Foraging Behaviors – Pheromone, Task Allocation, and Trophallaxis -A Relative Comparison for Robotic Swarm Foraging

Ralph Nicole R.Barcos, Angelo R. Dela Cruz, Edison A. Roxas, Argel A. Bandala, Laurence Gan Lim, Elmer P. Dadios and Ryan Rhay P. Vicerra

***Abstract* — A group of algorithm enhancing collective behavior is inspired by the animals working together as a group such as ants, bees, and etc. In connection, swarm is defined as a set of two or more independent homogenous or heterogeneous agents acting upon a common environment in a coherent fashion which generates emergent behavior. The development of artificial swarms or robotic swarms has attracted a lot of researchers in the last two decades including pheromone, trophallaxis and task allocation algorithms. However, among these swarm based algorithms, the most efficient in terms of group performance, efficiency and interference in collecting the dusts or objects in an environment with variable terrains has not been identified. With this, the researchers see the need to developed swarm simulation platform that would compare the swarm-behavior-based algorithms for an ideal use of robots in different environments in dust collection.**

***Index Terms* — Swarm intelligence, swarm foraging, swarm simulation platform development**

I. Introduction

Design and implementation of collective behavior of agents in accomplishing tasks are gaining popularity nowadays. Social animals and insects are the key inspiration of creating distributed behavior amongst independent agents. In connection, swarm is defined as a set of two or more independent homogeneous or heterogeneous agents acting in a common environment in a coherent fashion, which generates emergent behavior. The creation of artificial swarms or robotic swarms has attracted many researchers in the last two decades. Many studies have been undertaken using practical approaches to swarm construction such as investigating the navigation of the swarm, task allocation and elementary construction.

Examination of the behaviors of ants has led to the recently developed field of Swarm Intelligence. Ants can perform diverse collective tasks such as foraging, nest building, sorting, and cooperative transport. In this proposed study, three different behavior of ant system will be studied with certain given measures: speed, accuracy, efficiency, and collision avoidance. The goal is to obtain the best or most effective ant-based swarm behavior in foraging, specifically in dust collection, within a given platform environment.

A number of researchers have proposed many swarm robotics algorithm and the emerging trend in the field of interest are the study of ant-based algorithms like the pheromone, trophallaxis and task allocation. However, among these swarm based algorithms, the most efficient, in terms of speed, efficiency, and collision avoidance has not yet been determined. Thus, there is a need to develop a swarm simulation platform to compare the swarm behavior based algorithms for an ideal use of swarm robots in different environments for dust collection.

Ralph Nicole R. Barcos, Angelo R. Dela Cruz and Edison A. Roxas are affiliated with Faculty of Engineering University of Santo Tomas, Espana Blvd, Manila, 1008 Philippines

Ryan Rhay P. Vicerra, Argel A. Bandala, Laurence Gan Lim and Elmer P. Dadios are affiliated with De La Salle University, 2401 Taft Ave, Malate, Manila, 1004 Metro Manila (e-mail: ryan.vicerra@dlsu.edu.ph; argel.bandala@dlsu.edu.ph; laurence.ganlim@dlsu.edu.ph; elmer.dadios@dlsu.edu.ph)

II. Swarm Behaviors

This section will discuss the different swarm behavior algorithms as presented by select models. The first part presents the three ant-based swarm behaviors and their basic description; we also state the algorithm that will represent them in the study. The second part will present the strengths of each behavior as opposed to the others; need for the comparison of the behaviors.

A. Trophallaxis (mouth-to-mouth feeding)

Trophallaxis is a basic behavior observed in majority of social animals and insects [5]. Its main purpose is to provide nutrition to offspring in nursing stage. However, it is also observed in several colony-based insects, that trophallaxis occurs among adults to better distribute food. In [6, 7] it was shown that trophallaxis plays an important role in the regulation of collective foraging decisions in honeybees. Aside from its function to transfer nutrients, trophallaxis is also observed as a tool in exchanging information about available food sources in honeybees. A trophallaxis-based algorithm is a self-organized task exchange in a swarm of autonomous, movable, and reconfigurable agents. In our study, the trophallaxis-based algorithm will be the BEECLUST Algorithm [5].

B. Task Allocation

Task allocation and learning is normally quite important to a swarm of robots. Task decomposition and allocation can greatly improve efficiency for especially complex tasks. In [8], they compared the costs and benefits of different types of task allocation approaches in noisy world. Learning is also useful since the parameters of the control mechanism are hard to tune. With the help of self-adaptive learning and optimizing methods, the swarm shows better adaptability in the different environments [4]. Task allocation assigns the robot members among different tasks in an adaptive and flexible way [7]. Task allocation will be represented by the Task Partitioning Model in [10].

C. Pheromone

Ant colonies in the nature are famous for their navigation and migration behaviors with the help of pheromones. The researchers of the swarm robotics society employed such scheme into swarm robotics by simulating the pheromones using part of the robots in the swarm which serve as the beacons [4]. Usage of pheromone for swarm robotics vary with its purpose such as in [11] where it is unique to a location on a map which was used to create a path from garbage sites to dump sites or in [12] where it was a signal the robot always carried to be able to inform robots of its current position and heading when they are close enough to each other. It may also be a numerical value in shared memory such as in [9] that acts as a request to other robots for assistance in a task. In our study, Pheromone-based behavior will be represented by the model in [12].

III. Unique Characteristics

Pheromone-based swarm robots select different tasks of the same type with different probabilities according to the pheromone amounts to assign themselves in the performing of these tasks [9]. They can use simple attraction or repulsion behaviors and they also do not require distinct step of map generation [12]. Though communication is indirect, they move with respect to pheromones “left” by other bots and “drop” their own for other robots [4]. Thus Pheromone-based swarms provide a robust, scalable approach for achieving the swarm level behaviors using a large number of small-scale robots in tasks such as reconnaissance and path-finding [13][15]. Trophallaxis-based swarm robots do not require such information nor do they need to know the position of other robots meaning that there is no need for a collection of the information of bots [5], like in Pheromone-based swarms [9], since communication happens at the instance when two robots meet [5]. Trophallaxis-based robots can also exchange tasks between robots of different capabilities to reduce the energy consumption of reconfiguration when and if they meet [10]. Trophallaxis-based swarm robots also do not require additional hardware for onboard processing or memory [5]. Task-allocated swarms on the other hand boast high adaptability in different environments and greatly improve efficiency for especially complex tasks [4]. Taskallocated swarm bots are also specialized and can switch between specializations if the remaining tasks require more participants [9].

IV. SIMULATION PARAMETERS

Based on parameters used in [10], the measurement of group performance, individual efficiency, and interference:

Group Performance is the total number of prey objects collected within the time period while Individual Efficiency is the number of prey objects collected by each individual robot [10].

Interference is measured as the time spent performing actions not strictly related to the task, but rather those actions that are lost due to negative interactions with the environment (e.g., obstacle avoidance maneuvers) [10]

V. BEHAVIOR ALGORITHM

A. Trophallaxis

Figure 1 shows the logical representation of the flow of events in a BEECLUST algorithm. First, to achieve random movement, each bot is assigned a direction and moves in a straight line in that direction until it encounters an obstacle, boundary or other bot. Bot collisions are detected by determining whether bots are within a certain distance of each other. Then, the bot determines whether it has collided with an obstacle or another bot. Once a bot is stopped (as a result of collision with another bot), then it measures the value of the function at that location. Lastly, cluster finding is done when the search is terminated. In general, the bots begin to collide/stop/wait at the beginning of the search. Thus, the bots tend to cluster soon after the search begins so the search can be stopped at any time to observe the location(s) of the clusters.

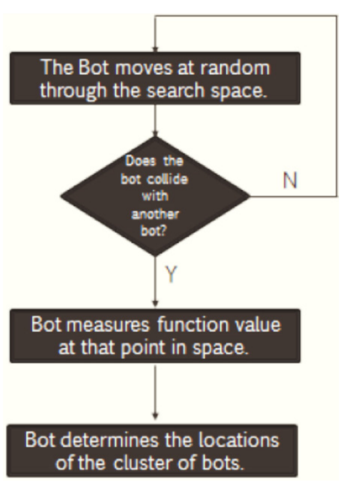

Fig. 1. Simplified state diagram of trophallaxis behavior based on BEECLUST algorithm

B. Task Allocation

Figure 2 shows the state diagram for the Task Allocation Algorithm [10]. Gray states belong to the harvest task, while the white states to the store task. The obstacle avoidance state has been omitted for clarity, as it is applicable in all states of the robot. t_w is the time spent in the exchange zone and $\boldsymbol{\theta}$ is the threshold.

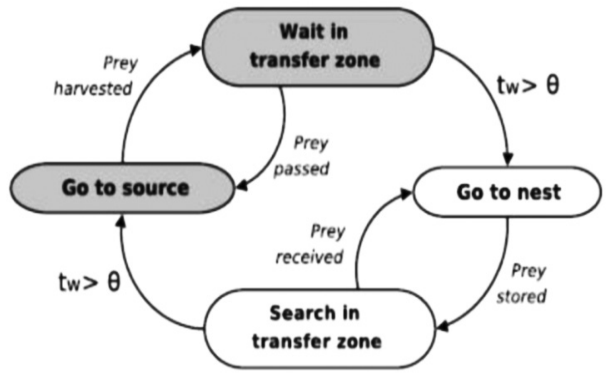

Fig. 2. Simplified state diagram of the controller of the robots

The first step of the algorithm is spatially partitioning the environment. The global foraging task is automatically partitioned into two subtasks particularly; harvesting prey objects from a harvesting area (source) and then transporting them to a home area (nest).

These subtasks have a sequential inter-dependency in the sense that they have to perform one after the other in order to complete the global task once: delivering a prey object to the home area [12].

A robot in the first subtask has to wait for a specific time before passing an object to the robot in the second subtask. A long waiting time could mean insufficient robots in subtask. The robots have the capability to switch a subtask. The waiting time could be used by the robot to decide whether it will switch a subtask or not.

C. Pheromone

Pheromones are locally transmitted without specifying a recipient. This obviates the need for unique identities that are impractical in a large group. Figure 3 shows the Pheromone's diffusion gradients that provide important navigational cues and it can also encode useful information about barriers in the environment that block pheromone propagation. But Pheromones decay over time, which reduces obsolete or irrelevant information and obstacles are sensed by the robot when pheromone messages bounce off them. The robot will remember the direction of the received virtual pheromone and will serve as guideposts for the following robots.

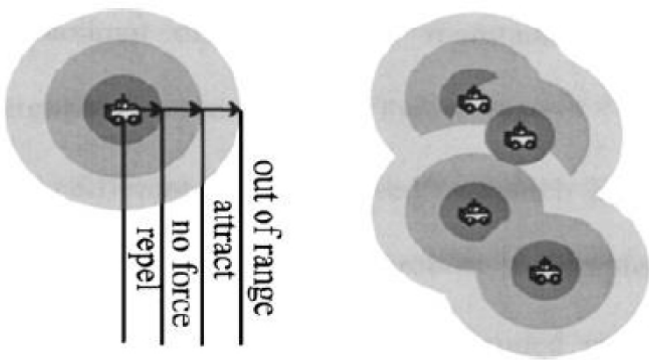

Fig. 3. Diagram of Pheromone field strength

VI. Simulation Environment

The testing environment has an area of 90 cm x 90cm, including the walls at the edge of the environment, divided into 3 cm x 3 cm sections called "patches". The environment will be leveled throughout. The test will also include static obstacles which will be expressed as a percent of the total area of the environment. In this study, the percentages will be: (a) 0% or no obstacles, (b) 25%, (c) 50% and (d) 75%. The environment will be generated so that all dust particles (not enclosed by obstacles) can be collected and that there are no single-block obstacles. The same generated environment will be used for testing all bots and algorithms so that all results can be compared. Each combination will be simulated 10 times with an upper limit of 40 if results are inconclusive. There will be a total of 50 dust particles in static tests and may be formed in clumps called "dust sites" or individual pieces simply called "dust". In dynamic tests, the first 25 dust particles will be placed on the field and the last 25 dust particles will be dropped randomly across the field.

VII. Dimensions and Specification of Swarm Robots

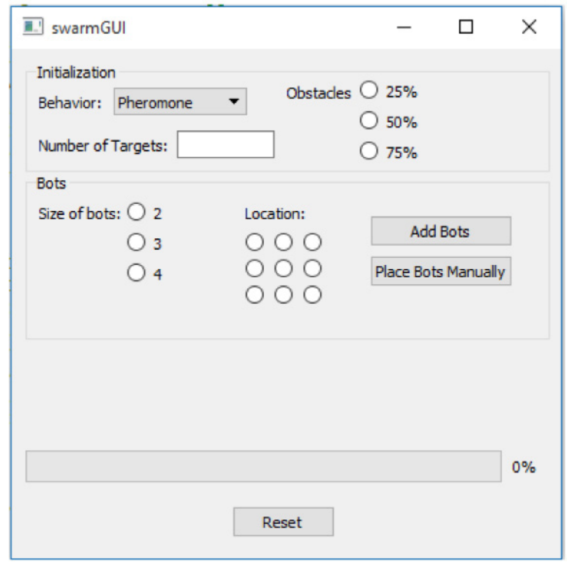

Fig. 4. Swarm robot and environment GUI

The robots in general will be able to move, rotate in place, detect the dust particle when they are near or over it, and carry the dust. Other specifications such as LEDs will be incorporated depending on the algorithm they will be using to perform the test.

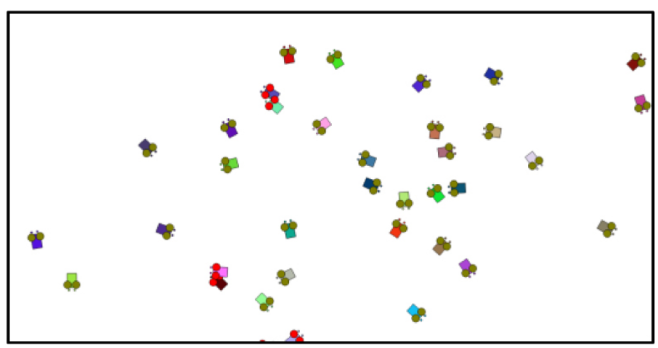

Fig. 5. Swarm robots in random movement (55 bots)

The size of the robots will be less than one patch, exactly the width of one patch and larger than one patch to allow for robot exploration. The diameter of the robots will be: (a) 2 cm, (b) 3 cm and (c) 4 cm. This study will use (a) 50 robots, (b) 100 robots, (c) 150 robots, (d) 200 robots of each size. The goal of the robots is to collect the dust scattered across the environment and bring it back to the nest located at the center. The dust will have no dimensions and no weight. The robots will be programmed to follow the algorithms presented in Chapter 2 to search for, collect and, return the dust.

The main algorithm used for the program of the graphical user interface and the environment is as follows:

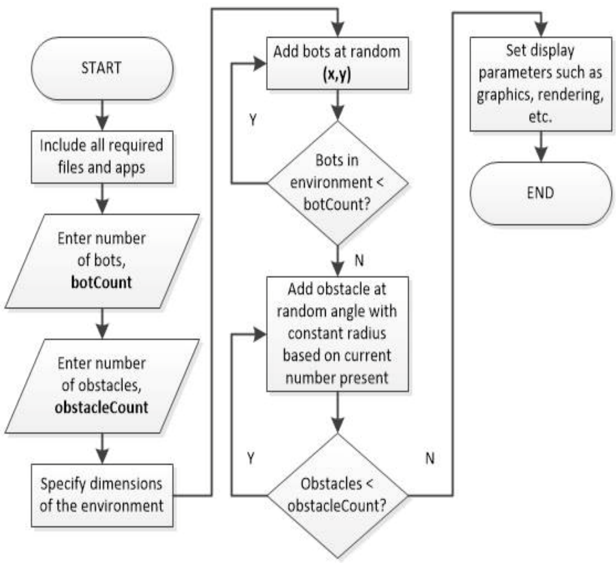

Fig. 6. Main algorithm of the simulation platform

VIII. SIMULATION RESULTS

In the simulation environment with 50 bots, it is observed that more or less 40 bots on average intersect at periods of random movements.

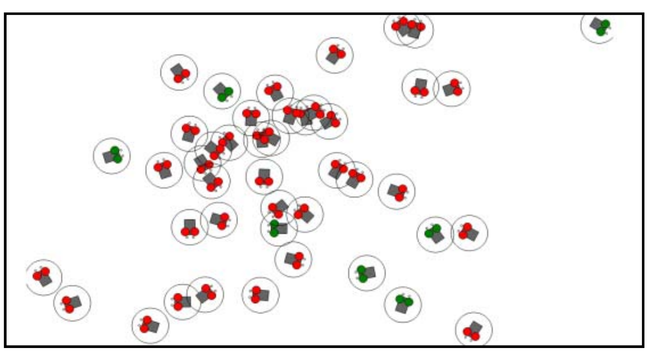

Fig. 7. Random movements of 50 Bots

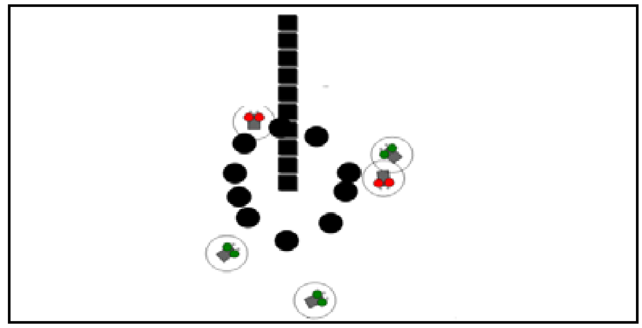

Fig. 8 Initial position of Dynamic Obstacles

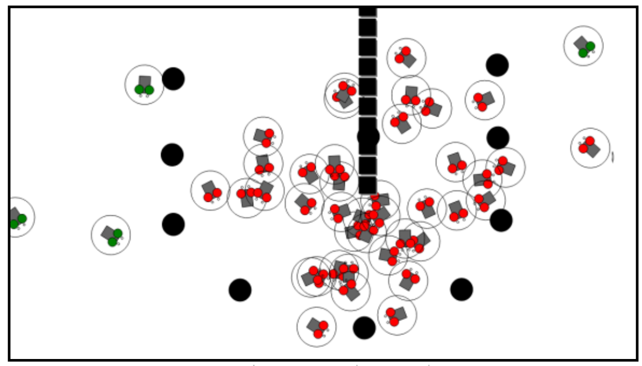

Figure 9. 50 bots used

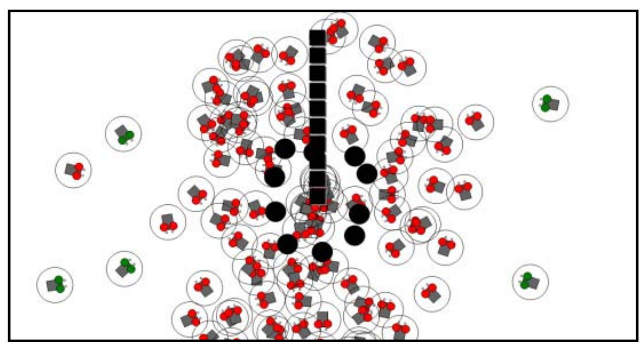

Figure 10. 100 bots used

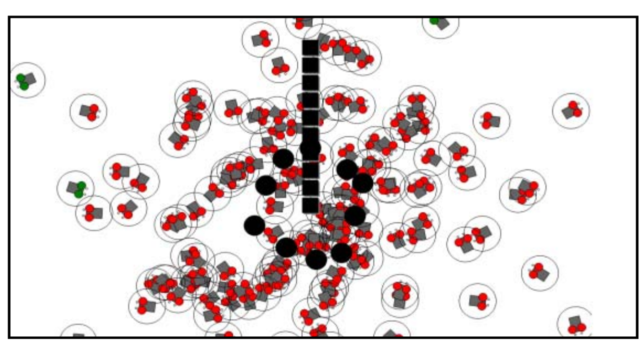

Figure 11. 150 bots used

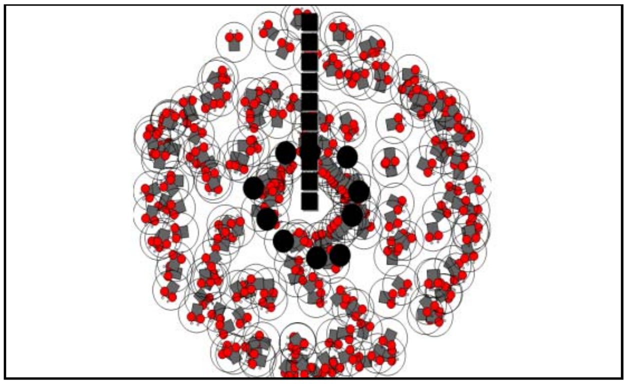

Figure 12. 200 bots used

Dynamic obstacles were added to the environment. These obstacles are movable depending on where the simulator/ user desires to put the obstacles in the environment area for testing purposes. 50 bots were first tested to walk-through the environment with the dynamic obstacles placed in the desired points by the user. It is observed that the bots avoid the obstacles (marked by black squares and circles) and somehow, tried to escape the area they are located in the environment. The bots still experience collisions with an average 40 bots per period. It was then followed up by using 100 bots, 150 bots and lastly, 200 bots.

IX. Conclusions

In the initial simulation of the swarm robots in an indefinite area, the source code is modifiable in terms of the number of bots to be added or robots present in the environment, the area of the environment is not yet determined since limitations is yet to be implemented in the area. The robots in the environment have collision detectors. In the simulation, it was able to send a signal (red blinking ears) that represents collision. The initial reaction of the robots is to avoid going to the path where the other bot it bumped into is heading. The robots are moving randomly and are initially in a circular formation. The GUI developed is to modify the size of each bot (as specified in the robots' specification) number of robots present in the environment, size of the environment, and the percentage of obstacles based in the platform. Initially, 50 bots were put in the environment with random movements and it shows normal movement without lag. As the number of bots is increased, it was observed that among the ranges 50-200 the most number of bots with normal movement and without lag is at 55-60 bots. Using a hundred bots in the environment, it experienced a lot of collisions in the obstacles and with bots. In spite of that, it still managed to explore the environment with a few bots not experiencing collisions.

With 150 and 200 bots in the environment, the bots experienced most of the collisions with all of the bots experiencing collisions at the initial position assigned to the bots. The simulation with 150 bots, however, gave a better response and less delay compared to the simulation with 200 bots.

References

[1] E. Bonabeau, G. Theraulaz, "Swarm smarts", *Scientific American, Inc.* , pp. 73–79, March 2000.

[2] W.A.F.W. Othman, B.P. Amavasai, S.P. McKibbin, F. Caparrelli, "An Analysis of Collective Movement Models for Robotic Swarms", *The International Conference on "Computer as a Tool"*, pp. 2373-2380, 2007.

[3] F. Gholami, M. J. Mahjoob, "An Investigation of Parameters in Ant Colony Optimization for a Path Optimization Algorithm", *International Conference on Mechatronics and Automation*, August 2007.

[4] Y. Tan, Z. Zheng, "Research Advance in Swarm Robotics", *Defence Technology vol. 9*, pp. 18 - 39, 2013.

[5] J. Hereford, "Analysis of a New Swarm Search Algorithm Based on Trophallaxis", *IEEE*, 2010.

[6] S. Camazine, "The Regulation of Pollen Foraging by Honey Bees: How Foragers Assess the Colony's Need for Pollen", *Behavioral Ecology and Sociobiology, vol. 32*, pp. 265 - 273, 1993.

[7] S. Camazine et al., "Protein Trophallaxis and the Regulation of Pollen Foraging by Honeybees (Apismellifera L.)", *Apidologie, vol. 29*, pp. 113-126, 1998.

[8] N. Kalra, A. Martinoli, "Comparative study of market-based and threshold-based task allocation", *Distributed Autonomous Robotic Systems vol. 7*, pp. 91 – 101, 2006.

[9] D. Zhang, "Adaptive task assignment for multiple mobile robots via swarm intelligence approach", *Robotics and Autonomous Systems 55*, 2007

[10] G. Pini, "Interference reduction through task partitioning in a robotic swarm", *IRIDIA Technical Report Series,* April 2009.

[11] D. O. Sales, M. A. Dias, F. S. Osorio, "Grid Ant Colony Optimization Applied to a Multi-robotic Garbage Collection System", *Robotics: SBR-LARS Robotics Symposium and Robocontrol,* pp. 187 – 192, October 2014.

[12] D. Payton, M, Daily, R. Estowski, M. Howard, C. Lee, "Pheromone Robotics", *Autonomous Robots 11*, pp. 319 – 324, 2001.

[13] D. Payton, R. Estkowski, H. Mike. "Compound behaviors in pheromone robotics", *Robot Autonomous Systems*, 2003.

[14] R. Moritz, "Trophallaxis-Inspired Self-Organized Task Exchange in Heterogeneous Swarms", *IEEE Conference,* 2011.

[15] A. Bandala, R. Vicerra and E. Dadios, "Swarming Algorithm for Unmanned Aerial Vehicle (UAV) Quadrotors in Swarm Behavior for Aggregation, Foraging, Formation and Tracking," Journal of Advanced Computational Intelligence and Intelligent Informatics, vol. 8, no. 9, 2014.