Ant Colony Optimization algorithm for breast cancer cells classification

Ahmed Nejmedine Machraoui, Mohamed Ali Cherni and Mounir Sayadi
SICISI Unit, University of Tunis
ESSTT, 5 Av. Taha Hussein 1008, Tunis, Tunisia
ahmed.machraoui@gmail.com, mohamed.ali.cherni@gmail.com, mounir.sayadi@esstt.rnu.tn

Abstract—Ant colony optimization (ACO) is a bio-inspired technique formalized into a meta-heuristic for combinatorial optimization problems. In this work, the ACO-Otsu segmentation method, based on ACO algorithm and Otsu’s method as a fitness function, is applied in classification and detection of breast cancer cells. Subsequently, this method is compared with the Otsu’s standard method. The experiments show the performance of this probabilistic search approach in such type of problems.

Keywords—Ant Colony Optimization, Ant System, Meta-heuristic, Cells classification, Optimization methods.

I. INTRODUCTION
Optimization is a well-used term in every engineer’s task. It designates the best solution that can be found for a specified problem. In literature, Garcia et al. [1] surveyed the different optimization methods and categorized them according to their application into linear and non-linear programming methods.

This last cited family of methods was also divided into two other categories: the deterministic methods and the stochastic methods. In this work we will study a stochastic method which is defined to be a probabilistic search method using only information from the objective function. Further details on the definition of this term and the others can be found on [1].

Otherwise, Kumar et al. [2] surveyed the stochastic methods and described the naturally inspired ones such as Particle Swarm Optimization (PSO), Tabu search (TS), Simulated Annealing (SA), Evolutionary algorithms (EA), Artificial Bee Colony (ABC) and Ant Colony Optimization (ACO). In the following, we will focus on Ant Colony Optimization as a meta-heuristic used in solving many complex problems in many fields. In Section II, we survey ACO algorithms and their main applications. Section III describes the improved ACO algorithm based on Otsu’s method and illustrates its application on breast cancer cells images and section IV provides an analysis study of the technique in image classification problem. Section V concludes the article.

II. ANT COLONY OPTIMIZATION TECHNIQUES
Ant Colony Optimization (ACO) is a probabilistic search approach based on an evolutionary process which is biologically inspired by the foraging behavior of some ant species colonies [3], [4] and [5].

The foraging behavior of ants consist on looking for food and when they find it they return to the nest following a random path by depositing on the ground an odorous substance called ”pheromone” [6]. When other ants find this path, they follow it. In fact, they especially tend to choose the track with the highest concentration of pheromone.

This allows them to find their way to the nest. On the other hand, the pheromone can be used by other ants to locate found food sources. Note that the pheromone evaporates over the time that makes disappear long ways. These paths will be less traveled by ants. The path traveled by more ants will be characterized by a high concentration of pheromone; this path is “the optimal path”.

Artificial ants are entities that operate in an environment by depositing an amount of pheromone according to certain data. This pheromone plays an integral role in changing the way in which the environment is perceived by the ants. Artificial ants move between adjacent sites as the real ants. Thus, the choice of the new recipient is random and depends on the heuristic information of the environment and information residing in the pheromone trails. The pheromone evaporates after a while promoting the development of new directions and therefore new solutions [7].

Based on this principle, Dorigo et al. [6] formulated for the first time the ant colony algorithm known as Ant colony optimization or simply Ant System (AS). Subsequently, several other studies have been done on this subject and the optimization algorithm inspired from the foraging behavior of ant colonies has been improved more than once to be suitable for application in many areas such as routing problems [8]–[10], assignment problems [11], [12], scheduling problems [13], [14], subset problems [15], [16], image processing [17] and so on. Among the well-used algorithms we cite the algorithm Ant System (AS) [6], [18], Ant-Q [19], [20], Ant Colony System (ACS) [21], [22] and Max-Min Ant System (MMAS) [23], [24]. These will be presented in the following.

A. Ant System (AS)
Ant System Algorithm has the first version proposed by M. Dorigo in 1991 [6]. It mainly consists of two phases: construction of solutions and updating pheromone trails. These two phases are performed iteratively until it reaches the
stop condition. The structure of the AS algorithm is presented by the chart in figure 1.

- **Initialization**: During this phase, we set the parameters of the algorithm, as number of ants \( m \), the pheromone evaporation rate \( \rho \) and the relative importance of heuristic information and pheromone information respectively \( \beta \) and \( \alpha \). We also initialize the pheromone trails \( \tau_{ij} \). Therefore, it is recommended in [25] to initialize the pheromone to a value \( \tau_0 \) slightly higher to the amount of pheromone deposited by an ant in one iteration according to the expression in equation 1.

\[
\tau_{ij} = \tau_0 = \frac{m}{C_{mn}}
\]  

(1)

where \( C_{mn} \) is the length of a turn processed by the nearest neighbor site following the heuristic factor. If we initialize \( \tau_0 \) to a very large value, we will have a proportionally large number of iterations before the tracks evaporate leaving only those belonging to the optimal solutions. On the other hand, if \( \tau_0 \) is set to a very small value, the search will be biased rapidly after the first round made by ants.

- **Construction of the solution**: in this step, the positions of ants in the search space are initialized randomly. Then these ants begin to move between sites. So ant \( k \) located in the site \( i \) chooses his destination \( j \) with the probability in equation 2.

\[
p_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{t \in N_i^k} [\tau_{it}]^\alpha [\eta_{it}]^\beta} \text{ if } j \in N_i^k
\]  

(2)

where \( \eta_{ij} \) is the heuristic information of the problem known a priori and \( N_i^k \) is the set of sites that can be accessed when ant \( k \) is at site \( i \).

AS per the algorithm, there are two ways to implement the construction phase:

- The parallel implementation: at each stage of construction, all the ants move from their sites to new ones.
- The sequential implementation: each ant must make a full turn (Solution) before the next ant makes his turn.

- **Updating pheromone trails**: After the construction phase, we proceeded to a step of evaluating the solutions found by the ants and update the new data collected. This is done in two steps first evaporation of pheromone trails and updating of these tracks by depositing a specified amount on them.

  - Pheromone trails evaporation: We Decreases the concentration of pheromones on all tracks according to expression in equation 3.

\[
\tau_{ij} = (1 - \rho) \tau_{ij}
\]  

(3)

where \( \rho \in [0,1] \) is the rate of vaporization of pheromone. This procedure encourages the discovery of new solutions by preventing the accumulation of pheromones on some tracks. Furthermore, it supports the elimination of non-optimal solutions.

- Deposition of new pheromone: After evaporation, ants lay an amount of pheromone on the trails visited during the construction phase. Each track, one of the solutions found by an ant, receives an amount of pheromone according to the expression in equation 4.

\[
\tau_{ij} = \tau_{ij} + \sum_{k=1}^m \Delta \tau_{ij}^k
\]  

(4)

where \( \Delta \tau_{ij}^k \) is the amount of pheromone deposited by ant \( k \). This parameter is related to the heuristic information of the problem.

- **Updating pheromone trails using a fitness function**

\[
\tau_{ij} = \tau_{ij} + \rho \cdot \tau_{ij} - \Delta \tau_{ij} = \rho \cdot \tau_{ij} (t) + (1 - \rho) \cdot \gamma \cdot \max_{t \leq t_0} \tau_{ij} (t)
\]  

(5)

where \( \gamma \) is an adjustable constant. At the end of the cycle we add to \( \tau_{ij} \) the term \( (1 - \rho) \cdot \Delta \tau_{ij} (t) \), where \( \Delta \tau_{ij} (t) \) is defined as for AS (see equation 4).

The purpose of this improvement is to update the pheromone trails with a value that presents a forecast of the quantities of pheromones that ants will find in the next destination. This way the ants that arrive at \( i \) will be informed if the displacement to \( j \) is justified or not. This algorithm has good results, but gives similar performance to ACS, while ACS algorithm is less time consuming.

- **Ant Colony System (ACS)**

Ant Colony System algorithm was proposed by Dorigo Gambardella in 1996 [2]. It keeps the same characteristics of the AS algorithm, while introducing a new parameter \( \psi_0 \). This parameter is used during the process of choosing a new destination for an ant \( k \). Thus, we assign a random value uniformly distributed between 0 and 1 to the variable \( \psi \). When \( \psi \leq \psi_0 \) we makes a choice based solely on heuristic factor or previous experience. For example, for the traveling salesman problem (TSP), we choose the cities already visited effectively. However, if \( \psi > \psi_0 \), the choice is performed in the
same way as for AS Algorithm. This way we can push the ants to focus on the best solutions or then to seek for other paths.

The ACS algorithm has also brought improvements concerning the update of pheromone trails. This algorithm allows the global update for tracks belonging to the best found solutions in the cycle. In addition, he defined a local update performed by each ant, following his move from site i to site j. This update tends to reduce the concentration of pheromones on this track via the expression of Equation 6.

\[ \tau_{ij} = (1 - \xi)\tau_{ij} + \xi\tau_0 \] (6)

where \( \xi \) is an adjustable constant with \( 0 < \xi < 1 \), and \( \tau_0 \) is the pheromone of the initial track. This aims to encourage the exploitation of new trails and therefore the search for new solutions.

The ACS algorithm based on the Ant-Q algorithm. He reintroduced the same expression of the update step-by-step (Local updates). However it reduces its expression by changing the term \( \gamma \max_{i \in EN(i)} h(i) \) with \( \tau_0 \). Thus it has allowed good results while increasing the performance of the algorithm.

**D. Max-Min Ant System (MMAS)**

This algorithm was proposed by Sttzle Hoos in 1996 [2]. It uses the same principle of global update as the ACS algorithm.

Thus, only the tracks belonging to the best solution in the cycle receives pheromone. In addition, we initialize the pheromone evaporation. Thus the amount of pheromone that is evaporated is proportional to its value at modification. More and more trails are strong their quantities of pheromone will be reduced.

**III. BREAST CANCER CELLS CLASSIFICATION USING ACO-Otsu ALGORITHM**

Image segmentation region-based approaches use the homogeneity of certain parameters to distinguish different regions of the image. In this context, the Otsu’s method is one of most commonly used techniques firstly for image binarization, and then for multi-levels classification.

In this section, we present the conventional method of Otsu and we describe the algorithm ACO-Otsu. Then, we apply this algorithm to classify breast cancer cells.

**A. Otsu’s method**

Otsu’s method is a reference method which consists in trying various threshold values and choose the one that maximizes the interclass variance between classes \( C_0 \) and \( C_1 \). Thus, computing different criteria is based on the density of probability of the pixels in the image using the expression in equation 7.

\[ p_i = \frac{h(i)}{\sum_{j=0}^{L-1} h(j)} \quad \text{with} \quad \sum_{i=0}^{L-1} p_i = 1 \] (7)

where \( L \) is the total number of gray levels in the image, and \( h(i) \) is the histogram of the image indicating the number of pixels with gray level \( i \).

The optimal threshold \( t^* \) is the one that maximizes the ratio of the interclass variance to the total variance (see equation 8).

\[ t^* = \arg_{t} \max \frac{\sigma_B^2(t)}{\sigma^2} \] (8)

Thus, we can define a measure of interclass variance by equation 9.

\[ \sigma_B^2(t) = P_t (P_t - 1) (\mu_t - \mu_0)^2 \] (9)

And the total variance is defined by equation 10.

\[ \sigma^2 = \sum_{i=0}^{L-1} p_i (i - \mu)^2 \] (10)

where \( \mu_0 \), \( \mu_1 \), and \( \mu_T \), respectively the first and the second class averages and the global average of the image, are defined in equations 11, 12 and 13.

\[ \mu_0 = \sum_{i=0}^{t-1} \frac{p_i}{P_t} \] (11)

\[ \mu_1 = \sum_{i=t+1}^{L-1} \frac{p_i}{1-P_t} \] (12)

\[ \mu_T = \mu_0 + \mu_1 \] (13)

Where \( P_t \) is defined in equation 14.

\[ P_t = \sum_{i=0}^{t} p_i \] (14)

And since the total variance for a given image is constant, then the problem is to maximize the interclass variance.

The major weakness of Otsu’s method is that it can be applied for classifying images containing just one type of object, and it aims to group the image pixels into two classes. The purpose here is to apply the Otsu’s method recursively to segment a single type of object at a time in the image. At the end the global segmentation is obtained by grouping all segmented objects [26].

In this case, the interclass variance to maximize is defined by equations 15, 16, 17 and 18.

\[ \sigma_{BC}^2 = \sum_{n=0}^{k} w_n (\mu_n - \mu_T)^2 \] (15)

where

\[ w_n = \sum_{i=t_n+1}^{t_{n+1}} P_i \] (16)

\[ \mu_n = \frac{\sum_{i=t_n+1}^{t_{n+1}} P_i}{w_n} \] (17)

\[ \mu_T = \sum_{i=0}^{L-1} P_i \] (18)

Thus, the optimal thresholds \( k^*_n \) are defined by equation 19.

\[ \sigma_{BC}^2(k_1^*, k_2^*, ..., k_L^*) = \max_{1 \leq k_1 < k_2 < ... < k_L} \sigma_{BC}^2(k_1, k_2, ..., k_L) \] (19)

**B. ACO-Otsu Algorithm**

The ACO algorithm used in this section is Ant System (AS). This approach is basic, easy to implement and contains the...
main advantages of ant’s algorithms. The ACO algorithm is used to search for optimal threshold $k$ for image classification where $k \in [0, L - 1]$. It groups the pixels of the image into classes based on the histogram $h(i)$ [26], [27].

ACO-Otsu algorithm uses artificial ants to mark the thresholds that maximize the interclass variance criterion. Each ant chooses the thresholds for classification, and then evaluates the found solution the interclass variance criterion. The construction of a solution is done according to the decision probability expressed by equation 20.

$$P_{ij} = \frac{(\tau_{ij})^\alpha}{\sum_{i=1}^{L-1} \sum_{j=1}^{L-1} (\tau_{ij})^\alpha} j \in \{1, 2, ..., UB_i - LB_i + 1\} \text{ for } i \in \{1, 2, ..., UB_i - LB_i + 1\}$$

(20)

where $i$ indicates the threshold classification index, $j$ indicates the grey level index for the $i^{th}$ threshold, and $\alpha$ the importance of pheromone information.

After the construction of a solution by each ant, ants evaluate their solution, then the one who found the best solution, performs an update of pheromone by adding an amount proportional to the interclass variance $\sigma_{BC}^2$ after having gone through the process of pheromone evaporation. The procedure for updating can be summarized by the expression in equation 21.

$$\tau_{ij}^{new} = (1 - \rho)\tau_{ij}^{old} + \rho \Delta^e$$

(21)

where $\rho \in [0,1]$ is the pheromone evaporation rate, and $\Delta^e$ is the amount of pheromone added by the ant that found the best solution. This amount is added to all combinations $(i,j)$ belonging to the best solution and is represented by the expression in equation 22.

$$\Delta^e = Q \sigma_{BC}^2$$

(22)

where Q is a constant controlling the additional rate of pheromone.

The ACO-Otsu algorithm is resumed by diagram in figure 2. The stop condition of the algorithm can be either a maximum number of iterations, or a number of iterations during which the solution found has not changed.

IV. RESULTS AND DISCUSSION

Breast cancer cells images studied in this paper are true color images obtained after immunohistochemical staining. They contain two types of cells:

- Positive cells corresponding to the marked cells which are red colored and are positive for hormone receptors.
- Negative cells corresponding to the unmarked cells which colors are blue. Those cells are negative for hormone receptor and for which there is no need to prescribe an anti-hormonal treatment.

In this section, we apply ACO-Otsu algorithm on breast cancer cells images in order to classify them into two types of cells. Note that the parameters of ACO-Otsu algorithm applied to these images are presented in Table1 and the classification results in figure 3.

| Parameters | Values |
|------------|--------|
| $A$        | 1      |
| $P$        | 0.9    |
| $Q$        | $10^{-4}$ |
| $\tau_0$  | $10^{-2}$ |

Tab. 1. Parameters of ACO-Otsu algorithm

The evolution of the best value of the interclass variance for each iteration (Fig. 3) shows the convergence speed of the ACO-Otsu algorithm to the maximum value of the variance from the first fifteen iterations.

![Fig. 3. Evolution of interclass variance values](image_url)

Curve of the evolution of the two searched thresholds (Fig. 4) also shows the speed of convergence of the algorithm. Furthermore, it proves that all the ants choose two thresholds (127-190) after the twentieth iteration. This is one of the advantages of AS algorithm that converges quickly to near-optimal thresholds and limits the choice of new solutions to these thresholds, which limit the use of new solutions. Indeed, this is due to the evaporation of pheromone that is almost nil on the tracks not selected by ants. This problem has been fixed in newer versions as MMAS defining a minimum value of pheromones still present on these tracks. Also note that, after
the eighteenth iteration, all ants choose only the quasi-optimum thresholds (127-190). The reason for this result is due to the amount of pheromone deposited on the tracks after the convergence of the algorithm (Fig. 5). Thus we note that the pheromone concentrates on two thresholds (127-190), and the remains thresholds (tracks) have almost zero quantity of pheromone.

Fig. 5. Pheromone trails after convergence of ACO-Otsu algorithm

Fig. 6. Histogram of image (1) and the founded thresholds

Tab. 2. Result comparisons among Otsu’s and ACO-Otsu methods for image (1)

|            | CPU Times (Sec.) | Population Sizes × Iteration |
|------------|------------------|-------------------------------|
| Otsu       | 126,190          | -                             |
| ACO-Otsu   | 127,190          | 0.249                         |

V. CONCLUSION

In this paper, we introduced the Ant Colony Optimization technique with description of the main algorithm (Ant System) and some of the best improved algorithms based on AS (Ant-Q, ACS and MMAS). The study provides also a hybrid approach based on AS algorithm and Otsu’s method. The approach is named ACO-Otsu, and is applied on breast cancer cells images after immunohistochemical staining to classify cells into positive marked cells and negative marked cells. The results show the efficacy of this algorithm to find quasi-optimum thresholds. The advantage of using such approach, and unlike to many conventional methods such Otsu’s method, is the possibility to classify several objects in image with considerably less time consuming. However, the drawback of the described approach lies in the detection of merged cells.

ACKNOWLEDGMENT

The authors would like to thank Dr. Khaled Ben Romdhane and Dr. Karima Mrad, from the department of pathology in Salah Azaiez Institute, Tunisia, for providing the microscopic breast cancer images used in this paper.

REFERENCES

[1] J. S. Garcia, L. Avila, and W. P. Carpes, “Introduction to optimization methods: a brief survey of methods,” *Population English Edition*, vol. 2, no. N. 2, pp. 1–5, 2006.
[2] Y. Kumar and D. Kumar, “Parametric analysis of nature inspired optimization techniques,” *International Journal of Computer Applications*, vol. 32, no. N. 3, pp. 42–49, October 2011.
[3] H. Duan, Y. Yu, J. Zou, and X. Feng, “Ant colony optimization-based bio-inspired hardware: Survey and prospect,” *Transactions of the Institute of Measurement and Control*, vol. 34, no. N. 2-3, pp. 318–333, 2012.
[4] B. K. Nanda and G. Das, “Ant colony optimization: A computational intelligence technique,” *International Journal of Computer Communication Technology (IJCCT)*, vol. 2, no. 6, pp. 105–110, 2011.
[5] O. A. M. Jafar and R. Sivakumar, “Ant-based clustering algorithms: A brief survey,” *International Journal*, vol. 2, no. 5, pp. 787–796, 2010.
[6] M. Dorigo, M. Birattari, and T. Stütze, “Ant colony optimization artificial ants as a computational intelligence technique,” *IEEE Computational Intelligence Magazine*, vol. 1, no. September, pp. 28–39, 2006.
[7] M. Divvyavani and T. Anudha, “Comparing the clustering efficiency of aco and k-harmonic means techniques,” *International Journal of Computer Science, Engineering and Applications (IJCSA)*, vol. 1, no. 4, pp. 91–106, 2011.
[8] M. Reimann, K. Doerner, and R. F. Hartl, “D-ants: Savings based ants divide and conquer the vehicle routing problem,” *Computers Operations Research*, vol. 31, no. 4, pp. 563–591, 2004.
[9] H. Hernández and C. Blum, “Ant colony optimization for multicasting in static wireless ad-hoc networks,” *Swarm Intelligence*, vol. 3, no. 2, pp. 125–148, 2009.
[10] G. Fuellner, K. Doerner, R. Hartl, and M. Iori, “Ant colony optimization for the two-dimensional loading vehicle routing problem,” *Computers Operations Research*, vol. 36, no. 3, pp. 655–673, 2009.
[11] M. Solimanpur, P. Vrat, and R. Shanker, “An ant algorithm for the single row layout problem in flexible manufacturing systems,” *Computers Operations Research*, vol. 32, no. 3, pp. 383–598, 2005.
[12] K. Socha, M. Sampels, and M. Manfrin, “Ant algorithms for the university course timetabling problem with regard to the state-of-the-art,” *Applications of Evolutionary Computing Proceedings of the EvoWorkshops 2003 LNCS 2611*, pp. 334–345, 2003.
[13] C. Blum, “Beam-aco hybridizing ant colony optimization with beam search: an application to open shop scheduling,” *Computers Operations Research*, vol. 32, no. 6, pp. 1565–1591, 2005.
Tab. 3. Microscopic breast cancer cells classification using ACO-Otsu algorithm (a) Original image (b) ACO-Otsu method at tri-level thresholding (c) 1st class extraction (d) elimination of detection errors using morphological operations (1–4) Image 1–4

[14] S. Morin, C. Gagn, and M. Gravel, “Ant colony optimization with a specialized pheromone trail for the car-sequenceing problem,” *European Journal Of Operational Research*, vol. 197, no. 3, pp. 1185–1191, 2009.

[15] C. Solnon and S. Fenet, “A study of aco capabilities for solving the maximum clique problem,” *Journal of Heuristics*, vol. 12, pp. 155–180, 2004.

[16] C. Blum and M. J. Blesa, “New metaheuristic approaches for the edgeweighted k-cardinality tree problem,” *Computers Operations Research*, vol. 32, no. 6, pp. 1355–1377, 2005.

[17] K. Thangavel and M. Kaman, “Meta-heuristic algorithms for automatic detection of microcalcifications in digital mammograms,” *ICGST International Journal on Graphics, Vision and Image Processing*, vol. 05, no. 7, pp. 41–55, July 2005.

[18] L. Cioni, “Some strategies for parallelizing ant systems,” 2005.

[19] I. Khamassi, M. Hammami, and K. Ghedira, “Ant-q hyper-heuristic approach for solving 2-dimensional cutting stock problem,” pp. 1–7, 2011.

[20] S. Lee, “A cooperation online reinforcement learning approach in ant-q,” *Lecture Notes in Computer Science*, vol. 4232, pp. 487–494, 2006.

[21] L. Mou, “An efficient ant colony system for solving the new generalized traveling salesman problem,” pp. 407–412, 2011.

[22] S.-C. Chu, J. F. Roddick, and J.-S. Pan, “Ant colony system with communication strategies,” *Information Sciences*, vol. 167, no. 1-4, pp. 63–76, 2004.

[23] P. C. S. P. C. See, K. Y. W. K. Y. Wong, and K. Komarudin, “A technique for improving the max-min ant system algorithm,” 2008.

[24] T. Sttzle and H. H. Hoos, “Max-min ant system,” *Future Generation Computer Systems*, vol. 16, pp. 889–914, 2000.

[25] M. Dorigo and T. Sttzle, “Ant colony optimization,” *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 28–39, 2004.

[26] K. Gao, M. Dong, L. Zhu, and M. Gao, “Image segmentation method based upon otsu aco algorithm,” in *Information and Automation - International Symposium, ISIA 2010, Guangzhou, China, November 10-11, 2010. Revised Selected Papers*, ser. Communications in Computer and Information Science, L. Qi, Ed., vol. 86. Springer, pp. 574–580, 2010.

[27] Y.-C. Liang, A. H.-L. Chen, and C.-C. Chyu, “Application of a hybrid ant colony optimization for the multilevel thresholding in image processing,” in *Proceedings of the 13th international conference on Neural Information Processing - Volume Part II*, ser. ICONIP’06. Berlin, Heidelberg: Springer-Verlag, pp. 1183–1192, 2006.