micromachines

MDPI

Article

The Multitasking System of Swarm Robot based on Null-Space-Behavioral Control Combined with Fuzzy Logic

Nga Le Thi Thuy [1] and Thang Nguyen Trong [2,*]

[1] Department of Cybernetics, University of Transport and Communication, Hanoi 122000, Vietnam; lethuynga@utc.edu.vn

[2] Department of Electrical Engineering and Automation, Haiphong Private University, Haiphong 181810, Vietnam

* Correspondence: thangnt@hpu.edu.vn; Tel.: +84-168-846-8555

Received: 26 November 2017; Accepted: 7 December 2017; Published: 9 December 2017

Abstract: A swarm robot is a collection of large numbers of simple robots used to perform complex tasks that a single robot cannot perform or only perform ineffectively. The swarm robot works successfully only when the cooperation mechanism among individual robots is satisfied. The cooperation mechanism studied in this article ensures the formation and the distance between each pair of individual robots while moving to their destination while avoiding obstacles. The solved problems in this article include; controlling the suction/thrust force between each pair of individual robots in the swarm based on the fuzzy logic structure of the Singer-Input-Singer-Output under Mamdani law; demonstrating the stability of the system based on the Lyapunov theory; and applying control to the multitasking system of the swarm robot based on Null-Space-Behavioral control. Finally, the simulation results make certain that all the individual robots assemble after moving and avoid obstacles.

Keywords: swarm robot; Lyapunov theory; fuzzy control; multitasking

1. Introduction

A flock is a gathering of a group of creatures, found in nature of many different species such as insect pests, ants, bees, termites, fish, etc. Flocking behavior can accomplish tasks that surpass the ability of the individual. So, researchers have modeled flocking activity, inheriting these advantages to apply to multi-robots. By the end of the 1980s, scientists had researched and built robotic teams with the capability of working and coordinating to perform a specific task [1].

The characteristics of the swarm robot are intellectual without requiring the complex manufacturing technology in the robot field [2,3]. The swarm robot can be used in many fields such as search [4,5], cleaning [6], and transportation [7,8]. So, swarm robots are increasingly attracting the interest of scientists around the world. A number of successful researches related to flock robots has brought swarm robots to more and more widespread applications in life, such as multi-robot system main principles [9–16], swarm robotics [17–20], human-multi-robot interaction [21,22], problem-specific works [23–26], and autonomous underwater vehicles [19,27].

In order that the robot individuals of the swarm can work together to perform a certain task, the first and most important problem that must be addressed is ensuring a collaborative mechanism among robots. When an individual robot performs a task, it must avoid obstacles, but it is not separated or cannot collide with others [28,29]. So, before researching the specialized applications of the swarm robot, the first problem that needs to be solved is ensuring the swarm and the distance among individuals while moving and avoiding obstacles.

Some researchers have solved the above problem as in references [30–33]. All of these studies are on the interaction between each pair of individuals and between the individual and the environment. The interactions are expressed by the suction/thrust force among individuals, usually explicit mathematical functions, but there is no convincing explanation. In reality, the working environment of the swarm robot is quite complex, frequently changing, and each robot consists of a collection of many details such as the engine, power circuit, control circuit, etc. so it is difficult to determine an explicit mathematical model, or even impossible to identify.

In order for the model system to get closer to nature, this article proposes a solution which uses fuzzy logic for controlling the suction/thrust force between each pair of individuals in the swarm. The advantage of this solution is that the object can be controlled easily without the requirement of knowing the object mathematical equation. Then, the authors applied this method to control the multitasking system of a swarm robot.

Multitasking in a swarm robot means that each individual robot in the swarm must perform multiple tasks at the same time; research [34] has also introduced two basic mechanisms to solve the multitasking problem as follows:

The first is an arbitration mechanism: the controller only obeys the output commands of the higher-priority behavior. The lower-priority behavior can only be performed if the output of the higher-priority behavior is equal to zero. The drawback of this mechanism is that when the behaviors are not inconsistent with each other they still cannot be done at the same time.

The second is the integration mechanism: The command is created based on a combination of several behaviors. The drawback of this mechanism is that when the behaviors are inconsistent with each other, they cannot be done at the same time.

Thus, the arbitration mechanism (competition) only allows one task at the same time, so this mechanism is rarely used. The integration mechanism allows for combining several tasks to achieve a different mission, but it is difficult to perform conflicting tasks. Research [35] has introduced a new method to solve the above limitations, which is the null space behavioral (NSB) control method. Based on this method, the complex task of the swarm robot can be divided into different basic tasks (behaviors), these tasks are properly combined to achieve the objective mission. The main steps of this method are determining the priority assignment of each basic task, then projection of the lower-priority tasks on the null-space of the higher-priority task.

So, the current authors combine the two best-fit methods to control the swarm robot: the first is using fuzzy logic to control the suction/thrust force among the individual robots. The advantage of this method is that the object can be controlled easily without the requirement of knowing the mathematical equation; the second is using the null-space behavioral control method for the multitasking system of swarm robot. The advantage of this method is that it allows the robot to perform many tasks easily at the same time.

2. Building the Function of the Suction/Thrust Ford Based on Fuzzy Logic

Considering a set of N individuals in n-dimensional Euclidean space ($n \leq 3$). Assume that each individual is a point and ignore their size and mass, the position of the individual (i) in the swarm is $p^i = \begin{bmatrix} p_1^i \\ p_2^i \\ \vdots \\ p_n^i \end{bmatrix} \in R^n$. The movement of individual robots in a homogeneous environment will depend on the interaction between the individual and all other individuals. The homogeneous environment is an environment where there is no obstruction, no external disturbance affecting the swarm. If pairs of individuals are far apart, they need to move toward each other by suction force in order to maintain the swarm. Conversely, if pairs of individuals are near together, they need to move away from each other by thrust force to avoid collisions. So, the interplay among individuals in the swarm will depend on the distance between the pairs of individuals. The interaction force between two individuals (i and j) is defined as follows:

$$f = f\left(\|p^j - p^i\|\right)$$

where: $\|p^j - p^i\|$ is the distance between two individuals i and j.

The actual distance between two robots (i and j):

$$\sigma_s = \|p^j - p^i\| = \sqrt{\left(p_1^j - p_1^i\right)^2 + \left(p_2^j - p_2^i\right)^2 + \cdots + \left(p_n^j - p_n^i\right)^2}$$

Named:

$$g\left(\|p^j - p^i\|\right) = \frac{f\left(\|p^j - p^i\|\right)}{\|p^j - p^i\|} = \frac{f(\sigma_s)}{\sigma_s} = g_a(\sigma_s) - g_r(\sigma_s) \tag{1}$$

where: $g(\cdot)$ is the suction/thrust force between two individuals (i, j), $g_a(\cdot)$ is the thrust force, $g_r(\cdot)$ is the suction force. $\sigma_s^* \in R$ is the distance between two individuals i and j, where the suction force and thrust between the two individuals are in equilibrium, which means:

$$(\sigma_s)\begin{cases} = 0 & \text{if } \sigma_s = \sigma_s^* \\ < 0 & \text{if } 0 < \sigma_s < \sigma_s^* \\ > 0 & \text{if } 0 < \sigma_s^* < \sigma_s \end{cases}$$

$\widetilde{\sigma}_s$ is the error between the real distance and the desired distance:

$$\widetilde{\sigma}_s = \sigma_s - \sigma_s^*$$

The interaction force $f(\sigma_s)$ among individuals in the swarm is a nonlinear function that depends on the distance between each pair of individuals (i, j). So, we can construct the function $f(\sigma_s)$ based on a Mamdani fuzzy system with the Singer-Input-Singer-Output (SISO) structure as follows:

- First step

  ➢ The input signal is $u = \widetilde{\sigma}_s = \sigma_s - \sigma_s^*$, assume that the value domain of u is $[\alpha_b, \beta_b] \in R$, divide this domain into 2Nf + 1 in the range B^k as shown in Figure 1.

  ➢ The output signal is $A = f(\sigma_s - \sigma_s^*)$ with the value domain $[\alpha_a, \beta_a]$, divide this value domain into $2N_f + 1$ in the range A^k as shown in Figure 2 (k = 1, 2, . . . , $2N_f + 1$). a^k is the focus of the fuzzy range A^k:

$$a^k\begin{cases} < 0 & \text{if } k = 1,\ 2, \ldots N_f \\ = 0 & \text{if } k = N_f + 1 \\ > 0 & \text{if } k = N_f + 2, \ldots,\ 2N_f + 1 \end{cases} \tag{2}$$

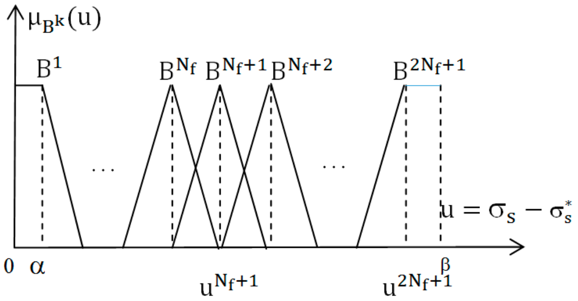


Figure 1. The membership function of the input.

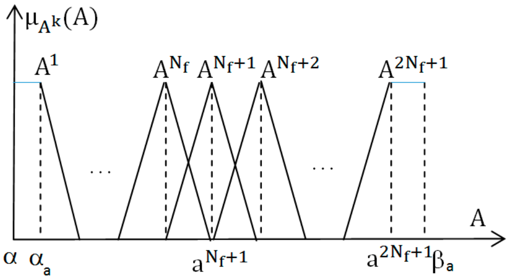


Figure 2. The membership function of the output.

- The second step: establishing 2Nf + 1 rule IF-THEN with the form: IF: $u = B^k$, THEN: $A = A^k$
- The third step: defuzzifier using the central area method, we have control laws as follows [36]:

$$f(u) = \frac{\sum_{k=1}^{2N_f+1} a^k \mu_{B^k}(u)}{\sum_{k=1}^{2N_f+1} \mu_{B^k}(u)} \tag{3}$$

With the solution to design the fuzzy control through the above three steps, the relationship between the input signal and the output signal is the relationship between the distance and the interaction force between individuals (i, j):

$$\begin{cases} f(\sigma_s) > 0, & if\ \sigma_s > \sigma_s^* \\ f(\sigma_s) < 0, & if\ \ 0 < \sigma_s < \sigma_s^* \\ f(\sigma_s) = 0, & if\ \sigma_s = \sigma_s^* \end{cases} \tag{4}$$

The fuzzy function $f(\sigma_s)$ is a continuous function which satisfies the following conditions:

- Upper and lower limits:

$$A_{min} \le f(\sigma_s) \le A_{max} \tag{5}$$

where: $A_{min} = a^1, \ \ A_{max} = a^{2N_f\,+\,1}$

- The equation of a part linearization:

$$f(\sigma_s) = \frac{(a^{k\,+\,1} - a^k)u + a^k u^{k\,+\,1} - a^{k\,+\,1}u^k}{u^{k\,+\,1} - u^k} \tag{6}$$

where: $u \in \left[u^k, \ \ u^{k\,+\,1}\right], k \in \{1, 2, \ldots, 2N_f\}$

G_{amin}, G_{amax} is the smallest and largest value of the suction force, G_{rmin}, G_{rmax} is the smallest and largest value of the thrust. From (6) we can find the limits of the function $g(\sigma_s)$ as follows:

$$0 \le G_{amin} \le g(\sigma_s) \le G_{amax}, \text{ if } \sigma_s > \sigma_s^*$$

$$-G_{rmin} \le g(\sigma_s) \le G_{rmax} < 0, \text{ if } 0 < \sigma_s < \sigma_s^* \tag{7}$$

where

$$G_{amax} = \max_{N_f+2 \le k \le 2N_f+1} \left[\frac{a^{k+1} - a^k}{u^{k+1} - u^k}\right]$$

$$G_{amin} = \min_{N_f+2 \le k \le 2N_f+1} \left[\frac{a^{k+1} - a^k}{u^{k+1} - u^k}\right]$$

$$G_{rmax} = \max_{1 \le k \le N_f} \left[\frac{a^{k+1} - a^k}{u^{k+1} - u^k}\right]$$

$$G_{rmin} = \min_{1 \le k \le N_f} \left[\frac{a^{k+1} - a^k}{u^{k+1} - u^k}\right]$$

3. The Stability of the System

Assume that the individuals move in sync and have no delay, all individuals in the swarm know exactly the relative position of all other individuals, the dynamic Equation (1) can be rewritten as follows:

$$\dot{p}^i = \sum_{j=1,j\neq i}^{N} f\left(\|p^j - p^i\|\right)\frac{(p^j - p^i)}{\|p^j - p^i\|} = \sum_{j=1,j\neq i}^{N} g\left(\|p^j - p^i\|\right)\left(p^j - p^i\right) \tag{8}$$

where: $\frac{(p^j-p^i)}{\|p^j-p^i\|}$ displays the direction of the force from the individual (i) to the individual (j), $f\left(\|p^j - p^i\|\right)$ is the interaction force depending on the distance between the pair of individuals (i, j).

If $g(\cdot) > 0$, this interaction is the suction force, if $g(\cdot) < 0$, this interaction is the thrust.

The center of the swarm is defined by the following formula:

$$p^c = \frac{1}{N}\sum_{i=1}^{N} p^i \tag{9}$$

The derivational of the center p^c:

$$\begin{aligned}\dot{p}^c &= \tfrac{1}{N}\sum_{i=1}^{N}\sum_{j=1,\, j\neq i}^{N} g(\|p^j - p^i\|)(p^j - p^i) \\ &= \tfrac{1}{N}\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\left[g(\|p^j - p^i\|)(p^j - p^i) + g(\|p^i - p^j\|)(p^i - p^j)\right] = 0\end{aligned} \tag{10}$$

The Equation (10) shows that the center of the swarm robot described by the Equation (8) where the suction/thrust force as Equation (3) is invariant.

The different position between individual (i) and the center is as follows:

$$e^i = p^i - p^c \ (i = 1,\ 2, \ldots, N)$$

The derivational of e^i:

$$\dot{e}^i = \dot{p}^i - \dot{p}^c = \dot{p}^i$$

Select the Lyapunov function for the individual robot (i):

$$V_i = \frac{1}{2}\|e^i\|^2 = \frac{1}{2}e^{iT}e^i$$

The derivational of V_i:

$$\dot{V}_i = \dot{e}^{iT}e^i = \dot{p}^{iT}e^i = \sum_{j=1}^{N} g\left(\|p^j - p^i\|\right)\left(p^j - p^i\right)^T e^i \tag{11}$$

The sum of Lyapunov functions of all individuals:

$$V = \sum_{i=1}^{N} V_i = \frac{1}{2}\sum_{i=1}^{N} e^{iT}e^i \tag{12}$$

The derivational of V:

$$\begin{aligned}\dot{V} &= \sum_{i=1}^{N}\sum_{j=1}^{N} g(\|p^j - p^i\|)(p^j - p^i)^T e^i \\ &= \sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\left[g(\|p^j - p^i\|)(p^j - p^i)^T e^i + g(\|p^i - p^j\|)(p^i - p^j)^T e^j\right]\end{aligned} \tag{13}$$

where: $p^j - p^i = (p^j - p^c) - (p^i - p^c) = e^j - e^i$.

Named: $e = e^j - e^i$

$$\begin{gathered} g\left(\left\|p^j - p^i\right\|\right)\left(p^j - p^i\right)^T e^i + g\left(\left\|p^i - p^j\right\|\right)\left(p^i - p^j\right)^T e^j \\ = g\left(\left\|p^j - p^i\right\|\right)\left[\left(p^j - p^i\right)^T e^i + \left(p^i - p^j\right)^T e^j\right] \\ = g\left(\left\|p^j - p^i\right\|\right)\left(p^i - p^j\right)^T\left(e^j - e^i\right) = -g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 \end{gathered}$$

So:

$$\begin{aligned} \dot{V} &= -\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 \\ &= -\tfrac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 \end{aligned} \tag{14}$$

Named: $S_1 = \left\{(i,\ j):\ \left\|p^j - p^i\right\| > \sigma_s^*\right\}$; $S_2 = \left\{(i,\ j):\ \left\|p^j - p^i\right\| < \sigma_s^*\right\}$

$$\sum_{S_1} = \sum_{i=1}^{N}\sum_{j=1}^{N},\quad (i,\ j) \in S_1;\ \sum_{S_2} = \sum_{i=1}^{N}\sum_{j=1}^{N},\quad (i,\ j) \in S_2$$

Equation (14) can be rewritten as follows:

$$\begin{gathered} \dot{V} = -\tfrac{1}{2}\sum_{S_1} g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 - \tfrac{1}{2}\sum_{S_2} g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 \\ = -\tfrac{1}{2}\left[\sum_{S_1} g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 + \sum_{S_2} -g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|\left\|p^j - p^i\right\|^2\right] \\ -\tfrac{1}{2}\left[\sum_{S_2} g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 - \sum_{S_2} -g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|\left\|p^j - p^i\right\|^2\right] \\ = -\tfrac{1}{2}\left[\sum_{S_1} g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 + \sum_{S_2} -f\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2\right] \\ -\tfrac{1}{2}\left[\sum_{S_2} g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 + \sum_{S_2} f\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2\right] \end{gathered} \tag{15}$$

From Condition (5), we infer:

$$\begin{gathered} \sum_{S_2} -f\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 \le A_{min}\sum_{S_2}\left\|p^j - p^i\right\|^2; \\ \sum_{S_1} g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 \ge G_{amin}\sum_{S_1}\left\|p^j - p^i\right\|^2 \end{gathered}$$

Considering the second component of Equation (15):

$$\begin{gathered} \sum_{S_2} g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 - \sum_{S_2} -f\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 \\ \ge \sum_{S_2} g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 - \sum_{S_2} A_{min}\left\|p^j - p^i\right\|^2 \end{gathered} \tag{16}$$

The left side of the Inequality (16):

$$\begin{gathered} \sum_{S_2} g\left(\left\|p^j - p^i\right\|\right)\left\|p^j - p^i\right\|^2 - \sum_{S_2} A_{min}\left\|p^j - p^i\right\|^2 \\ = \sum_{S_2}\frac{f\left(\left\|p^j - p^i\right\|\right) - \underline{A}\left\|p^j - p^i\right\|}{\left\|p^j - p^i\right\|}\left\|p^j - p^i\right\|^2 \end{gathered}$$

Named: $f_{S_2} = -f\left(\left\|p^j - p^i\right\|\right) + A_{min}\left\|p^j - p^i\right\|$
Set β is the largest value of f_{S_2} in the domain S_2. The graph of total f_{S_2} is shown in Figure 3.
Named $\beta = A_{min}\sigma_s^*$. So:

$$-\sum_{S_2}\frac{-f\left(\left\|p^j - p^i\right\|\right) + \underline{A}\left\|p^j - p^i\right\|}{\left\|p^j - p^i\right\|}\left\|p^j - p^i\right\|^2 \le \beta\sum_{S_2}\left\|p^j - p^i\right\|$$

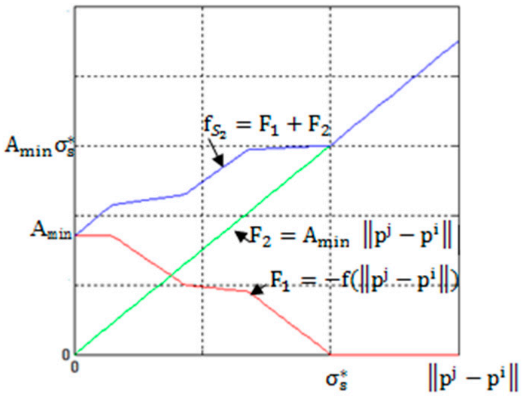


Figure 3. The graph of total f_{S_2}.

Thus, the Inequality (16) is equivalent to:

$$\sum_{S_2} g\left(\|p^j - p^i\|\right)\|p^j - p^i\|^2 - \sum_{S_2} -f\left(\|p^j - p^i\|\right)\|p^j - p^i\|^2 \geq -A_{min}\sigma_s^* \sum_{S_2}\|p^j - p^i\| \tag{17}$$

The first component of Equation (15):

$$\begin{aligned} &\sum_{S_1} g(\|p^j - p^i\|)\|p^j - p^i\|^2 + \sum_{S_2} -f(\|p^j - p^i\|)\|p^j - p^i\|^2 \\ &= \sum_{S_1} g(\|p^j - p^i\|)\|p^j - p^i\|^2 + \sum_{S_2} A_{min}\|p^j - p^i\|^2 \end{aligned}$$

Named $\alpha = \min\{G_{amin}, A_{min}\}$, we have:

$$\sum_{S_1} g\left(\|p^j - p^i\|\right)\|p^j - p^i\|^2 + \sum_{S_2} -f\left(\|p^j - p^i\|\right)\|p^j - p^i\|^2 \geq \alpha \sum_{S_1 \cup S_2}\|p^j - p^i\|^2 \tag{18}$$

Combination of Inequalities (16) and (18), with $\|p^j - p^i\| \leq \sigma_s^*$, $(i, j) \in S_2$, we have:

$$\dot{V} \leq -\alpha \sum_{i=1}^{N}\sum_{j=1}^{N}\|p^j - p^i\|^2 + A_{min}\sigma_s^{*2} \tag{19}$$

From the definition of the swarm robot center, we have:

$$\sum_{j=1}^{N} p^j = Np^c \tag{20}$$

Two sides of (20) minus Ne^i, we have:

$$\sum_{j=1}^{N}\left(p^i - p^j\right) = N\left(p^i - p^c\right)$$

Thus, the sum of the squared deviations is given by the formula:

$$\begin{aligned} \sum_{i=1}^{N}\|e^i\|^2 &= \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}(p^i - p^j)^T e^i = \frac{1}{N}\sum_{i=1}^{N-1}\sum_{j=1}^{N}\|p^j - p^i\|^2 \\ &= \frac{1}{2N}\sum_{i=1}^{N}\sum_{j=1}^{N}\|p^j - p^i\|^2 \end{aligned} \tag{21}$$

Combining Equations (17) and (21) we have:

$$\dot{V} \leq -2N\alpha\sum_{i=1}^{N}\|e^i\|^2 + A_{min}\sigma_s^{*2}$$

$\dot{V} < 0$ when $\sum_{i=1}^{N}\|e^i\|^2 > \frac{A_{min}\sigma_s^{*2}}{2N\alpha}$. So, we can conclude with the following theorem:

Theorem: *Swarm robots are modeled by Equation (8) with the fuzzy control law of suction/thrust force constructed according to (3), satisfying Conditions (4). After a period, all individuals of the swarm will be converged in a restricted area by:*

$$\Omega_\sigma = \left\{ \sum \|p^j - p^i\|^2 \le \sigma^2 \right\} \tag{22}$$

where $\sigma = \sqrt{\frac{A_{min}\sigma_s^{*2}}{2N\alpha}} = \alpha_s^* \sqrt{\frac{A_{min}}{2N\alpha}}$ is the convergent radius of the swarm.

The effect of the parameters on the restricted area of the swarm (Ω_σ) is as follows:

- If A_{min} increases, the thrust increases, the restricted area of the swarm robot increases.
- If α increases, the restricted area of the swarm robot decreases.
- If the size of the swarm (N) is bigger, the restricted area is lower.

4. Multitasking-Control System of Swarm Robot

When the robots perform the task of moving to a destination, on the way they must avoid obstacles. So, each robot in the swarm has three tasks as follows:

- The first task is avoiding obstacles
- The second task is moving to the destination.
- The third task is maintaining the swarm: Avoiding collisions among individuals in the swarm, but not splitting the group.

In order for the robot to perform the above tasks, the supervisor selects the priority of the tasks. In this study, the priority of the tasks in order is: Avoiding obstacles, moving to the destination, and maintaining the swarm. Assume that the obstacles are static and are known then the speed vector of each robot based on the null- space behavioral control technique [36] is calculated according to Figure 4.

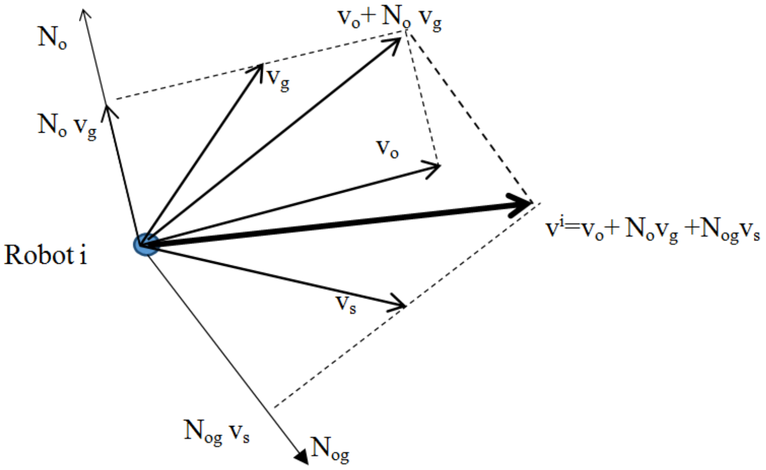


Figure 4. The speed vector of each robot based on the null-space behavioral control technique.

The speed of the robot (i) is determined as follows:

$$v^i = v_o + N_o v_g + N_{og} v_s$$

where v_o, v_g, v_s are the speed vectors performing the tasks: Avoiding obstacles, moving to the destination and maintaining the swarm. N_o, N_{og} are the projection matrixes according to the priority of the tasks.

4.1. Determining the Speed Component Avoiding Obstacles

Assume that in the working environment of the swarm the robot has M obstructions, $\mathrm{p}^{\mathrm{o}_m} = \begin{bmatrix} \mathrm{p}_1^{\mathrm{o}_m} \\ \mathrm{p}_2^{\mathrm{o}_m} \\ \vdots \\ \mathrm{p}_\mathrm{n}^{\mathrm{o}_m} \end{bmatrix} \in \mathrm{R}^{\mathrm{n}\times 1}$ is the position of the obstruction (m) in n-dimensional space, (m = 1 ÷ M).

$\sigma_\mathrm{o} \in \mathrm{R}$ is the actual distance between the individual robot (i) and the obstacle (m):

$$\sigma_\mathrm{o} = \|\mathrm{p}^{\mathrm{o}_m} - \mathrm{p}^\mathrm{i}\| = \sqrt{\left(\mathrm{p}_1^{\mathrm{o}_m} - \mathrm{p}_1^\mathrm{i}\right)^2 + \left(\mathrm{p}_2^{\mathrm{o}_m} - \mathrm{p}_2^\mathrm{i}\right)^2 + \cdots + \left(\mathrm{p}_\mathrm{n}^{\mathrm{o}_m} - \mathrm{p}_\mathrm{n}^\mathrm{i}\right)^2}$$

The purpose of the speed component avoiding obstacle is that if the obstacle lies in the moving way, the robot must be kept away from the obstacle at a safe distance $\sigma_{\mathrm{o,d}} = \sigma_\mathrm{o}^*$, if the obstacle is outside the safe area of the robot, the obstacle does not affect the movement speed of the robot. So, the movement speed of the robot depends on the distance between the robot and the obstacle.

The Jacobi Matrix $\mathrm{J}_\mathrm{o} \in \mathrm{R}^{\mathrm{M}\times\mathrm{n}}$ shows the movement speed of the robot avoiding obstacles:

$$\mathrm{J}_\mathrm{o} = \begin{bmatrix} \left[\frac{\mathrm{p}^{\mathrm{o}_1} - \mathrm{p}^\mathrm{i}}{\|\mathrm{p}^{\mathrm{o}_1} - \mathrm{p}^\mathrm{i}\|}\right]^\mathrm{T} \\ \left[\frac{\mathrm{p}^{\mathrm{o}_2} - \mathrm{p}^\mathrm{i}}{\|\mathrm{p}^{\mathrm{o}_2} - \mathrm{p}^\mathrm{i}\|}\right]^\mathrm{T} \\ \vdots \\ \left[\frac{\mathrm{p}^{\mathrm{o}_\mathrm{M}} - \mathrm{p}^\mathrm{i}}{\|\mathrm{p}^{\mathrm{o}_\mathrm{M}} - \mathrm{p}^\mathrm{i}\|}\right]^\mathrm{T} \end{bmatrix} = \hat{\mathrm{p}}_\mathrm{io}^\mathrm{T} \tag{23}$$

The Matrix inverse of J_o:

$$\mathrm{J}_\mathrm{o}^+ = \hat{\mathrm{p}}_\mathrm{io},\ \mathrm{J}_\mathrm{o}^+ \in \mathrm{R}^{\mathrm{n}\times\mathrm{M}}$$

The projection matrix of J_o:

$$\mathrm{N}_\mathrm{o} = \mathrm{I}_\mathrm{n} - \hat{\mathrm{p}}_\mathrm{io}\hat{\mathrm{p}}_\mathrm{io}^\mathrm{T},\ \mathrm{N}_\mathrm{o} \in \mathrm{R}^{\mathrm{n}\,\times\,\mathrm{n}} \tag{24}$$

where I_n is the unit matrix.

The speed component avoiding the obstacle is defined as follows:

$$\mathrm{v}_\mathrm{o} = -\mathrm{k}_\mathrm{vo}\mathrm{J}_\mathrm{o}^+\left(\sigma_\mathrm{o} - \sigma_\mathrm{o}^*\right) = -\mathrm{k}_\mathrm{vo}\mathrm{J}_\mathrm{o}^+\widetilde{\sigma}_\mathrm{o} \tag{25}$$

where k_vo is a negative coefficient, $\widetilde{\sigma}_\mathrm{o} = \sigma_\mathrm{o} - \sigma_\mathrm{o}^*$ is the error between the actual distance and the desired distance from the robot to the obstacle.

4.2. Determining the Speed Component Moving to the Target

Named $\mathrm{p}^\mathrm{g} = \begin{bmatrix} \mathrm{p}_1^\mathrm{g} \\ \mathrm{p}_2^\mathrm{g} \\ \vdots \\ \mathrm{p}_n^\mathrm{g} \end{bmatrix} \in \mathrm{R}^{\mathrm{n}\times 1}$ is the position of the target, $\sigma_\mathrm{g} \in \mathrm{R}$ is the actual distance between the robot (i) and the target, σ_g is calculated according to the formula:

$$\sigma_\mathrm{g} = \|\mathrm{p}^\mathrm{g} - \mathrm{p}^\mathrm{i}\| = \sqrt{\left(\mathrm{p}_1^\mathrm{g} - \mathrm{p}_1^\mathrm{i}\right)^2 + \left(\mathrm{p}_2^\mathrm{g} - \mathrm{p}_2^\mathrm{i}\right)^2 + \cdots + \left(\mathrm{p}_\mathrm{n}^\mathrm{g} - \mathrm{p}_\mathrm{n}^\mathrm{i}\right)^2}$$

The purpose of the speed component moving to the target is that the desired distance (σ_g^*) is equal to 0:

$$\sigma_{\mathrm{g,d}} = \sigma_\mathrm{g}^* = 0$$

The Jacobi Matrix $J_g \in R^{1\times n}$:

$$J_g = \left[\frac{p^g - p^i}{\|p^g - p^i\|}\right]^T = \hat{p}_{ig}^T \tag{26}$$

The Matrix inverse of J_g:

$$J_g^+ = \hat{p}_{ig}, \quad J_g^+ \in R^{n\times 1}$$

The projection matrix of J_g:

$$N_g = I_n - \hat{p}_{ig}\hat{p}_{ig}^T, N_g \in R^{n\times n} \tag{27}$$

The speed component moving to the target (i) is rewritten as follows:

$$v_g = k_{vg}J_g^+\left(\sigma_g - \sigma_g^*\right) = k_{vg}J_g^+\widetilde{\sigma}_g \tag{28}$$

where k_{vg} is a positive coefficient, $\widetilde{\sigma}_g = \sigma_g - \sigma_g^*$ is the error between the actual distance and the desired distance from the robot to the target:

$$\widetilde{\sigma}_g = \sigma_g - \sigma_g^* = \sigma_g$$

4.3. Determining the Maintained-Swarm Speed Component

The purpose of the maintained-swarm speed component is keeping $\sigma_s = \sigma_s^* \in R$. In this condition, the difference between the actual distance and the desired distance is:

$$\widetilde{\sigma}_s = \sigma_s - \sigma_s^*$$

From the model Equation (8) of the individual robot (i), the Jacobi matrix J_s:

$$J_s = \hat{p}_s^T = \begin{bmatrix} J_{s1} \\ J_{s2} \\ \vdots \\ J_{sN} \end{bmatrix} = \begin{bmatrix} \hat{p}_{s1}^T \\ \hat{p}_{s2}^T \\ \vdots \\ \hat{p}_{sN}^T \end{bmatrix} = \begin{bmatrix} \left[\frac{p^1-p^i}{\|p^1-p^i\|}\right]^T \\ \left[\frac{p^2-p^i}{\|p^2-p^i\|}\right]^T \\ \vdots \\ \left[\frac{p^N-p^i}{\|p^N-p^i\|}\right]^T \end{bmatrix} \in R^{N\times n} \tag{29}$$

The Matrix inverse of J_s:

$$J_s^+ = \hat{p}_s = \begin{bmatrix} J_{s1} \\ J_{s2} \\ \vdots \\ J_{sN} \end{bmatrix}^T = \begin{bmatrix} \hat{p}_{s1}^T \\ \hat{p}_{s2}^T \\ \vdots \\ \hat{p}_{sN}^T \end{bmatrix}^T = \begin{bmatrix} \left[\frac{p^1-p^i}{\|p^1-p^i\|}\right]^T \\ \left[\frac{p^2-p^i}{\|p^2-p^i\|}\right]^T \\ \vdots \\ \left[\frac{p^N-p^i}{\|p^N-p^i\|}\right]^T \end{bmatrix}^T \in R^{n\times N} \tag{30}$$

The projection matrix of J_s:

$$N_s = I_n - \hat{p}_s\hat{p}_s^T, N_s \in R^{n\times n} \tag{31}$$

The maintained-swarm speed component of the individual robot (i) is defined as follows:

$$v_s = J_s^+ f(\widetilde{\sigma}_s) \in R^{n\times 1} \tag{32}$$

Combining all the speed vectors of each robot when performing all three tasks based on the NSB method is shown in Figure 4:

$$v^i = v_o + N_o v_g + N_{og} v_s = -k_{vo} J_o^+ \widetilde{\sigma}_o + k_{vg} N_o J_g^+ \widetilde{\sigma}_g + N_{og} J_s^+ f(\widetilde{\sigma}_s) \tag{33}$$

where: $v^i \in R^{n\times 1}$, $J_{og} = \begin{bmatrix} J_o \\ J_g \end{bmatrix}$, $J_{og} \in R^{(M+1)\times n}$, $N_{og} = I_n - J_{og}^+ J_{og}$, $N_{og} \in R^{n\times n}$.

4.4. The Algorithm of Swarm Robot Control for Performing Multiple-Task

The algorithm of swarm robot control for performing multiple-tasks includes the following steps:

- The first step
  - ➢ Enter the number (N) of robots in the swarm.
  - ➢ Enter the number (M) of obstacles in the moving space.
  - ➢ Initially the position of individual robots in n-dimensional space:

$$p^1 = \begin{bmatrix} p_1^1 \\ p_2^1 \\ \vdots \\ p_n^1 \end{bmatrix}, p^2 = \begin{bmatrix} p_1^2 \\ p_2^2 \\ \vdots \\ p_n^2 \end{bmatrix}, \ldots p^N = \begin{bmatrix} p_1^N \\ p_2^N \\ \vdots \\ p_n^N \end{bmatrix}$$

  - ➢ Placement of M obstacles and the destination (g) in n-dimensional space:

$$p^{o1} = \begin{bmatrix} p_1^{o1} \\ p_2^{o1} \\ \vdots \\ p_n^{o1} \end{bmatrix}, p^{o2} = \begin{bmatrix} p_1^{o2} \\ p_2^{o2} \\ \vdots \\ p_n^{o2} \end{bmatrix}, \ldots p^{oM} = \begin{bmatrix} p_1^{oM} \\ p_2^{oM} \\ \vdots \\ p_n^{oM} \end{bmatrix}, p^g = \begin{bmatrix} p_1^g \\ p_2^g \\ \vdots \\ p_n^g \end{bmatrix}$$

  - ➢ Enter the safe distance between the individual robot and obstacle σ_o^*, the safe distance among robot individuals σ_s^*
  - ➢ Enter the coefficients k_{vo} and k_{vg}
  - ➢ Enter the number of steps to calculate (K).
- The second step
  - ➢ Calculating the distance between each robot (i) and each obstacle σ_o, the distance between each robot and target, the distance between robot (i) and robot (j).
  - ➢ Calculating the suction/thrust force $f(\sigma_s)$ according to Equation (3), satisfying Condition (4).
- The third step
  - ➢ Comparing the actual distance and safe distance from the robot (i) to the obstacle (m):
    - ■ If $\sigma_o \geq \sigma_o^*$, the robot (i) does not need to avoid the obstacle (o), it means $J_o = [0]$.
    - ■ If $\sigma_o < \sigma_o^*$, the robot (i) needs to avoid the obstacle (o), calculating J_o by Formula (26). Calculating $J_o^+,\ N_o,\ v_o$.
  - ➢ Comparing the actual distance and the desired distance from the robot (i) to the target:
    - ■ If $\sigma_g = 0$, the robot (i) reached the target g, $J_g = [0]$.
    - ■ If $\sigma_g > 0$, the robot (i) has not reached the target, calculating J_g by the Formula (24).
    - ■ Calculating J_g^+, N_g, v_g, calculating: J_{og}, J_{og}^+, N_{og}.

- Comparing the actual distance and the desired distance from the robot (i) to the robot (j):
    - If $\sigma_s > \sigma_s^*$, the robot (i) and the robot (j) move towards each other by the suction force $f(\sigma_s) > 0$.
    - If $\sigma_s < \sigma_s^*$, the robot (i) and the robot (j) move away from each other by the thrust force $f(\sigma_s) < 0$.

    i. If $\sigma_s = \sigma_s^*$, the robot (i) and the robot (j) keep their route because of $f(\sigma_{s_i}) = 0$.

    ii. Calculating J_s, J_s^+, v_s.

- The fourth step
    - The speed of the individual (i) at the step k ($k = 0 \div K - 1$) is determined by the formula:

$$v^i[k] = v_o[k] + N_o[k]v_g[k] + N_{og}[k]v_s[k]$$

    - The distance moved of the robot (i) in a step time(Δt):

$$\Delta S^i[k+1] = \Delta S^i[k] + v^i[k] \times \Delta t$$

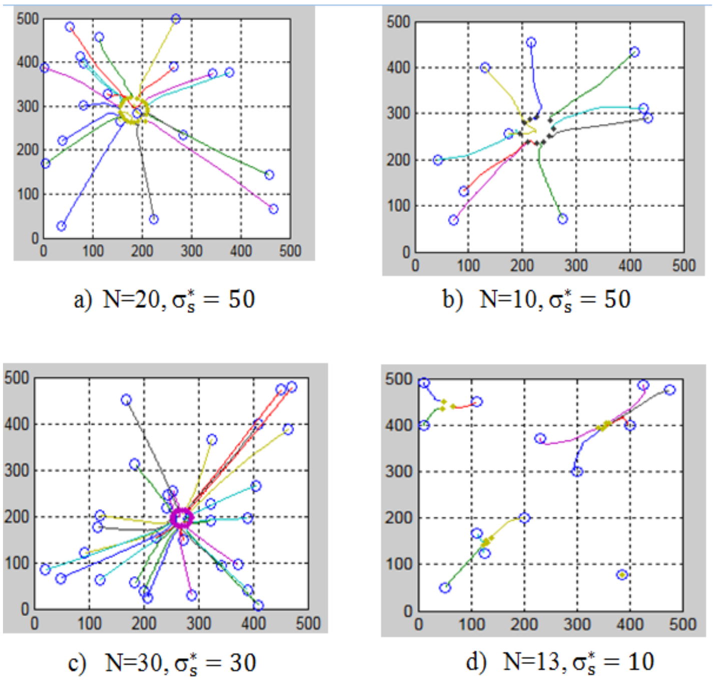
    - The new position of the robot (i) after k + 1 steps:

$$p^i[k+1] = p^i[k] + \Delta S^i[k+1]$$

Repeat from the second step to the fourth step until all individuals converge at the target and after K calculation steps.

5. Simulation Results and Analysis

We ran the simulation in the two-dimensional coordinate system [500, 500]. The initial position of the robots, obstacles, and targets are random. The convergence process of the swarm robot without obstacles is shown in Figure 5.



a) N=20, $\sigma_s^* = 50$

b) N=10, $\sigma_s^* = 50$

c) N=30, $\sigma_s^* = 30$

d) N=13, $\sigma_s^* = 10$

Figure 5. The convergence process of the swarm robot without obstacles.

R is the actual convergence radius of the swarm robot, R is also the distance from the nearest robot to the center of the swarm. The results of the calculation of the parameters in certain cases are presented in Table 1.

Table 1. The parameters in certain cases.

N	σ_s^*	α	β	σ	**R**
10	10	0.02	9.74	15.61	6.18
10	30	0.02	25.37	43.62	17.06
20	10	0.02	8.40	10.25	4.10
20	30	0.02	25.33	30.82	15.18
30	10	0.02	8.23	8.77	3.91
30	30	0.02	25.25	25.23	15.04

The results in Table 1 show that:

- If the size of the swarm (N) increases, the convergence radius decreases;
- If the safe distance (σ_s^*) increases, the convergence radius increases;
- The actual convergent radius (R) is always smaller than the calculated value (σ).

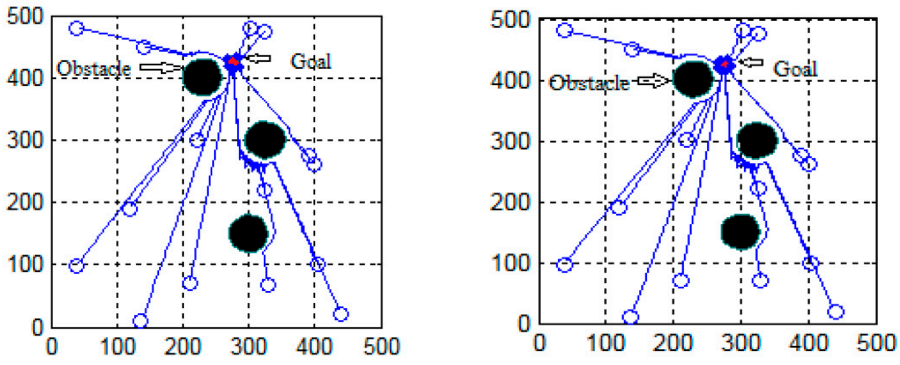
The simulation results are consistent with the theorem that the authors expressed above.

Run the swarm robot model with the obstacles, then the simulation results when the number of robots are changing and the priority coefficient is changing are shown in Figures 6 and 7.

Figure 6 shows that all individual robots have moved to their destination and avoided obstacles along the way. After individual robots converge to the destination, they only move around the destination area and do not move away.

The simulation results when changing the priority coefficients k_{vo} and k_{vg} are shown in Figure 7. From the simulation results, we see:

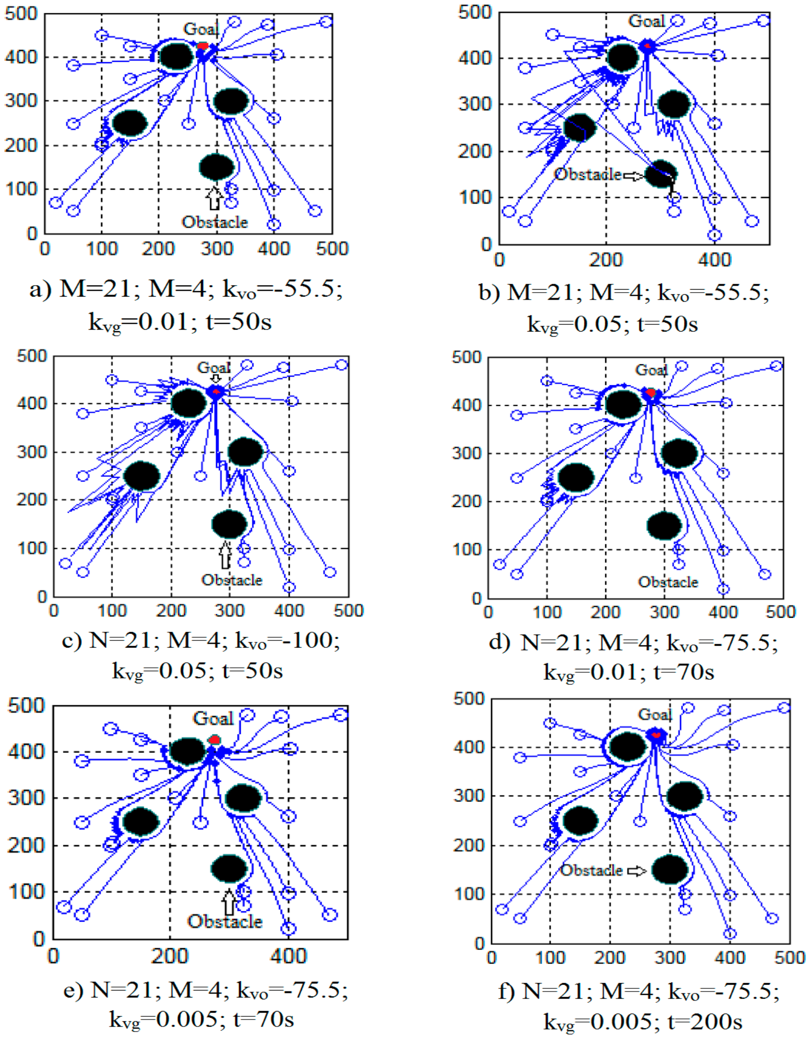
- If the coefficient k_{vg} is larger, the individual movement to the target is faster.
- If we want to increase the coefficient k_{vg} but not let the robot collide with obstacles, we must reduce the coefficient k_{vo}. This means, if k_{vg} is more positive, k_{vo} must be more negative.
- If the number of obstacles (M) is bigger, the avoid-obstacle-coefficient (k_{vo}) must be more negative. If the coefficient k_{vo} is more negative, the ability of the robots to avoid obstacles is better, but the moving time to the destination will be longer.



a) N=15; M=3; k_{vo}=-55.5; k_{vg}=0.05; t=50s

b) N=15; M=3; k_{vo}=-55.5; k_{vg}=0.05; t=100s

Figure 6. The simulation results of the multi task process after a period.



a) M=21; M=4; k_{vo}=-55.5; k_{vg}=0.01; t=50s

b) M=21; M=4; k_{vo}=-55.5; k_{vg}=0.05; t=50s

c) N=21; M=4; k_{vo}=-100; k_{vg}=0.05; t=50s

d) N=21; M=4; k_{vo}=-75.5; k_{vg}=0.01; t=70s

e) N=21; M=4; k_{vo}=-75.5; k_{vg}=0.005; t=70s

f) N=21; M=4; k_{vo}=-75.5; k_{vg}=0.005; t=200s

Figure 7. The simulation results of the multitask process when the coefficients k_{vo} and k_{vg} are changing.

6. Conclusions

In this paper, the authors applied both fuzzy logic to control the suction/thrust force between individuals in the swarm, and the null-space behavioral control technique to determine the total speed vector of each robot. The proposed solution is close to nature with high flexibility by selecting the input/output, defuzzifier, and the fuzzy rules. The simulation results coincide with the theories proposed by the authors. Compared with the previous methods, the advantage of our proposed method is that the object can be controlled easily without the requirement of knowing the mathematical equation. We controlled effectively a swarm robot without the object model equations. All individuals of the swarm move together to the destination without collision, and ensure the task of avoiding obstacles.

Author Contributions: Nga Le Thi Thuy proposed the initial idea. Thang Nguyen Trong and Nga Le Thi Thuy developed the research and analyzed the results together. Thang Nguyen Trong wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Darmanin, R.N.; Bugeja, M.K. A review on multi-robot systems categorised by application domain. In Proceedings of the 2017 25th Mediterranean Conference on Control and Automation (MED), Valletta, Malta, 3–6 July 2017; pp. 701–706.
2. Li, D.; Fan, Q.; Dai, X. Research status of multi-robot systems task allocation and uncertainty treatment. *J. Phys. Conf. Ser.* **2017**, *887*, 012081. [CrossRef]
3. Gautam, A.; Mohan, S. A review of research in multi-robot systems. In Proceedings of the 2012 7th IEEE International Conference on Industrial and Information Systems (ICIIS), Chennai, India, 6–9 August 2012; pp. 1–5.
4. Bruemmer, D.J.; Dudenhoeffer, D.D.; McKay, M.D.; Anderson, M.O. A Robotic Swarm for Spill Finding and Perimeter Formation. 2002. Available online: https://pdfs.semanticscholar.org/baa7/5fc5db081a730a01ecfccb42167c62d39c58.pdf (access on 9 December 2017).
5. Mataric, M.J. Behaviour-based control: Examples from navigation, learning, and group behaviour. *J. Exp. Theor. Artif. Intell.* **1997**, *9*, 323–336. [CrossRef]
6. Correll, N.; Martinoli, A. Multirobot inspection of industrial machinery. *IEEE Robot. Autom. Mag.* **2009**, *16*, 103–112. [CrossRef]
7. Kube, C.R.; Bonabeau, E. Cooperative transport by ants and robots. *Robot. Auton. Syst.* **2000**, *30*, 85–101. [CrossRef]
8. Miyata, N.; Ota, J.; Arai, T.; Asama, H. Cooperative transport by multiple mobile robots in unknown static environments associated with real-time task assignment. *IEEE Trans. Robot. Autom.* **2002**, *18*, 769–780. [CrossRef]
9. Farinelli, A.; Iocchi, L.; Nardi, D. Multirobot systems: A classification focused on coordination. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* **2004**, *34*, 2015–2028. [CrossRef]
10. Parker, L.E. Multiple mobile robot systems. In *Springer Handbook of Robotics*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 921–941.
11. Cai, Y.; Yang, S.X. A survey on multi-robot systems. In Proceedings of the World Automation Congress (WAC), Puerto Vallarta, Mexico, 24–28 June 2012; pp. 1–6.
12. Yan, Z.; Jouandeau, N.; Cherif, A.A. A survey and analysis of multi-robot coordination. *Int. J. Adv. Robot. Syst.* **2013**, *10*, 399. [CrossRef]
13. Cao, Y.; Yu, W.; Ren, W.; Chen, G. An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Trans. Ind. Inf.* **2013**, *9*, 427–438. [CrossRef]
14. Gerkey, B.P.; Matarić, M.J. A formal analysis and taxonomy of task allocation in multi-robot systems. *Int. J. Robot. Res.* **2004**, *23*, 939–954. [CrossRef]
15. Khamis, A.; Hussein, A.; Elmogy, A. Multi-robot task allocation: A review of the state-of-the-art. *Coop. Robot. Sens. Netw.* **2015**, *604*, 31–51.
16. Dias, M.B.; Zlot, R.; Kalra, N.; Stentz, A. Market-based multirobot coordination: A survey and analysis. *Proc. IEEE* **2006**, *94*, 1257–1270. [CrossRef]
17. Brambilla, M.; Ferrante, E.; Birattari, M.; Dorigo, M. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intell.* **2013**, *7*, 1–41. [CrossRef]
18. Dorigo, M.; Floreano, D.; Gambardella, L.M.; Mondada, F.; Nolfi, S.; Baaboura, T.; Burnier, D. Swarmanoid: A novel concept for the study of heterogeneous robotic swarms. *IEEE Robot. Autom. Mag.* **2013**, *20*, 60–71. [CrossRef]
19. Champion, B.T.; Joordens, M.A. Underwater swarm robotics review. In Proceedings of the 2015 10th System of Systems Engineering Conference (SoSE), San Antonio, Texas, USA, 17–20 May 2015; pp. 111–116.
20. Son, J.H.; Ahn, H.S. Cooperative reinforcement learning: brief survey and application to bio-insect and artificial robot interaction. In Proceedings of the IEEE/ASME International Conference on Mechtronic and Embedded Systems and Applications, 2008 (MESA 2008), Beijing, China, 12–15 October 2008; pp. 71–76.
21. Goodrich, M.A.; Schultz, A.C. Human-robot interaction: A survey. *Found. Trends Hum.-Comput. Interact.* **2007**, *1*, 203–275. [CrossRef]
22. Chen, J.Y.; Barnes, M.J. Human-agent teaming for multirobot control: A review of human factors issues. *IEEE Trans. Hum.-Mach. Syst.* **2014**, *44*, 13–29. [CrossRef]

23. Guanghua, W.; Deyi, L.; Wenyan, G.; Peng, J. Study on formation control of multi-robot systems. In Proceedings of the 2013 Third International Conference on Intelligent System Design and Engineering Applications (ISDEA), Hong Kong, China, 16–18 January 2013; pp. 1335–1339.
24. Yang, M.; Yan, G.G.; Tian, Y.T. A review of studies in flocking for multi-robot system. In Proceedings of the 2010 International Conference on Computer, Mechatronics, Control and Electronic Engineering (CMCE), Changchun, China, 24–26 August 2010; pp. 28–31.
25. Portugal, D.; Rocha, R. A Survey on Multi-Robot Patrolling Algorithms. 2011. Available online: http://ap.isr.uc.pt/archive/dportugalPR11_Doceis2011.pdf (access on 9 December 2017).
26. Liu, Y.; Nejat, G. Robotic urban search and rescue: A survey from the control perspective. *J. Intell. Robot. Syst.* **2013**, *72*, 147. [CrossRef]
27. Cook, D.; Vardy, A.; Lewis, R. A survey of AUV and robot simulators for multi-vehicle operations. In Proceedings of the 2014 IEEE/OES Autonomous Underwater Vehicles (AUV), Oxford, MS, USA, 6–9 October 2014; pp. 1–8.
28. Xiong, N.; He, J.; Yang, Y.; He, Y.; Kim, T.H.; Lin, C. A survey on decentralized flocking schemes for a set of autonomous mobile robots. *JCM* **2010**, *5*, 31–38. [CrossRef]
29. Olfati-Saber, R. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Trans. Autom. Contr.* **2006**, *51*, 401–420. [CrossRef]
30. Gazi, V.; Passino, K.M. Stability analysis of swarms. *IEEE Trans. Autom. Contr.* **2003**, *48*, 692–697. [CrossRef]
31. Wang, L.; Fang, H. Stability analysis of practical anisotropie swarms. In Proceedings of the 2010 11th International Conference on Control Automation Robotics & Vision (ICARCV), Singapore, 7–10 December 2010; pp. 768–772.
32. Chen, X.B.; Pan, F.; Li, L.; Fang, H. Practical stability analysis for swarm systems. In Proceedings of the IECON 2006-32nd Annual Conference on IEEE Industrial Electronics, Paris, France, 6–10 November 2006; pp. 3904–3909.
33. Xue, Z.; Zeng, J.; Feng, C.; Liu, Z. Swarm target tracking collective behavior control with formation coverage search agents & globally asymptotically stable analysis of stochastic swarm. *JCP* **2011**, *6*, 1772–1780.
34. Brooks, R. A robust layered control system for a mobile robot. *IEEE J. Robot. Autom.* **1986**, *2*, 14–23. [CrossRef]
35. Marino, A.; Parker, L.E.; Antonelli, G.; Caccavale, F. A decentralized architecture for multi-robot systems based on the null-space-behavioral control with application to multi-robot border patrolling. *J. Intell. Robot. Syst.* **2013**, *71*, 423–444. [CrossRef]
36. Wolkenhauer, O. A course in fuzzy systems and control. *Int. J. Electr. Eng. Educ.* **1997**, *34*, 282. [CrossRef]