

Research Article

ACOustic: A Nature-Inspired Exploration Indicator for Ant Colony Optimization

Rafid Sagban,¹ Ku Ruhana Ku-Mahamud,² and Muhamad Shahbani Abu Bakar²

¹Computer Science Department, University of Babylon, Babylon, Iraq

²School of Computing, College of Arts and Sciences, Universiti Utara Malaysia, 06010 Sintok, Kedah, Malaysia

Correspondence should be addressed to Rafid Sagban; rsagban@uobabylon.edu.iq

Received 26 December 2014; Accepted 12 March 2015

Academic Editor: Patricia Melin

Copyright © 2015 Rafid Sagban et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A statistical machine learning indicator, ACOustic, is proposed to evaluate the exploration behavior in the iterations of ant colony optimization algorithms. This idea is inspired by the behavior of some parasites in their mimicry to the queens' acoustics of their ant hosts. The parasites' reaction results from their ability to indicate the state of penetration. The proposed indicator solves the problem of robustness that results from the difference of magnitudes in the distance's matrix, especially when combinatorial optimization problems with rugged fitness landscape are applied. The performance of the proposed indicator is evaluated against the existing indicators in six variants of ant colony optimization algorithms. Instances for travelling salesman problem and quadratic assignment problem are used in the experimental evaluation. The analytical results showed that the proposed indicator is more informative and more robust.

1. Introduction

Since the day of modelling "swarm intelligence" (SI) [1] as a computational technique for global optimization, ant colony optimization (ACO) algorithms went beyond being specific purpose methods. The first ACO algorithm, namely, ant system, was introduced as a novel SI metaheuristic for solving hard combinatorial optimization (CO) problems [2]. The idea is to consider the artificial ant processes that build solutions with the help of a common memory. The memory is readable and modifiable by all artificial ants. In ACO modelling, translating this natural optimization process into a combinatorial optimization system without monitoring causes one of the following situations. The first situation is that all ants may follow the same path. This results in an overexploitation state where the local optimum is occurring. The second situation is that the ants just perform a random walk. This results in an overexploration state [3]. The exploration and exploitation balance is needed for effective search not in ACO level, but at SI metaheuristic level.

In metaheuristics, higher level strategies of exploration are the typical solutions to escape from local optima and performing a robust search of a solution space [4]. Although

the relevance of this concept is commonly agreed, so far, there is a limited unifying description to be found in the literature, while most of the work that deals with this topic focused on specific procedures [5]. Recently, reactive search [6] has emerged as a generic technique to improve the internal behavior of metaheuristics. It relies on exploration indicators to promote the exploration needed. In SI algorithms, prominent indicators are typically used for learning parameters [7, 8], determining when to restart the search [9, 10], or analyzing the explorative behavior of the algorithm [11, 12].

So far, utilizing traditional indicators does not satisfy the requirements of reactive search because they are simply not machine learning methods, except the indicator developed by Pellegrini et al. [13]. It utilizes an agglomerative clustering to quantify the exploration as the number of clusters of solutions visited. Two closest clusters can be concluded when the Euclidean distance between them is greater than a predefine threshold ϵ_x where $x\%$ of their solution components do not exist in the cluster. However, the definition needs to be reconsidered in terms of robustness against various circumstances because of the coherence with the magnitude of the distance matrix. For example, in ACO, the threshold ϵ_x is equal to 7.8 without local search, equal to 17.5 in with

2-opt local search, and equal to 35.8 with 3-opt local search. Therefore, this situation leads to an unstable measurement, especially when more rugged CO problem instances, such as QAP, need to be solved by SI algorithm.

This paper introduces, systematically, a step forward in proposing a statistical machine learning technique for indicating the exploration amount in ACO algorithm during the run. The idea of this work is simulated by a natural strategy, namely, *acoustical mimicry*, in which the signals exchanged between nest-mates in the nest may be mimicked by other insects that coexist with ants or interact with them as social parasites. In some way, the larva of the parasite is carried into the ant nest, where the ants treat it as a nest-mate. This biological strategy is utilized by the parasites to survive inside their host ant nests [14]. In this paper, the exploration conducted by artificial ants in the optimization process is the analogy of leaving the nest before parasites are killed by the real ants, that is, characterizing the current penetration state. The mechanism in nature is modelled into several statistical machine learning algorithms to detect the performance of the ACO algorithm.

The rest of the paper is organized as follows. The exploration measurement in ACO is overviewed in Section 2. The biological idea, the modelling, and the implementation of the proposed exploration indicator are illustrated in Section 3. The experimental setting is described in Section 4. The computational results are presented in Section 5. Finally, the conclusions and the suggested directions for future research are highlighted in Section 6.

2. Exploration Indicators in ACO

This section is to overview the existing techniques to indicate the exploration amount in ACO algorithm. Towards a reactive-based ACO search, by tuning an ACO algorithm, setting the parameters to values which allow an efficient search, describing the amount of exploration that an ACO algorithm performs, and detecting stagnation situations, the numerical exploration indicators are proposed [3, 6, 15]. The distance between solutions is one of the simplest indicators that calculate the number of arcs that are contained in one solution, but not in the other. It focuses on the duplicated arcs between two solutions, and if the degree of similarity is high, then the exploration amount is low [15]. Its disadvantage is being computationally expensive. Pellegrini [16] introduced the use of entropy indicator for characterizing the amount of exploration in ACO algorithm. Colas and Monmarch [17] employed entropy as a feedback indicator in their parameters control method proposal. This indicator can be applied simultaneously during the solution construction as follows. At each node, the ant calculates the selection probabilities of all other nodes in the following manner:

$$\varepsilon_i = - \sum_{j=1}^l p_{ij} \log p_{ij}, \quad (1)$$

where p_{ij} is the probability of choosing arc (i, j) when being in node i and $l, 1 \leq l \leq n - 1$, is the number of possible choices. Therefore, finding the average of all entropies will

be in the same way. Another indicator, convergence factor, is introduced by Blum [18] to be used in a hypercube framework [19]. It is used to track the convergence of ants, which is the converging phase of all ants. The convergence factor is calculated as follows:

$$\frac{\sum_{\tau_{ij} \in T} \min \{ \tau_{\max} - \tau_{ij}, \tau_{ij} - \tau_{\min} \}}{n^2}, \quad (2)$$

where T is the pheromone matrix and τ_{ij} is the pheromone trails. Another tool for indication is acceptance criteria (AC), which is used with restart strategies to increase the exploration amount in ACO. It is defined by Stützle [20] as such: if the i_{last} was the iteration counter i in which the best-iteration solution has been found, then the optimal point for restarting the search can be calculated as follows:

$$\text{AC} = \begin{cases} s'' & \text{if } f(s'') < f(s) \\ s''' & \text{if } f(s'') \geq f(s), i - i_{\text{last}} > i_r \\ s & \text{otherwise.} \end{cases} \quad (3)$$

Based on this formula, the value of s''' is generated randomly by a new initial solution which corresponds to a restart of the algorithm. The amount of exploration will increase without guaranteeing that the same regions will not be visited again. The problem has been solved by Sagban et al. [10]. Solnon and Fenet [12] used the similarity ratio to indicate how much the solutions are similar. This is done by finding the final solutions generated by ants and then applying some statistics tools such as the standard deviation and the variation coefficient. One example of these statistics is the standard deviation of the objective function of solutions constructed. The exploration amount will be high if the standard deviation is near to one and low if near to zero. Gambardella and Dorigo [21] introduced the average lambda branching factor (λ -branching) which depends directly on the pheromone trail values and makes it more suitable for tracking the ant's behavior while the computation is going on. This technique measures the diversity of the pheromone trail values in a more direct way. It does not change much from iteration to iteration. The branching factor can be defined as follows. Let $ph_max(i, j)$ and $ph_min(i, j)$ be the maximum and minimum pheromone amount, respectively, of all the arcs that exit from node i . Let d be the difference between two amounts. The branching factor of node i is the number of arcs that is greater than $\lambda * d + ph_min(i, j)$, where $0 \leq \lambda \leq 1$. The average of the lambda branching factor of all nodes gives an indication of the amount of exploration conducted by each ant. The λ -branching indicator is very important in the experimental history of improving the ACO metaheuristic. Recently, Neyoy et al. [22] and Neyoy et al. [7] used this indicator in designing fuzzy controllers for parameters adaptation in ACO algorithms. The disadvantage of the branching factor is its dependency on the value of parameter λ . Moreover, it is not applied for other CO problems such as quadratic assignment problem. Pellegrini and Favaretto [5] quantified the exploration as the number of clusters of solutions visited: $E(B, I, R, h) = |L(B, I, R, h)|$,

where L is the set of clusters resulting from the solutions visited by the algorithm B when solving the instance I using the resources R and the seed h . Given such definition, it is possible to observe the behavior of any metaheuristic without needing a deep knowledge on the internal state of the algorithm.

3. The Proposed Indicator: ACOustic

The nature-inspired indicator, denoted as (ACOustic), is proposed to characterize the exploration in the ACO algorithm. To present the idea, the following subsections are discussed: the biological schema, the modelling, and the implementation.

3.1. The Biological Schema. Rapid and effective communication between ants is a key attribute that enables them to live in dominant, fiercely protected societies. *Myrmica* ant colonies, in particular, are exploited by social parasites called *Maculinea* butterflies [23]. The process of *Trophallaxis* (i.e., distributing liquid food from the “social stomach”) between attendance worker and other nest-mates is the main process in food foraging behavior of ants. The worker ants produce acoustics during the process. The *Maculinea* larvae are interfering with the *Myrmica* system and produce similar acoustics to that of the colony. The high number of worker ants leads to a low relatedness between nest-mates. A greater variance in nest-mates acoustic signals leads to a higher likelihood of being infested [14]. Through this indicator, the larva can decide the optimal point to leave the colony before it is discovered by other ants as depicted in Figure 1.

The larva is able to evaluate the situation inside the nest whether to leave or to stay. If the relatedness between nest-mates becomes high, then the likelihood of being clustered around the larvae will become low. This is an indication for the larva to explore another nest before being killed; otherwise the larva will continue to exploit the current nest until further notice. The acoustic reaction in this process can be simplified in three basic components as shown in Figure 2.

In ACO modelling, a colony of artificial ants inspires its characteristics from the real ant’s foraging behavior. The construction graph simulates the environment that ants and larvae agents are moving on. For larvae agents, the interaction with the new environment is highly related to the state of penetration, that is, the learning process. The agents can decide whether to continue with the current exploitation or to explore another environment. To simulate the process of characterizing the state of penetration, statistical analyzing and agglomerative clustering algorithms are developed in this paper.

3.2. The Modelling. In this section, the way of characterizing the state of penetration is used as a didactic tool to explain the idea behind the ACOustic’s proposal. The behavior of ACO algorithm is described in terms of exploration and exploitation processes. According to the scheme described in Figure 2, the natural scheme in parasites-ants system translates into problem solving models as follows.

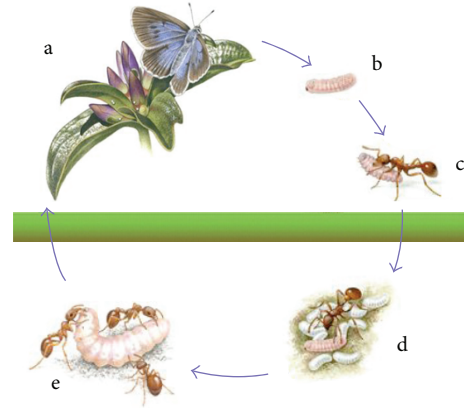


FIGURE 1: The *Myrmica* ants-*Maculinea* larvae system.

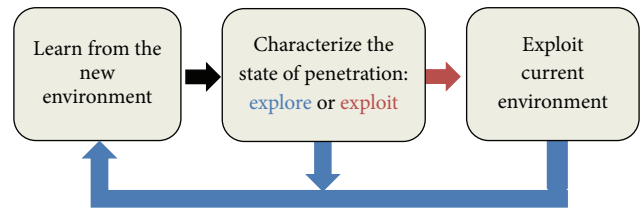


FIGURE 2: The scheme of acoustical indication in nature.

Let a construction graph $G = (N, A)$ represent a CO problem, where N is the set of nodes; A is the set of arcs; $|A| = a$ and $|N| = n$. The fitness landscape of the given CO problem is defined by the following: P is a population set which includes all solutions to the CO problem, where each solution $s \in P$ assigns a fitness value $f(s)$ and has a structure of neighborhood $N \subseteq P \times P$. A colony of artificial ants performs a biased walk in this landscape with the goal of finding low $f(s)$ (in case of minimization problems). The set $C_p(t)$ represents the collection of acoustics (sounds) that emanate from the landscape traversed by the ants of a perspective colony at time t where $C_p(t) \subseteq P(t) \times P(t)$ where c_i and c_{i+1} are two acoustics belonging to $C_p(t)$ where $c_i = \{x_1, x_2, \dots, x_a\}$; $c_{i+1} = \{y_1, y_2, \dots, y_a\}$ where the length of each acoustic signal is equal to a . The relatedness between two nest-mates is defined by the similarity between their acoustics. Two acoustics c_{i+1} and c_i are considered as being similar if their similarity neighborhood SN is below a predefined threshold X :

$$SN(c_{i+1}, C_p(t)) = \min d(c_{i+1}, c_i), \tag{4}$$

$$c \in C_p(t), \quad t = 0 \dots \text{current iteration and } c_{i+1} \neq c_i,$$

$$SN(c_{i+1}, C_p(t)) > X \quad (\text{exploration}), \tag{5}$$

$$SN(c_{i+1}, C_p(t)) \leq X \quad (\text{exploitation}), \tag{6}$$

where d is the Euclidian distance between two acoustics in $C_p(t)$ within Euclidian space R^n . The exploration occurs when SN of the two acoustics is greater than the boundary

of neighbourhood threshold (X); otherwise it is identified as exploitation.

A population-based memory scheme is used to record the best-iteration solutions produced by the algorithm during the run. An agglomerative clustering procedure is applied to the recorded population every ten (10) iterations. This is to determine the similarity features of the population through its acoustics during the past ten iterations. A matrix of distances is defined to conduct the clustering and then detect the number of clusters. The Euclidean distance d between c_i and c_{i+1} is a common way for finding similarity as follows:

$$d(c_i, c_{i+1}) = \sqrt{\sum_{j=1}^a (x_j - y_j)^2}. \quad (7)$$

The quantity d may have different magnitudes so that it is normalized to the size of population:

$$d_{\text{norm}} = \frac{d}{|P|}. \quad (8)$$

Three statistics medians (mean, variance, and standard deviation) are derived in (9), (10), and (11), respectively:

$$\text{mr}(t) = \sum_{i=1}^{\max-1} \sum_{j=i+1}^{\max} \frac{d_{\text{norm}}}{((\max^2 - \max)/2)}, \quad (9)$$

$$\text{vr}(t) = \sum_{i=1}^{\max-1} \sum_{j=i+1}^{\max} (C_{ij} - \text{mr})^2, \quad (10)$$

$$\text{stdr}(t) = \sqrt{\frac{\text{vr}}{(\max-1)}}, \quad (11)$$

where \max is the maximum size of the distance matrix. In order to minimize the computational efforts and keep the algorithm nonweights, the size of matrix fixes to (10). The agglomerative hierarchical technique uses for calculating the number of clusters are as follows:

$$C_{\text{Num}}(C_p(t)) = |L(C_p(t))|, \quad (12)$$

where L is the set of clusters resulting from the solutions visited by the ants. The statistics and clustering information are combined. The relatedness between ants can be calculated by finding the difference between the mean of distances and the number of clusters by the standard deviation of distances as follows:

$$\text{rltdnss} = \frac{(\text{mr} - C_{\text{Num}})}{\text{stdr}}. \quad (13)$$

The definition of exploration and exploitation in (5) and (6) can be reformulated based on the relatedness between acoustics (rltdnss) as follows:

$$\begin{aligned} \text{rltdnss} > NC_{\text{threshold}} & \quad (\text{exploration}), \\ \text{rltdnss} \leq NC_{\text{threshold}} & \quad (\text{exploitation}), \end{aligned} \quad (14)$$

where $NC_{\text{threshold}}$ is the lowest degree of clustering. It is detected by capturing the first value of rltdnss within the first ten iterations. For instance, when C_{Num} is decreased from 10 to 8, this indicates that the exploitation has occurred. In contrast, if C_{Num} stays as it is, this indicates that the exploration has occurred. The assignment of $NC_{\text{threshold}}$ has to be complete within the first ten iterations. Hereafter, each new value of rltdnss will be characterized as either exploration or exploitation accordingly.

3.3. The Implementation. This section walks through the implementation of ACOustic algorithm. The pseudocode of the algorithm is illustrated in Pseudocode 1.

The nearest neighborhood threshold X is entered, the vector of acoustics clusters C_i is defined, and other variables such as miniDist, C_{Num} , $NC_{\text{threshold}}$, and \max are initialized as in Procedure 1.

In findSimilarities algorithm, a matrix of Euclidian distances between acoustics is generated and the statistical medians are calculated as in Pseudocode 2.

In determineRelatedness, the minimum distance miniDist is calculated from the distance matrix that is generated earlier. The nearest two clusters are united, the distance matrix is recalculated, and finally miniDist and the number of clusters C_{Num} are updated as in Pseudocode 3.

In Pseudocode 3, the number of clusters and the statistics collected earlier are combined and returned as a relatedness quantifier denoted as rltdnss.

4. Experimental Setting

In this section, the experimental design for applying the ACOustic for several standard ACO algorithms is described. The aim of the application is to examine the ability of ACOustic for the following: (i) monitoring its performance against different ACO algorithms, (ii) monitoring its performance against different fitness landscapes, and (iii) evaluating its performance against the state-of-the-art measurement tools in ACO. The implemented ACO algorithms are ant system (AS), elitist ant system (EAS), ant colony system (ACS), rank-based ant system (RAS), max-min ant system (MMAS), and best-worst ant system (BWAS). Two different fitness landscapes are used: travelling salesman problem (TSP) and quadratic assignment problem (QAP). The performance is reported to have been compared with average λ -branching and exploration quantifier as measurement tools. In the comparisons, the effect of the parameters of MMAS algorithm on the exploration and exploitation mechanisms is analyzed; several scenarios have been considered and the effect of the raggedness of fitness landscape is analyzed.

The analyzed parameters are pheromone intensity (α), preheuristic importance (β), evaporation rate (ρ), and the number of ants (m). The parameters setting suggested by Pellegrini et al. [24] has been considered. The stopping criteria are either the completion of 350 sec (only the first 3000 iterations are reported) for large instances or finding the optimal solution for small instances. The C coding is used in the implemented algorithms. The experiments are conducted on a Windows 8 64-bit operating system, processor Intel Core

```

Procedure initialization()
Input:  $X$ 
Define:  $C = \{C_1, C_2, \dots, C_{|a|}\} = \{\{c_1\}, \{c_2\}, \dots, \{c_{|a|}\}\}$ 
Initialize: miniDist,  $C_{Num}$ ,  $NC_{threshold}$ , max
end-procedure
    
```

PROCEDURE 1: The initialization procedure.

```

ACOustic()
initialization()
while (not terminate()) do
    {mr, stdr} = findSimilarities()
    rltdnss = determineRelatedness (mr, stdr)
end-while
Return rltdnss
    
```

PSEUDOCODE 1: The pseudocode of ACOustic algorithm.

```

findSimilarities()
foreach  $C_h, C_k \in C$  do
     $D = d_{c_h, c_k} = \sqrt{\sum_{i=1}^a (x_i^h - x_i^k)^2}$ 
end-foreach
 $mr = (\sum_{i=1}^{\max-1} \sum_{j=i+1}^{\max} (D_{i,j}/m)) / ((\max^2 - \max)/2)$ 
 $vr = \sum_{i=1}^{\max-1} \sum_{j=i+1}^{\max} (C_{ij} - mr)^2$ 
 $stdr = \sqrt{vr / \max - 1}$ 
    
```

PSEUDOCODE 2: The pseudocode of findSimilarities algorithm.

```

determineRelatedness()
miniDist =  $\min_{c_h, c_k \in C} d_{c_h, c_k}$ 
repeat
     $C_h = C_k \cup C_h$ 
     $C = C \setminus \{C_k\}$ 
foreach  $C_w \in C \setminus \{C_h\}$  do
     $d_{c_h, c_w} = d_{c_w, c_h} = \min\{d_{c_h, c_w}, d_{c_k, c_w}\}$ 
end-foreach
miniDist =  $\min_{c_h, c_k \in C} d_{c_h, c_k}$ 
 $C_{Num} = |C|$ 
until (miniDist  $\leq X$ )
rltdnss  $\leftarrow mr - C_{Num}/stdr$ 
return rltdnss
    
```

PSEUDOCODE 3: The pseudocode of determineRelatedness algorithm.

i3-3217U with CPU @ 1.80 GHz, RAM 4 GB. Each experiment is executed ten times to avoid the stochastic behavior. The main results of this application are depicted as below. The TSP instances used in the experiments are selected from TSPLIB repository and from the 8th DIMACS challenge. Following the TSPLIB format, d198 instance is selected. Following the DIMACS format, one random instance is generated using *portgen*; the instance generator is adopted in the 8th DIMACS

TSP challenge. It is generated with size = 2000 and seed = 39200. The *kra30a.qap* instance used in the experiments is selected from QAPLIB repository.

5. Computational Results

This section presents the results of two types of experiments that have been conducted to evaluate the performance of

ACOustic indicator. The first type of results reported the robustness of the proposed tool against the difference in the raggedness of TSP and QAP fitness landscapes (see Figures 3–8). The second type of results reported the ability of the proposed tool to analyze the convergence behavior under two conditions, the convergence of different ACO algorithms (see Figure 9) and the convergence of different parameters settings for MMAS, the prominent ACO algorithm (see Figures 10–13).

In Figures 3–8, the y -axis visualizes the explorative behavior of MMAS using the ACOustic indicator compared with λ -branching indicator and exploration indicator. The λ -branching indicator is selected for comparison due to its theoretical and practical importance as a traditional indicator in ACO literature. The exploration indicator is selected because of its similarity to ACOustic in the context of being a machine learning method. The x -axis represents the number of objective function evaluations. The general performance of the MMAS algorithm is reported for TSP and QAP.

Figures 3–5 showed that when different values of neighborhood threshold are used, the proposed indicator gives the same insights into the shape of fitness landscape for TSP that other traditional indicators gave. In fact, traditional indicators have no problem of robustness when applied for TSP problem solving. Therefore, when 80%, 70%, or 60% of the components of one TSP solution do not exist in the other solution, the indication is similar. This confirms that ACOustic retains the same quality of indication of other indicators.

Figures 6–8 showed that, for QAP, the ACOustic indicator shows significant robustness against the branching indicator and exploration indicator. In Figure 6, when 80% of the components of one QAP solution do not exist in the other solution, the indication of ACOustic and exploration ranges between zero (0) and eight (8). In Figure 7, when the rate is deduced to 70%, the disparity in the indication of ACOustic and exploration begins where the range of the exploration amount in the latter indicator tends to be more towards ten (10). In Figure 8, when the rate is reduced to 60%, the sensitivity for exploration behaviour was at its lowest level. On the other hand, the branching factor in all three cases is the same, that is, equal to one (1), while the relatedness of ACOustic ranges between four (4) and eight (8).

As shown, different values of neighborhood threshold give the same insights into the shape of fitness landscape for QAP that other traditional indicators did not give. In fact, the behavior of ACOustic relies strongly on the normalized quantities of the distance between solutions and the number of clusters. These are statistical and clustering information about the current population. Hence, by normalization, the influence of the difference in magnitudes of the distance matrix has been reduced. This confirms the problem of robustness in traditional indicators when applied for QAP problem solving.

In Figures 9–13, the general performance of AS, EAS, RAS, ACS, and BWAS algorithms and the effect of parameters on MMAS behavior are reported. The y -axis visualizes the exploration using ACOustic indicator and λ -branching indicator for TSP. The x -axis represents the number of iterations.

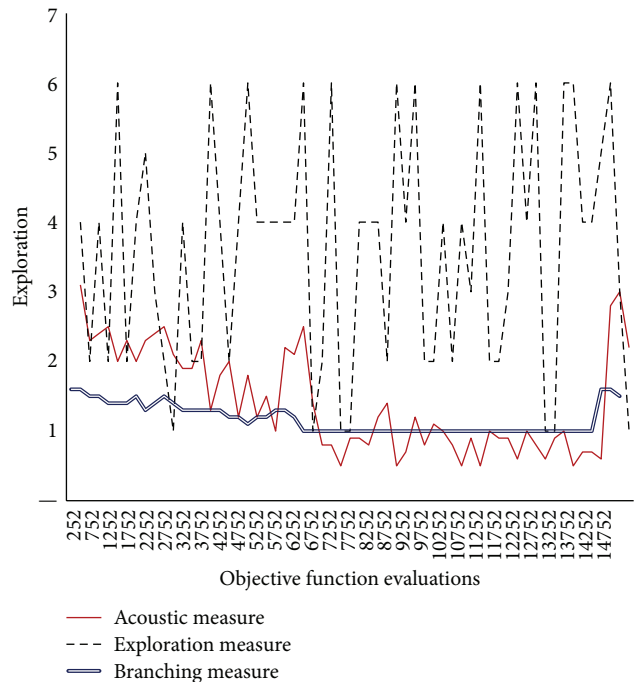


FIGURE 3: Evaluating ACOustic against traditional indicators in ACO for the TSP instance d198 with nearest neighbourhood threshold = ϵ 8.

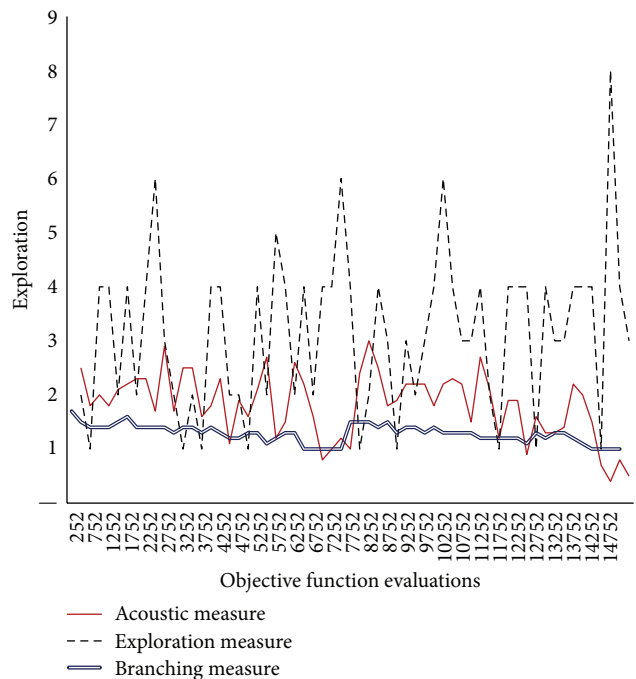


FIGURE 4: Evaluating ACOustic against traditional indicators in ACO for the TSP instance d198 with nearest neighbourhood threshold = ϵ 7.

In Figure 9, the results of analyzing using the proposed indicator showed that AS tends to be a very explorative algorithm. The rest of the tested algorithms either start with

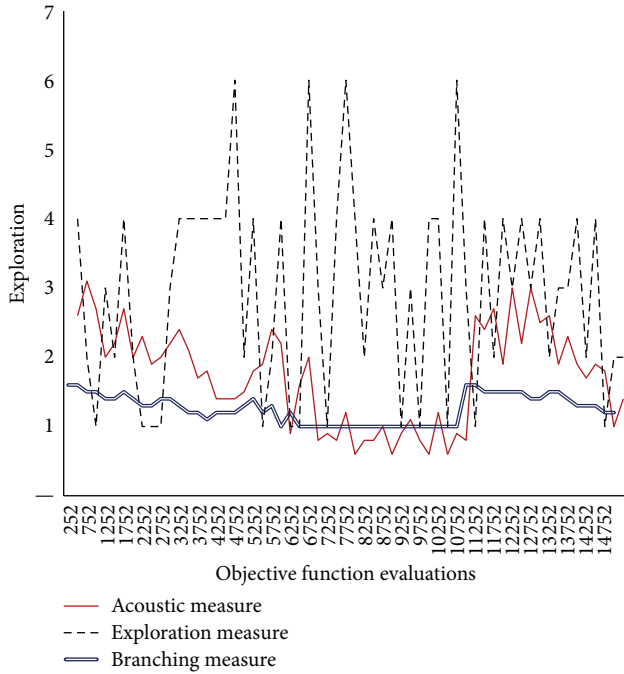


FIGURE 5: Evaluating *ACO*ustic against traditional indicators in *ACO* for the TSP instance d198 with nearest neighborhood threshold = $\epsilon 6$.

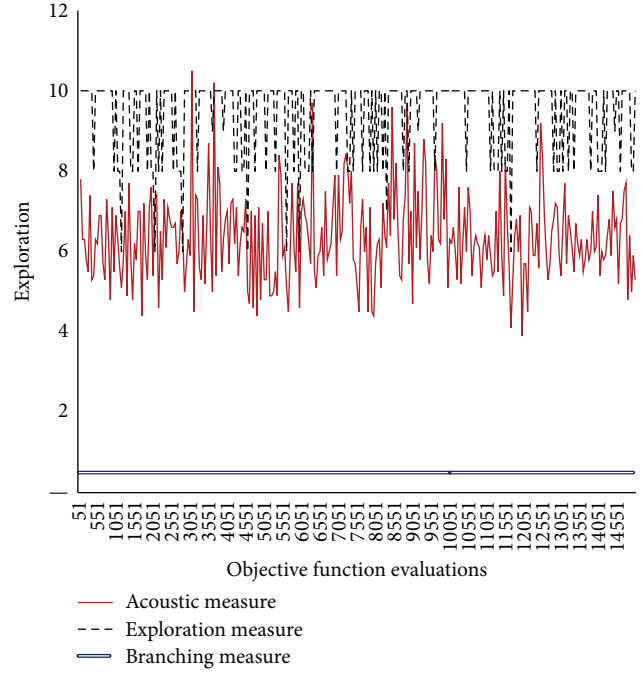


FIGURE 7: Evaluating *ACO*ustic against traditional indicators in *ACO* for the QAP instance kra30a with nearest neighborhood threshold = $\epsilon 7$.

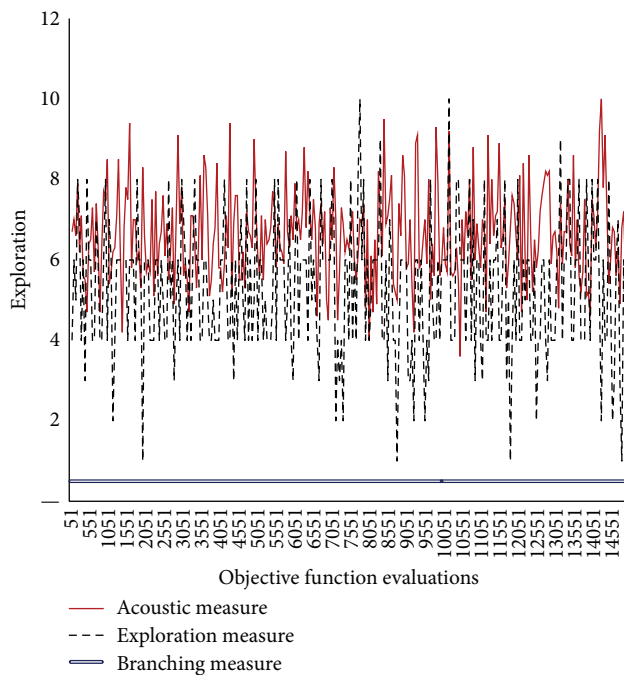


FIGURE 6: Evaluating *ACO*ustic against traditional indicators in *ACO* for the QAP instance kra30a with nearest neighborhood threshold = $\epsilon 8$.

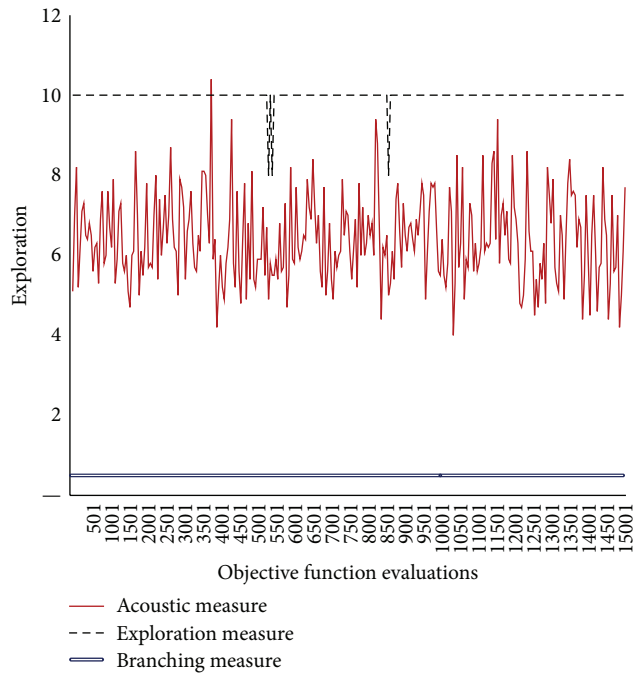


FIGURE 8: Evaluating *ACO*ustic against traditional indicators in *ACO* for the QAP instance kra30a with nearest neighborhood threshold = $\epsilon 6$.

a very short exploration phase followed by a very aggressive exploitation phase (e.g., EAS and RAS) or skip the initial exploration phase (e.g., ACS and BWAS). This is mainly

achieved by a stronger emphasis given to the best tours found during the search. In general, when compared with λ -branching indicator, the proposed indicator draws the same shape for TSP.

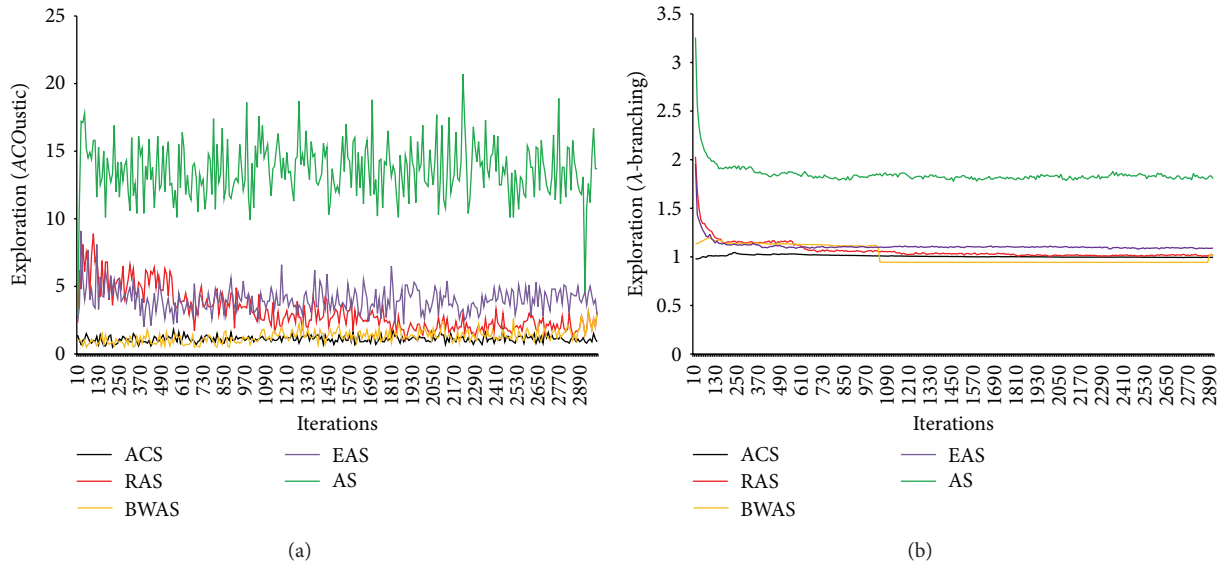


FIGURE 9: The performance of five ACO algorithms using the ACOustic (a) and branching factor indicator (b) for a DIMACS instance size = 2000 and seed = 39200 generated using portgen.

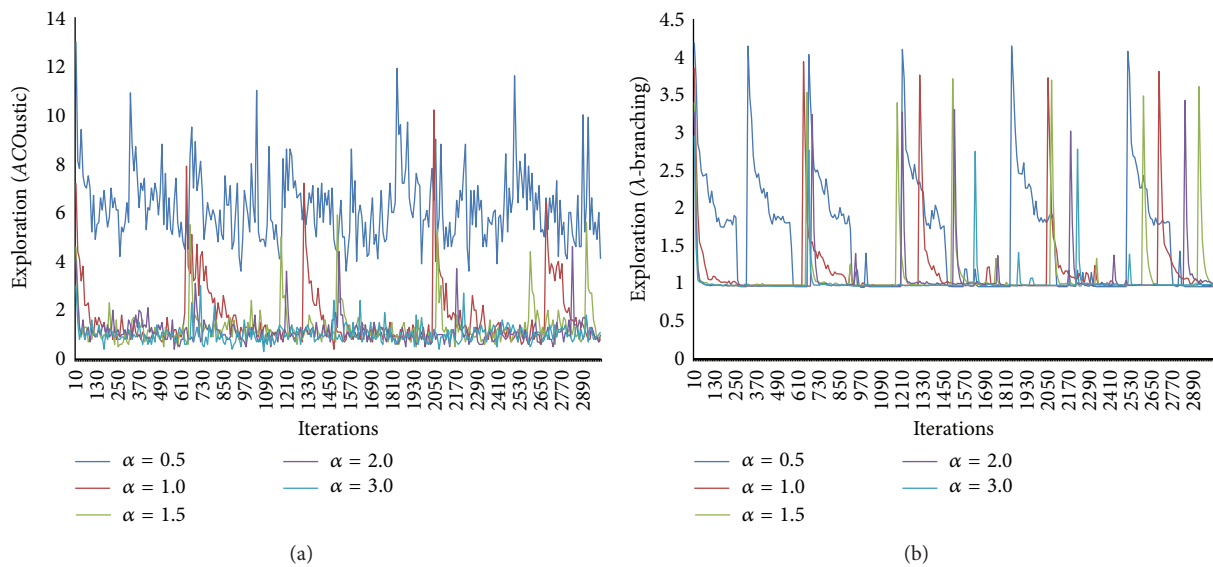


FIGURE 10: The effect of pheromone intensity (alpha) measured by ACOustic (a) and measured by λ -branching (b).

In Figure 10, the effect of varying the main parameters on the explorative and exploitative behaviors of MMAS algorithm is characterized. The proposed indicator can detect the relationship between parameter values and local search. Figure 10(a) showed clearly that the higher the value of pheromone intensity, the lower the exploration. The value $\alpha = 1.0$ is the ideal value to achieve a moderate behavior.

In Figure 11, the influence of preheuristics parameter is tested. The higher the value of the parameter (β) is, the more greedy the behavior is recorded to become at its peak when ($\beta = 10$).

In Figure 12, the evaporation ratio as a trail learning factor has distinct impact on the way of search. Increasing the value of ρ results in slow learning of pheromone trail parameters, and thereby the chance of forgetting the previous search

experience will increase. In this way, the value of $\rho = 0.7$ leads to the increase of exploitation while the value of $\rho = 0.5$ seems ideal.

Figure 13 showed that the exploration decreases with respect to the number of ants. The greater the number of ants, the lower the number of iterations performed in a run and consequently the lower the number of different probability distributions used. A high value of m implies that the likely edges are often the same. In general, when compared with λ -branching indicator, the proposed indicator draws the same shape for TSP which reflects the explorative/exploitative behavior influenced by parameter tuning.

As shown, these insights are compatible with the common beliefs among ACO researchers. The ACOustic indicator is a very convenient tool for characterizing the exploration

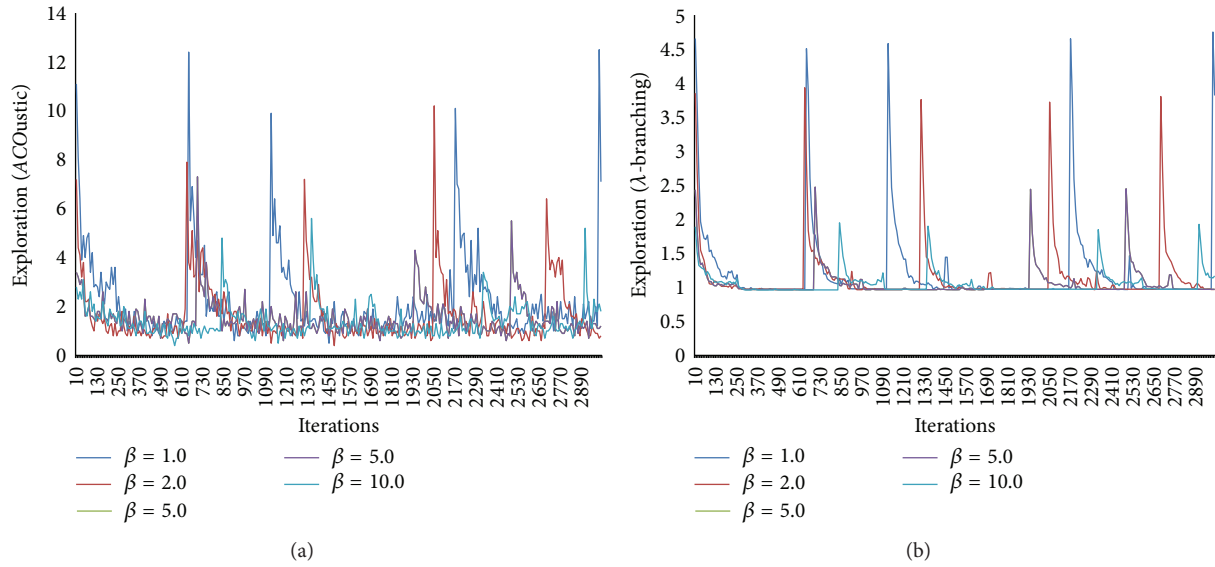


FIGURE 11: The effect of preheuristic (beta) measured by ACOustic (a) and measured by λ -branching (b).

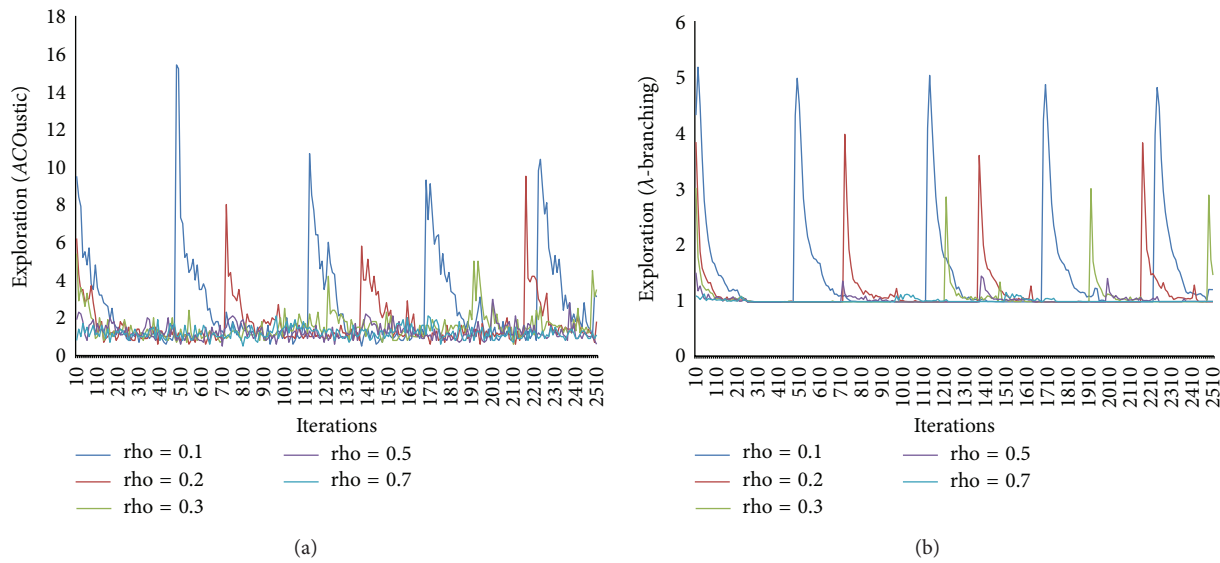


FIGURE 12: The effect of preheuristic (beta) measured by ACOustic (a) and measured by λ -branching (b).

on-demand. This conclusion came as a result of its robustness against the state-of-the-art indicators in ACO and of using the statistical machine learning paradigm.

6. Conclusion

Towards a reactive-based ACO search, the ACOustic is proposed. It is a nature-inspired exploration indicator that helps in measuring the amount of exploration, detecting stagnation, tuning the algorithm, adapting parameters in an online manner, and analyzing the behavior of the algorithm. The biological idea that inspired the proposed indicator

is presented, modelled, and implemented. The comparison with the state-of-the-art indicators in ACO showed that the ACOustic is more informative and more robust.

For future work, the proposed exploration measurement will be harnessed in automating the exploration and exploitation in reactive-based ACO algorithms. The proposed tool can be easily applied for other metaheuristics and local search algorithms.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

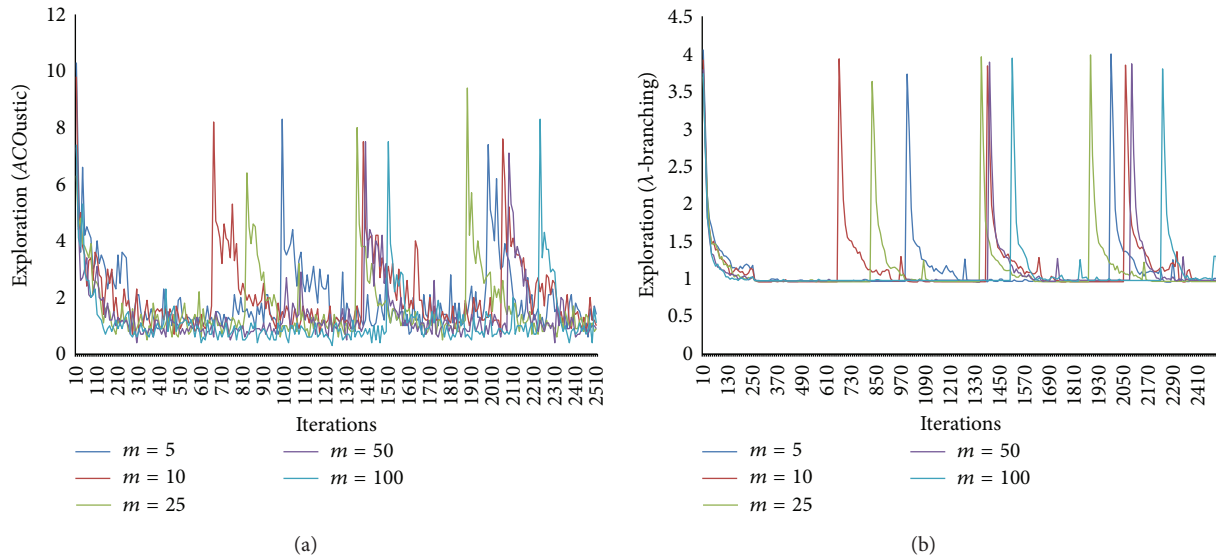


FIGURE 13: The effect of number of ants (m) measured by ACOustic (a) and measured by λ -branching (b).

Acknowledgment

The authors wish to thank the School of Computing in Universiti Utara Malaysia, Kedah, Malaysia, for the funding and administration of this study.

References

- [1] G. Beni and J. Wang, "Swarm intelligence in cellular robotic systems," in *Robots and Biological Systems: Towards a New Bionics?* P. Dario, G. Sandini, and P. Aebischer, Eds., vol. 102, pp. 703–712, Springer, Berlin, Germany, 1993.
- [2] M. Dorigo, V. Maniezzo, and A. Colorni, "Positive feedback as a search strategy," Tech. Rep. 91-016, Dipartimento di Elettronica, Politecnico di Milano, Laboratorio di Calcolatori, Milano, Italy, 1991.
- [3] C. Solnon, *Ant Colony Optimization and Constraint Programming*, John Wiley & Sons, New York, NY, USA, 2010.
- [4] M. Gendreau and J.-Y. Potvin, *Handbook in Metaheuristics*, Springer, New York, NY, USA, 2010.
- [5] P. Pellegrini and D. Favaretto, "Quantifying the exploration performed by metaheuristics," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 24, no. 2, pp. 247–266, 2012.
- [6] R. Battiti, M. Brunato, and F. Mascia, *Reactive Search and Intelligent Optimization*, vol. 45 of *Operations Research/Computer Science Interfaces Series*, Springer, New York, NY, USA, 2009.
- [7] H. Neyoy, O. Castillo, and J. Soria, "Fuzzy logic for dynamic parameter tuning in ACO and its application in optimal fuzzy logic controller design," in *Fuzzy Logic Augmentation of Nature-Inspired Optimization Metaheuristics*, O. Castillo and P. Melin, Eds., vol. 574 of *Studies in Computational Intelligence*, pp. 3–28, Springer, 2015.
- [8] J. P. Queiroz dos Santos, J. D. de Melo, A. D. Duarte Neto, and D. Aloise, "Reactive search strategies using reinforcement learning, local search algorithms and variable neighborhood search," *Expert Systems with Applications*, vol. 41, no. 10, pp. 4939–4949, 2014.
- [9] H. R. Ahmed, "An efficient fitness-based stagnation detection method for particle swarm optimization," in *In Proceedings of the conference companion on Genetic and evolutionary computation companion (GECCO Comp '14)*, pp. 1029–1032, ACM Press, New York, NY, USA, 2014.
- [10] R. Sagban, K. R. Ku-Mahamud, and M. S. A. Bakar, "Reactive memory model for ant colony optimization and its application to TSP," in *Proceedings of the 4th International Conference on Control System, Computing and Engineering (ICCSCE '14)*, IEEE, Penang, Malaysia, 2014.
- [11] D. Favaretto, E. Moretti, and P. Pellegrini, "On the explorative behavior of MAX-MIN ant system," in *Engineering Stochastic Local Search Algorithms. Designing, Implementing and Analyzing Effective Heuristics*, T. Stützle, M. Birattari, and H. H. Hoos, Eds., vol. 5752 of *Lecture Notes in Computer Science*, pp. 115–119, Springer, Berlin, Germany, 2009.
- [12] C. Solnon and S. Fenet, "A study of ACO capabilities for solving the maximum clique problem," *Journal of Heuristics*, vol. 12, no. 3, pp. 155–180, 2006.
- [13] P. Pellegrini, D. Favaretto, and E. Moretti, "Exploration in stochastic algorithms: an application on MAX-MIN ant system," in *Nature Inspired Cooperative Strategies for Optimization (NICSO 2008)*, N. Krasnogor, M. B. Melián-Batista, J. A. M. Pérez, J. M. Moreno-Vega, and D. A. Pelta, Eds., vol. 236 of *Studies in Computational Intelligence*, pp. 1–13, Springer, Berlin, Germany, 2009.
- [14] F. Barbero, S. Bonelli, J. A. Thomas, E. Balletto, and K. Schönrogge, "Acoustical mimicry in a predatory social parasite of ants," *The Journal of Experimental Biology*, vol. 212, no. 24, pp. 4084–4090, 2009.
- [15] M. Dorigo and T. Stützle, *Ant Colony Optimization*, MIT Press, Cambridge, Mass, USA, 2004.
- [16] P. Pellegrini, *ACO: parameters, exploration and quality of solutions [dissertation]*, Università Ca' Foscari Venezia, Venezia, Italy, 2006.
- [17] S. Colas and N. Monmarché, "Artificial ants for the optimization of virtual keyboard arrangement for disabled people," in *Artificial Evolution*, N. Monmarché, E.-G. Talbi, P. Collet, M.

- Schoenauer, and E. Lutton, Eds., vol. 4926 of *Lecture Notes in Computer Science*, pp. 87–99, Springer, Heidelberg, Germany, 2008.
- [18] C. Blum, “ACO applied to group shop scheduling: a case study on intensification and diversification,” in *Ant Algorithms*, M. Dorigo, G. Di Caro, and M. Sampels, Eds., vol. 2463 of *Lecture Notes in Computer Science*, pp. 14–27, Springer, Berlin, Germany, 2002.
- [19] C. Blum and M. Dorigo, “The hyper-cube framework for ant colony optimization,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 2, pp. 1161–1172, 2004.
- [20] T. Stützle, *Local search algorithms for combinatorial problems: analysis, improvements, and new applications [Doctoral dissertation]*, Technische Universität Darmstadt, Germany, 1999.
- [21] L. M. Gambardella and M. Dorigo, “Ant-Q: a reinforcement learning approach to the traveling salesman problem,” in *Proceedings of the 12th International Conference on Machine Learning*, A. Prieditis and S. J. Russell, Eds., The Morgan Kaufmann Series in Machine Learning, pp. 252–260, Morgan Kaufmann, Boston, Mass, USA, 1995.
- [22] H. Neyoy, O. Castillo, and J. Soria, “Dynamic fuzzy logic parameter tuning for ACO and its application in TSP problems,” in *Recent Advances on Hybrid Intelligent Systems: Studies in Computational Intelligence*, O. Castillo, P. Melin, and J. Kacprzyk, Eds., vol. 451, pp. 259–271, Springer, Berlin, Germany, 2013.
- [23] F. Barbero, D. Patricelli, M. Witek et al., “Myrmica ants and their butterfly parasites with special focus on the acoustic communication,” *Psyche*, vol. 2012, Article ID 725237, 11 pages, 2012.
- [24] P. Pellegrini, T. Stützle, and M. Birattari, “A critical analysis of parameter adaptation in ant colony optimization,” *Swarm Intelligence*, vol. 6, no. 1, pp. 23–48, 2012.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

