ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

A Novel Control Algorithm for Multi-Robot Pattern Formation

[I]Emmanuel Gbenga Dada, [II]Effirul Ikhwan Ramlana

[I]Dept. of Computer Engineering, Faculty of Engineering, University of Maiduguri, P.M.B 1069, Maiduguri, Borno State, Nigeria

[II]Natural Computing Laboratory, Department of Artificial Intelligence, Faculty of Computer Science and Information Technology, University of Malaya, 50603 Kuala Lumpur, Malaysia

Abstract

Pattern formation basically involves forming a particular shape by altering the positions of each robots in a swarm. The most common problem in the field of swarm robotics is dealing with managing and directing how a sizable number of robots navigate in the search environment to accomplish an assignment in unity. This type of task is normally impracticable, demanding and laborious for a particular set of robots to accomplish. The inspiration of swarm robots is fundamentally drawn from the study of behaviour of animals in group like a flock of birds, a herd of cattle, and a shoal of fish. In this paper, we used our novel hybrid algorithm called Primal Dual PSO (pdPSO) to solve the pattern formation problem in swarm robotics. Our simulation result provides a clear indication of the effectiveness of the algorithm. The results of our simulations show that our hybrid algorithm's performance is robust and scalable. It has great pattern formation capability for homogenous set of swarm robots compared to some other variants of PSO.

Keywords

Particle Swarm Optimization (PSO), Primal-Dual Particle Swarm Optimization (pdPSO), Pattern formation, Pheromone, Swarm Robots, Interior Point Method, Primal-Dual, gbest, and lbest

I. Introduction

Pattern formation basically involves changing the position of each robots in a swarm based on some predefined rules in order to form a particular shape [1]. This is also a non-regular arrangement behaviour that can be utilized to arrange robots in an orderly and rhythmic manner. According to Meinhardt [2], the inspiration of designing algorithms to make a swarm of robots to form a particular pattern came from natural processes like the development of colour patterns in animals. Another example, as stated by Langer [3], where the inspiration for pattern formation was drawn is in such physical activity such as mineral development.

Egerstedt and Hu [4] proposed a pattern formation method that uses the organization approach for a swarm of robots to form a given shape. The technique was used in a simulated environment that took obstacle avoidance into consideration to control the movement of robots in order to form a triangular shape. The strength of this approach is that the tracking of robots was properly done thereby stabilizing the pattern formation error.

Koo and Shahruz [5] presented another pattern formation approach that uses a set of unmanned aerial vehicles (UAVs) to form a preferred shape using centralized pattern formation method. The main emphasis of their research work is the computation of the route through which the unmanned aerial vehicles will travel through.

Belta and Kumar [6] proposed a method that uses an invariable kinetic energy measurement to create plane paths for a swarm of robots to navigate. The closeness between one robot and another in the swarm can be regulated using some parameters. However, their approach failed to consider obstacle avoidance and it is not scalable. Krishnanand and Ghose [7] proposed a pattern formations algorithms of simple robots using indigenous prototypes and attitudinally distributed communications. Ikemoto et al. [8] presented a pattern formation approach based on steady spatial formation of similar robots into a desired shape. Chen et al. [9] also proposed a decentralized pattern formation algorithm to enable mobile robots in forming a given shape. Elor and Bruckstein [10] proposed a pattern formation approach that deploys multiple identical robots in a swarm to form a particular shape.

From the research work that we have considered so far, there is no guarantee that the robot will converge to form a desired pattern [11]. Even if the given shape is formed, it is a weak formation that have no firm symmetrical shape. Many of the determining factors about the formation of patterns are based on presumptions. Examples of such deductions include the detection, direction-finding, interaction and computational competences of the robots. The robots also have negligible or no perceptive of other robots in the search space. Furthermore, there is insignificant or no interaction among the robots [11]. The performance of the swarm at a global level will be largely influenced by the performance of the individual agent at the local level. Some of the traits of swarm robotics that have been extensively investigated are convergence, foraging [12], pattern formation [1], flocking, aggregation and segregation [14], box-pushing [15], cooperative mapping [16], soccer tournaments [17], site preparation [18] and sorting [19].

Recent researchers have shifted much of their attention to do extensive study on multi-robot systems from the perspective of engineering and artificial intelligence. Pattern formation has been considered an initial phase for an effective flocking for many reasons including corresponding carriage of loads, preventing intrusion and etc. The aim of the research on solving pattern formation problem is to provide some guidelines and principles that will make the robots in the swarm to work as a group from a local level to achieve complex tasks overall.

Multi-robot pattern formation is an extremely needed solution in majority of the problem areas where a swarm of robots are employed and it is essential to organize them in a particular manner. The formation can be described as an arrangement in a constrained working area, in which individual robot is given a predefined gap between them and their neighbours. Multi-robot pattern formation is defined as a configuration in a bounded workspace, where each robot is at a desired distance from its neighbors. The desired formation is specified in terms of relative distances, so that the formation can be achieved in any part of the workspace and at any orientation.

In this paper, we propose a solution to some of the drawback of pattern formation for an identical set of robot using Primal-

© 2013, IJARCST All Rights Reserved

Dual Particle Swarm Optimization (*pd*PSO) algorithm and artificial pheromone. Fundamentally, the artificial pheromone trail integrated into the *pdPSO* algorithm serves as the medium through which messages are transferred from one robot to another. The communication at the local level enables the robots to make decisions on the next point to move to in the map. We employed the operational rule in [19] for the robot. The *pd*PSO method aids the assignment of different robots to various sections of the given pattern. The different experiments conducted proves the ability of our method to make the robots form the given pattern. It also indicate the tremendous scalability of our algorithm.

The virtual pheromone was introduced to guarantee the effectiveness of harmonization among the robots and to prevent the robots from gathering in a certain part of the pattern and being totally absent in some other parts. For the start, the initial value of the pheromone is set to 0. The robot mimic the natural boosting action of pheromone in biology by updating the pheromone and propagating the information to its neighbours that are within the limited communication range anytime it locates a grid that is still available for occupation. The maximum value of pheromone is set to prevent run-off. The virtual pheromone level will continue to decrease with time until it is finally eradicated. Figure 1 below is a pictorial representation of how the pheromone works.

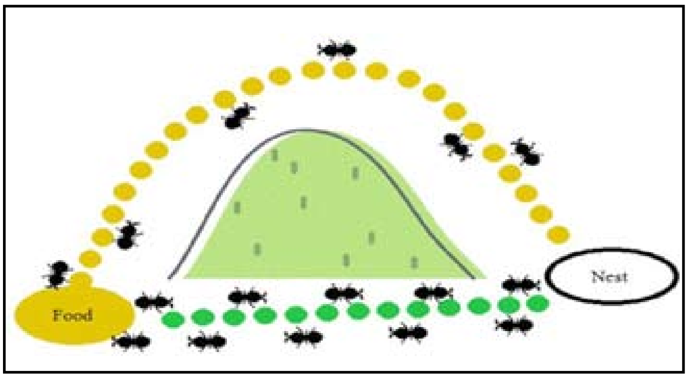

Fig.1: Pictorial Representation of Pheromone

II. World Definition

We presume that the environment consists of an M×N grid-based map. We vary the number of agents distributed in the map to 50, 100, 150, 200, and 250. Our method uses one type of pheromone and also holds the map of pattern (location of the grids of pattern found by the current agent or agents near to the current agent) in its memory. The pheromone is used for exploration (searching the map) and the map of pattern is used for convergence (converging to the grids of pattern found). The pheromone is updated when an agent moves in the map and the map of pattern is updated when an agent found a grid of pattern. However, agents communicate with other agents nearby and try to update pheromone and map of the pattern. In exploration mode, an agent moves in a deterministic manner based on the pheromone (moving to the grids with lower pheromone to explore unvisited grids). In convergence mode, agent moves based on map of pattern semi-stochastically (moving to the grids with higher pheromone to approach to the grids of the pattern). In this method, we use sub-area (blocks) for pheromone rather than sub-area (blocks) for map of pattern unlike the approach used in the work of Xu et al. [20]. This means that the agent must hold a matrix of pheromone, and the size of this matrix must be smaller than that of the map. It must also hold a matrix for map of pattern so that the size of the matrix is equal to the size of the map. Some of the presumptions that guide the work of the agents of this swarm robotic system are as follow:

(1) Every agent in the system is indistinctive and similar; meaning they cannot be differentiated by their exterior look;
(2) There is a limited range of communication for individual agents;
(3) There is an accurate recording of the decentralization and navigation of agents in the grids;
(4) Agent to agent, or agent to obstruction collisions are insignificant.

In figure 2 below, the flowchart for the operation of pheromone is presented.

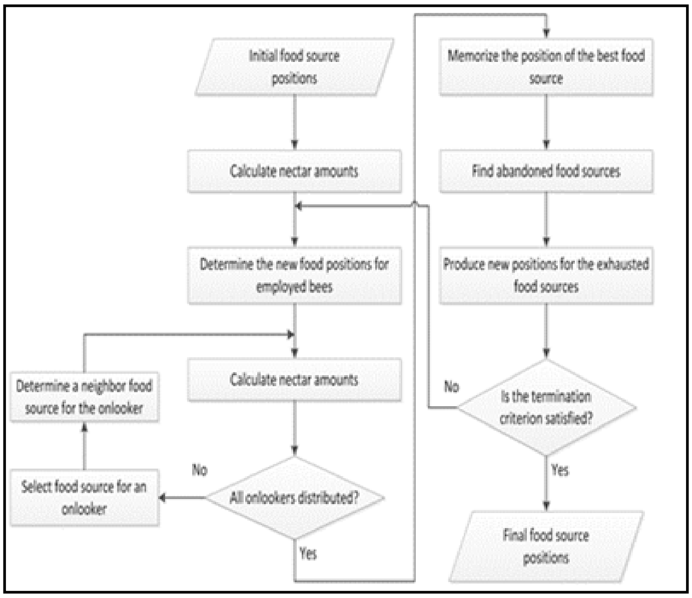

Fig. 2: Pheromone Flowchart

In figure 3 below is the pictorial representation of the virtual pheromone grid map.

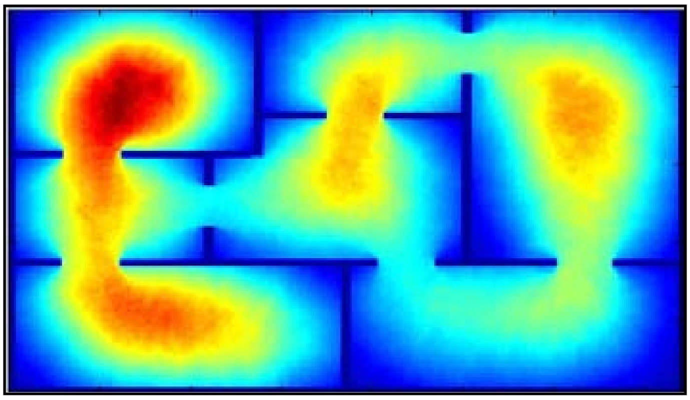

Fig. 3: Virtual Pheromone Map

III. Problem Statement

Assuming that we have an ellipse pattern on the grid of map, which is a group of coordinates. The pattern formation problem is the problem of designing an algorithm that is decentralized, in a way that, as the agents are distributed randomly, they will finally form the coveted ellipse pattern. Here, we have an unfamiliar area and a pattern defined beforehand. The swarm which is initially started randomly, searches grids for the coordinates that represent the pattern and ultimately complete taking the shape of the preferred pattern The aim of our experiment is to build an ellipse pattern formation algorithm that can allow more agents to be added to the number of agents in the swarm and thereby allowing the swarm

© All Rights Reserved, IJARCST 2013

robot system within the least time frame to form the predefined pattern.

IV. Pattern Formation Algorithm

At the onset, all the robots are haphazardly positioned on the map, and some of the grids on the map are designated as the predefined pattern where the information are stored in the matrix. A local search is first conducted since the robots are positioned randomly on the map with the robots set to the spreading activated mode searching for an unpopulated designated grid. With the spreading mode activated, each agent uses the virtual pheromone to communicate the information of its own local area in the grid with its neighbours, thereby increasing learning. A Primal-Dual PSO (*pd*PSO) [20] based searching approach that uses pheromone information is used for allotting local jobs among robots. The algorithm uses the value of distance that exists amid grids as a guide to update the location of the agent on the map. The exploration mode is activated whenever the agent notices that there is no vacant designated grid in the sub-area, enabling it to find a nearby occupied sub-area that have a limited number robots now occupying it. This operation continues, pending the time when all the robots sent to the designated grids and the pattern specified by the designated grids is created by the robots in the swarm. We invented two approaches for switching between exploration and convergence:

1 A certain number of iterations (e.g. 80) for exploration and then a certain number iterations (e.g. 20) for convergence. Of course, the iterations for exploration are always performed before the iterations for convergence.
2 Stochastic switching between exploration and convergence. In this approach, the probability of exploration mode in the first iteration is very high and gradually, the probability of convergence mode is increased. In the last iterations, the probability of convergence mode is very high. We combine the two approaches mentioned above. There is stochastic switching between exploration and convergence while the probability of exploration mode is zero if the number of iterations is more than the threshold. In this case, the probability of convergence mode is very low when the number of iterations is less than the threshold.

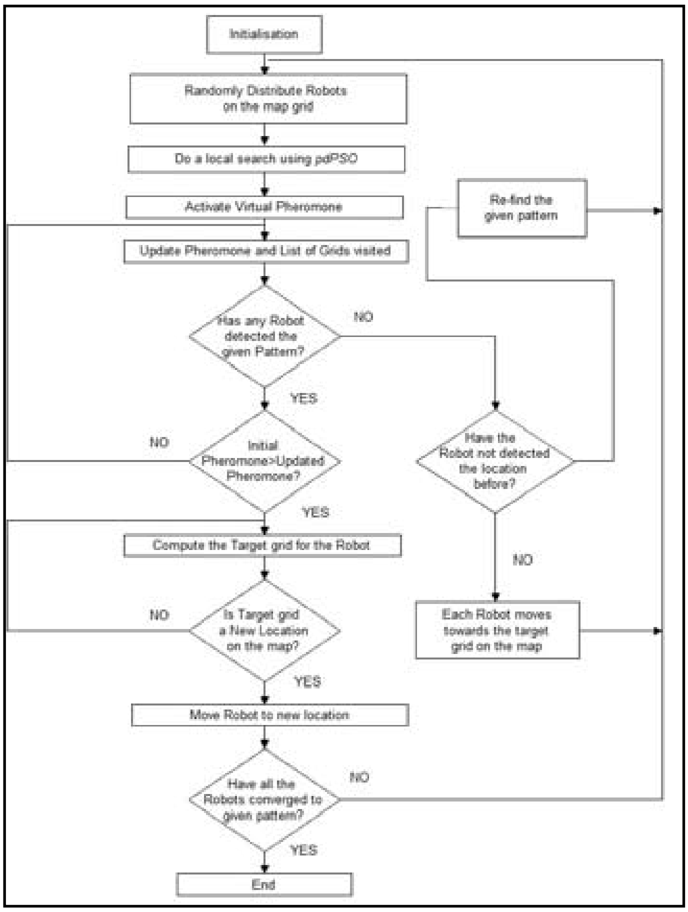

Fig. 4: Flowchart of Pattern Formation Algorithm

V. Particle Swarm Optimization (PSO) Algorithm

In the PSO algorithm, all the particles are randomly introduced and evaluated to calculate fitness of the particles in the swarm. It also computes the local best which is the best value of individual particle and global best which is the best value of particle in the entire swarm [21], [22]. To get the optimum solution, some iterative steps are involved. During the looping process, the velocity of the particles is first updated by the local and global bests. After this, the position of the individual particle is then updated by the up-to-date velocity of the particle. Once the stopping criteria, which has already been predetermined is satisfied, the loop will be terminated. The representation of PSO particle position and velocity update is shown in figure 4.

Where

X_i = the position of a particle
V_i = the velocity of the particle
N = the number of particles in the swarm
i = the particle's number (where i = 1…N)

The i[th] particle is represented as $X_I = (X_{i1}, X_{i2}, ..., X_N)$. The velocity is the rate at which the next position is changing with respect to the current position. $V_I = (V_{i1}, V_{i2}, ..., V_N)$ represents the velocity for the particle, i. At the start of the algorithm, initial numerical values of the position and velocity of particles are assigned haphazardly. This is followed by using equations (1) and (2) to update the position and velocity of the particles, after subsequent iterations are conducted during the search process.

© 2013, IJARCST All Rights Reserved

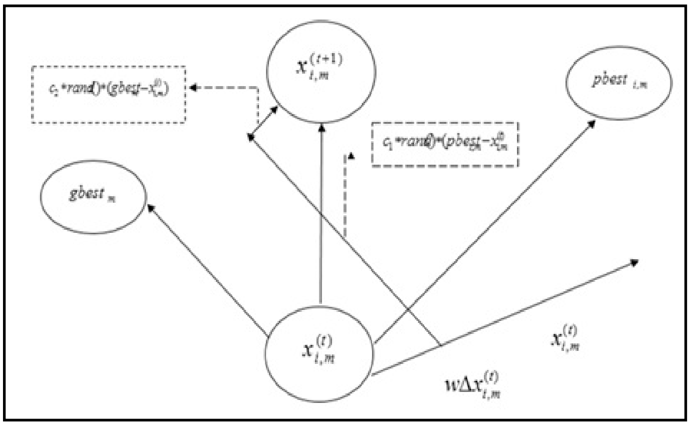

Fig. 4: Diagrammatic Representation of PSO Particle Position and Velocity Update

In PSO, (X_i) represents the position of a particle, and (V_i) the velocity of the particle. The particle's number is *i*, where ($i = 1,\ldots,N$), and *N* is the number of particles in the swarm. The i^{th} particle is represented as, $X_I = (X_{i1}, X_{i2},\ldots,X_N)$ whereas the velocity is the rate at which the next position is changing with respect to the current position. $V_I = (V_{i1}, V_{i2},\ldots,V_N)$ represents the velocity for the particle, *i*. At the start of the algorithm, initial numerical values of the position and velocity of the particles are assigned randomly. Equations (1) and (2) will then update the position and velocity of the particles for subsequent iterations during the search process.

$$v_{i,m}^{(t+1)} = w * v_{i,m}^{(t)} + c_1 * rand1() * (pbest_{i,m} - x_{i,m}^{(t)}) + c_2 * rand2() * (gbest_m - x_{i,m}^{(t)}) \quad (1)$$

$$x_{i,m}^{(t+1)} = x_{i,m}^{(t)} + v_{i,m}^{(t+1)} \quad (2)$$

According to Shi and Eberhart [22], to avert eruption, the value $v_{i,m}^{(t+1)}$ is fixed at $\pm v_{max}$. This is because the value of v_{max} will be too large if the search range is very wide. If v_{max} is too small, the scope of the search will be excessively limited, thereby forcing the particles to support local exploration. The "*w*" is the inertia weight (constriction factor) and this regulates the algorithm searching properties.

Shi and Eberhart [22] suggested that a large inertia value (a more global search) initiation will be dynamically reduced towards the end of the optimization (a more local search). The use of small inertia weight usually guarantees quick convergence as the time spent on the exploration of the global space is reduced [23]. The inclusion of *w* in the equation is to provide equilibrium between the global and local search capability of the particles. The positive constant c_1 and c_2, representing the cognitive scaling and social scaling factors, are set to the value of 2 [24].

Both the cognitive and social scaling factors assist the PSO to successfully build the local bests into the global best [24]. The stochastic variables $rand1()$ and $rand2()$ have uniform distribution. These random variables are independent functions that provide energy to the particles. The best position found so far by the particle is represented as $pbest_{i,m}$. The best position attained by the neighboring particles is represented as $gbest_m$. There are two types of neighborhood particles in PSO, and the type of neighborhood is what determines the value of $gbest_m$. The two types of neighborhood are:

1. *gBest* (Global neighborhood) – There is a full connection among the particles, and the exploration of swarm is controlled by the best particle in the swarm.
2. *lBest* (Local neighborhood) – There is no full connection among the particles in the swarm, rather they are connected only to their neighbors.

Equation (2) is used in updating the position of the particles whereby the velocity is added together with the earlier position and a new search is started from its former position. Eberhart and Shi [22], in their work, bounded $x_{i,m}^{(t+1)}$ to avoid a situation whereby particles are spending too much time in infeasible region. A problem dependent fitness function is used in evaluating the superiority of $x_{i,m}^{(t+1)}$. Assuming the present solution is superior to the fitness of $pbest_{i,m}$ or $gbest_m$, then the new position will replace $pbest_{i,m}$ or $gbest_m$ accordingly. Unless the condition for ending the search (either the iteration has reached its peak or we have gotten the desired solution), this updating process will continue. The optimal solution is where the best particle is found when the stopping criterion is satisfied [24], [25].

VI. Primal-Dual Interior Point Method

The primal-dual interior-point (PDIP) method is an excellent example of an algorithm that uses the constraint-reduction methods. Mehrotra [26], in his research work, developed the Mehrotra's Predictor-Corrector PDIP algorithm, which has been executed in the majority of the interior-point software suite for solving both linear and convex-conic problems [27]. The primal-dual methods are a new category in interior-point methods that have recently been practically employed for solving large-scale nonlinear optimization problems [28]. Contrary to the traditional primal method, primal-dual methods evaluates both the primal variables x and dual Lagrange multipliers λ relating to the constraints concurrently. The disconcerted Karush-Kuhn-Tucker (KKT) equation below can be solved using the precise primal-dual solution (x^*_μ, λ^*_μ) at a given parameter μ

$$\begin{cases} \nabla F(x) - C^T\lambda = 0 \\ \lambda_i C_i(x) = \mu, i = 1,\ldots,m \end{cases} \quad (3)$$

with the constraint $(C(x),\lambda) \geq 0$.

The Newton's algorithm and line search approach are employed to recursively solve any primal or primal-dual sub-problems for a given μ value [29] [30]. Feasibility and convergence is enforced in the algorithm by selecting the size of step in the iteration. This can be achieved by appropriately reducing the merit function used in gauging degree of advance to the solution. The dual variables of the primal-dual can be protected by using $F\mu$ as a function that can incorporate the primal and dual variables [31] and at the same time measuring the harmony between data and the fitting model for a particular choice of the parameter [32]. The major setback of the barrier functions is the ineffectiveness of the traditional line exploration methods, thereby necessitating the development of more efficient line search [32].

According to [33], primal-dual method can efficiently handle large linear programming problems (the bigger the problem size, the more noticeable the efficiency of the primal-dual algorithm). The algorithm is not susceptible to degradation and the number of iterations do not depend on the number of vertices in the feasible search space [26]. Primal-dual algorithm uses considerably less iterations as compared to the simplex method. The algorithm is

© All Rights Reserved, IJARCST 2013

Vol. 4, Issue 3 (Jul. - Sept. 2016)
ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

able to generate ideal solutions for a linear programming problem in less than 100 iterations regardless of huge number of variables involved in nearly all its implementations [34].
However, the primal-dual method is hindered by its inability to detect the possibility of having unbounded status of the problem (to a certain extent, the method is labeled as incomplete). This issue has been addressed thoroughly using undiversified model as suggested in [34, 31].

VII. Pattern Formation Results

The figures below shows the results of the pattern formation of the robots using our algorithm.

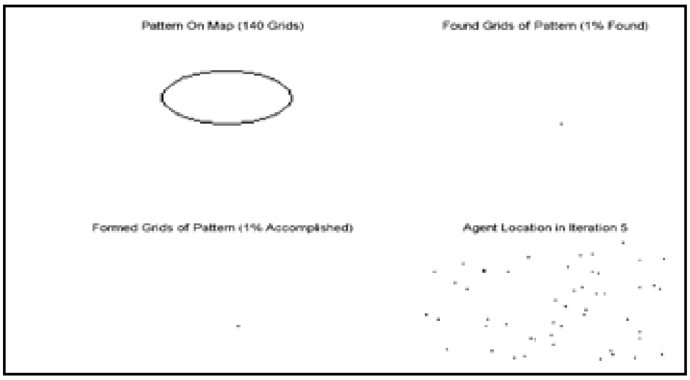

Fig. 5: 50-Agent Pattern Formation (black spots indicate robots)

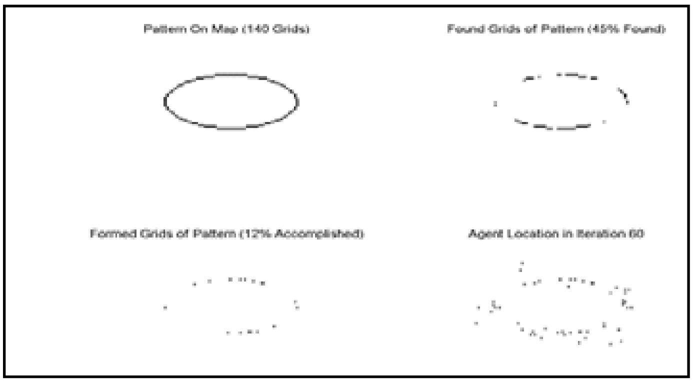

Fig. 6: 100-Agent Pattern Formation (black spots stands for the robots)

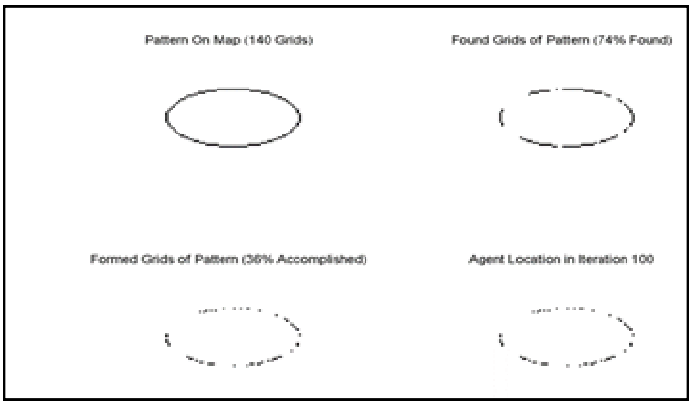

Figure 7: 150-Agent Pattern Formation (black spots stands for the robots)

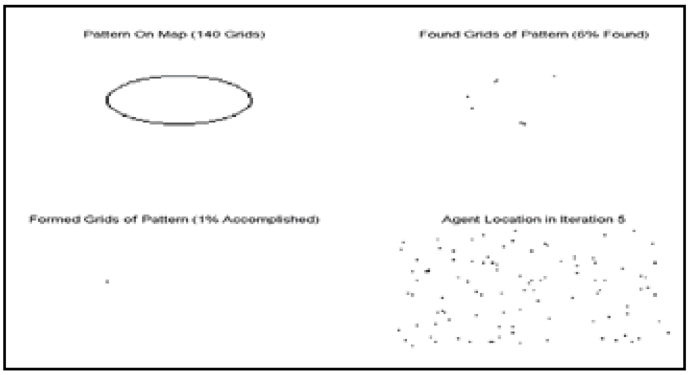

Fig. 8: 200-Agent Pattern Formation (black spots stands for the robots)

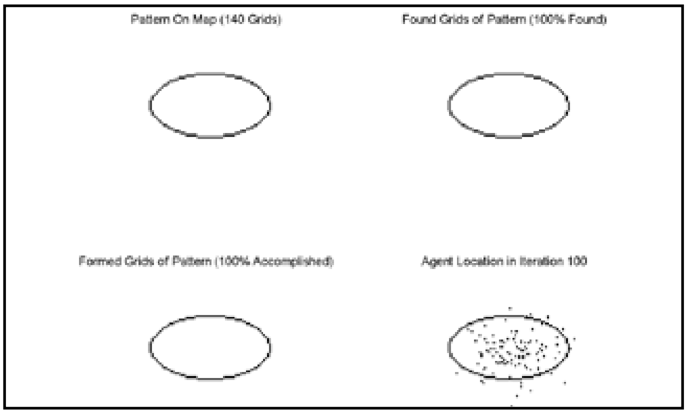

Fig. 9: 250-Agent Pattern Formation (black spots stands for the robots)

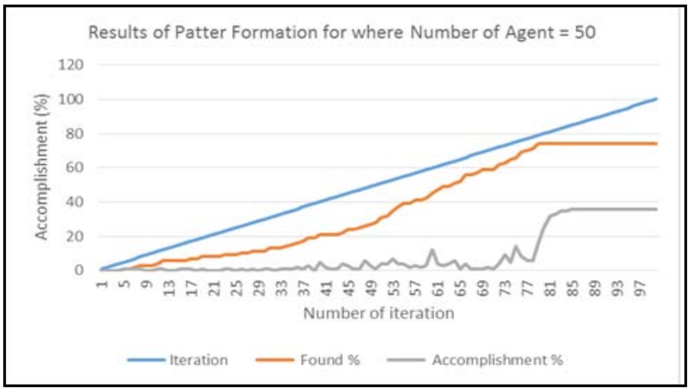

Fig. 10(a): Graph of Pattern Formation Using 50 Agents

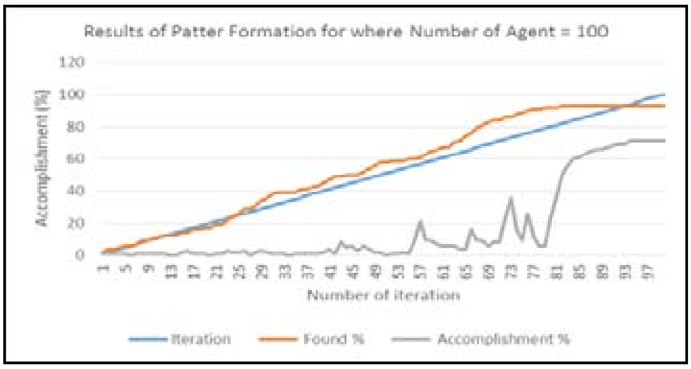

Fig.10(b): Graph of Pattern Formation Using 100 Agents

© 2013, IJARCST All Rights Reserved

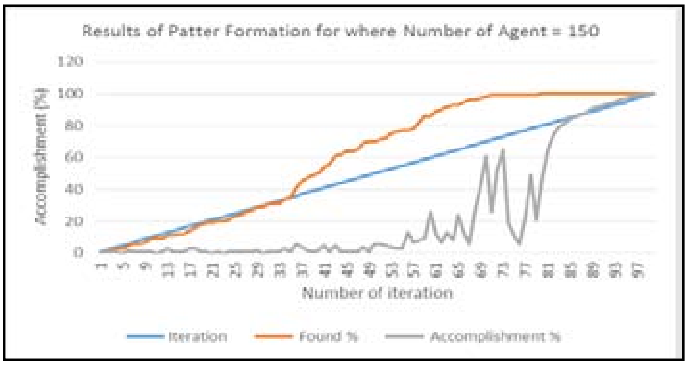

Fig. 10(c) : Graph of Pattern Formation Using 150 Agents

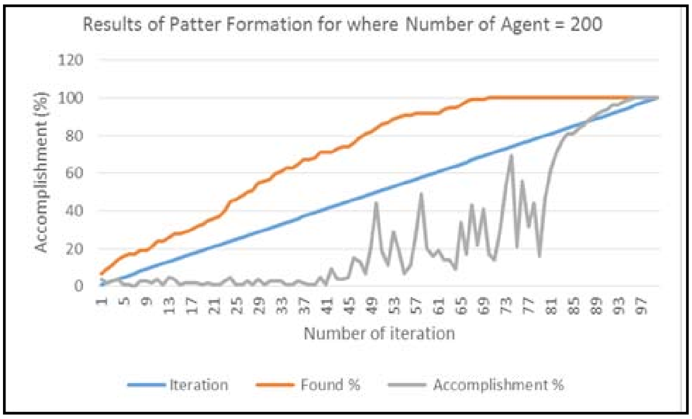

Fig. 10(d): Graph of Pattern Formation Using 200 Agents

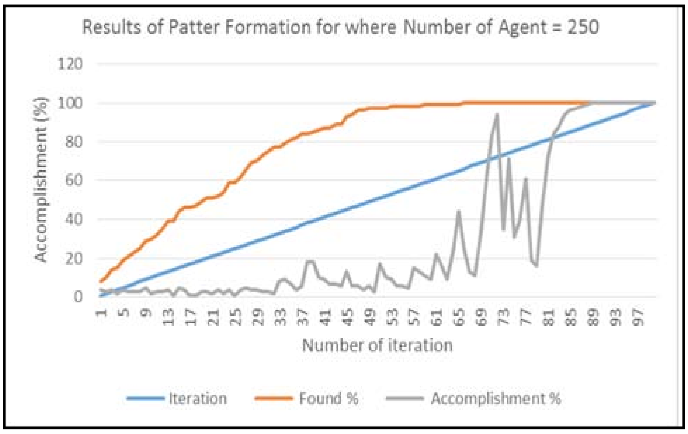

Fig. 10 (e): Graph of Pattern Formation Using 250 Agents

A. Discussion

The results of our simulation of 50-agents, 100-agents, 150-agents, 200-agents, and 250-agents are presented in figures 5–9. The figures illustrate the percentage (%) of accomplishment in the process of forming ellipse with different number of robots that comprise the swarm. We set the pheromone attenuation rate and pheromone accumulation rate to 0.01. The maximum value for pheromone is set to 1000. The effectiveness of our algorithm largely depends on the values of the pheromone parameters and the pdPSO. We discovered from our experiment that the attenuation factor in pheromone has a great effect on the ability of our algorithm to form a desired pattern. The PSO acceleration factor for local search and global search is set to 2 .0 and the inertia parameter to 0.1. Figures 10a - 10e depict the graphical representations of our results. As illustrated in the graphs above, the percentage of accomplishment will reach 100% as the process continues except where we have 50-agents and 100-agents. This explains that the number of agents have a great influence on the efficiency of our algorithm in forming a desired pattern. Our simulation results also signify that the scalability capacity of our approach is of commendable level.

VIII. Conclusions

In this paper, we proposed a hybrid algorithm strategy for solving pattern formation problem of swarm robots. This algorithm combines the explorative ability of PSO with the exploitative capacity of the Primal-Dual Interior-Point Method thereby possessing a strong capacity of avoiding premature convergence, making the robots to form the desired pattern. For the sake of effective coordination and communication among the robots, the virtual pheromone was introduced. The use of *pdPSO,* which is a PSO-based technique, ensures the efficiency of the pattern formation. We carried out some simulation to test the ability of our proposed algorithm to solve this problem and to evaluate its performance. In our future work, we plan to implement our algorithms on hardware of physical swarm robots.

IX. AcknowledgEment

Emmanuel Gbenga Dada wants to thank the University of Maiduguri, Nigeria for granting him the study fellowship, enabling him to come to University Malaya, Malaysia for his PhD study. Effirul Ikhwan Ramlan is taking this opportunity to appreciate University Malaya as this research is supported by the High Impact Research MoE Grant UM.C/625/1/HIR/MoE/FCSIT/002 from the Ministry of Education, Malaysia and University Malaya.

References

[1] *Yamauchi, Y. A Survey on Pattern Formation of Autonomous Mobile Robots: Asynchrony, Obliviousness and Visibility. J. Phys.: Conf. Ser. Journal of Physics: Conference Series 473 (2013): 012016.*

[2] *Meinhardt H. Models of biological pattern formation. London: Academic Press; 1982.*

[3] *Langer, J. S. Interfacial Instabilities and Dendritic Crystal Growth. Prog. Theor. Phys. Suppl. Progress of Theoretical Physics Supplement 64 (1978): 463-76.*

[4] *Egerstedt, M., and Hu., X. Formation Constrained Multi-agent Control. Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164).*

[5] *Koo, T.J., and Shahruz, S.M. "Formation of a Group of Unmanned Aerial Vehicles (UAVs)." Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148) (2001).*

[6] *Belta, C., and Kumar., V. "Trajectory Design for Formations of Robots by Kinetic Energy Shaping." Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292).*

[7] *Krishnanand, K.N., and Ghose., D. "Formations of Minimalist Mobile Robots Using Local-templates and Spatially Distributed Interactions." Robotics and Autonomous Systems 53.3-4 (2005): 194-213.*

[8] *Ikemoto Y, Hasegawa Y, Fukuda T, Matsuda K. Gradual spatial pattern formation of homogeneous robot group. Information Sciences. 2005;171: 431–445. doi:10.1016/j.ins.2004.09.013.*

[9] *Chen F, Chen Z, Liu Z, Xiang L, Yuan Z. Decentralized formation control of mobile agents: A unified framework.*

© All Rights Reserved, IJARCST 2013

Physica A: Statistical Mechanics and its Applications. 2008;387: 4917–4926. doi:10.1016/j.physa.2008.04.018.

[10] Elor, Y., and Bruckstein., A.M. "Uniform Multi-agent Deployment on a Ring." Theoretical Computer Science 412.8-10 (2011): 783-95.

[11] Gautam, A., and Mohan, S. "A Distributed Algorithm for Circle Formation by Multiple Mobile Robots." 2013 International Conference on Control, Automation, Robotics and Embedded Systems (CARE) (2013).

[12] Krieger, M., Billeter, J. B. and Keller, L. "Ant-like Task Allocation and Recruitment in Cooperative Robots." Nature, (2000) 406: 992-995.

[13] Balch, T., and Arkin., R.C. "Behavior-based Formation Control for Multirobot Teams." IEEE Trans. Robot. Automat. IEEE Transactions on Robotics and Automation 14.6 (1998): 926-39.

[14] Martinoli, A., Ijspeert, A.J., and Mondada., F. "Understanding Collective Aggregation Mechanisms: From Probabilistic Modelling to Experiments with Real Robots." Robotics and Autonomous Systems 29.1 (1999): 51-63.

[15] Mataric, M.J., Nilsson, M., and Simsarin, K.T. "Cooperative Multi-robot Box-pushing." Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots.

[16] Yamauchi, B. "Decentralized Coordination for Multirobot Exploration."Robotics and Autonomous Systems 29.2-3 (1999): 111-18.

[17] Weigel T, Gutmann J-S, Dietl M, Kleiner A, Nebel B. CS Freiburg: coordinating robots for successful soccer playing. IEEE Trans Robot Automat IEEE Transactions on Robotics and Automation. 2002;18: 685–699. doi:10.1109/tra.2002.804041

[18] Holland O, Melhuish C. Stigmergy, Self-Organization, and Sorting in Collective Robotics. Artificial Life. 1999;5: 173–202. doi:10.1162/106454699568737.

[19] Xu H, Guan H, Liang A, Yan X. A Multi-robot Pattern Formation Algorithm Based on Distributed Swarm Intelligence. 2010 Second International Conference on Computer Engineering and Applications. 2010; doi:10.1109/iccea.2010.22

[20] Dada E G, Ramlan E I. Primal-Dual Interior Point Method Particle Swarm Optimization (pdipmPSO) Algorithm. 3rd International Conference on Advances in Engineering Sciences and Applied Mathematics (ICAESAM'2015), March 23-24, 2015 London (UK). 2015; doi:10.15242/iie.e0315072

[21] Talukder, S. Mathematical modelling and applications of particle swarm optimization. Master's thesis, Blekinge Institute of Technology, The School of Engineering, 2011.

[22] Shi, Y., and Eberhart., R.C. "Parameter Selection in Particle Swarm Optimization." Lecture Notes in Computer Science Evolutionary Programming VII (1998): 591-600.

[23] Aziz, N.A.A., and Ibrahim., Z. "Asynchronous Particle Swarm Optimization for Swarm Robotics." Procedia Engineering 41 (2012): 951-57.

[24] Kennedy, J., Eberhart, R.C., and Shi., Y. "The Particle Swarm." Swarm Intelligence (2001): 287-325.

[25] Civicioglu, P., and Erkan, B."A Conceptual Comparison of the Cuckoo-search, Particle Swarm Optimization, Differential Evolution and Artificial Bee Colony Algorithms." Artificial Intelligence Review Artif Intell Rev 39.4 (2011): 315-46.

[26] Mehrotra, S. "On the Implementation of a Primal-Dual Interior Point Method." SIAM J. Optim. SIAM Journal on Optimization 2.4 (1992): 575-601.

[27] The logarithmic potential method of convex programming: with a particular application to the dynamics of planning for national development; synopsis of a communication to be presented at the International Colloquium of Econometrics in Paris, 23-2851955. Oslo; 1955.

[28] Wright SJ. Primal-dual interior-point methods. Primal-dual interior-point methods. Philadelphia: Society for Industrial and Applied Mathematics; 1997.

[29] Boyd S P, Vandenberghe L. Convex optimization. Convex optimization. Cambridge: Cambridge Univ. Pr.; 2011.

[30] V. H. Quintana, G. L. Torres. Introduction. Primal-Dual Interior-Point Methods. 1997;: 1–19. doi:10.1137/1.9781611971453.ch1.

[31] Johnson, C.A., Seidel, J., and Sofer. A. "Interior-point Methodology for 3-D PET Reconstruction." IEEE Transactions on Medical Imaging IEEE Trans. Med. Imaging 19.4 (2000): 271-85.

[32] Chouzenoux, E., Moussaoui, S., and Idier., J. "Efficiency of Line Search Strategies in Interior Point Methods for Linearly Constrained Signal Restoration." 2011 IEEE Statistical Signal Processing Workshop (SSP) (2011).

[33] Glavic M, Wehenkel L. A Survey, Short Survey of Applications to Power Systems, and Research Opportunities.Technical Report. University of Liège Electrical Engineering and Computer Science Department Sart Tilman B-28 4000 Liege, Belgium, 2004.

[34] Armand, P., Gilbert, J.C., and Jan-Jégou, S. "A Feasible BFGS Interior Point Algorithm for Solving Convex Minimization Problems." SIAM J. Optim. SIAM Journal on Optimization 11.1 (2000): 199-222.

Authors' Profile

Emmanuel Gbenga Dada is a PhD holder in Computer Science. He received his Doctoral degree from the Department of Artificial Intelligence, Faculty of Computer Science and Information Technology at University of Malaya, Malaysia. He received his Master of Computer Science in 2010 at the University of Ibadan, Nigeria. His current research interests include Soft Computing Algorithms (Particle Swarm Optimization, Fuzzy Logic, Neural Computing, Evolutionary Computation, Machine Learning, and Probabilistic Reasoning), Swarm Robotic, Optimization Techniques, and Information Security (Email: gbengadada2004@yahoo.com).

Effirul Ikhwan Ramlan received his PhD in Computer Science from Southampton University, UK in 2011. He graduated with BSc (Hon) in Intelligent Systems from Universiti Teknologi Mara (UITM) in 2003. He is a Senior Lecturer at the Faculty of Computer Science and Information Technology in University of Malaya, Kuala Lumpur. His research interests include Molecular Computing (Ribonucleic acids (RNA) / Deoxyribonucleic acids

© 2013, IJARCST All Rights Reserved

(DNA) Computing, Biomolecular Computing, Unconventional / Natural Computing), Algorithms (Machine Learning, Complexity Studies, Evolutionary Algorithms, Bio-inspired Algorithms), Software Development (Next generation sequencing analysis and Ribonucleic acid profiling). (Email: effirul@um.edu.my).

© All Rights Reserved, IJARCST 2013