Burma14	14	3323
Ulysses22	22	7013
Berlin52	52	7542
Eil76	76	538
kroA100	100	21,282

Table 2
Parameters used for each ACO variant.

ACO	α	β	ρ	m	τ_0
AS	1	2	0.5	n	m/C^m
AS _{Rank}	1	2	0.1	n	$0.5r(r-1)/\rho C^m$
EAS	1	2	0.5	n	$(e+m)/\rho C^m$

$m = n$.

$C^m = 20$ for each tsp except burma14 where $C^m = 10$.

EAS: $e = 6$.

AS_{Rank}: $r = w - 1$; $w = 6$.

Table 3
Performance obtained for the TSP instance Burma14.

ACO	Best	Average	Successful runs
AS	3323	3323	30/30
AS _{Rank}	3323	3329	19/30
EAS	3323	3323	30/30

Table 4
Performance obtained for the Ulysses22 TSP instance.

ACO	Best	Average	Successful runs
AS	7013	7022	30/30
AS _{Rank}	7013	7067	19/30
EAS	7013	7018	30/30

Table 5
Performance obtained for the Berlin52 TSP instance.

ACO	Best	Average	Successful runs
AS	7542	7557	2/30
AS _{Rank}	7542	7580	17/30
EAS	7542	7554	6/30

(5) w represents a number of ants considered in the ranking and r is an index for the ants in this set of w ants.

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{r=1}^{w-1} (w-r) \Delta \tau_{ij}^r + \Delta \tau_{ij}^{bs} \quad (5)$$

3. Performance analysis of ACO

To analyze the performance of the AS, EAS and AS_{Rank} variants, 30 experiments were performed by method for each instance of the examined TSP (Table 1), which are in the range of 14–100 cities, all extracted from TSPLIB [33], using the parameters recommended by the literature (Table 2) [8].

The behavior of AS and EAS is very similar in all experiments (Tables 3–7), the performance of the three variants began to worsen

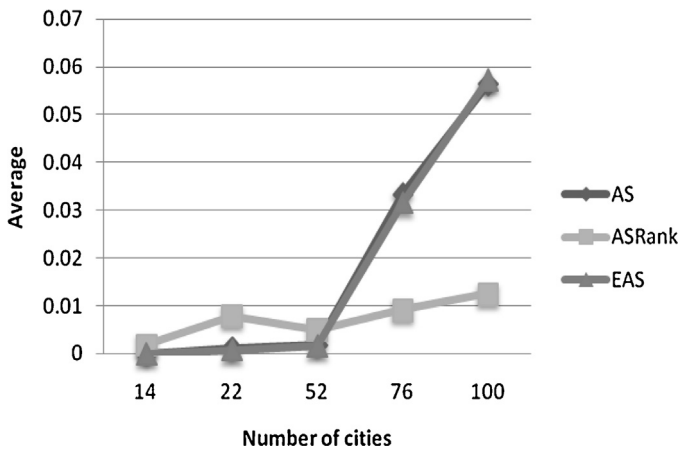
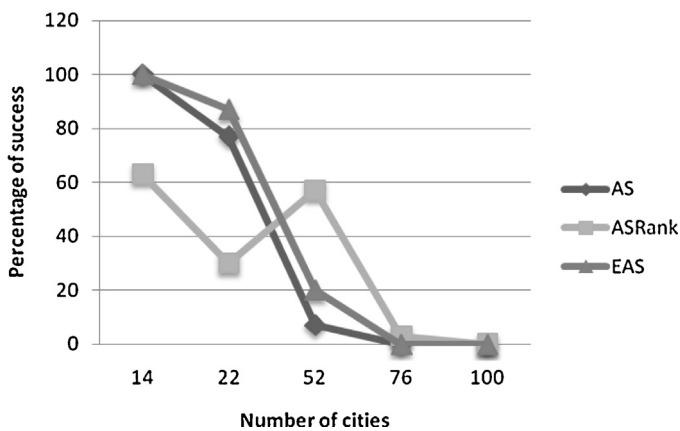
Table 6
Performance obtained for the Eil76 TSP instance.

ACO	Best	Average	Successful runs
AS	547	556	0/30
AS _{Rank}	538	543	1/30
EAS	544	555	0/30

Table 7

Performance obtained for the KroA100 TSP instance.

ACO	Best	Average	Successful runs
AS	22,305	22,483	0/30
AS _{Rank}	21,304	21,549	0/30
EAS	22,054	22,500	0/30

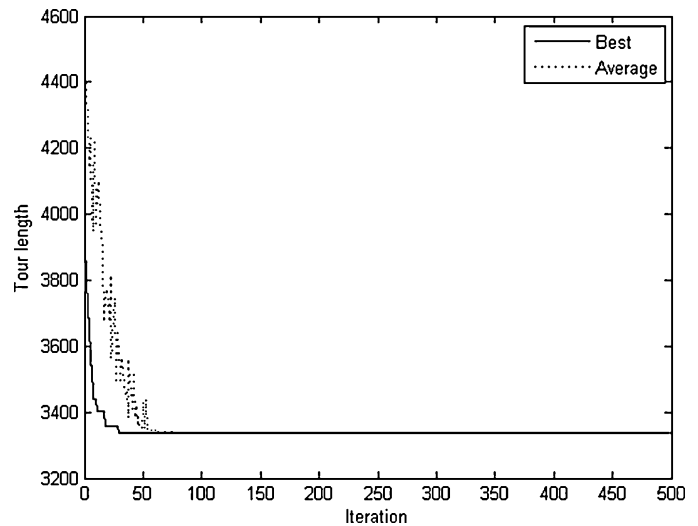
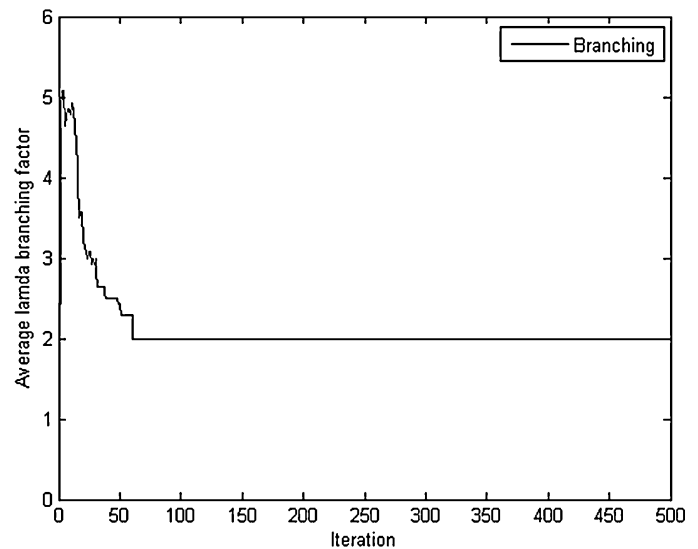
**Fig. 1.** Average results of each presented approach.**Fig. 2.** Percentage of success in finding the global minimum of each presented approach.

by increasing the problem complexity, however the AS_{Rank} performance decreased to a lesser extent than their counterparts when the number of cities was greater than 50 (Tables 5–7).

Since AS_{Rank} has more success finding the minimum and achieved lower averages with more complex TSP instances than the other approaches presented (Figs. 1 and 2). It can be concluded that AS and EAS have better performance when the number of cities is low, unlike AS_{Rank} that works better when the number of cities is not too small due to the pheromone deposit mechanism of this approach, where only the $w - 1$ ants with the shorter tours and the ant with the best so far tour are allowed to deposit pheromone. This strategy can lead to a stagnation situation where all the ants follow the same path and construct the same tour [8] as a result of excessive increase in the pheromone trails of suboptimal routes (Figs. 3 and 4).

4. Fuzzy logic convergence controllers

Based on the obtained results it was decided to use AS_{Rank} as the basis for our proposed ACO variant. The main idea is to prevent

**Fig. 3.** Convergence plot of the ACO algorithm AS_{Rank} variant.**Fig. 4.** Behavior of the average lambda branching factor during the execution of the ACO algorithm AS_{Rank} variant.

or stop the total convergence through the dynamic variation of the alpha parameter.

Alpha has a large effect in the diversity. Is recommended to keep α in the range of $0 < \alpha < 1$ [8]. A value closer to 1 will emphasize better paths but reduce diversity, while lower α will keep more diversity but reduce selective pressure [26].

However, it appears impossible to fix a universally best α . In most approaches it is taken to be 1, so that the selection probability is linear in the pheromone level.

An adaptive parameter control strategy is proposed in this paper; this takes place when there is some form of feedback from the search that is used to determine the direction and/or magnitude of the change to the strategy parameter [9]. In our case, the average lambda branching factor is used, and this factor measures the distribution of the values of the pheromone trails and provides an indication of the size of the search space effectively explored [8].

A convergence fuzzy controller to prevent or delay the full convergence of the algorithm was created in this work (Fig. 5). Fuzzy control can be viewed as the translation of external performance specifications and observations of a plant behavior into a rule based linguistic control strategy [40].

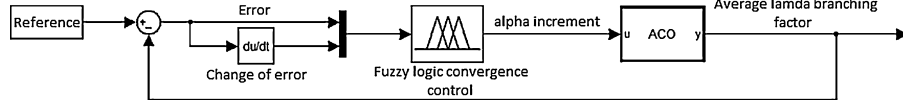


Fig. 5. Block diagram of the proposed system to control the convergence of the ACO algorithm variant AS_{Rank} .

If (error is P) and (error_change is P) then (alpha increment is N)
 If (error is N) and (error_change is N) then (alpha increment is P)
 If (error is P) and (error_change is Z) then (alpha increment is N)
 If (error is N) and (error_change is Z) then (alpha increment is P)
 If (error is P) and (error_change is N) then (alpha increment is Z)
 If (error is N) and (error_change is P) then (alpha increment is Z)
 If (error is Z) and (error_change is Z) then (alpha increment is Z)
 If (error is Z) and (error_change is N) then (alpha increment is P)
 If (error is Z) and (error_change is P) then (alpha increment is N)

Fig. 6. Rules of the proposed fuzzy system to control the convergence of the ACO algorithm.

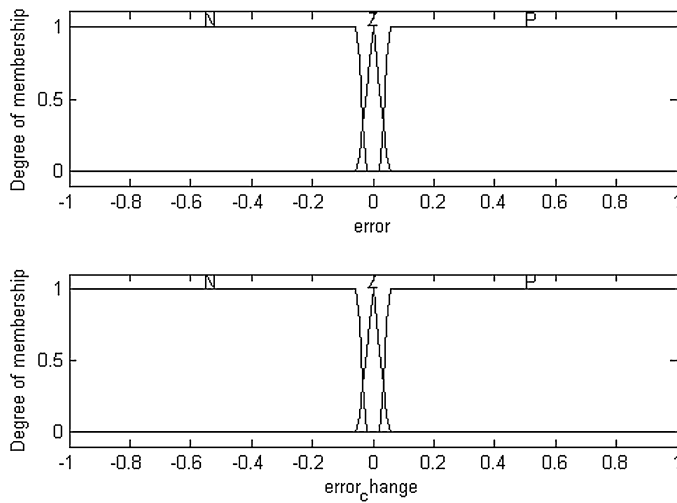


Fig. 7. Membership functions of the input variables of the proposed fuzzy system to control the convergence of the ACO algorithm.

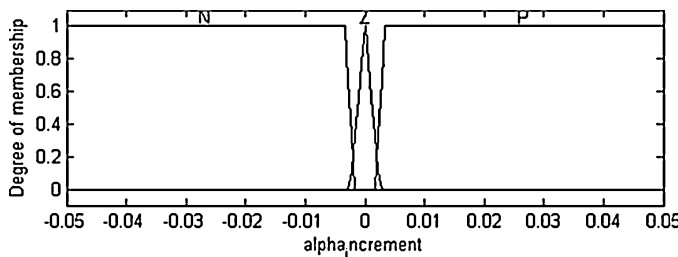


Fig. 8. Membership functions of the output variables of the proposed fuzzy system to control the convergence of the ACO algorithm.

The objective of the controller is to maintain the average lambda branching factor at a certain level to avoid a premature convergence, so its rules are designed to fulfill this goal (Fig. 6).

The controller of the α increment uses as inputs the error and change of error (Fig. 7) with respect to an average lambda branching factor reference level (this is the objective value) and provides as output an increase in the value of the parameter alpha (Fig. 8). Of course, in Fig. 7 the membership functions are defined over normalized ranges of values between -1 and 1 , and in Fig. 8 the range of α increment is defined between -0.05 and 0.05 , which was found by performing previous experiments to be a good interval.

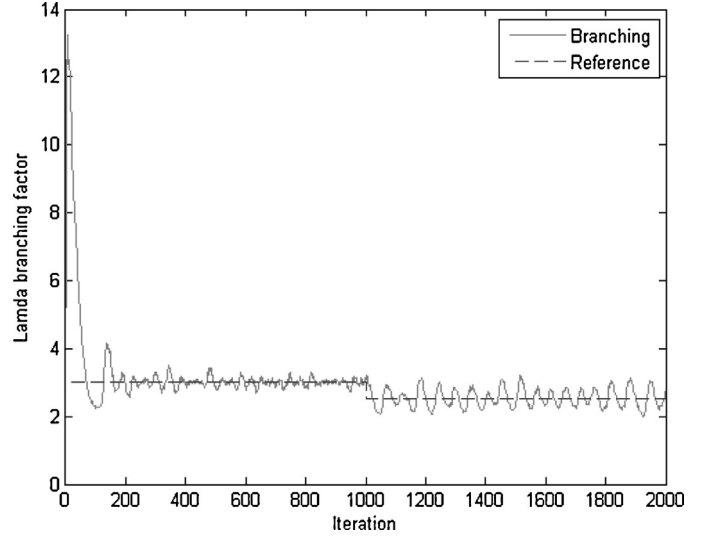


Fig. 9. Behavior of the average lambda branching factor during the execution of the developed approach.

Table 8

Performance obtained by the strategy proposed in the instances discussed above.

TSP	Best	Average	Successful runs
Burma14	3323	3323	30/30
Ulysses22	7013	7013	30/30
Berlin52	7542	7543	26/30
Eil76	538	539	21/30
KroA100	21,292	21,344	0/30

Table 9

Null and alternative hypothesis for the statistical hypothesis testing performed for TSP problems.

Case	Null hypothesis (H_0)	Alternative hypothesis (H_a)
1	$\mu_{AS} \leq \mu_{ASRank+ConvCont}$	$\mu_{AS} > \mu_{ASRank+ConvCont}$
2	$\mu_{EAS} \leq \mu_{ASRank+ConvCont}$	$\mu_{EAS} > \mu_{ASRank+ConvCont}$
3	$\mu_{ASRank} \leq \mu_{ASRank+ConvCont}$	$\mu_{ASRank} > \mu_{ASRank+ConvCont}$

5. Simulation in TSP problems

The fuzzy controller is able to maintain diversity in a more appropriate level, thus avoiding the full convergence of the algorithm (Fig. 9).

The same number of experiments mentioned in the above analysis is performed and we obtained the following results. Table 8 shows the performance of the proposed method in the same instances of TSP.

It was found that the proposed method is able to improve the results of the strategies studied, obtaining lower averages (Fig. 10) and reaching the global minimum on more occasions than the analyzed variants (Fig. 11).

To verify the above results in a more formal way a Z test for means of two samples is performed (Table 9).

The 3 ACO variants mentioned above are analyzed in addition to the approach developed in 5 instances of the TSP, 30 experiments are performed for each instance, 150 experiments are made in total of we extracted a 30 data random sample for each method.

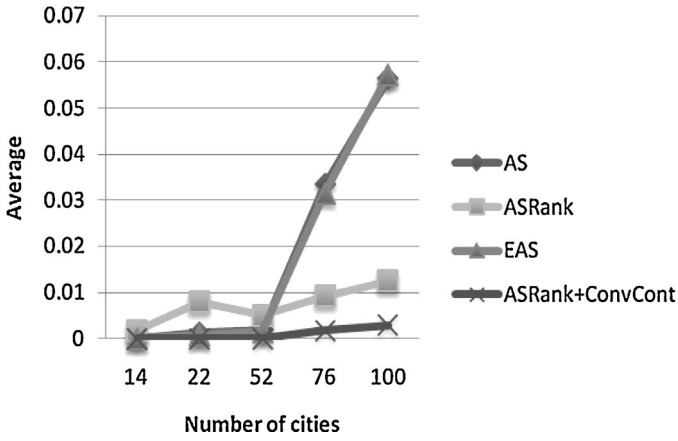


Fig. 10. Average of the results obtained by the proposal and each approach under review.

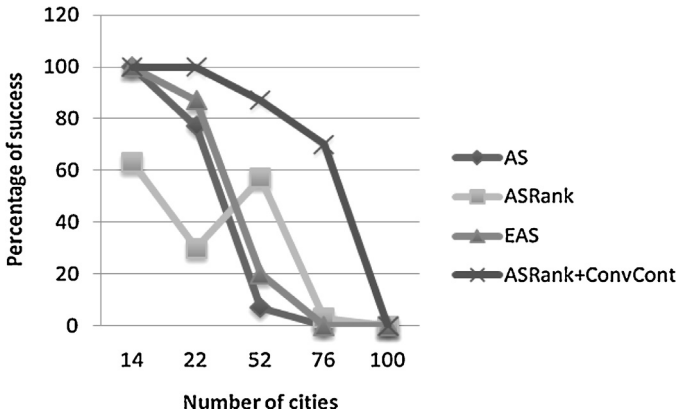


Fig. 11. Percentage of success in finding the global minimum of the proposal and each approach under review.

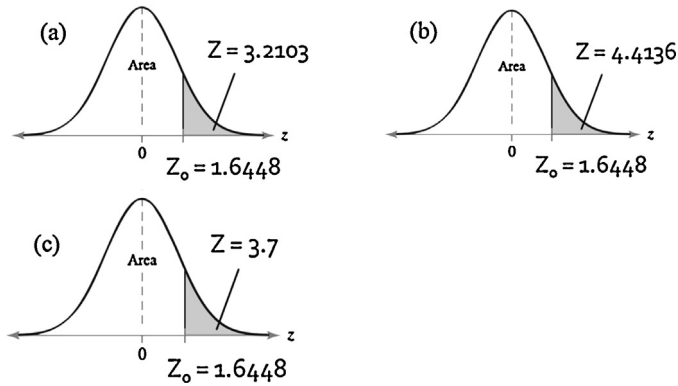


Fig. 12. Results of the statistical hypothesis testing performed for (a) AS vs. ASRank + ConvCont, (b) EAS vs. ASRank + ConvCont, (c) ASRank vs. ASRank + ConvCont for TSP problems.

With a significance level of 5% it was found sufficient statistical evidence to claim that the averages of AS (Fig. 12a), EAS (Fig. 12b) and ASRank (Fig. 12c) are higher than the one obtained for ASRank + ConvCont in the experiments, which means that the proposed approach improved the performance of the discussed variants on the studied problems, as had been observed in the first analysis.

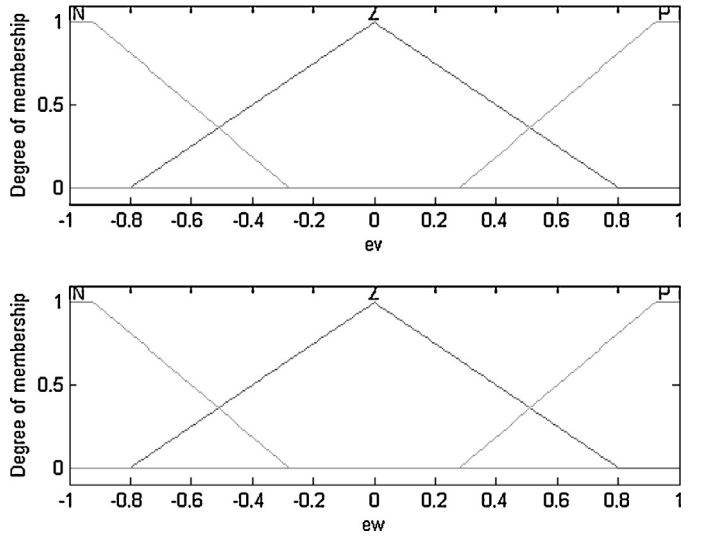


Fig. 13. Membership functions of the fuzzy trajectory controller input variables.

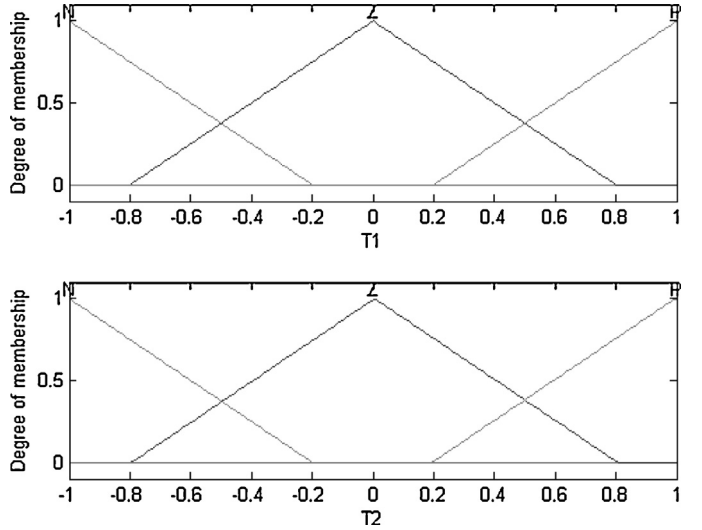


Fig. 14. Membership functions of the fuzzy trajectory controller output variables.

6. Fuzzy trajectory controllers for a unicycle mobile robot

It was decided to optimize a fuzzy trajectory controller for a unicycle mobile robot to test the developed method in a more complex problem. The control proposal for the mobile robot is as follows: Given a path $q_d(t)$ and a desired orientation, a fuzzy logic controller must be designed to apply an adequate torque τ , such that measured positions $q(t)$ reach the reference trajectory $q_d(t)$. That is:

$$\lim_{t \rightarrow \infty} \|q_d(t) - q(t)\| = 0 \quad (6)$$

The fuzzy system to be optimized [23] is of Takagi-Sugeno type, and for simplicity it was decided to modify and convert it into a Mamdani type controller so that the input and output parameters are represented by linguistic variables.

The controller receives as input variables the error in the linear (e_v) and angular (e_w) velocities (Fig. 13), that is, the difference between the predefined desired speed and the actual speed of the plant, and as output variables, the right (τ_1) and left (τ_2) torques of the mentioned robot (Fig. 14).

The membership functions of the input variables are trapezoidal for the negative (N) and positive (P) linguistic terms, and

If (e_v is N) and (e_w is N) then (τ_1 is N)(τ_2 is N)
 If (e_v is N) and (e_w is Z) then (τ_1 is N)(τ_2 is Z)
 If (e_v is N) and (e_w is P) then (τ_1 is N)(τ_2 is P)
 If (e_v is Z) and (e_w is N) then (τ_1 is Z)(τ_2 is N)
 If (e_v is Z) and (e_w is Z) then (τ_1 is Z)(τ_2 is Z)
 If (e_v is Z) and (e_w is P) then (τ_1 is Z)(τ_2 is P)
 If (e_v is P) and (e_w is N) then (τ_1 is P)(τ_2 is N)
 If (e_v is P) and (e_w is Z) then (τ_1 is P)(τ_2 is Z)
 If (e_v is P) and (e_w is P) then (τ_1 is P)(τ_2 is P)

Fig. 15. Rules of the of the fuzzy trajectory controller discussed.

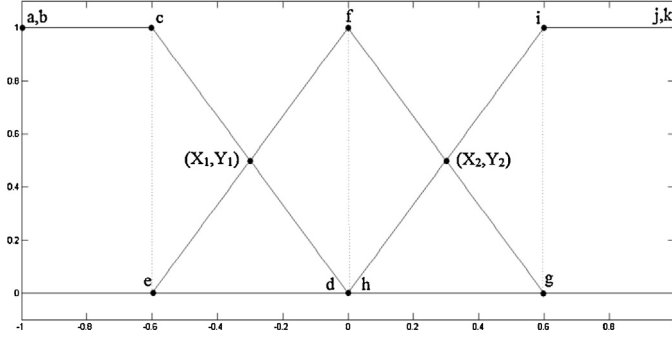


Fig. 16. Membership functions of the input variables of the fuzzy system to control the robot trajectory.

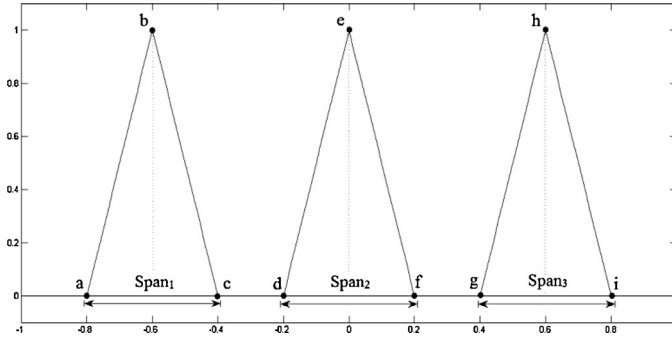


Fig. 17. Membership functions of the output variables of the fuzzy system to control the robot trajectory.

triangular for the zero (Z) linguistic term. The output variables have three membership functions, negative (N), zero (Z), positive (P) of triangular shape and the fuzzy system uses nine fuzzy rules which are shown below (Fig. 15):

7. ACO for membership functions optimization

ACO is used to find the membership functions optimal parameters through its adjustment and by the subsequently evaluation of the system.

The parameters a, b, f, j, k corresponding to the membership functions of the input variables remain fixed to simplify the problem. The algorithm will find the optimal values of the parameters c, i in a straightforward manner and, through the optimum position of the intersection points $(X_1, Y_1), (X_2, Y_2)$, the value of the parameters d, e, g, h (Fig. 16).

Regarding the membership functions of the output variables, the algorithm will search for the optimum center (b, h , except e that remains fixed for simplicity) and span of each one (a, c, d, f, g, i) (Fig. 17).

The application of ACO to optimize membership functions involves some considerations. First, encode all parameters in a weighted graph. For this goal we choose a complete graph of 43

Table 10

Relation variable weight for the linear velocity error input of the fuzzy system to optimize.

Variable	Relation
c	$c = -1 + 0.475 \left(\frac{(d_{1j_1} + d_{2j_2})}{2} \right) + 0.475$
X_1	$X_1 = c + \left(m_1 \left(\frac{(d_{3j_3} + d_{4j_4})}{2} \right) + C_1 \right)$ $m_1 = -\frac{c}{2}$ $C_1 = -(c + m_1)$
Y_1	$Y_1 = 0.5 \left(\frac{(d_{5j_5} + d_{6j_6})}{2} \right) + 0.5$
i	$i = 1 - 0.475 \left(\frac{(d_{11j_{11}} + d_{12j_{12}})}{2} \right) + 0.475$
X_2	$X_2 = i + \left(m_2 \left(\frac{(d_{7j_7} + d_{8j_8})}{2} \right) + C_2 \right)$ $m_2 = \frac{c}{2}$ $C_2 = i - m_2$
Y_2	$Y_2 = 0.5 \left(\frac{(d_{9j_9} + d_{10j_{10}})}{2} \right) + 0.5$

Table 11

Relation variable weight for the angular velocity error input of the fuzzy system to optimize.

Variable	Relation
c	$c = -1 + 0.475 \left(\frac{(d_{13j_{13}} + d_{14j_{14}})}{2} \right) + 0.47$
X_1	$X_1 = c + \left(m_1 \left(\frac{(d_{15j_{15}} + d_{16j_{16}})}{2} \right) + C_3 \right)$ $m_3 = -\frac{c}{2}$ $C_3 = -(c + m_3)$
Y_1	$Y_1 = 0.5 \left(\frac{(d_{17j_{17}} + d_{18j_{18}})}{2} \right) + 0.5$
i	$i = 1 - 0.475 \left(\frac{(d_{23j_{23}} + d_{24j_{24}})}{2} \right) + 0.475$
X_2	$X_2 = i + \left(m_4 \left(\frac{(d_{19j_{19}} + d_{20j_{20}})}{2} \right) + C_4 \right)$ $m_4 = \frac{c}{2}$ $C_4 = i - m_4$
Y_2	$Y_2 = 0.5 \left(\frac{(d_{21j_{21}} + d_{22j_{22}})}{2} \right) + 0.5$

Table 12

Relation variable weight for the right torque output of the fuzzy system to optimize.

Variable	Relation
b	$b = 0.5 \left(\frac{(d_{25j_{25}} + d_{26j_{26}})}{2} \right) - 0.5$
span_1	$\text{span}_1 = 0.475 \left(\frac{(d_{27j_{27}} + d_{28j_{28}})}{2} \right) + 0.525$
span_2	$\text{span}_2 = 0.475(d_{29j_{29}}) + 0.525$
h	$h = 0.5 \left(\frac{(d_{30j_{30}} + d_{31j_{31}})}{2} \right) + 0.5$
span_3	$\text{span}_3 = 0.475 \left(\frac{(d_{32j_{32}} + d_{33j_{33}})}{2} \right) + 0.525$

nodes to maintain the similarity of the problem with a classical TSP where a minimum Hamiltonian circuit is searched.

The range of each variable was discretized in 22 normalized values in the range $[-1, 1]$, and a symmetric data matrix of 43×43 with the distance between nodes was created. The parameters of the membership functions of the fuzzy system are obtained through the distance between two nodes using the relations of Tables 10–13.

The algorithm will find the optimal values of c, i in a direct manner and using the optimal positions of the intersection points $(X_1,$

Table 13
Relation variable weight for the left torque output of the fuzzy system to optimize.

Variable	Relation
b	$b = 0.5 \left(\frac{(d_{34,j_{34}} + d_{35,j_{35}})}{2} \right) - 0.5$
span_1	$\text{span}_1 = 0.475 \left(\frac{(d_{36,j_{36}} + d_{37,j_{37}})}{2} \right) + 0.525$
span_2	$\text{span}_2 = 0.475(d_{33,j_{33}}) + 0.525$
h	$h = 0.5 \left(\frac{(d_{39,j_{39}} + d_{40,j_{40}})}{2} \right) + 0.5$
span_3	$\text{span}_3 = 0.475 \left(\frac{(d_{41,j_{41}} + d_{42,j_{42}})}{2} \right) + 0.525$

Y_1 , (X_2, Y_2) the values of parameters d, e, g, h where:

$$d = \frac{(0 - C_1)}{m_1}, \quad e = \frac{(0 - C_2)}{m_2},$$

$$m_1 = \frac{(Y_1 - 1)}{(X_1 - c)}, \quad m_2 = \frac{(1 - Y_1)}{(0 - X_1)},$$

$$C_1 = 1 - m_1 c, \quad C_2 = 1,$$

$$-1 < c < -0.05, \quad c < X_1 < 0, \quad 0 < Y_1 < 1,$$

$$g = \frac{(0 - C_3)}{m_3}, \quad h = \frac{(0 - C_4)}{m_4},$$

$$m_3 = \frac{(Y_2 - 1)}{(X_2 - 0)}, \quad m_4 = \frac{(1 - Y_2)}{(i - X_2)},$$

$$C_3 = 1, \quad C_4 = 1 - m_4 i,$$

$$0.05 < i < 1, \quad 0 < X_2 < i, \quad 0 < Y_2 < 1$$

where m_1, m_2, m_3 and m_4 are the slopes in Fig. 16.

The next step is to define an appropriate objective function to evaluate the performance of ACO. The objective function represents the quality of the solution, and acts as an interface between the optimization algorithm and the considered problem. The mean square error is used to evaluate the fitness of the fuzzy system.

$$\text{MSE} = \frac{1}{N} \sum_{k=1}^N [y(k) - \tilde{y}(k)]^2 \quad (7)$$

where $y(k)$ = Reference value at instant k ; $\tilde{y}(k)$ = Computed output of the system at instant k ; N = Number of samples considered.

Since the system is responsible for controlling the linear (v) and angular (w) velocities of the plant, the overall error is given by:

$$\text{MSE}_v = \frac{1}{N} \sum_{k=1}^N [v(k) - \tilde{v}(k)]^2$$

$$\text{MSE}_w = \frac{1}{N} \sum_{k=1}^N [w(k) - \tilde{w}(k)]^2$$

$$\text{Error}_{\text{global}} = \text{MSE}_v + \text{MSE}_w$$

This is used to represent the entire length of each ant generated graph.

8. AS_{Rank} + ConvCont for membership functions optimization

Due to the nature of the problem, we do not have previous heuristic information to make a balance between the influence of the knowledge we have a priori of the problem and the pheromone trails that ants have generated, thus the dynamic variation of the alpha parameter had a null effect on the convergence of the algorithm when applied to the optimization of membership functions (Fig. 18).

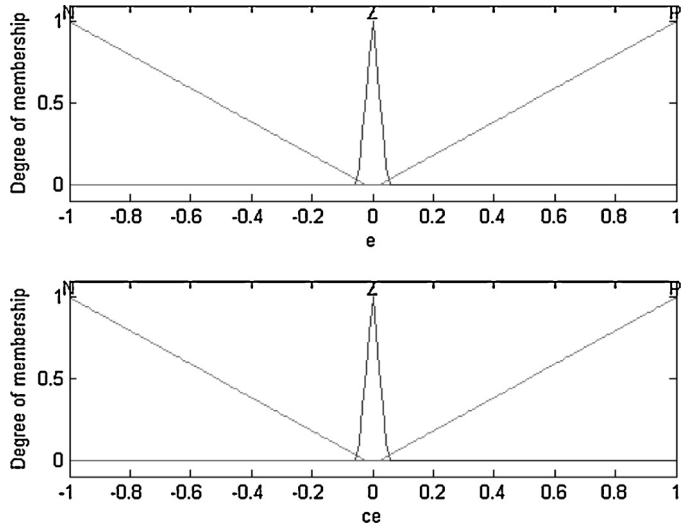


Fig. 18. Membership functions of the input variables of the fuzzy system proposed to control the convergence of the ACO algorithm without heuristic information.

If (error is P) and (error_change is P) then $(\Delta_u^{bs} N) (\Delta_u P) (\rho^{bs} P) (\rho N)$
 If (error is N) and (error_change is N) then $(\Delta_u^{bs} P) (\Delta_u N) (\rho^{bs} N) (\rho P)$
 If (error is P) and (error_change is Z) then $(\Delta_u^{bs} N) (\Delta_u P) (\rho^{bs} P) (\rho N)$
 If (error is N) and (error_change is Z) then $(\Delta_u^{bs} N) (\Delta_u P) (\rho^{bs} P) (\rho N)$
 If (error is P) and (error_change is N) then $(\Delta_u^{bs} N) (\Delta_u P) (\rho^{bs} P) (\rho N)$
 If (error is N) and (error_change is P) then $(\Delta_u^{bs} N) (\Delta_u P) (\rho^{bs} P) (\rho N)$
 If (error is Z) and (error_change is Z) then $(\Delta_u^{bs} N) (\Delta_u P) (\rho^{bs} P) (\rho N)$
 If (error is Z) and (error_change is N) then $(\Delta_u^{bs} N) (\Delta_u P) (\rho^{bs} P) (\rho N)$
 If (error is Z) and (error_change is P) then $(\Delta_u^{bs} N) (\Delta_u P) (\rho^{bs} P) (\rho N)$

Fig. 19. Rules of the proposed fuzzy system to control the convergence of the ACO algorithm without heuristic information.

Then it was decided to continue with the same strategy of convergence control, but this time by varying the evaporation rate (ρ) and the weight to be given to the amount of pheromone that each ant leaves on its trail (w) to control diversity, so another fuzzy system is implemented for this task.

The controller now uses as inputs, the error (e) and change of error (ce) with respect to an average lambda branching factor reference level (Fig. 18) and provides as output the evaporation rate corresponding to arcs, which belong (ρ^{bs}) and do not belong (ρ) to the best so far tour, in addition to an increase in the weight that is given to the pheromone increment of the arcs that form part of the best so far tour (u^{bs}) and the remaining arcs (u) in AS_{Rank} (Fig. 19).

Again the rules are created with the intention to keep the average lambda branching factor at some particular level to slow the convergence process and are shown below (Fig. 20):

Thus Eqs (2) and (4) corresponding to the evaporation and pheromone deposit process in AS_{Rank} become:

$$\begin{aligned} \tau_{ij}^{bs} &\leftarrow (1 - \rho^{bs}) \tau_{ij}^{bs}, \quad \forall (i, j) \in T^{bs} \\ \tau_{ij} &\leftarrow (1 - \rho) \tau_{ij}, \quad \forall (i, j) \notin T^{bs} \\ \tau_{ij} &\leftarrow \tau_{ij} \sum_{r=1}^w \frac{(w-r)(u)}{(w-1)} \Delta \tau_{ij}^r + u^{bs} \Delta \tau_{ij}^{bs} \\ \Delta \tau_{ij}^r &= \frac{1}{c^r} \quad \text{and} \quad \Delta \tau_{ij}^{bs} = \frac{1}{c^{bs}} \end{aligned}$$

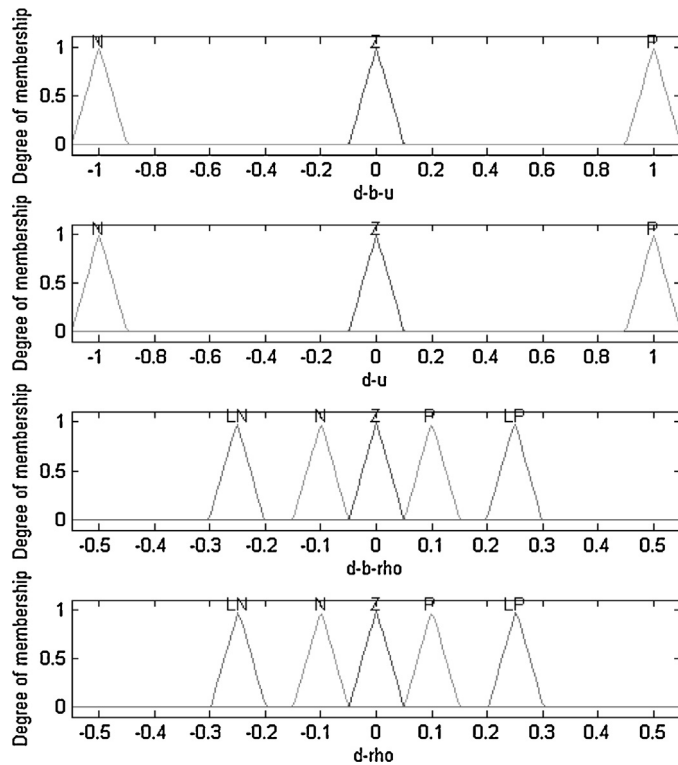


Fig. 20. Membership functions of the output variables of the proposed fuzzy system to control the convergence of the ACO algorithm without heuristic information.

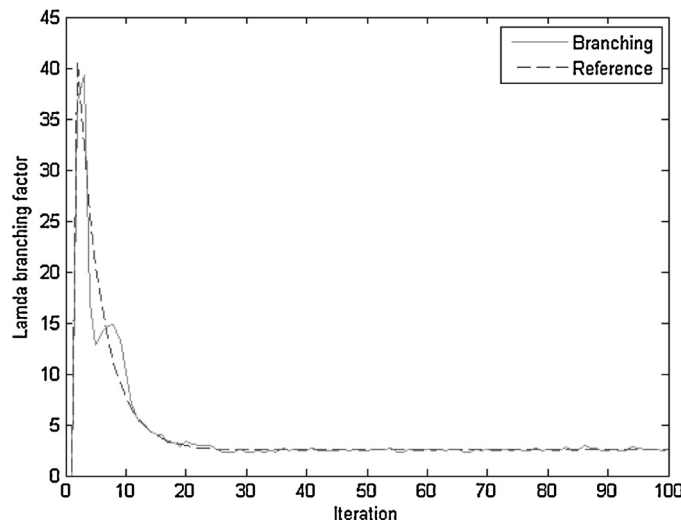


Fig. 21. Behavior of the average lambda branching factor during the execution of the developed approach to control the convergence of the ACO algorithm without heuristic information.

9. Simulation in membership functions optimization problem

The model of the mobile robot and the path used in the simulations performed by the ACO algorithm are defined in [23]. The approach described in the previous section is able to maintain diversity at the required level (Fig. 21) unlike the convergence controller that was tested in Section 5.

In this case 30 experiments were performed with the proposed approach (Table 15) to compare the performance of classical approaches with the developed proposal. The parameters used in the experiments are presented in Table 14.

Table 14

Parameters used for each ACO algorithm in the membership function optimization problem.

ACO	α	β	ρ	m	τ_0
AS	1	0	0.5	n	m/C^n
AS _{Rank}	1	0	0.1	n	$0.5r(r-1)/\rho C^n$
EAS	1	0	0.5	n	$(e+m)/\rho C^n$
AS _{Rank} + CONVCONT	1	0	Dynamic	n	0.1

$m = n$.

C^n = length of a tour generated by a nearest-neighbor heuristic.

EAS: $e = 6$.

AS_{Rank}, AS_{Rank} + CONVCONT: $r = w - 1$; $w = 6$.

Table 15

Results obtained by the proposal and each approach under review algorithm in the membership function optimization problem.

ACO	Best	Average
AS	0.0015	0.0172
EAS	0.00013	0.0161
AS _{Rank}	0.00015	0.0572
AS _{Rank} + CONVCONT	0.00029	0.0131

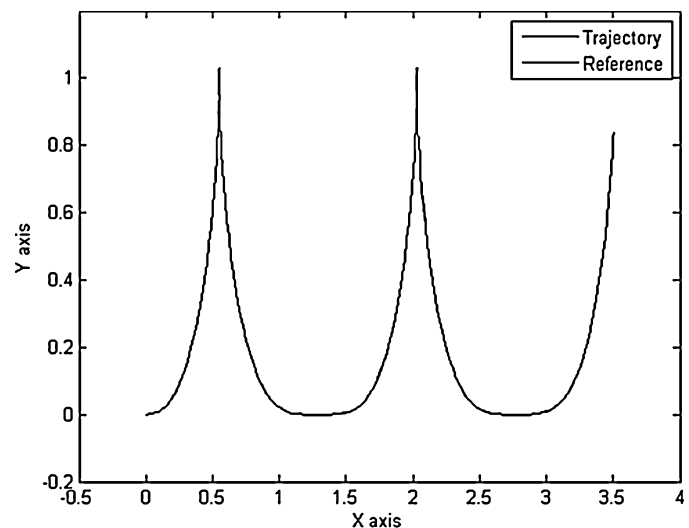


Fig. 22. Trajectory obtained by the best generated controller.

Table 16

Null and alternative hypothesis for the statistical hypothesis testing performed for membership function optimization problem.

Case	Null hypothesis (H_0)	Alternative hypothesis (H_a)
1	$\mu_{AS} \leq \mu_{ASRank+ConvCont}$	$\mu_{AS} > \mu_{ASRank+ConvCont}$
2	$\mu_{EAS} \leq \mu_{ASRank+ConvCont}$	$\mu_{EAS} > \mu_{ASRank+ConvCont}$
3	$\mu_{ASRank} \leq \mu_{ASRank+ConvCont}$	$\mu_{ASRank} > \mu_{ASRank+ConvCont}$

With the exception of AS_{Rank}, the average simulation results obtained are very similar. The proposal obtained the lowest average, but despite this it was EAS which generated the lowest MSE controller (Fig. 23) and therefore the more accurate trajectory (Fig. 22).

It is difficult to determine whether the proposal improved over the classical approaches with the above analysis, so a Z test for two samples means is performed to achieve a conclusion (Table 16).

No statistical evidence was found with a significance level of 5% that the average of AS or EAS is greater than the average of AS_{Rank} + CONVCONT (Fig. 24a and b).

With a significance level of 5%, there is statistical evidence that the average of the results of simulations of AS_{Rank} is greater than

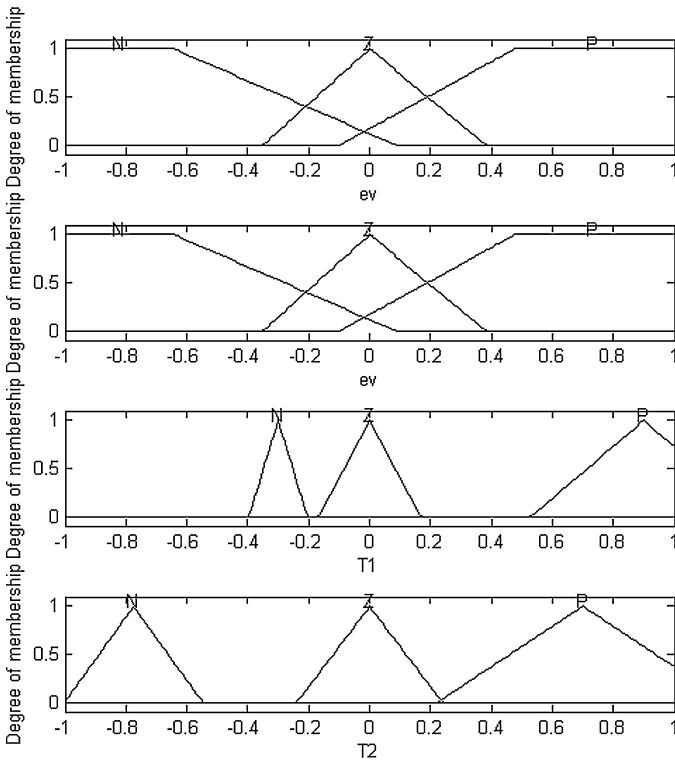


Fig. 23. Membership functions of the best generated controller.

Table 17

Performance obtained by AS_{Rank} + CONVCONT and S-ACO in the membership function optimization problem.

ACO	Best	Average
AS_{Rank} + CONVCONT	0.00029	0.0131
S-ACO	0.0982	0.1199

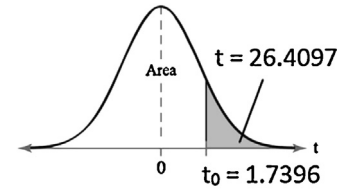


Fig. 26. Results of the statistical hypothesis testing performed for (a) S-ACO vs. AS_{Rank} + ConvCont.

AS_{Rank} + CONVCONT (Fig. 24c), that is, the proposal was only able to outperform the AS_{Rank} variant.

10. AS_{Rank} + ConvCont vs. S-ACO

The results obtained with the developed proposed approach are compared with the ones obtained by [5], where the same membership function optimization problem was considered for the same fuzzy trajectory controller and unicycle mobile robot model, the difference lies in S-ACO as strategy used to solve the problem and the directed graph of 12 nodes chosen to represent it.

At first glance it can be observed that the best result of AS_{Rank} + CONVCONT is significantly lower than S-ACO as well as the average of the results obtained in the experiments (Table 17), this is reflected in the path generated by each controller (Fig. 25), therefore we conclude that its performance is higher.

To support the above a t -test for means of two samples is performed, for which we considered a random sample of 10 experiments per technique to compare their performance.

The null hypothesis claims that the average of S-ACO is less than or equal to AS_{Rank} + CONVCONT.

$$H_0: \mu_{S-ACO} \leq \mu_{AS_{Rank}+ConvCont}$$

$$H_a: \mu_{S-ACO} > \mu_{AS_{Rank}+ConvCont}$$

$$\alpha = 0.05$$

Since t is located at the rejection zone with a significance level of 5% and 9 degrees of freedom there is sufficient statistical evidence to prove that the average of S-ACO is greater than AS_{Rank} + ConvCont

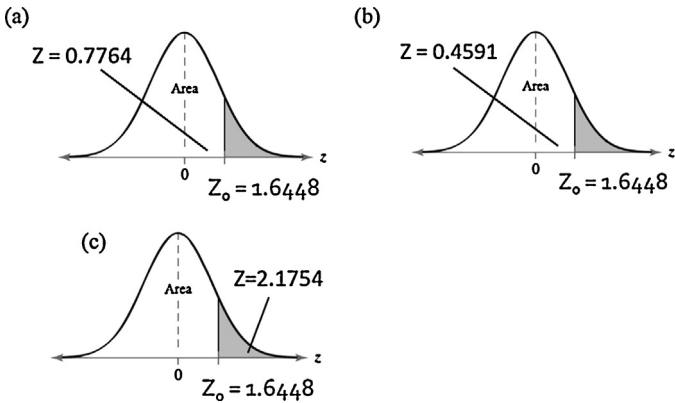


Fig. 24. Results of the statistical hypothesis testing performed for (a) AS vs. AS_{Rank} + ConvCont, (b) EAS vs. AS_{Rank} + ConvCont, (c) AS_{Rank} vs. AS_{Rank} + ConvCont for membership functions optimization problem.

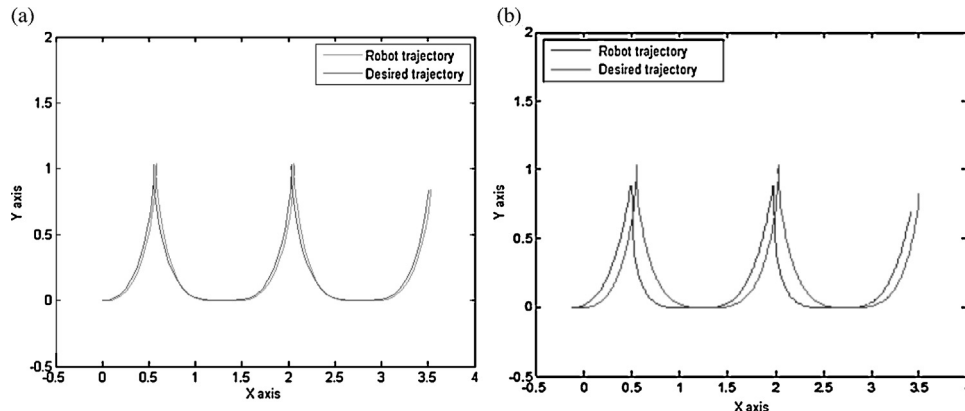


Fig. 25. Trajectories generated by the controller obtained by the best of experiments performed with: (a) AS_{Rank} + ConvCont, (b) S-ACO.

(Fig. 26), that is, the developed approach outperforms the method used by [5] and therefore likewise AS and EAS by the analysis of Section 9.

11. Conclusions

Maintaining diversity is important for good performance in the ACO algorithm. An adaptive control strategy of the parameter alpha for this purpose is proposed, which is embodied in a diversity fuzzy controller, which allows avoiding, or delaying the total convergence and thereby controlling the exploration and exploitation capabilities of the algorithm.

The proposed strategy is compared with 3 variants of the ACO algorithm on several instances of the TSP taken from TSPLIB. An improvement is observed by dynamically changing the alpha parameter value, as is noted in the statistical analysis performed, where the proposed approach outperforms the classical strategies.

It was found that the alpha parameter is not the most appropriate when there is no heuristic information to guide the search as is the case with the optimization of membership functions, since it is not possible to balance between the previous knowledge of the problem and by the generated by the algorithm itself during its execution and thus control the convergence of the algorithm. So it was decided to continue with the same strategy for this kind of problem, but varying the evaporation rate and the weight, which is given to the amount of pheromone which each ant deposited, and this allowed controlling the convergence of the algorithm without heuristic information. This modification improved the performance of AS_{Rank}, however since this variant scored the lowest performance, is probably not the most appropriate in these cases.

The formulated strategy is outperformed by AS and EAS in the membership functions optimization problem, but managed to outperform the method developed in [5], so it was concluded that the improvement could not come from the convergence control made and is attributed to the way in which the problem is encoded.

As future work, we intend to apply convergence control to other variants of ACO algorithm. Modify the reference, and thus diversity in an intelligent way, depending of the search progress or some other performance measure. Look for heuristic information relevant to the membership functions optimization problem that drives the search process in early iterations of the algorithm, making it possible to use the strategy of dynamic variation of the parameter alpha and an analysis in presence of noise of the generated controller by ACO algorithm. Finally, the proposed approach could also be extended as a multi-objective optimization method to improve results [41].

References

- [5] O. Castillo, R. Martinez-Marroquin, P. Melin, F. Valdez, J. Soria, Comparative study of bio inspired algorithms applied to the optimization of type-1 and type-2 fuzzy controllers for an autonomous mobile robot, *Inform. Sci.* 192 (2012) 19–38.
- [8] M. Dorigo, T. Stutzle, *Ant Colony Optimization*, MIT Press, Cambridge, MA, 2004.
- [9] A.E. Eiben, R. Hinterding, Z. Michalewicz, Parameter control in evolutionary algorithms, *IEEE Trans. Evol. Comput.* 3 (2) (1999) 124–141.
- [23] R. Martínez, O. Castillo, L. Aguilar, Optimization of interval type-2 fuzzy logic controllers for a perturbed autonomous wheeled mobile robot using genetic algorithms, *Inform. Sci.* 179 (2009) 2158–2174.
- [25] D. Merkle, M. Middendorf, Prospects for dynamic algorithm control: lessons from the phase structure of ant scheduling algorithms, in: R.B. Heckendorn (Ed.), *Proceedings of the 2000 Genetic and Evolutionary Computation Conference – Workshop Program. Workshop “The Next Ten Years of Scheduling Research”*, Morgan Kaufmann Publishers, Las Vegas, Nevada, USA, 2001.
- [26] D. Merkle, M. Middendorf, H. Schmeck, Ant colony optimization for resource-constrained project scheduling, *IEEE Trans. Evol. Comput.* 6 (2002) 333–346.
- [27] B. Meyer, Convergence control in ACO, in: *Genetic and Evolutionary Computation Conference (GECCO)*, Seattle, WA, 2004.
- [28] M. Mohamad, W. Dunningan, Ant colony robot motion planning, in: *International Conference on Computer as a Tool, EUROCON, Serbia & Montenegro*, 2005.
- [33] G. Reinelt, *TSPLIB*, University of Heidelberg, 1995, <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/>
- [40] J. Yen, R. Langari, *Fuzzy Logic: Intelligence, Control and Information*, Prentice Hall, 1999.
- [41] A. Ochoa, M. De, L.M. Fuentes, J. Arreola, G.O. Gutiérrez, G. García, F. Maldonado, Using multi-objective optimization to design parameters in electro-discharge machining by wire, in: *DCAI, 2014*, pp. 493–502.