

AS-PSO, Ant Supervised by PSO Meta-heuristic with Application to TSP.

Nizar Rokbani^{*1}, Arsene L. Momasso^{*2}, Adel.M Alimi^{*3}

[#]REGIM-Lab, Research Groups on Intelligent Machine

University of Sfax, Tunisia

¹nizar.rokbani@ieee.org

³adel.alimi@ieee.org

Abstract— Optimization problems occupy a prominent place in engineering, management, process control, it consist in finding a fair solution of a problem in accordance with specific environment, and respecting a given list of constraints. Whether it is to determine the best chemical balance of a product, or to predict future market trends, we need optimization methods. Bio-inspired techniques and swarm intelligence as well as various numerical techniques, are used to solve problems increasingly difficult. This paper investigates a new Meta-heuristic, called AS-PSO, Ant Supervised by Particle Swarm Optimization, based on the famous ant colony, ACO, and particle swarm optimization. AS-PSO is an adaptive heuristic, since the user is not in need to fit any parameter of the search strategy. In AS-PSO, the ACO algorithm is in charge with the problem solving, while the PSO is managing the optimality of the ACO parameters. The paper includes also an application of AS-PSO to the travelling Salesman Problem (TSP).

Keywords— PSO, ACO, AS-PSO, Optimization, Heuristics, TSP

I. INTRODUCTION

The importance of optimization in economics, industry, science and engineering is well established today. Indeed, we live in a world more and more optimized. The main challenges is to find a quality result and/or a reliable prediction for a specific problem what ever its nature, Optimization process should be adapted in order to fit the problem requirements' and specific attributes [1], That depends essentially on the main topic of application example robotics, mechanical design, soft computing ...etc.

To find an optimal solution, several approaches could be investigated, including exact algorithms that work in order to have an optimal solution, with the disadvantage of reaching slowly to the solution, and heuristics that provide a feasible solution quickly, but not necessarily optimal, it is in this context that ant colony and particle swarm optimization are involved.

Searching for an optimum, is a challenging task starting by defining what makes a solutions optimal, optimality is quantified using mathematical and numerical criteria that should be fitted with respect to constraints, These criteria are expressed as a set of mathematical functions, also called objective functions.

Form this point of view, a problem could be solved by different techniques including heuristics and several analyses are needed to consider the best solution among them. Even

within a specific class of heuristics, the optimality of a solution depends on the heuristic parameters; adaptive methods are a sub-class of optimizer's able to self tune their parameters.

Within optimization methods we distinguish deterministic and probabilistic algorithms; PSO, particle swarm optimization [2], and ACO, Ant Colony Optimization [3], are probabilistic methods; they belong also evolutionary techniques.

The hybrid methods can be classified into two groups: the hybrid metaheuristics that combine several heuristics [4], and the group combining exact methods and heuristic ones. It is possible to classify the hybrid methods, following the taxonomy of Talbi [ref] that provides qualitative comparison of hybrid methods. The hierarchical heuristics are seen as :

- Low-level hybridization, in witch a function of an heuristic is replaced by another heuristic.
- High_level hybridation, where two heuristics are hybridized without their internal functioning is related.

Within these classes, two additional mechanisms could be involved:

- The Hierarchical mechanism, when heuristics are executed sequentially, one using the output of the previous as an input, there is hybridization with relay.
- Co-evolutionary mechanism, qualify an evolution in witch agents cooperate in parallel to explore the space of solutions. [5].

In this paper and according to the previous classification, AS-PSO, is observed as High-level Hierarchical, HLH, meta-heuristic. Initially the AS-PSO were proposed by Elloumi et al, 2009 in [6], in this paper, we investigate an aspect of AS-PSO, focusing only on the self adaptation of (α, β) of the ACO, with a constrain weight PSO. The remaining of this paper is organized as follows : Paragraph II, review the PSO and ACO algorithms. Paragraph III, is dedicated to the presentation of AS-PSO, then the application of AS-PSO to the travelling salesman problem, around Tunisian cities, is presented. The paper is ended by results discussions and further works openings.

II. PSO AND ACO

A. Particle swarm Optimization

This optimization method is based on collaboration between individuals. It also has some similarities with the algorithm of ant colony, which also relies on the concept of self-organization. The idea is that a group of individuals which have a little intelligence can create a complex global organization. Thus, through simple rules of movement (in the space of solutions), the particles can gradually converge to a local minimum. Eberhart and Kennedy were inspired by socio-psychological behavior of certain groups of animals such as migratory birds, schools of fish and bees to create the ‘‘psa’’. A swarm of particles, which are the potential solutions to the optimization problem, ‘‘flies’’ the search space, searching for the global optimum.

Each particle is informed of the best known point in its neighbourhood and it will tend to move towards this point and the best of the swarm towards which all particles. The equations implemented for the movement of particles are :

$$\vec{V}_{i+1} = w\vec{V}_i + c_1 * rand() * (\vec{x}_{lbest} - \vec{x}_i) + c_2 * rand() * (\vec{x}_{gbest} - \vec{x}_i) \quad (1)$$

$$\vec{x}_{i+1} = \vec{x}_i + \vec{V}_{i+1} \quad (2)$$

In equation (1) w is the inertia term that moderates the current position of the particle, The second term, is the cognitive component of displacement. $c1$ controls cognitive behaviour of the particle. The impact of this component is that the particle tends to move towards the best by which it has already passed.

The third term is the social component of displacement. $c2$ controls the social competence of the particle. The particle tends to rely on the experience of its congeners, and thus to move towards the best already reached by its neighbours.

At the beginning of the algorithm, the particles are dispersed randomly or not in space research. Then, the particles form a ‘‘bench’’ and explore the search space while maintaining cohesion between them and gathering around the optimum. They no longer run away from this optimum. Depending on the configuration of the algorithm, the particles end up in the same spot, which highlights a global trend to move towards the optimum.

B. Ant Colony Optimization.

Ant colony optimization (ACO) is inspired from the natural behaviours of ants’ colony and their capacity in solving food search and management problem. The ACO focus on the use of phenomenon in order to mark the prospective paths, which should be followed by other members of the colony. It aims to iteratively achieve the optimal solution of a target problem through a guided search (i.e. the movements of a number of ants) over the solution space. The pheromone is used as a tagging procedure and also as a communication medium between colony members.

Assume that (K ants) are involved to find the optimal solution in a space χ that consists of $M1 \times M2$ nodes. the procedure of ACO can be summarized as in Fig 1.

In ACO, the displacement of an ant depends on the transition matrix $p(n)$, and the pheromone state matrix $\tau(n)$. The nodes of the transition matrix are computed iteratively using equation (3)., they define the probabilistic action rules.

$$p_{i,j}^{(n)} = \frac{(\tau_{i,j}^{(n-1)})^\alpha (\eta_{i,j})^\beta}{\sum_{j \in \Omega_i} (\tau_{i,j}^{(n-1)})^\alpha (\eta_{i,j})^\beta}, \text{ if } j \in \Omega_i \quad (3)$$

Where $\tau(i,j)$ represents the pheromone value of the arc linking the node (i) to the node (j); Ω_i is the neighbourhood nodes for a given ant $a(k)$. The constants α and β represent the moderation of pheromone information and heuristic information $\eta(i,j)$, allowing to go from node (i) to node (j). The pheromone matrix is updated during the ACO procedure, this update is performed after the displacement of ants, according to equation (4), where ρ is the evaporation rate.

$$\tau_{i,j}^{(n-1)} = \begin{cases} (1-\rho) \cdot \tau_{i,j}^{(n-1)} + \rho \cdot \Delta_{i,j}^{(k)}, & \text{if } (i,j) \text{ belongs to the best tour} \\ \tau_{i,j}^{(n-1)}, & \text{otherwise.} \end{cases} \quad (4)$$

Furthermore, the determination of best tour is subject to the user-defined criterion, it could be either the best tour found in the current construction-step, or the best solution found since the start of the algorithm, or a combination of both. The second update is performed after the move of all (k) ants within each construction-step; and the pheromone matrix is updated as:

$$\tau^{(n)} = (1 - \psi) \cdot \tau^{(n-1)} + \psi \cdot \tau^{(0)} \quad (5)$$

(Ψ) is the pheromone decay coefficient. Note that the ant colony system performs two update operations for updating the pheromone matrix, while the ant system only performs one operation.

Algorithm 1 Ant colony	
ACO (K, N : Integer)	
1	Initialize the positions of totally K ants, as well as the pheromone matrix $\tau^{(0)}$
2	For the construction-step index $n = 1 : N$
•	For the ant index $k = 1 : K$
•	Consecutively move the k -th ant for L steps, according to a probabilistic matrix $p^{(n)}$ (with a size of $M_1 M_2 \times M_1 M_2$)
•	update the pheromone matrix $\tau^{(n)}$
3	Make the solution decision according to the final pheromone matrix $\tau^{(N)}$

Fig. 1 ANT algorithm.

III. AS-PSO, ANT SUPERVISED BY PSO.

A. Motivation of AS-PSO.

The hybrid met-heuristics have emerged along with the paradigm itself. They are gaining popularity now because they have worked well in some hard optimization problems such as routing of the cars.

To design swarm systems, the main difficulty that arises is to determine the individual behaviour, the environment and the dynamic that will govern the operation of the system to produce the desired collective response. The methods of swarm intelligence have been very successful in the field of optimization, which is of great importance for industry and science. The most popular, because of their variety and quantity of reference documents, are probably particle swarm optimization and ant

In fact, the main idea that underlies the design of hybrid algorithms is simple: for a given optimization problem, we have two algorithms, each with its strengths and weaknesses. We want to create a powerful algorithm that combines the strengths of both, or improve an algorithm using the other to optimize its parameters.

B. The AS-PSO meta-heuristic.

Separately, PSO and ACO showed great potential in solving a wide range of optimization problems. Why not use both to solve optimization problems? This is what Elloumi et al have tried to do in [6]. The idea is to allow PSO to optimize the optimizer (ACO), knowing that ACO is used for discrete problems and PSO generally for continuous problems, and considering their strengths and their weaknesses. The figure 1 shows roughly the organization of AsPSO.

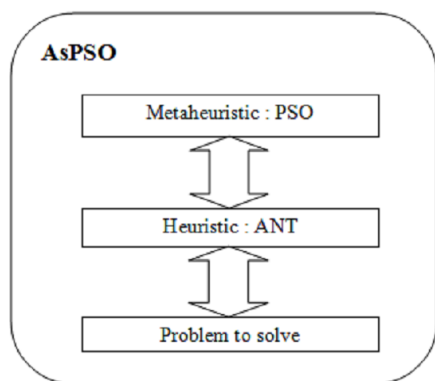


Fig. 2 AS-PSO Architectural Overview.

The heuristic is directly related to the physical problem and try to solve it, while the heuristic adjusts the parameters of the heuristics. So the operation of the heuristic, ACO in our case, is either guided or supervised by the meta-heuristic, PSO. From this organization, we can deduce several variants of AS-PSO.

The running of the classical ACO is based on parameters that are often set by the user of the algorithm. Thus, to find parameters that are appropriate for a problem, the user needs to perform many tests. With AsPSO, the user does not need to

look ACO settings by trial and error. Indeed, the system is responsible for providing all the necessary parameters using PSO. Convergence may refer to the swarm's best known position approaching (converging to) the optimum of the problem, regardless of how the swarm behaves. Convergence may also refer to a swarm collapse in which all particles have converged to a point in the search-space, which may or may not be the optimum. Figure 2, shows the flowchart of the AS-PSO algorithm in which we can identify the various stages of operation of the algorithm.

Algorithm 2 Ant Supervised by PSO

AsPSO (InitialSettings: settings; N: Integer)

- 1 Initialize the settings of each PSO particle
- 2 For step index $n = 1 : N$
 - For each particle index $k = 1 : K$
Consecutively runs ACO ($settings_k$)
 - Update all the settings
- 3 Save the best settings.

Fig. 3 AS-PSO Algorithm overview

During the initialization, the functional parameters of the PSO algorithm are enabled. Each particle provides a setting. ACO retrieves these parameters and runs in order to find the solution from the parameters proposed by each particle.

Once the research phase of the solution is complete, each setting is updated using the PSO and the best solution are saved. The PSO here help to ACO finding its best parameters. And the PSO particle stands as :

$$X = [\alpha, \beta] \quad (6)$$

Classically in ACO, the constants α and β represent the influence of pheromone information and heuristic information. In AS-PSO they are no more constant but subject to an optimization process in witch PSO is used.

IV. APPLICATION OD AS-PSO TO TSP PROBLEM

A. The TSP Problem.

The travelling salesman problem, TSP, is one which has commanded much attention of mathematicians and computer scientists, TSP is so easy to describe and so difficult to solve. The problem can simply be stated as: if a travelling salesman wishes to visit exactly once each of a list of n cities and then return to home city, what is the least costly route the travelling salesman can take [7]?

The figure 4 is an illustration of this problem with 5 cities. This figure shows two possible solutions, one in red and the other in green colour. The two routes don't have the same length. A travelling salesman will choose the shortest path to reduce the cost of the travel. However, the TSP is said np-complete. In fact, for n cities the number of possible route is equal to $(n - 1)!/2$.

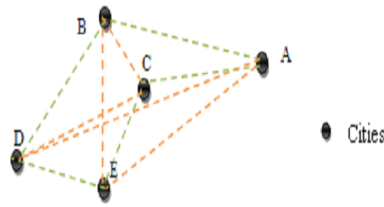


Fig. 4. TSP possible solutions for a simplified cities representation, here the number of cities is limited to 5.

To get an idea of what might make this algorithm in practice, merchant has to move in 17 cities of Tunisia. He would like to minimize the cost of transport (assuming also that the shortest path is the least expensive). AS-PSO algorithm is asked to give the most advantageous path, solution is displayed in figure 5.



Fig. 5. AS-PSO applied to TSP of 17 Tunisian cities.

B. Comparative results Ant System to AS-PSO.

In order to evaluate the AS-PSO performance, a comparative study is conducted to the original Ant System algorithm, AS. Results of each algorithm for solving the problem, when the travelling salesman must visit 10, and 20 cities, are showed respectively in fig 6 and 7. The experiment is done under the following conditions:

- Number of ants = 50,
- number of particles = 10,

The algorithms run ten times for the same problem and the average of the best results is calculated. The costs returned are in fact the minimum distances of a path found by an ant for a given test.

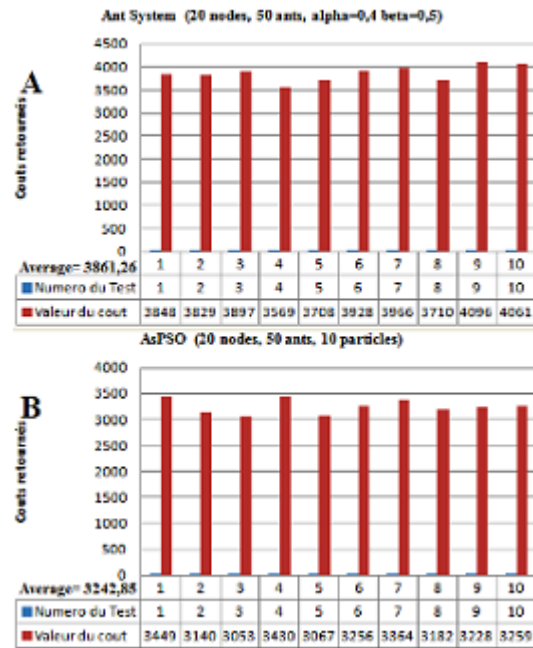


Fig. 6. Ant System and AS-PSO comparative results for 10 cities, (A) Ant system results, (B) AS-PSO results

The figure6, shows the results obtained by AS and AsPSO for ten cities. AS-PSO clearly gives better results with an average 964.42, while AS gives an average of 1165.28. This tendency becomes clearer when we increase the number of cities. Thus, with 20 cities AS gives 3861.26, while AsPSO gives 3242.85, as we can observe it on the fig 7.

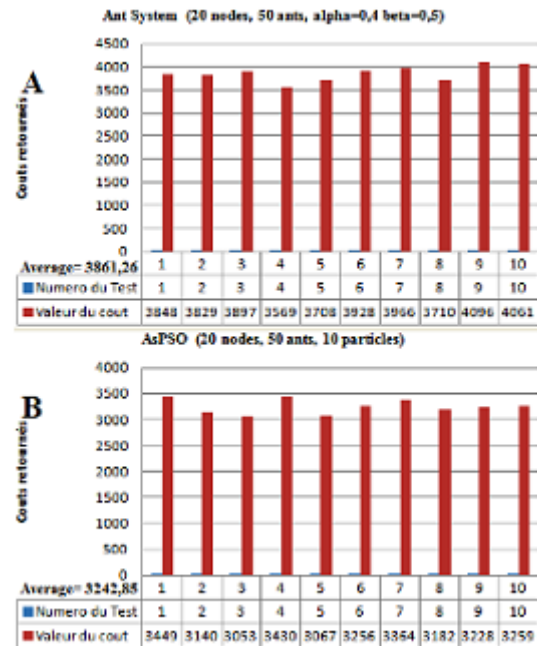


Fig. 7 Ant System and AS-PSO comparative results for 20 cities, (A) Ant system results, (B) AS-PSO results

V. CONCLUSIONS

In a classical schema, Ant System, AS, is applied several times with the same parameters and the same initialization of ants. Then the best results of the attempts is assumed to be the problem solution. With AS-PSO the parameters are re-evaluated by each particle for each iteration (for each (i) iterations and (p) particles, AS-PSO use a new set of parameters while AS use just use one. Thus, for the TSP problem, it is experienced here, the initial set of parameters used in AS-PSO is not very important, since the algorithm proceed with self adaptation, this minimizes the impact of a bad parameter choice, AS-PSO acts like an Adaptive ACO, in witch the adaption of ACO parameters are made by a PSO.

The goal of this work is not to prove that AS-PSO is better than AS. The performances of the both depend on the size of the problem to solve, of the execution time that is available and the nature of the problem itself. But we observed that the difference between the averages increases when we increase the number of cities. So, for a big problem for which we search good settings and a good accuracy, we will choose AS-PSO.

In this paper only a limited Aspect of AS-PSO is investigated involving the self tuning of ($\alpha\beta$) ACO parameters. More investigated are needed to cover the remaining parameters as well as the possible combinations' of them, and their impact on the end-solution quality.

ACKNOWLEDGMENT

The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

REFERENCES

- [1] N. Rokbani and A. M Alimi. "IK-PSO, PSO Inverse Kinematics Solver with Application to Biped Gait Generation.International", Journal of Computer Applications, vol 58, number (22):33-39, November 2012.
- [2] Y. Shi, and R. Eberhart, 1998. A modified particle swarm optimizer. In Proc of the 1998 IEEE World Congress on Computational Intelligence and IEEE International Conference on Evolutionary Computation, pp: 69-73.
- [3] M. Dorigo, M. Birattari, and T. Stutzle. "Ant colony optimization.", IEEE Computational Intelligence Magazine (2006), 28-39, 2006.
- [4] M. Reimann, and M. Laumanns. "A hybrid aco algorithm for the capacitated minimum spanning tree problem." Proceedings of first international workshop on hybrid metaheuristics. 2004.
- [5] P. Winker, P and M. Gilli, "Applications of optimization heuristics to estimation and modelling problems". Computational Statistics & Data Analysis,47(2), 211-223, 2004.
- [6] W. Elloumi, N. Rokbani and A.M. Alimi, " Ant supervised by PSO", . In Proc of International symposium on Computational Intelligence and Intelligent Informatics, pp : 161-166, 2009.
- [7] B. Gavish, & S. C. Graves, "The travelling salesman problem and related problems", (1978).