

# Swarm Robotics with Circular Formation Motion Including Obstacles Avoidance

*Nabil M. Hewahi*

Department of Computer Science  
College of Information Technology  
University of Bahrain, Bahrain  
nhewahi@uob.edu.bh

*Abed Alkader A. Almobayed*

Department of Computer Science  
College of Information Technology  
Islamic University of Gaza, Palestine  
abedk2006@gmail.com

## Abstract

The robots science has been developed over the past few years, where robots have become used to accomplish difficult, repetitive or accurate tasks, which are very hard for humans to carry out.

In this paper, we propose an algorithm to control the motion of a swarm of robots and make them able to avoid obstacles. The proposed solution is based on forming the robots in circular fashion. A group set of robots consists of multiple groups of robots, each group of robots consists of robots forming a circular shape and each group set is a circular form of robots. The proposed algorithm is concerned with first locating the randomly generated robots in groups and secondly with the swarm robot motion and finally with the swarm obstacle avoidance and swarm reorganization after crossing the obstacle.

The proposed algorithm has been simulated with five different obstacles with various numbers of randomly generated robots. The results show that the swarm in the circular form can deal with the obstacles very effectively by passing the obstacles smoothly. The proposed algorithm has been compared with flocking algorithm and it is shown that the circular formation algorithm does not need extensive computation after obstacle avoidance whereas the flocking algorithm needs extensive computation. In addition, the circular formation algorithm maintains every robot in its group after avoiding the obstacles whereas with flocking algorithm does not.

**Keywords:** Robotics, Swarm Robotics, Circular Formation, Obstacle Avoidance.

## 1. Introduction

Due to heavy duties or sometimes very accurate duties (duties performed in very narrow areas), human being cannot perform those tasks and it becomes very important to use robots.

With the revolution of technology and the expanding of discoveries and science, the human's passion to know far things as in seas or in space is increasing. Also, humans want to reach undiscovered locations and difficult to reach places. This made robots become one of those possible potential solutions. Robots are pre-programmed or controlled and can carry out tedious, boring or dangerous tasks, and also able to do the needed works accurately. In addition, robots might involve in searching for survivors and rescue during natural disasters and wars. Other robots might be used to carry out surgery tasks or household works. The robot's work might vary from simple to complex task.

Fortunately, robots have begun working in these places, where most of them have the ability to move with wires or wireless like Uav planes which flies without a pilot, submarine or mobile

robot (Hereford & Blum, 2011). For such tasks one robot will not be enough and there is a need for more robots to work together.

To study the swarm's behavior, scientists study the behavior of ants, bees and fish, and how they work together to find food or solve their problems. Based on the motion of swarms and swimming of flocks, researchers build their algorithms that behave like fish schools, bird flocks, and ants (Geunho & Chong, 2008),(Hanada, Lee, & Chong, 2007),(Hereford & Blum, 2011),(La & Sheng, 2009),(Lee & Chong, 2008),(Scanniello & Erra, 2013).

Swarm robotics is the name which is given to a group of robots working together in Autonomous; robots that perform behaviors or tasks which are particularly desirable in fields such as; space exploration, cleaning floors, mowing lawns, waste water treatment, rescue after any kind of disaster, and discovery of areas damaged with Gas or contaminated with nuclear radiation(Cheng, Wang, & Dasgupta, 2009),(Stormont & Kutiyawala, 2007).

Researchers have worked in several fields related to swarm robotics; the way robotics can connect to each other by using simple techniques with little resources and how they share big tasks and solve them (Stormont & Kutiyawala, 2007). Researchers proposed several algorithms for swarm robotics to have one standalone algorithm for each robot; most of these approaches are based on fish school and motion of the birds including v formation and triangular formation (Meng & Gan, 2008).

In this study, we develop a swarm robotics motion based on circular formation that will help them move fast in order to be able to avoid obstacles and to easy reform themselves.

Swarm robotics is a group of robots that moves in a particular format based on a simple connection to carry out a certain task, each robot in the group works based on a certain algorithm to interact with other robots to help in achieving the group task. This task can be achieved by assigning a subtask for each robot in the group. There are many different kinds of robots that are used in swarms of robots, including:

- **Ant robotics:** This model follows the traditional way of ant life and communication, by producing pheromones to give signals to other ants. In 1991, James McLurkin, provided the robot with special sensors, infrared and communication system to be able to follow the signs and interpret the signals received from other robots. An algorithm was developed to mimic the ant behavior on robots and several experiments were conducted to test and discover the distribution of the ants workload performed(McLurkin, 1995).
- **Kilobot:** In 2010, a Small robot, which does not exceed a length of 3.3 cm, was produced in the Self-Organizing Systems Research Group at Harvard University. The cost for the robot does not exceed \$ 15 and these robots can interact in doing tasks as a group (Rubenstein, Ahler, & Nagpal, 2012).
- **Nanorobotics:** This technology is still in the process of growing up, where the scientists target is to have a nanometer-sized robot industry to be used in medicine operations by sending the robot into the bloodstream in the human body. Some minor attempts in this area succeeded (Freitas, 2005).
- **Microbotics:** An idea to design a robot less than 1 mm size and has a simple characteristics for use in a swarm of robots (Lehr et al., 1996).

## 2. Related works

Swarm robotics is a new research field, where researchers seek to develop small robots that have the potential to communicate with each other with the least amount of resources consumption to carry out major operations that cannot be performed by a single robot. There are different approaches which have been adopted for swarm robotics. These approaches can be classified into three types, communication (Guo, Meng, & Jin, 2010), flocking (Folino & Spezzano, 2002; Geunho

& Chong, 2008; Hanada et al., 2007; Hereford & Blum, 2011; La & Sheng, 2009; Lee & Chong, 2008; Reynolds, 1987; Turgut, Çelikkanat, Gökçe, & Şahin, 2008) and formation (Défago & Konagaya, 2002; Lee & Chong, 2008; Yun, Alptekin, & Albayrak, 1997).

### **2.1. Communication**

Swarm robotics is not always about the control and coordination of mobile agents, it entails the development of infrastructure for proper functioning such as communication, sensing, processing.

Nouyan and Dorigo (Nouyan & Dorigo, 2004) implemented a chain formation behavior. Initially the robots search for other chain members or the nest. Once a robot finds a chain member or the nest, it becomes a chain member depending on two predefined timeouts. The robots distinguish chain members and the nest based on the color of the LED ring around their body. A chain member can have three different colors: blue, green and red. It activates the color blue, if it connects to the nest or to a red chain member. It activates the color green, if it connects to a blue chain member and color red otherwise. This coloring mechanism allows robots to find the direction of the chain. Since having a long chain instead of a chain with several branches is preferred, the robot follows the color to reach to the end of the chain to connect.

### **2.2. Flocking**

What distinguishes a swarm of robots is the movement, which is either organized or simulates swarms of nature like a swarm of fish or flock of birds. In the case of nature like swarm, a robot is considered independent and moves autonomously. Every robot is having its own number despite that in the simulation environment the robot is considered a point in a two-dimensional space.

The main problem facing these robots is that it is separated from the rest of the swarm and do not take complex communication devices. On the other hand, the main characteristics of this approach are the simplicity in the process of communication, movement independently and carrying a decentralized algorithm.

The goal of flocking is to simplify the coordination of the communication process “low sensing” between the robots and to allow them to interact with each other.

Lee and Chong (Lee & Chong, 2008) proposed robot swarms in triangular shapes. This triangular formation is used to facilitate the movement of the swarm concurrently and simulates the movement of swarms of fish. The process of triangular formation is called "local interaction algorithm". In addition, another algorithm to make the swarm move forward simultaneously is developed. The swarm of robots keeps the organization of triangular formation during movement, which leads to a consumption of a lot of processes. Several experiments were conducted mainly to avoid obstacles by a swarm of robots.

### **2.3. Formation**

A fundamental problem in collective robotics is to have the group organize into global formations or patterns. These include simple patterns like circles, lines, uniform distribution within a circle or square, etc. In the presence of a central controller, these tasks are trivial, but this is not the case in a distributed system. The main goal is to have self-autonomy robots, where each behaves independently from the others based its surrounding environment including other robots' behavior. Each robot might not be aware that it works within a group.

Ducatelle et al.(Ducatelle, Di Caro, Pinciroli, Mondada, & Gambardella, 2011) proposed a collective behavior based on network routing, capable of guiding a robot from a source area to a target. Similarly to what happens in packet routing, the robots keep a table of the distance of other robots with respect to the target. A robot can then use the entries in the table and reach the target.

The authors studied two different experimental setups. In the first, only a single robot is moving while the others act as beacons to guide it. In the second, all the robots are moving between two points, creating a dynamic chain.

The previous works and studies have talked about swarm robotics motion in different formations like V and Fish school in addition to areas covering capability for searching and saving missions. Our proposed approach is an attempt to reduce the computational complexity of some of the previous approaches and to make robots motion more simple and capable of avoiding obstacles regardless of the obstacles structure.

### 3. Methodology and the proposed model

The General diagram of the proposed approach is shown in Figure 1 and presented below.

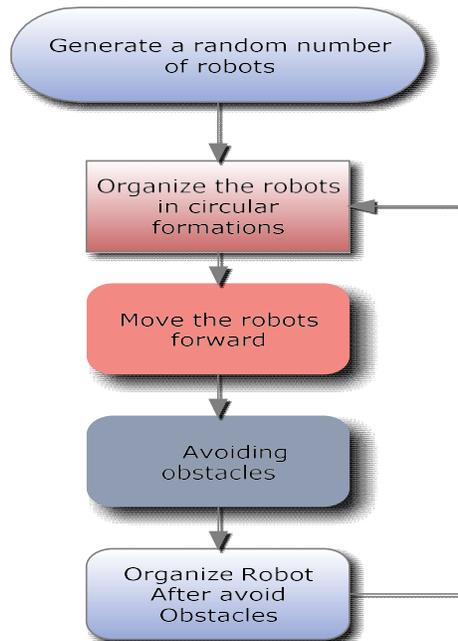


Figure 1. The five steps of Methodology Approach

- a) Generate a random number of robots to have a swarm.
- b) Organize the robots in circular formations, where every robot will be in a group related to circular formation. This means every circular formation is a cycle that has cyclic groups of robots on its circumference as shown in Figure 2 and Figure 3. We might have more than one circular formation.
- c) Move the robots forward in a steady state.
- d) Avoiding obstacles in case of facing an obstacle, and the swarm must adapt itself based on the type of the obstacle. Various types of obstacles will be considered.
- e) The swarm reorganizes itself after avoiding the obstacle in the same way as it was before facing the obstacle.

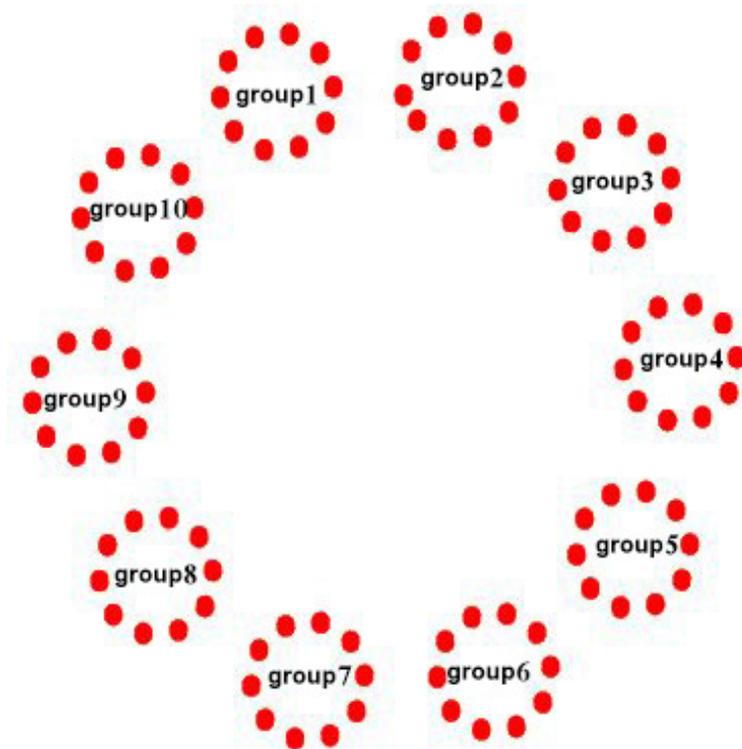


Figure 2: A robot set with ten groups of robots

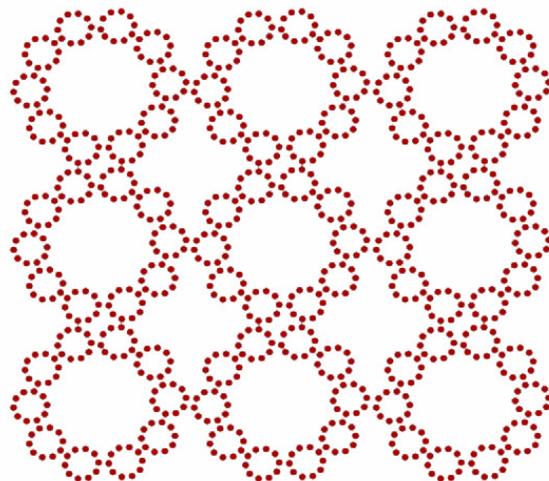


Figure 1: Circular formation for nine hundred robots

The detailed methodology approach is given in Figure 4.

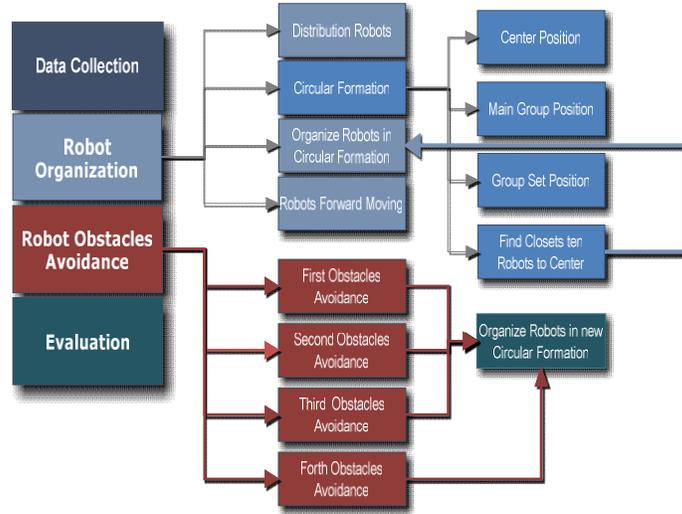


Figure 4: Methodology flowchart

### 3.1. Data collection

One of the major steps in our proposed approach is to have a certain formation for the robots to allow them to move smoothly and to maintain easy connection between them. The robot organization is based on circular formation in which robots are organized in circular groups. To obtain the circular formation, sequences of mathematical formulas have to be followed as shown in the next sections.

### 3.2. Circle circumferences points

Formulas (1) and (2) are used to determine the new points on the circumference of the circle to form the circular formation shown in Figure 5.

$$\text{newpoint}(x', y') \gg x' = \text{Distance} * \cos \frac{\text{degree} * \pi}{180} + x \quad \text{Formula (1)}$$

$$\text{newpoint}(x', y') \gg y' = \text{Distance} * \sin \frac{\text{degree} * \pi}{180} + y \quad \text{Formula (2)}$$

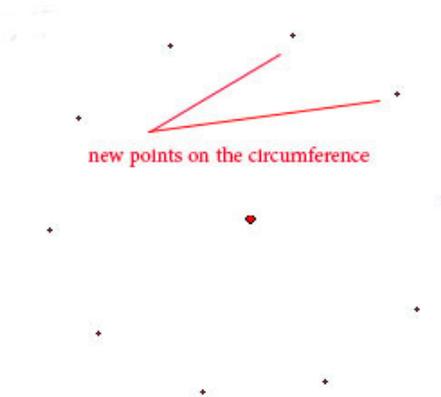


Figure 5: Graph Illustrate the new points on the circumference of the circle

### 3.3. Circle circumference calculation

The pseudo code description of circumference of circle calculation is given in Algorithm 1.

**Algorithm 1: Calculate circumference of circle**  
 Purpose: Calculate the point position on circumference of circle  
 Input: (R,degree, Center.x, Center.y, pn)  
 Output:  $x^n, y^n$   
 Procedure: { // BEGIN  
                     Degree=360/pn  
                     For K=1..pn Do  
                         Degree=degree\*pn  
                          $x^n=R*(\text{Cos}(\text{degree}*\pi/180)+x$ ;  
                          $y^n=R*(\text{sin}(\text{degree}*\pi/180)+y$ ;  
                     } //END

### 3.4. Circle center calculation

The pseudo code description of calculating the center of circle from three points is given in Algorithm 2.

**Algorithm 2: Calculate the center of circle from three points**  
 Purpose: Calculate the center of circle from position of three points  
 Input: (p1x,p1y,p2x,p2y,p3x,p3y) /\*position x,y of three point \*/  
 Output: Center.x, Center.y  
 Procedure: { //BEGIN  
                      $\lambda y1=p2y-p1y$   
                      $\lambda x1=p2x-x1$   
                      $\lambda y2=p3y-p2y$   
                      $\lambda x2=p3x-p2x$   
                      $a=\lambda y1/\lambda x1$   
                      $b=\lambda y2/\lambda x2$   
                     Center.x= (a\*b\*(P1y-P3y)+b\*(P1x-P2x)-a\*(P2x+P3x))/(2\*(a-b)  
                     Center.y=-1\*( Center.x-(P1x+P2x)/2)/a+(P1y+P2y)/2  
                     } // END

### 3.5. Swarm robotics circular formation

The pseudo code description to calculate the Swarm robotics circular formation is given in Algorithm 3 and Figure 6.

**Algorithm 3: Swarm robotics circular formation**

**Purpose:**

Swarm robotics circular formation (code executed by a robot  $r_i$  at the point  $p_i$ )

**Input:**

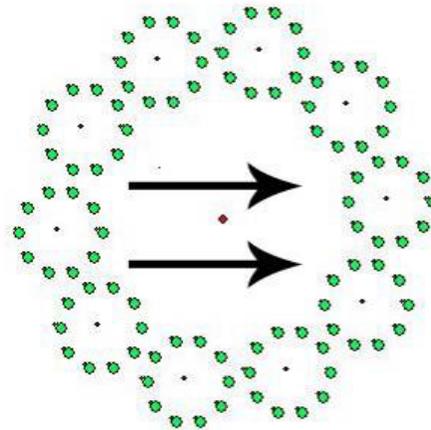
$\{P_1, \dots, P_n\}$  set of position for robots  $\{r_1, \dots, r_n\}$ , Cxy Circular formation center

**Output:**

**Procedure**

```

{
  //BEGIN
  Perceived in the current position for every robots.
  IF  $r_n < 16$  THEN
    FOR  $k_1=1, \dots, \text{GSn}$  DO
       $\text{CP}_n := \text{Calculate New position of } r_n \text{ on circumference of circle } \in \text{Cxy}$ 
       $\vec{M} := \text{move } r_n \text{ from } P_n \text{ to } \text{CP}_n$ 
    END FOR
  ELSE
    Calculate center of group sets on circumference of circle
     $\text{CPG}_n := \text{calculate new position of } r_n \text{ on circumference of group set}$ 
    FOR  $k_2=1, \dots,$  DO
       $\text{CP}_n := \text{Calculate New position of } r_n \text{ on circumference of circle of group set}$ 
    END FOR
    FOR  $k_3=1, \dots, \text{GN}$  DO /*group set number */
      FOR  $k_4=1, \dots, r_n - \text{Cpn}$  DO /* CPn number of robot in every group set*/
        /* calculate min distance between robots n center of group set */
         $\text{Psn} := \min[\text{dist}(P_n, \text{CPG}_n)]$ 
      END FOR
    END FOR
     $\vec{M} := \text{move } r_n \text{ from } P_n \text{ to } \text{CP}_n \in \text{Psn}$ 
  END IF
} // END
  
```



*Figure 6: Swarm Robotics Move Forward*

### 3.6. Group set movement

The pseudo code description to Move Group Set is given in Algorithm 4.

#### Algorithm 4: Move Group Set

**Purpose:** Move Group Set

**Input:** Constant Pd = destination point  
 {p1,p2,p3} position of three robots on circumference of group set circle  
 GSN group set number  
 {P1,...,Pn} set of position for robots on group set {r1,...,rn} ∈ GSN

**Output:** Cg:=cx,cy center of group set

**Procedure:** `//BEGIN`  
     REPEAT  
         Cg(cx,cy) := Centerf3p(p1,p2,p3)  
         /\* Algorithm 3-2 find center of group set from three point\*/  
          $\vec{M}$  =move (cx,cy) to pd  
         FOR k1=1,..., ∈ GSn DO  
             CPn:=Calculate new position of rn on circumference of group set circle ∈ Cg  
              $\vec{M}$  :=move rn from Pn to CPn  
         END FOR  
     UNTIL ((cx,cy)= Pd)  
`// END`

### 3.7. First obstacle crossing

The pseudo code description to move the group set to avoid the predefined first obstacle as shown later in experimentation section. First obstacle crossing is given in Algorithm 5 and Figure 7.

#### Algorithm 5: Move Swarm Robotics Over First Obstacles

**Purpose:** Avoid First Obstacles one robot can pass through obstacles

**Input:** Constant W= Width Of Gap on Obstacles, Oc= Center of Gap begin of Obstacle  
 Oc2= Center of Gap end of Obstacle  
 {p1,p2,p3} position of three robots on circumference of group set circle  
 GSN group set number  
 {P1,...,Pn} set of position for robots on group set {r1,...,rn} ∈ GSN  
 DRH Distance Swarm Robotics Will Stop in Before Obstacles  
 Cxy' New Circular Formation Center After Avoid Obstacles  
 Cg:=cx, cy center of group set

**Procedure:** `//BEGIN`  
     Repeat  
         Cg (cx, cy) := Centerf3p(p1,p2,p3)  
         /\*find center of group set from three point\*/  
          $\vec{M}$  =move (cx, cy) to DRH  
         FOR k1=1,..., ∈ GSn DO  
             CPn:=Calculate New position of rn On Circumference Of group set Circle ∈ Cg  
              $\vec{M}$  :=move rn from Pn to CPn  
         END FOR  
     Until (Oc=DRH) /\*Stop move of Group set \*/  
     n1=1;  
     Repeat  
          $\vec{M}_1$  move rn1 to Oc  
         IF rn1 reach to Oc THEN n1=n1+1

```

 $\overrightarrow{M}_2$  = move  $r_{n1-1}$  to Oc2
Until  $r_{n1}=r_n$ 
IF reach to Oc2 THEN
    Cxy' = DRH2
END IF
Organizing Robots in circular formation
} // END
    
```

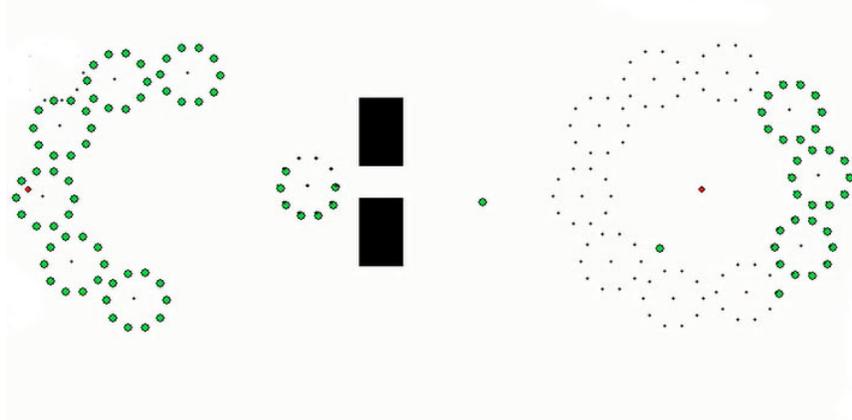


Figure 7. The first obstacle with 100 robots after passing some robots

#### 4. Experimentation and results

We have developed an algorithm that allows a swarm of robots to move forward in a circular formation with a capability of passing obstacles. Therefore, a simulation has been generated to simulate the swarm robotics in its movement and the way it passes the obstacles. All experiments have been carried out on Asus laptop processing power of 2.40 GHz CPU with 16GB RAM. RAD Studio XE7 is used as an IDE for Delphi programming language.

The experiments were based on five different types of obstacles, where each one of them has its own specifications. For every type of obstacles, we implement four experiments using different number of robots as shown in Table 1.

Table 1. Every type of the five obstacles has four different number of robots as experiments

<b>EXP 1</b>	Using 100 Robots
<b>EXP 2</b>	Using 50 Robots
<b>EXP 3</b>	Using 25 Robots
<b>EXP 4</b>	Using 16 Robots

In each experiment, we apply the number of robots on five types of predefined obstacles. Here we show EXP1 (100 robots) with predefined five obstacles. The robots are distributed to 10 robots in each group having ten groups.

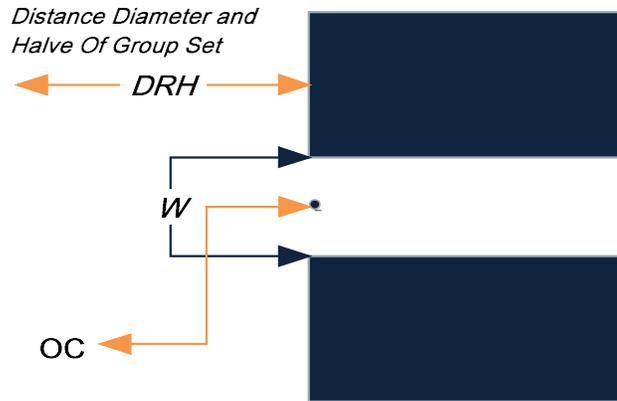


Figure 8 : The first obstacle (allows one robot to pass at a time)

The swarm movement and obstacle avoidance are shown in Figures 8, 9 and 10 for the first obstacle. Figures 11, 12 and 13 are to present the second obstacle and its avoidance. Figures 14, 15, 16 and 17 are to present the third obstacle and its avoidance. Figures 18, 19, 20 and 21 are to present the fourth obstacle and its avoidance. Figures 22, 23, 24 and 25 present the fifth obstacle and its avoidance.

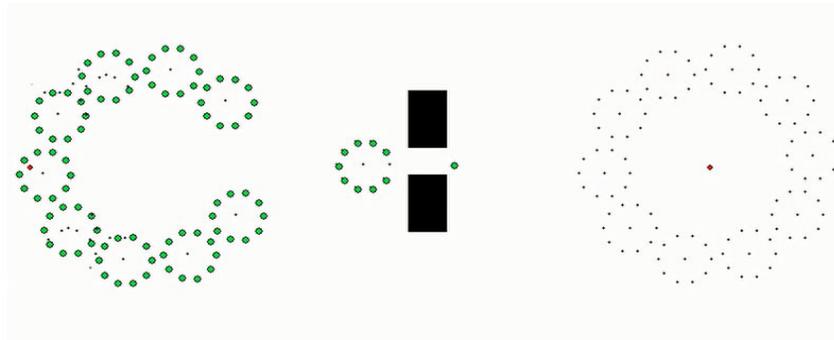


Figure 9: First obstacle with 100 robots before passing any robot

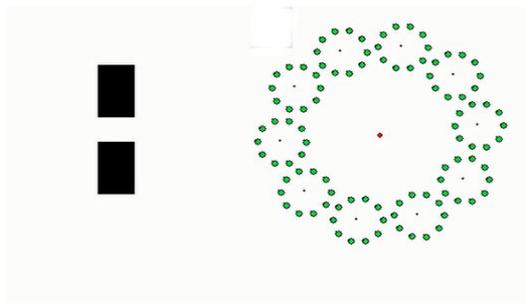


Figure 10: First obstacle with 100 robots after passing all the robots

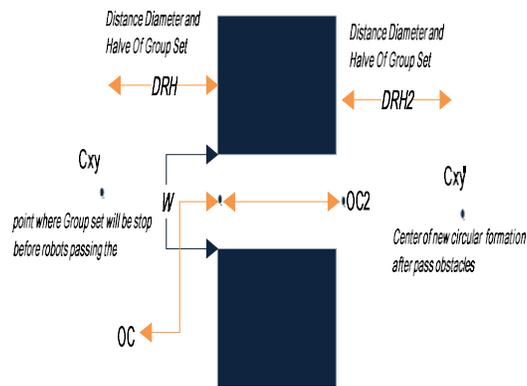


Figure 11: The second obstacle

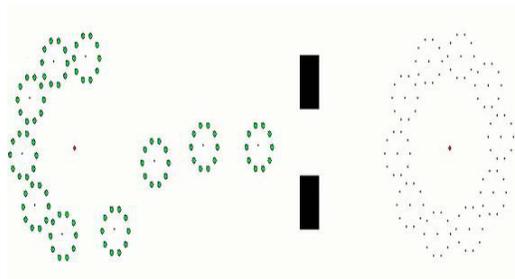


Figure 12: Second obstacle with 100 robots before passing any robot (allows one group set to pass at a time)

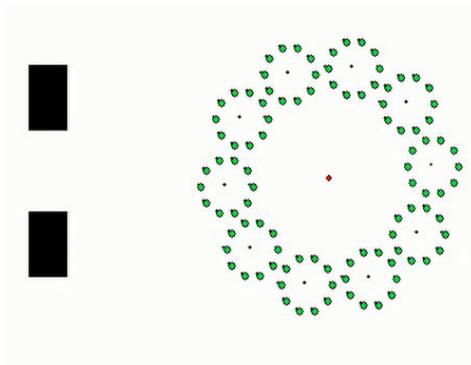


Figure 13: second obstacle with 100 robots after passing all robots

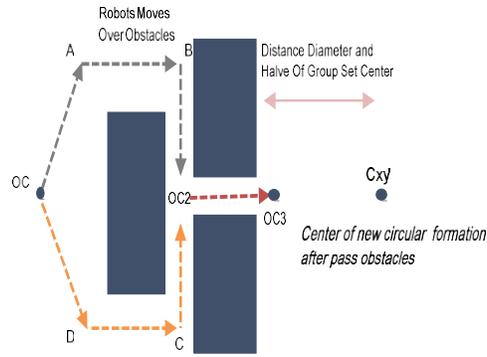


Figure 14: Illustrate the Path of Move for robots on swarm robotics with the third obstacle type (allows one robot to pass at a time with a complex obstacle)

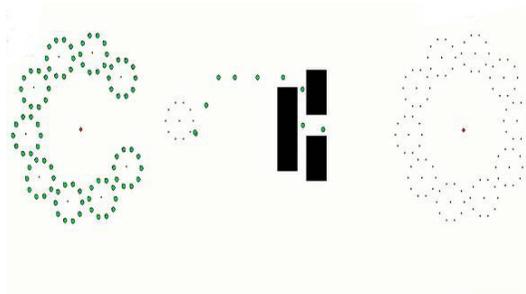


Figure 15: third obstacle with 100 robots before passing any robot

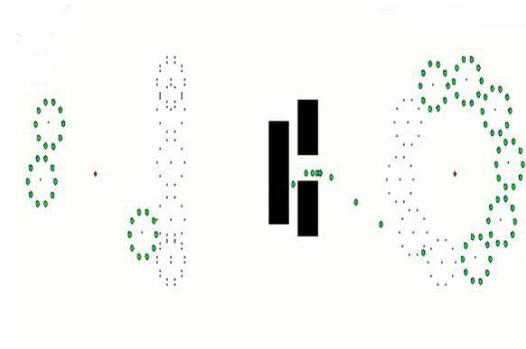


Figure 16: The third obstacle with 100 robots after passing some robots

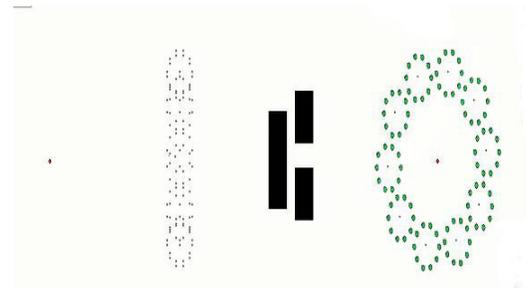


Figure 17: The third obstacle with 100 robots after passing all robots

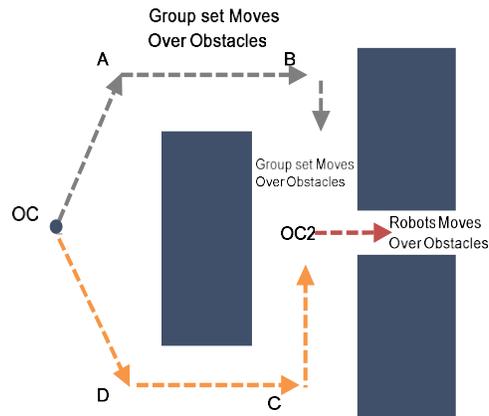


Figure 18: Fourth Obstacle (a part of the obstacle can be passed by a group set but the other part can be passed by only one robot at a time)

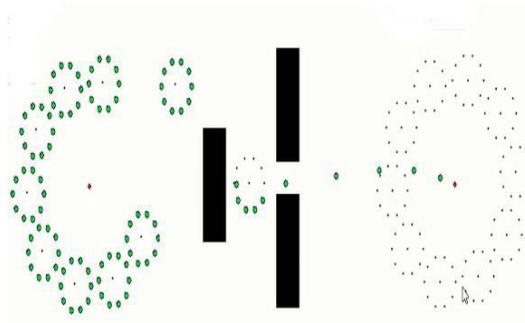


Figure 19: The fourth obstacle with 100 robots before passing any robot

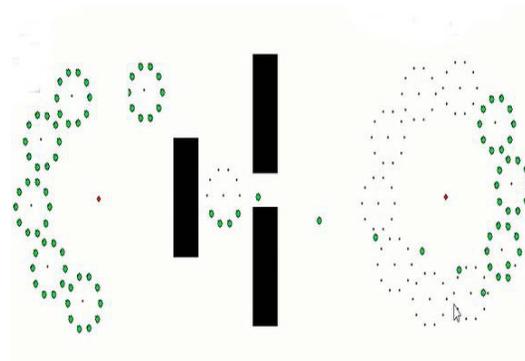


Figure 20: The fourth obstacle with 100 robots after passing some robots

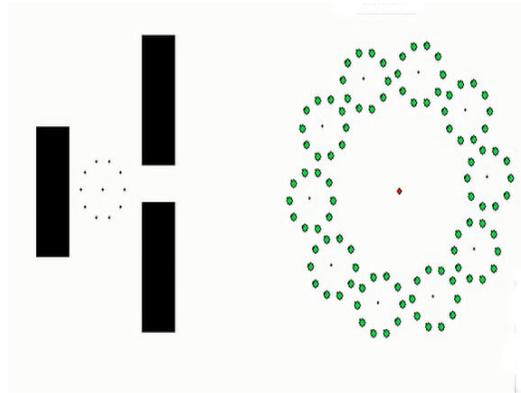


Figure 21: The fourth obstacle with 100 robots after passing all robots

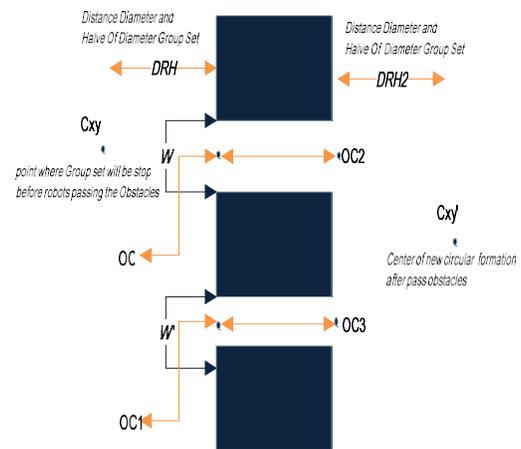


Figure 22: Fifth Obstacle (obstacle with two entries that each allow the passing of one robot)

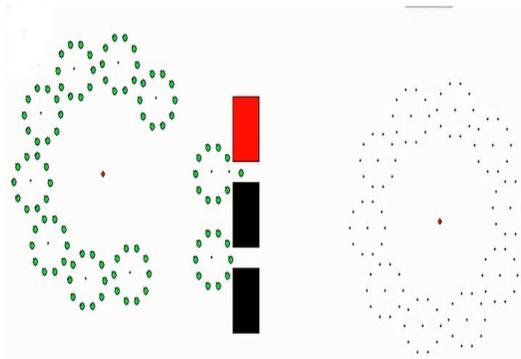


Figure 23: The fifth obstacle with 100 robots before passing any robot

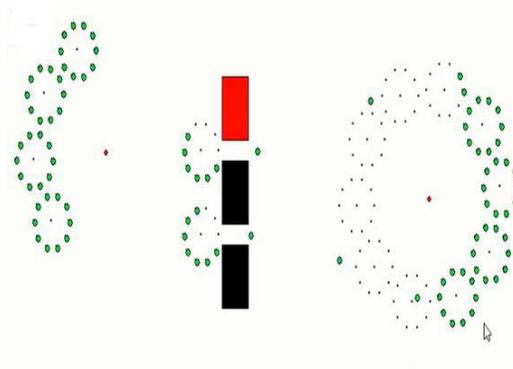


Figure 24: The fifth obstacle with 100 robots after passing some robots

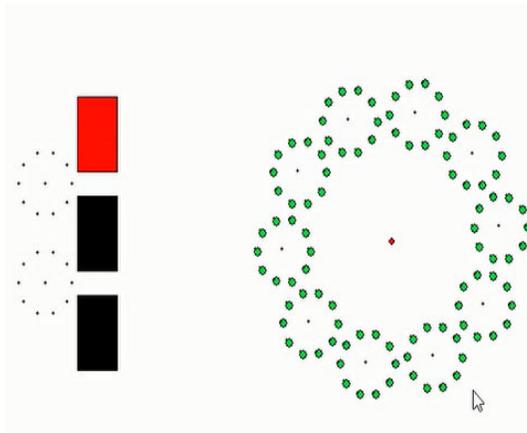


Figure 25: The fifth obstacle with 100 robots after passing all robots

As shown above, with 100 robots, we get 10 group sets, each with 10 robots. For the case of 50 robots, the algorithm will form 10 group sets, each with 5 robots. For the case of 25 robots, the algorithm will form 6 group sets, where 5 of them will have 4 robots each, and the last group set will have 5 robots. For the case of 16 robots, the algorithm will form 4 group sets, each with 4 robots. The algorithm behaves like this because the minimum number of robots to form a group set is 4 robots and the maximum number of robots is 10. As stated before if the number of robots exceeds 100 robots another circular formation is created.

#### 4.1. Discussion

Comparing the circular formation algorithm with the flocking algorithm (Lee & Chong, 2008), the circular formation algorithm performs the process of organizing the swarm only once, and does not need to repeat the process and computations related to it. While the flocking algorithm repeats the processes of forming and reorganizing the swarm so it needs more computations.

Regarding the obstacle crossing, the circular formation is characterized by the possibility of separation for each group where it can move independently to the obstacle so it does not require complex computational operations, but the flocking algorithm performs full adaptation for the swarm and thus it needs computations for each robot.

In circular formation, the swarm reorganization (post obstacle organization) does not need computations after crossing the obstacle where its groups automatically organized and identified.

The only change will be the coordinates for the relocation of these groups. With flocking algorithm, it is forced to fully reorganize the swarm after crossing the obstacle.

The possibility of separating the group sets from the swarm enables the circular formation to implement small tasks efficiently while maintaining the rest of the swarm away from danger. Also, it is more efficient in searching tasks to find foods or even wounded people.

The flocking algorithm could implement tasks that need a large number of robots working at the same time and tasks related to water environment due to the nature of the swarm which simulates the fish swarm, but the flocking algorithm uses the whole swarm to accomplish the tasks, putting the entire swarm at risk when dealing with dangerous environment. Table 2 summarizes the difference between circular formation algorithm and flocking algorithm.

Table 2. The difference between Circular formation algorithm and flocking algorithm.

Point	Circular formation algorithm	Flocking algorithm
<b>Initial Organization</b>	<ul style="list-style-type: none"> <li>- It distributes the robots randomly.</li> <li>- Heuristic way is used to organize the robots in a circular formation.</li> </ul>	<ul style="list-style-type: none"> <li>- It distributes the robots randomly.</li> <li>- "local active interaction" algorithm is used to organize the robots by making triangular formation.</li> </ul>
Movement	After organizing the robots in a circular formation, the swarm robotics start moving forward for all robots in the swarm robotics in the same speed.	After organizing the swarm robotics in triangular network, the robots start moving forward in the same speed.
<b>Obstacles Avoidance</b>	In this algorithm, we will define the obstacles and the points of the path that the robots will move through as an obstacle, if the obstacles allow the group set of the swarm robotics to pass through, then the group set will separate from the swarm robotics to avoid the obstacles alone. if the obstacles allow just one robot pass through it then the group set will again split into just one robot to move through the obstacle.	In this algorithm, the obstacles will be defined then an adaptation for robots in swarm robotics is going to be split into multiple smaller swarms to let every new swarm to move through the obstacles.
<b>Post Obstacle Organization</b>	After avoiding the obstacles, a center of the circular formation is created to be adjacent to the center of obstacles far away from the obstacles as a diameter of the circular formation and a half, to let the robots organize in a new circular formation without calculating the new position, because every robot will move to its previous known position.	After avoiding the obstacles, the swarms will merge in a single swarm by calculating the distance between the nearest robots from every swarm by using sensing boundary between robots and calculating the smallest distance between other robots. In this step, it will repeat the local interaction algorithm to recalculate the distances and to make a new triangular network.
Drawbacks	<ul style="list-style-type: none"> <li>- The initial organization of the swarm takes a considerable time.</li> <li>- Determining the points to define the swarm path through the obstacle takes a considerable time.</li> </ul>	<ul style="list-style-type: none"> <li>- The initial organization of the swarm takes a considerable time.</li> <li>- Also, pre obstacle swarm splitting takes a considerable time.</li> <li>- Post obstacle splitting takes a considerable time.</li> <li>- High processing is needed to maintain the network of triangles in the transit called "Adapting" during the crossing.</li> <li>- Post obstacle formed triangles may contain different robots that were in the same triangle form, which means inconsistency if every robot of a triangle of robots have different duty.</li> </ul>

## 5. Conclusion

Our study focused on designing and developing an algorithm that guides and controls the swarm robotics motion including obstacle avoidance based on circular formation to allow robots to move, organize and reorganize themselves in a systematic way and to reduce the reorganizing time after avoiding the obstacles. The swarm of robots contains a group set that might face any of five defined different obstacles.

A simulation process has been performed for the algorithm of circular formation to show the behavior of the swarm of robots during motion and obstacle avoidance.

The algorithm deals with a minimum of four robots to consider them as a swarm, if the number is less than four, the proposed algorithm does not work as it is and a certain amendment is needed. If the number of robots exceeds 100, a new additional circular formation will be formed.

Twenty experiments have been conducted with five different types of obstacles. Each type of obstacles has four experiments with 100, 50, 25 and 16 robots. These numbers cover the majority of the possibilities for the robots' distribution in the circular formation. Table 1 shows the main points used to differentiate between circular formation and flocking algorithm behaviors.

The behavior of the swarm forming circular formation is considered by gathering the robots from their random distribution into a circular formation and moving towards the obstacles; and passing them. Based on the nature and the shape of the obstacle, one robot might pass at a time, two robots might pass at a time, a group set might pass at a time, or could be any number of robots less than a group set.

The obtained results have been generally compared with flocking algorithm. The circular formation movement of a swarm of robots using a circular configuration is better in terms of movement and organization. Regarding the obstacle crossing, the circular formation is characterized by the possibility of separation for each group where it can move independently to the obstacle so it does not require complex computational operations, but the flocking algorithm performs full adaptation for the swarm and thus it needs computations for each robot. One more advantage for circular formation over flocking algorithm is that each robot before passing the obstacle will be in the same group after passing the obstacle. This will be good for two reasons; the first one is what we can call group harmony and the second one is the robot specific duty within the group.

Some of the future directions are:

- 1) Simulate the circular formation algorithm in 3-D environment;
- 2) Consider more types of obstacles;
- 3) Considering various jobs for each group set;
- 4) Considering different job for each robot;
- 5) Considering different sizes and capabilities of the robots in the swarm;
- 6) Considering various jobs for swarm robotics in space

## References

- Cheng, K., Y. Wang, and P. Dasgupta (2009). *Distributed area coverage using robot flocks*, in Nature & Biologically Inspired Computing, NaBIC 2009.
- Défago, X. and A. Konagaya (2002). *Circle formation for oblivious anonymous mobile robots with no common sense of orientation*, in Proceedings of the second ACM international workshop on Principles of mobile computing.
- Ducatelle, F., Di Caro, G. A., Pinciroli, C., Mondada, F., & Gambardella, L. (2011). *Communication assisted navigation in robotic swarms: self-organization and cooperation*, in proceedings of international conference on Intelligent Robots and Systems (IROS).
- Folino, G. and G. Spezzano (2002). *An adaptive flocking algorithm for spatial clustering*, in *Parallel Problem Solving from Nature—PPSN VII. 2002*, Springer. p. 924-933.

- Freitas, R.A. (2005). *Current status of nanomedicine and medical nanorobotics*. Journal of Computational and Theoretical Nanoscience, 2(1): p. 1-25.
- Geunho, L. and N.Y. Chong (2008). *Adaptive flocking of robot swarms: Algorithms and properties*. IEICE transactions on Communications, 91(9): p. 2848-2855.
- Guo, H., Y. Meng, and Y. Jin. (2010) *Analysis of local communication load in shape formation of a distributed morphogenetic swarm robotic system*, in Evolutionary Computation (CEC), IEEE Congress.
- Hanada, Y., G. Lee, and N.Y. Chong (2007). *Adaptive flocking of a swarm of robots based on local interactions*, in Swarm Intelligence Symposium.
- Hereford, J. and C. Blum (2011). *FlockOpt: A new swarm optimization algorithm based on collective behavior of starling birds*, in Nature and Biologically Inspired Computing (NaBIC).
- La, H.M. and W. Sheng (2009). *Adaptive flocking control for dynamic target tracking in mobile sensor networks*, in Intelligent Robots and Systems, IROS 2009.
- Lee, G. and N.Y. Chong (2008). *Flocking controls for swarms of mobile robots inspired by fish schools*, INTECH Open Access Publisher.
- Lehr, H., Abel, S., Doepper, J., Ehrfeld, W., Hagemann, B., Kaemper, K. P., . . . Theurigen, C. (1996). *Microactuators as driving units for microbotic systems*. Paper presented at the Photonics East'96.
- McLurkin, J. (1995). *The ants: a community of microrobots*. Massachusetts Institute of Technology. Bachelor's Thesis.
- Meng, Y. and J. Gan (2008). *Self-adaptive distributed multi-task allocation in a multi-robot system*, in Evolutionary Computation. CEC 2008 (IEEE World Congress on Computational Intelligence).
- Nouyan, S. and M. Dorigo (2004), *Chain formation in a swarm of robots*. IRIDIA, Université Libre de Bruxelles, Tech. Rep. TR/IRIDIA/2004-18.
- Reynolds, C.W. (1987). *Flocks, herds and schools: A distributed behavioral model*, in ACM Siggraph Computer Graphics.
- Rubenstein, M., C. Ahler, and R. Nagpal (2012). *Kilobot: A low cost scalable robot system for collective behaviors*, in Robotics and Automation (ICRA).
- Scanniello, G. and U. Erra (2013). *Software entities as bird flocks and fish schools*, in Software Visualization (VISSOFT).
- Stormont, D.P. and A. Kutiyawala (2007). *Localization using triangulation in swarms of autonomous rescue robots*, in Safety, Security and Rescue Robotics.
- Turgut, A. E., Çelikkanat, H., Gökçe, F., & Şahin, E. (2008). *Self-organized flocking in mobile robot swarms*. Swarm Intelligence, 2(2-4), 97-120.
- Xiaoping, Y., Gokhan, A., & Okay, A. (1997). *Line and circle formation of distributed physical smobile robots*, Journal of Robotic Systems, 14(2), 63-76.