Hunting strategy for multi-robot based on wolf swarm algorithm and artificial potential field

Oussama Hamed, Mohamed Hamlich, Mohamed Ennaji
Complex Cyber Physical Systems Laboratory, National Higher School of Arts and Crafts (ENSAM), University of Hassan II, Casablanca, Morocco

Article Info

Article history:

Received Jun 20, 2021
Revised Oct 22, 2021
Accepted Nov 24, 2021

Keywords:

Cooperative systems
Mobile robots
Multi-robot systems
Optimization
Path planning

ABSTRACT

The cooperation and coordination in multi-robot systems is a popular topic in the field of robotics and artificial intelligence, thanks to its important role in solving problems that are better solved by several robots compared to a single robot. Cooperative hunting is one of the important problems that exist in many areas such as military and industry, requiring cooperation between robots in order to accomplish the hunting process effectively. This paper proposed a cooperative hunting strategy for a multi-robot system based on wolf swarm algorithm (WSA) and artificial potential field (APF) in order to hunt by several robots a dynamic target whose behavior is unexpected. The formation of the robots within the multi-robot system contains three types of roles: the leader, the follower, and the antagonist. Each role is characterized by a different cognitive behavior. The robots arrive at the hunting point accurately and rapidly while avoiding static and dynamic obstacles through the artificial potential field algorithm to hunt the moving target. Simulation results are given in this paper to demonstrate the validity and the effectiveness of the proposed strategy.

Corresponding Author:

Oussama Hamed
CCPS Laboratory, ENSAM, University of Hassan II
150 Street Nile Sidi Othm Casablanca, Morocco
Email: oussamahamed11@gmail.com

1. INTRODUCTION

The cooperative hunting problem is an ideal problem for the research on multi-robot cooperation and coordination [1]-[3]. These robots work together to hunt a single target or a group of targets. Due to several factors, including uncertainty, target's velocity, target's behavior, lack of information, and the nature of the search space, it is difficult to perform complex tasks such as hunting with a single robot [4]-[7]. Conversely, in order to compensate for the shortage and weaknesses of using a single robot for hunting, a multi-robot system which is composed of many robots is employed instead. Multi-robot hunting is one of the most important and complex challenges in the field of multi-robot cooperation due to the fact that it includes all multi-robot system subproblems such as task allocation, target localization, collaborative pursuing, obstacle and collisions avoidance. The research on multi-robot system cooperative hunting covers many disciplines and domain knowledge, such as optimization algorithms [8], real-time dynamic path planning [9]-[10], multi-robot coordination [11]-[13], planning and learning [14], and communication [15].

There are many methods to solve the problem of multi-robot cooperative hunting. These methods are classified into two categories. The first category is cooperative hunting strategies based on artificial intelligence algorithms that are being used to hunt the target whose location is previously known to hunters [16]. For the second category, the hunting methods are based on sensor information, where the location of the target is not predefined or the environment is unknown [17]. Both categories have some disadvantages such as in the first category methods that mainly depend on the information provided to the hunters in advance about the location of the target, which is impractical when the target is constantly moving in a random way. Whereas, the second category methods are mainly depending on the information provided by the sensors, therefore this information is not always reliable and do not produce robust results. The information that the sensors detect about research space is used by robots to capture the target [18]. This information is analyzed in order to find out the location of the robots, track the target, and detect obstacles.

At present, the hunting with multi-robot attracts the attention of many researchers, as it is both significant and complex. As a result of this, many researches have been done that propose solutions for the problems of multi-robot cooperative hunting methods. Yan Wang *et al.* [19] proposed an improved Voronoi graph self-organizing cooperative hunting algorithm for the problem that capturing random targets with a conventional distributed algorithm based on Voronoi graph has a low efficiency and a high standard deviation. In this algorithm, Hunters find their own targets by choosing the nearest midpoint of a typical Voronoi graph boundary, and form a hunter's alliance for the same target. To hunt, each hunter in the group employs a goal Voronoi graph area minimization technique. In [20], a cooperative hunting method inspired by the behavior of biological societies is proposed which uses four parameters that are repulsion, attraction, orientation, and influence. While attempting to catch the prey, the hunter robots create a cooperative swarm, and changing the parameters allows for decentralized control of the swarm's formation and behavior. The researchers in [21]-[23] propose solutions to solve cooperative hunting with multi-robot based on reinforcement learning.

In this paper, the cooperative hunting multi-robot system is studied when the target moves with an unknown behavior, and a hunting strategy based on wolf swarm algorithm and artificial potential field is proposed. This hunting strategy is geared towards solving the problems of target escaping from hunters and long hunting time. In order to capture a target which is constantly moving at different speeds and in different directions with the presence of obstacles within the search space, the hunter robots must have the ability to keep the hunting formation and self-adjust while interacting with the environment. Moreover, this hunting process is independent of dynamic constraints related to the robot model. The specific hunting process is as follows: firstly, with moving of the target, the robots behave with the hunting method for wolves' swarm using the wolf swarm algorithm which is to enable robots to work under a certain configuration. In the wolf swarm algorithm, the multi-robot system contains three kinds of robots, the leader, the antagonist and the follower that are for preventing the target's escape when they detect and catch it as a collaborative swarm. Robots take turns during the hunt, according to the position of the robot with the target. Each wolf (robot) has a specific cognitive behavior that enables it to work with the swarm, track, and hunt the prey. Secondly, the robots reach the desired point rapidly and safely through the artificial potential field which is an important algorithm for avoiding obstacles and collisions with other robots. The simulation results show that the proposed algorithm can accurately and quickly hunt the moving target with an unknown behavior in an environment that contains obstacles. The main contributions of this paper are summarized as follows: i) Design of the simulation model of efficient and robust cooperation and coordination of multi-robot systems for hunting algorithm; ii) This hunting method is proposed to find an optimal, accurate, and safe path to catch the dynamic target by avoiding obstacles; iii) This method relies on properties that enable the robots' movements to change in real time to rapidly adapt to changes in the pursuit; and iv) Reducing the resources required for the implementation of the hunting process based on the uncomplicated calculations in this proposed strategy, which must be calculated at the real time. The rest of the article is structured as follows: section 2 discusses some related works, section 3 introduces the basic theoretical knowledge of methods used, section 4, presents the proposed hunting strategy of dynamic target for multi-robot systems and how it solves the cooperative hunting problem, section 5 gives the simulation test and verifies the feasibility of the algorithm. In section 6, conclusion and future works are discussed.

2. RELATED WORKS

This section gives a quick overview of some of the recent methods proposed for solving the multi-robot hunting problem. In the multi-autonomous underwater vehicle (multi-AUV) hunting algorithm, a leader-follower formation algorithm [24] is proposed by Cao and Guo to solve the problems of hunting a target. This method is based on leader-follower formation. The process of hunting is composed of three phases. The initial phase is task assignment, then formation tracking phase, and finally the phase of target capture. With this proposed algorithm, all hunters arrive at the same time at the hunting locations, cutting down on hunting time and preventing the escape of the target. In our work, we added a new role in the swarm in addition to the leader and the follower in order to find the target quickly, and change the formation to adapt to the hunting situation, and prevent the target escaping.

Cao and Xu [25] proposed a multi-AUV hunting algorithm based on dynamic prediction for the trajectory of the moving target to increase the performance of target hunting of multi-AUV. At the beginning of the hunting process, the prediction of the possible position of the target is performed while the target is moving using sample points that are updated dynamically in a short delay by using the fitting of a polynomial. Then, to assign the intended hunting points for each AUV, the method of negotiation is used. This method uses a deep reinforcement learning (DRL) algorithm to help the AUVs to arrive rapidly at targeted hunting spots. This method has some limitations as hunting a target with intelligent behavior who can escape easily from the hunters.

In [26] they proposed the adaptive bio-inspired neural network (ABNN) method based on ABNN which is addressed to solve repeated and inefficient cooperative hunting of many targets by multi-robot problem in a workspace containing big-size obstacles using the new shunting equation which allows to optimize the hunting performance of the targets by the multiple hunters. This method is designed to catch targets by predicting their paths while avoiding the big sized obstacles. Hamed and Hamlich [27] presented a new strategy based on the wolf swarm algorithm for hunting a target with unexpected behavior by means of a cooperative multi-robot system. This method which is inspired from wolf's behavior contains three types of wolves helps to reduce the time needed to capture the dynamic target and prevent it escape. However, this method has some limitations such as it is used only to capture a single target, and it is not practical when the research space contains obstacles.

3. THE PROPOSED STRATEGY

Our idea came from both combining an improvised multi-robot cooperation strategy for hunting a dynamic target [26] and the artificial potential fields method [28]. The objective is to hunt a target with random behavior and velocity by several robots. This hunting process is based on a strategy inspired from wolf swarm algorithm [6] where all robots in this multi-robot system have a specific role. This role is changeable according to the hunt situation. Each robot has to follow the hunting strategy while avoiding collisions with obstacles and other robots. We will give a brief introduction on an improvised multi-robot cooperation strategy for hunting a dynamic target and the artificial potential field.

3.1. Improvised multi-robot cooperation strategy for hunting a dynamic target

This paper adopts this improvised multi-robot cooperation strategy for hunting a dynamic target which is an efficient method to solve this kind of multi-robot tasks. This method is adapted from the way which the wolves use for hunting with some modifications to the wolves' behavior. The robots in this method are divided in three roles, the leader wolf, the antagonist wolves and the follower wolves. The N robots use for hunting the odor concentration of the prey which is expressed with this fitness function Ø (R_r, T), where T is the target vector position and r = (1…. N). The closer the robot is to the target, the greater the fitness function Ø (R_r, T) value. The function value Ø (R_r, T) is independent of the number of dimensions of the search space(n) Ø: $IR^n \rightarrow IR$.

3.1.1. The leader

The leader wolf R^l is supposed to have a great deal of knowledge about prey's position, and being the closest one to the prey. This leader starts howling and all the follower wolves join their leader to chase the target. The roles exchange between wolves is as follows: if Ø (R_r, T) > Ø (R^l, T) then the Robot R_r becomes the new leader and the ex-leader becomes an antagonist.

3.1.2. The antagonist

The robots with the role of the antagonist wolf follow the method described below in order to track the target without following the leader. Within the swarm, the number of antagonists (A_wolves) is ranging between [(N-1)/(λ +1) , (N-1)/λ], where λ = [1, N/2] is the antagonism proportion factor. The research area is determined from the position of the leader, because this leader is always closest to the prey. This wolf only searches in a specific angle. The measure of this angle which is between the x axis and the straight line from the antagonist and the leader, is calculated in the following way:

$$\Theta = 2\text{artan}\ \frac{R_y^l - R_{ky}^a}{\sqrt{(R_x^l - R_{kx}^a + 1)^2 + (R_y^l - R_{ky}^a)^2} + R_x^l - R_{kx}^a + 1} \quad (1)$$

R_{kx}^a and R_{ky}^a are the coordinates of the antagonist wolf R_k^a with $1 \leq k \leq$ (A_wolves), and the angle Θ value is by radian. The value of this angle is converted according the (2) to adapt it with the algorithmic method.

$$\Theta' = \begin{cases} \Theta, \Theta \in [0,\pi] \\ \Theta + 2\pi, \Theta \notin [0,\pi] \end{cases} \quad (2)$$

Using the aforementioned angle measure, the antagonist wolf examines the odor of prey in certain positions. These temporary positions $\left(\widetilde{R^{a}_{kx}}, \widetilde{R^{a}_{ky}}\right)^{t}_{\varphi}$ are calculated by the (3).

$$\begin{pmatrix}\widetilde{R^{a}_{kx}}\\ \widetilde{R^{a}_{ky}}\end{pmatrix}_{\varphi} = \begin{pmatrix}R^{a}_{kx}\\ R^{a}_{ky}\end{pmatrix} + \begin{pmatrix}\beta_1\\ \beta_2\end{pmatrix} \odot \begin{pmatrix}\cos\left(\frac{2\pi\varphi}{\psi}\right)\\ \sin\left(\frac{2\pi\varphi}{\psi}\right)\end{pmatrix} \tag{3}$$

The vector β is the seeking vector $(\beta_1, \beta_2)^t$ which is expressed in the (8). The factor ψ is the amount of the global directions whither the antagonist R^{a}_{k} will search for the prey, and φ is the factor for the advancing direction where:

$$\varphi = \left[\frac{\psi \times \Theta'}{2\pi} - l_1 \, , \frac{\psi \times \Theta'}{2\pi} + l_2\right] \tag{4}$$

The two integers l_1 and l_2 are responsible for limiting the seeking area. Their values are in interval $[1, \psi]$. The antagonist R_k updates its position to a temporary position where it senses the highest value of the prey's odor concentration provided $\varnothing(\widetilde{R^{a}_{k\varphi}}, T) > \varnothing(R^{a}_{k}, T)$. The antagonist wins the leader position if $\varnothing(R^{a}_{k}, T) > \varnothing(R^{l}, T)$.

3.1.3. The follower

In this swarm, there are N-A_wolves-1 followers. This hunting strategy defines two types of behavior for each follower. The follower wolf switches between the two behaviors according to the distance between this wolf and the leader D_{conv}. The distance of convergence D_{conv} is described as:

$$Dconv = \left(\frac{1}{\max(xs) - \min(xs)} + \frac{1}{\max(ys) - \min(ys)}\right)^{-1} \times \sigma \tag{5}$$

where $\boldsymbol{\sigma} =]0, 1]$ is the convergence factor. (xs, ys) are the boundaries of the search space. The robot is in the summoning behavior when the distance between this follower and the leader is greater than D_{conv}. The follower wolf advances to the leader, according the (6):

$$\begin{pmatrix}R^{f}_{jx}(i+1)\\ R^{f}_{jy}(i+1)\end{pmatrix} = \begin{pmatrix}R^{f}_{jx}(i)\\ R^{f}_{jy}(i)\end{pmatrix} + \begin{pmatrix}\alpha_1\\ \alpha_2\end{pmatrix} \odot \begin{pmatrix}R^{l}_{x} - R^{f}_{jx}(i)\\ R^{l}_{y} - R^{f}_{jy}(i)\end{pmatrix} \odot \begin{pmatrix}\varepsilon_1\\ \varepsilon_2\end{pmatrix} \odot \begin{pmatrix}\left|R^{l}_{x} - R^{f}_{jx}(i)\right|^{-1}\\ \left|R^{l}_{y} - R^{f}_{jy}(i)\right|^{-1}\end{pmatrix} \tag{6}$$

The vector $(\varepsilon_1, \varepsilon_2)^t$ is the deviation vector where $\varepsilon_1, \varepsilon_2$ are random numbers in $0 < \varepsilon_1, \varepsilon_2 \leq 1$, and i is the iteration index. α is the advancing vector which is expressed in the (8). The follower wolf behaves with preying behavior when the distance between it and the leader is less than D_{conv}. The follower with this behavior gradually encircles and grasps the prey. The update of follower's position is expressed by the expression:

$$\begin{pmatrix}R^{f}_{jx}(i+1)\\ R^{f}_{jy}(i+1)\end{pmatrix} = \begin{pmatrix}R^{f}_{jx}(i)\\ R^{f}_{jy}(i)\end{pmatrix} + \begin{pmatrix}\gamma_1\\ \gamma_2\end{pmatrix} \odot \begin{pmatrix}R^{l}_{x} - R^{f}_{jx}(i)\\ R^{l}_{y} - R^{f}_{jy}(i)\end{pmatrix} \odot \begin{pmatrix}\tau_1\\ \tau_2\end{pmatrix} \tag{7}$$

where $(\tau_1, \tau_2)^t$ is the encirclement vector, whereas $0 \leq \tau_1, \tau_2 \leq 2 - (i/max\_Iteration)$, and γ is the preying vector which is expressed in the (8).

The vectors α, β and γ are calculated as follows:

$$\begin{pmatrix}\beta_1\\ \beta_2\end{pmatrix} = \frac{4}{3} \cdot \begin{pmatrix}\gamma_1\\ \gamma_2\end{pmatrix} = \frac{1}{2} \cdot \begin{pmatrix}\alpha_1\\ \alpha_2\end{pmatrix} = \begin{pmatrix}\max(xs) - \min(xs)\\ \max(ys) - \min(ys)\end{pmatrix} . S \tag{8}$$

The scalar S is the step factor where $S \in]0,1[$.

3.2. Artificial potential fields (APF)

The path planning problem is to develop a strategy for determining a suitable path for the robot. In the case of a mobile robot, the robot should move in the workspace from the initial position to its target or final position. In this paper, we adopt the APF [28], [29] in order to plan the path, the robot will follow. In the same workspace which is a two-dimensional space, we have N robots. Each robot has coordinates that are noted by q (x, y).

In this method, we consider that the robot moves in a potential field of two virtual forces. The repulsive potential field U_{rep} (q) is generated by the obstacles to help collision avoidance. The attractive potential field U_{att} (q) is generated by the target to attract the robot to its goal destination. The expression of the attractive parabolic potential field U_{att} (q) is calculated depending on the position of the robot and the target.

$$Uatt(x,y)= \frac{1}{2}\,\xi\sqrt{(x-\delta_x)^2+(y-\delta_y)^2} \tag{9}$$

Where δx , δy is the position of the target. The scalar ξ is the attracting factor where ξ > 0.

The potential is greater when the robot is far from the target, and it decreases when the distance between the robot and the target decreases. The attractive force applied to the robot is calculated by (10.a):

$$F_{att}\,(x,y) = -\nabla Uatt\,(x,y) \tag{10.a}$$

By partially deriving the potential attractive equation to the axis x and y as follows:

$$F_{xatt}(x,y)= \frac{\partial U_{att}(x,y)}{\partial x} \tag{10.b}$$

$$F_{yatt}(x,y)= \frac{\partial U_{att}(x,y)}{\partial y} \tag{10.c}$$

The attractive force F_{att} equation on the x axis and y axis is expressed as follows:

$$F_{xatt}(x,y) = -\xi\,(x-\delta x) \tag{11.a}$$

$$F_{yatt}(x,y) = -\xi\,(y-\delta y) \tag{11.b}$$

The expression of the repulsive potential field U_{rep} (x,y) is calculated depending on the position of the robot and the obstacle. The other robots in the multi-robot system are also considered as obstacles.

$$U_{rep}(x,y) = Urepo\,(x,y) + Urepr\,(x,y) \tag{12}$$

Where U_{repo} (x,y) is the repulsive potential field generated by the obstacles. U_{repr} (x,y) is the repulsive potential field generated by the other robots.

$$Urepo(qi)=\begin{cases}\frac{1}{2}\ Krep\ \left(\frac{1}{d(qi,\upsilon oj)}-\frac{1}{\rho o}\right), & d(qi,\upsilon oj)<\rho o\\ 0, & d(qi,\upsilon oj)\geq\rho o\end{cases} \tag{13}$$

$$Urepr(qi)=\begin{cases}\frac{1}{2}\ Krep\ \left(\frac{1}{d(qi,\upsilon rk)}-\frac{1}{\rho r}\right), & d(qi,\upsilon rk)<\rho r\\ 0, & d(qi,\upsilon rk)\geq\rho r\end{cases} \tag{14}$$

Where d(qi,voj) is the Euclidean distance between the robot i (i = 1,2,….,N) and the obstacle j (j = 1,….,M), d(qi,υrk) is the Euclidean distance between the robot i and k, ρo is the safety distance from the obstacle, ρr is the safety distance from the other robot.

The repulsive force applied to the robot is calculated by the (15.a):

$$F_{rep}\,(x,y) = -\nabla U_{rep}\,(x,y) \tag{15.a}$$

By partially deriving the potential repulsive equation to the axis x and y axis, the repulsive forces are expressed as follows:

$$F_{xrepo}(x,y)= \frac{\partial U_{repo}(x,y)}{\partial x} \tag{15.b}$$

$$F_{yrepo}(x,y)= \frac{\partial U_{repo}(x,y)}{\partial y} \quad (15.c)$$

$$F_{xrepr}(x,y)= \frac{\partial U_{repr}(x,y)}{\partial x} \quad (15.d)$$

$$F_{yrepr}(x,y)= \frac{\partial U_{repr}(x,y)}{\partial y} \quad (15.e)$$

The repulsive forces F_{repo} and F_{repr} equations on the x axis and y axis are expressed as follows:

$$\text{Frepox}(qi)=-\begin{cases} \text{Krep}\left(\frac{1}{d(qi,\upsilon xoj)}-\frac{1}{\rho o}\right)\frac{(\upsilon xoj-xi)}{d(qi,\upsilon xoj)^3}, & d(qi,\upsilon xoj)<\rho o \\ 0, & d(qi,\upsilon xoj)\geq\rho o \end{cases} \quad (16.a)$$

$$\text{Frepoy}(qi)=-\begin{cases} \text{Krep}\left(\frac{1}{d(qi,\upsilon yoj)}-\frac{1}{\rho o}\right)\frac{(\upsilon yoj-yi)}{d(qi,\upsilon yoj)^3}, & d(qi,\upsilon yoj)<\rho o \\ 0, & d(qi,\upsilon yoj)\geq\rho o \end{cases} \quad (16.b)$$

$$\text{Freprx}(qi)=-\begin{cases} \text{Krep}\left(\frac{1}{d(qi,\upsilon xrk)}-\frac{1}{\rho r}\right)\frac{(\upsilon xrk-xi)}{d(qi,\upsilon xrk)^3}, & d(qi,\upsilon xrk)<\rho r \\ 0, & d(qi,\upsilon xrk)\geq\rho r \end{cases} \quad (16.c)$$

$$\text{Frepry}(qi)=-\begin{cases} \text{Krep}\left(\frac{1}{d(qi,\upsilon yrk)}-\frac{1}{\rho r}\right)\frac{(\upsilon yrk-yi)}{d(qi,\upsilon yrk)^3}, & d(qi,\upsilon yrk)<\rho r \\ 0, & d(qi,\upsilon yrk)\geq\rho r \end{cases} \quad (16.d)$$

3.3. Proposed cooperative hunting strategy

For hunting in multi-robot systems, one of the most important considerations is to avoid colliding with obstacles while maintaining the hunting strategy. In this part, we explain the hunting strategy which combines the artificial potential field and an improvised multi-robot cooperation strategy for hunting a dynamic target. In the presented strategy in the section (III.A), there is no consideration for any obstacle whether fixed or moving obstacles (other robots) Figure 1. The robots are looking for the target T in a workspace W that does not include any obstacle.

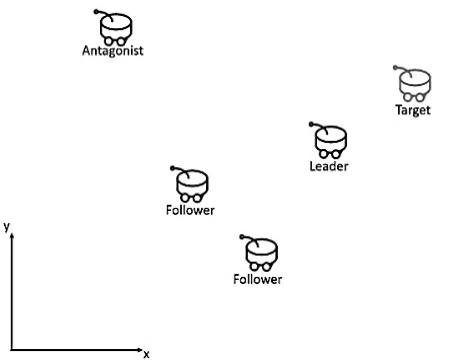

Figure 1. Hunters looking for the target

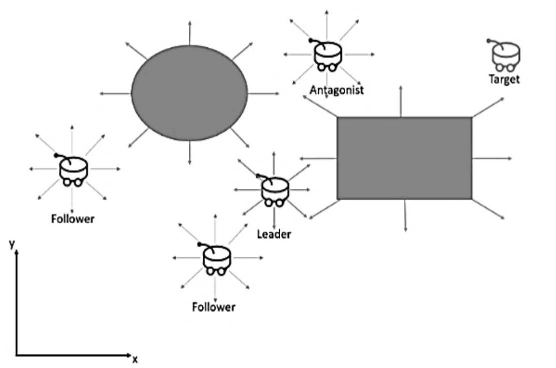

Figure 2. Repulsive potential field generated by obstacles and robots

During the hunt, there is a high possibility of a collision between robots. The workspace is not always free from obstacles. In the presented paper, static and moved obstacles are taken into consideration. By using the repulsive potential field Urep (q) which is generated by the obstacles and the other robots, the hunter robots avoid collision with any obstacle (Figure 2).

In this method, the attractive potential field Uatt (q) is generated by a virtual target instead of the target. The virtual target's position is calculated by the motion control ons (3), (6-7) used in the introduced hunting method. This method is based on dividing the path followed by the robot towards its target into many virtual target's positions in order to guide the hunter to the prey through a safe path. In the Figure 3, the follower advances towards the leader, and its next position is calculated with random parameters using (6) into area ($\varepsilon_1 \times \alpha_1 \times \alpha_2 \times \varepsilon_2$). For each iteration, the robot calculates its next position using the appropriate equation. This next position is updated each iteration is the virtual target position.

As shown in Figure 4 the robot's path towards the target is a set of virtual target's position that the robot will pass through. The equations used to calculate the virtual target's positions contain random parameters. Therefore, for the same situation, the robot can pass through different paths towards the target. When at least one virtual target's position is located in a moving obstacle (Figure 5) or a static obstacle area (Figure 6) the path is modified in a way that the robot does not collide with the obstacle. As shown in Figures 7 and 8, the robot tries to reach the virtual target due to the attractive force that it generates, but the repulsive force of the obstacles drives the robot away.

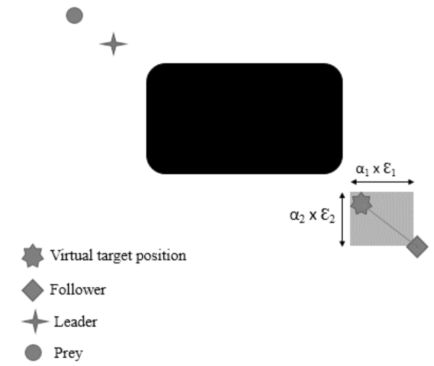

Figure 3. Follower's advancing process towards the leader

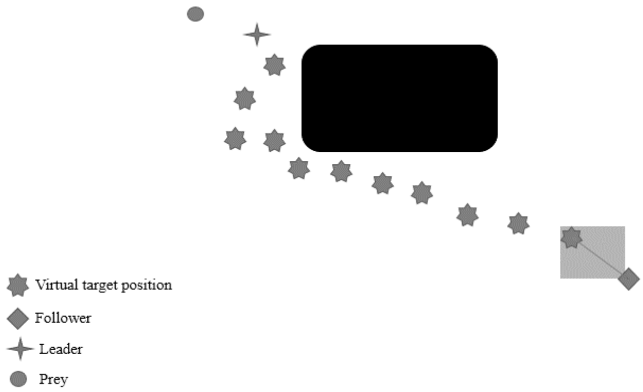

Figure 4. The path of the follower towards the leader

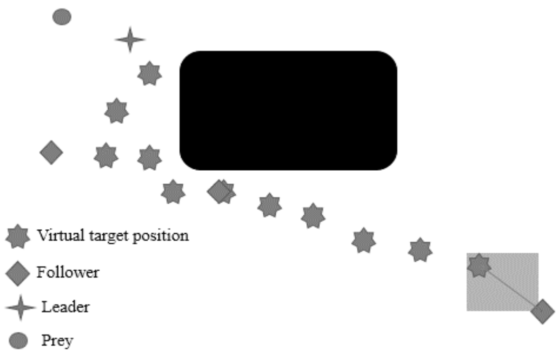

Figure 5. The follower's path intersects with another robot

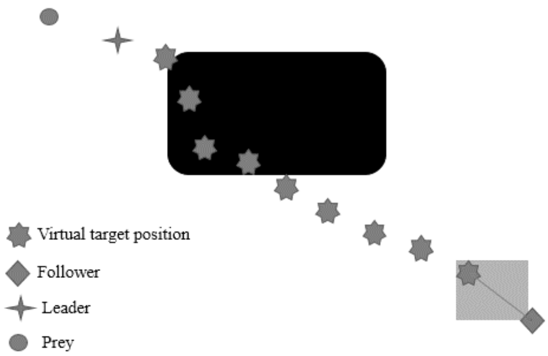

Figure 6. The generated path passes through the obstacle

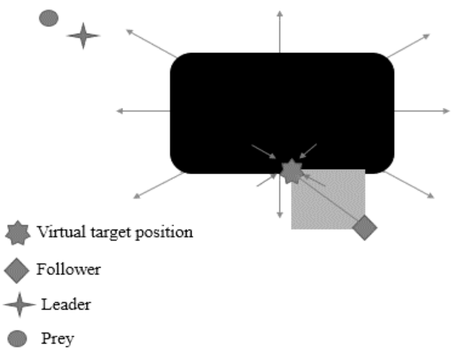

Figure 7. The repulsive force generated by the obstacle prevents the robot to reach the position of the virtual target

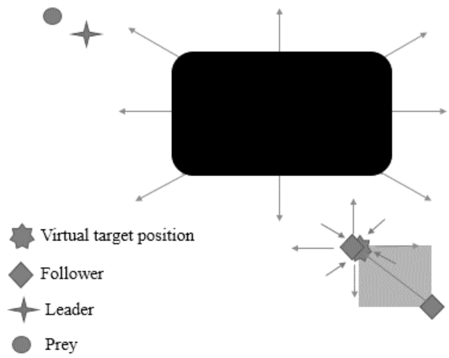

Figure 8. The repulsive force generated by another robot prevents the robot to reach the position of the virtual target

The robot's motion control equations have been combined with the equations of the artificial potential field. The forces are calculated using values of the virtual target's position. The attractive force for the virtual target when the robot is a follower is different when the robot is an antagonist (17.c). The same for the two behaviors of the follower. The attractive force for the virtual target when the follower robot is in summoning behavior (17.a) is different when the behavior of the follower robot is preying (17.b).

In the (11.a) and (11.b) the value δ_x and δ_y are replaced with the values of the virtual target's position according to the role and the behavior of the robot.

$$F_{att}(x,y) = (-\xi\,(x - \{R^f_{jx} + \alpha_1.\varepsilon_1.(R^l_x - R^f_{jx}).|R^l_x - R^f_{jx}|^{-1}\}), -\xi\,(y - \{R^f_{jy} + \alpha_2.\,\varepsilon_2.(R^l_y - R^f_{jy}).|R^l_y - R^f_{jy}|^{-1}\}) \quad (17.a)$$

$$F_{att}(x,y) = (-\xi\,(x - \{R^f_{jx} + \gamma_1.\,\tau_1.(R^l_x - R^f_{jx})\}), -\xi\,(y - \{R^f_{jy} + \gamma_2.\,\tau_2.(R^l_y - R^f_{jy})\}) \quad (17.b)$$

$$F_{att}(x,y) = (-\xi\,(x - \{R^a_{kx} + \beta_1.\cos(\frac{2\pi\varphi}{\psi})\}),\ -\xi\,(y - \{R^a_{ky} + \beta_2.\sin(\frac{2\pi\varphi}{\psi})\}) \quad (17.c)$$

The repulsive forces are always the same for both types of obstacles. The equations of the repulsive forces are expressed as:

$$Frepo(qi)=Frepox(qi), Frepoy(qi) \quad (18.a)$$

$$Frepr(qi)=Freprx(qi), Frepry(qi) \quad (18.b)$$

$$Frepx(qi)= Frepox(qi)+ Freprx(qi) \quad (18.c)$$

$$Frepy(qi)= Frepoy(qi)+ Frepry(qi) \quad (18.d)$$

From the (18.c)-(18.d), if $d(qi,\upsilon oj) < \rho o$ the distance between the robot and the obstacle is less than the safety distance ρo and $d(qi,\upsilon rk) < \rho r$ the distance between the robot and the other robots is less than the safety distance ρr, then the equations of the repulsive forces will be expressed as:

$$Frepx(qi)= -\sum_{j=0}^{M} Krep\left(\frac{1}{d(qi,\upsilon xoj)} - \frac{1}{\rho o}\right)\frac{(\upsilon xoj\text{-}xi)}{d(qi,\upsilon xoj)^3} - \sum_{k=0}^{N} Krep\left(\frac{1}{d(qi,\upsilon xrk)} - \frac{1}{\rho r}\right)\frac{(\upsilon xrk\text{-}xi)}{d(qi,\upsilon xrk)^3} \quad (19.a)$$

$$Frepy(qi)= -\sum_{j=0}^{M} Krep\left(\frac{1}{d(qi,\upsilon yoj)} - \frac{1}{\rho o}\right)\frac{(\upsilon yoj\text{-}yi)}{d(qi,\upsilon yoj)^3} - \sum_{k=0}^{N} Krep\left(\frac{1}{d(qi,\upsilon yrk)} - \frac{1}{\rho r}\right)\frac{(\upsilon yrk\text{-}yi)}{d(qi,\upsilon yrk)^3} \quad (19.b)$$

Using this strategy, the robot will adjust its path, and avoid going to the virtual target's position that was calculated, if this position would cause a collision between the robot and moving or static obstacles. This new path ensures that the robot reaches the target without hitting any obstacle. As shown in Figure 9, when the virtual target's position is within or close to the obstacle $d(qi,\upsilon oj)<\rho o$ or $d(qi,\upsilon rk)<\rho r$, the robot does not go to this position.

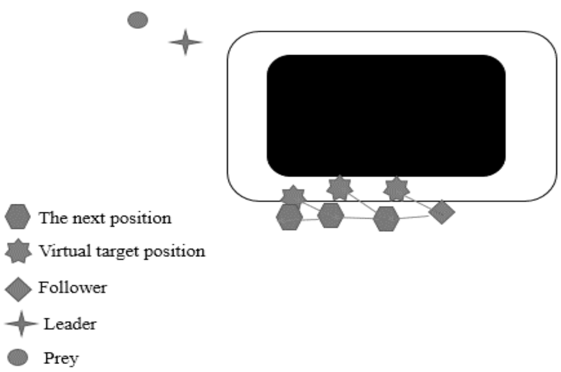

Figure 9. The robot adjusts its next position to avoid the obstacle

4. RESULTS AND DISCUSSION

In order to verify the effectiveness and the performance of the proposed strategy, the simulation experiment of a hunting process is given. This hunting process is performed by means of a set of robots, which will determine the path in order to capture the moving target. The target and the robots are within the same search space which contains several obstacles. The simulation software is MATLAB. In this experiment, we assume that the two-dimensional workspace 20×20 contains several robots that are moving all the time (Table 1), and several static obstacles (Table 2). The objective here is to use the hunting strategy to plan the path which the robots will follow in order to capture the target without colliding with obstacles. Therefore, each robot take off from the starting point towards the endpoint which is fundamentally dependent on the behavior of the target. The target updates its position randomly and unpredictably (Figure 10), where Px and Py are the coordinates of the target: Px (i+1) = Px (i) + µ1 and Py (i+1) = Py (i) + µ2. µ1 and µ2 are random numbers, where - 1 ≤ µ1,µ2 ≤ 1.

We start the simulation by initializing randomly the location of the robots and the target as shown in Table 1 and Figure 11. The multi robot system is consisted of four mobile robots, one of them is the antagonist (∇) and the three others (X) are the followers and the leader, and in addition to the target (O). It is worth noting that the process of capturing the target only takes place when at least three robots encircle the prey.

In this random situation, the objective of the robots is planning the path to the target which is always in motion at random velocities. The path of the robots has to be admissible. These robots have to avoid collision between them and with the obstacles. The simulation results are shown in the Figures 12 and 13. The robots hunt the target at the position [12.3395, 11.0244] as shown in Figure 14. The hunting process is successfully performed.

Table 1. The initial positions of the robots and target

Parameter	Initial
Target	9 ; 11.5
Robot 1	17.0606 ; 12.4411
Robot 2	7.0190 ; 10.2650
Robot 3	8.0362 ; 1.5193
Robot 4	4.7983 ; 2.4664

Table 2. The initial positions and the radius of obstacles

Parameter	Position	Radius
Obstacle 1	(10 , 10)	0.7
Obstacle 2	(8 , 12)	0.7
Obstacle 3	(7 , 9)	0.7
Obstacle 4	(8 , 14)	0.7
Obstacle 5	(15 , 14)	0.7
Obstacle 6	(11 , 14)	0.7

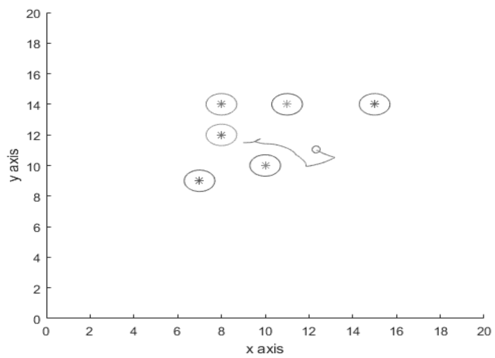

Figure 10. The path of the target during hunting process

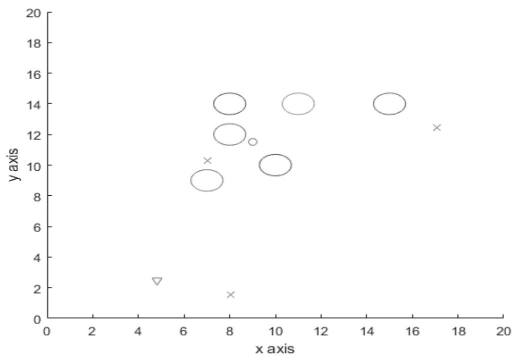

Figure 11. Initial position of the robot and the target in the workspace

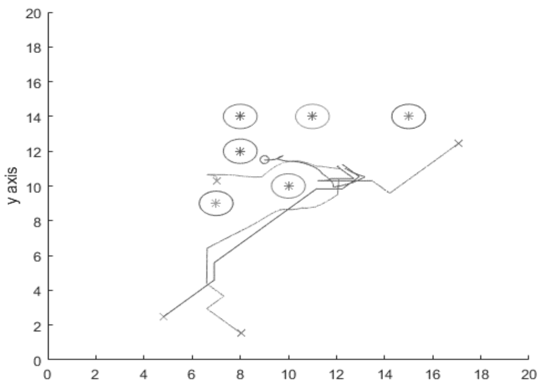

Figure 12. The path of each robot during hunting process

As shown in Figure 13, the evolution of the best cost function depends on the closest wolf to the target. This function increases and decreases in different intervals, and that is mainly due to the dynamic of the target, which sometimes moves away from the hunter robots. In the fifth iteration, the function decreased significantly, as a result of the target moving faster and encountering another hunter who became much closer to the prey. This sharp decrease, which helps to catch the prey quickly, is a result of the well-distributed hunters in addition to the presence of the antagonist. The value of the best cost function tends towards zero on the 16th iterations, where the robots hunt the target.

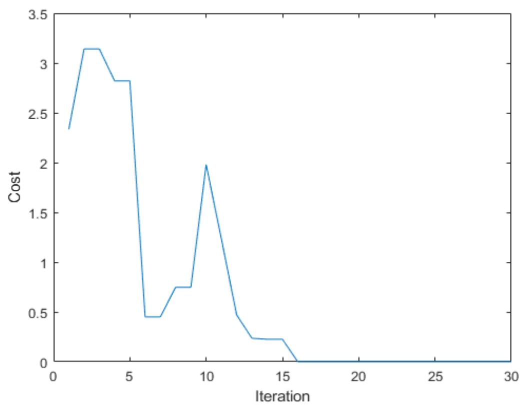

Figure 13. Best cost function progression

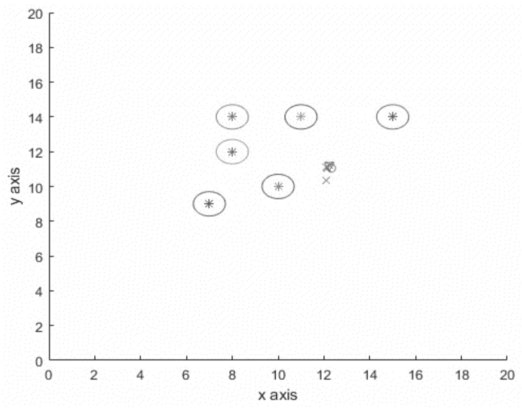

Figure 14. Robots hunt successfully the target

The trajectories of each robot are secured and eligible towards the target as shown in the Figures 15 to 18. The coordination and cooperation between the robots are well organized. As shown in the Figure 19, the first robot is constantly approaching the target, and the noticeable decrease between the first and sixth repetition is because the target and the robot take opposite paths, which makes them approach each other, and in the 16th iteration it catches the prey. In Figure 20, the second robot makes a safe distance between it and the obstacles (ob1–ob6) in order to make the path more secure. We used large numbers in the x axis to make the progression of the distance in y axis more accurate. The robot's path is usually not straight, as random values used in the equations of motion control affect its velocity.

Based on these experiments, the hunter robots were able to search and chase the target without deviating from their main path despite the presence of some small biases in their behaviors. This strategy ensured the continuous pursuit of the prey despite its random behavior and the presence of obstacles. The results of the simulation experiments in the hunting strategy based on WSA and APF show that the proposed approach can satisfy the cooperative hunting task for a dynamic target by multiple hunter robots.

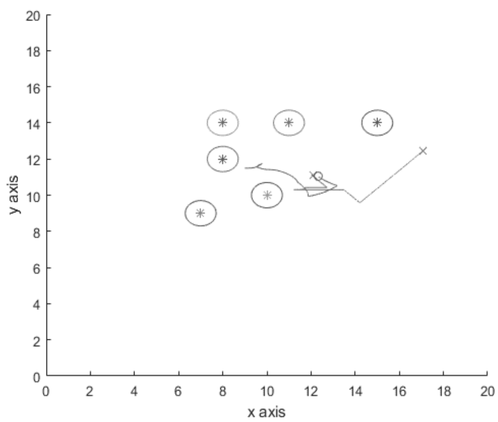

Figure 15. The path of the robot 1 during hunting process

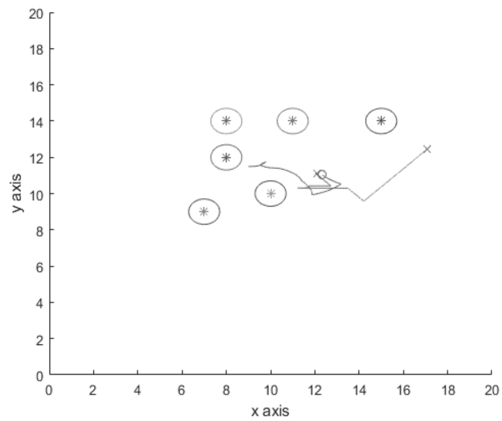

Figure 16. The path of the robot 2 during hunting process

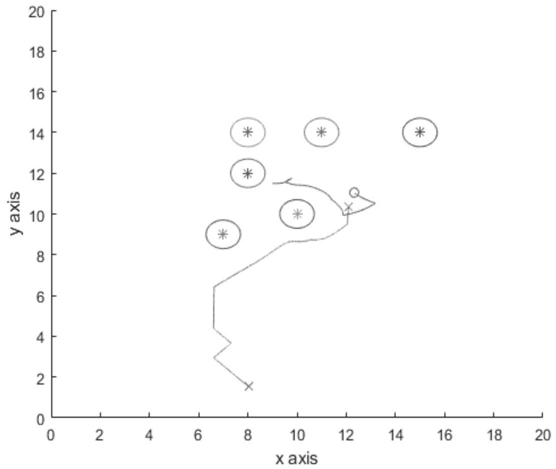

Figure 17. The path of the robot 3 during hunting process

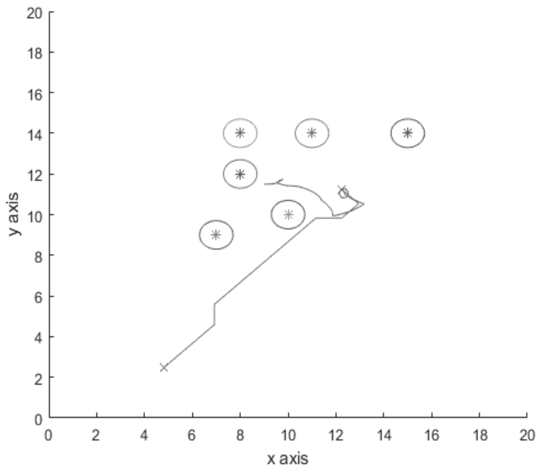

Figure 18. The path of the robot 4 during hunting process

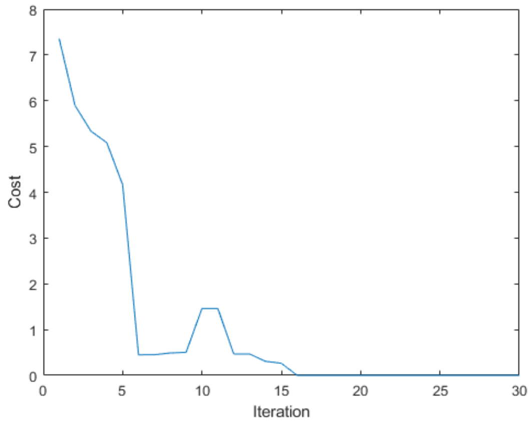

Figure 19. Cost function progression for robot 1

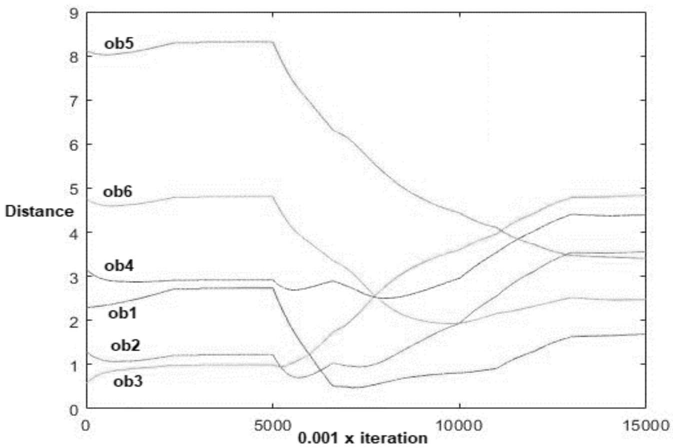

Figure 20. Distances between the robot 2 and the obstacles

5. CONCLUSION

In this paper, we proposed a new cooperative hunting strategy for a multi-robot system based on wolf swarm algorithm and artificial potential field. This strategy is dedicated to hunt targets with random behavior. This cooperative hunting is performed by several robots that are divided into three roles: the leader, the follower, and the antagonist. To avoid static or dynamic obstacles and reach the hunting point rapidly, the artificial potential field algorithm is used.

In complex search spaces, when the target is moving in random and unpredictable patterns, the problem of continually escaping prey from hunter robots may occur in a multi-robot system. Moreover, they are vulnerable to disturbances and loss of prey trace due to random prey behavior and the presence of obstacles in the search space that can cause the hunters to lose formation. In this paper, we study how to improve the hunting process by increasing the decision control to make the robots avoid collisions with obstacles while keeping the hunting process. The algorithm proposed in this paper is verified by numerical simulation. From the results of the simulation, robots can find and capture prey within the search space faster and without letting the prey escape thanks to the cognitive behavior of hunters. In order to make the behavior and characteristics of the robot flexible and effective during the hunting process, we added the artificial potential field to cooperative hunting strategy. The simulation showed that the APF leads to an optimal safe path towards the target. However, in this paper, the hunting process is performed only for a single target. Therefore, in our future research, we will modify the proposed hunting strategy to make it applicable for hunting multi-dynamic targets.

REFERENCES

[1] Z. Feng, G. Hu, Y. Sun, and J. Soon, "An overview of collaborative robotic manipulation in multi-robot systems," *Annual Reviews in Control*, vol. 49, pp. 113-127, 2020, doi: 10.1016/j.arcontrol.2020.02.002.

[2] Y. Rizk, M. Awad, and E. W. Tunstel, "Cooperative heterogeneous multi-robot systems: A survey," *ACM Computing Surveys* (CSUR), vol. 52, no 2, 2019, pp 1-31, doi: 10.1145/3303848.

[3] E. Olcay, F. Schuhmann, and B. Lohmann, "Collective navigation of a multi-robot system in an unknown environment," *Robotics and Autonomous Systems*, vol. 132, p. 103604, 2020, doi: 10.1016/j.robot.2020.103604.

[4] M. Dadvar, S. Moazami, H. R Myler, and H. Zargarzadeh, "Multiagent Task Allocation in Complementary Teams: A Hunter-and-Gatherer Approach," *Complexity*, vol. 2020, 2020, p. 1752571, doi: 10.1155/2020/1752571.

[5] O. Hamed and M. Hamlich, "Improvised multi-robot cooperation strategy for hunting a dynamic target," *2020 International Symposium on Advanced Electrical and Communication Technologies (ISAECT)*, vol. 6, 2020, pp. 1-4, doi: 10.1109/ISAECT50560.2020.9523684.

[6] X. Luan and Y. Sun, "Research on Coperative Encirclement Strategy of Multiple Underwater Robots based on Wolf Swarm Algorithm," *J. Phys: Conf. Ser*, vol. 1570, no. 1, p. 012017, 2020, doi: 10.1088/1742-6596/1570/1/012017.

[7] H. Liang, Y. Fu, F. Kang, J. Gao, and N. Qiang, "A Behavior-Driven Coordination Control Framework for Target Hunting by UUV Intelligent Swarm," *IEEE Access*, vol. 8, pp. 4838-4859, 2020, doi: 10.1109/ACCESS.2019.2962728.

[8] G. Rossides, B. Metcalfe, and A. Hunter, "Particle Swarm Optimization—An Adaptation for the Control of Robotic Swarms," *Robotics*, vol. 10, no. 02, pp. 58, 2021, doi: 10.3390/robotics10020.

[9] Á. Madridano, A. Al-Kaff, D. Martín, and A. de la Escalera, "Trajectory planning for multi-robot systems: Methods and applications," *Expert Systems with Applications*, vol. 173, pp 114660, 2021, doi: 10.1016/j.eswa.2021.114660.

[10] N. Alexandros, H. A. Shahab, and D. V. Dimos, "Scalable time-constrained planning of multi-robot systems," *Autonomous Robots*, vol. 44, no 8, p. 1451-1467, 2020, doi: 10.1007/s10514-020-09937-6.

[11] A. N. Jati, R. E. Saputra, M. G. Nurcahyadi, and N. T. A. Ghifary, "A multi-robot system coordination design and analysis on wall follower robot group," *Int. J. Electr. Comput. Eng. (IJECE),* vol. 8, no. 6, pp. 5098, Dec. 2018, doi: 10.11591/ijece.v8i6.pp5098-5106.

[12] M. S. Gharajeh and H. B. Jond, "Hybrid Global Positioning System-Adaptive Neuro-Fuzzy Inference System based autonomous mobile robot navigation," *Robotics and Autonomous Systems*, vol. 134, pp. 103669, 2020, doi: 10.1016/j.robot.2020.103669.

[13] M. H. M. Nor, Z. H. Ismail, and M. A. Ahmad, "Broadcast control of multi-robot systems with norm-limited update vector," *International Journal of Advanced Robotic Systems*, vol. 17, p. 1729881420945958, July 2020, doi: 10.1177/1729881420945958.

[14] Y. Chen, U. Rosolia, and A. D. Ames, "Decentralized Task and Path Planning for Multi-Robot Systems," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4337-4344, July 2021, doi: 10.1109/LRA.2021.3068103.

[15] Y. Msala, M. Hamlich, and A. Mouchtachi, "A new Robust Heterogeneous Multi-Robot Approach Based on Cloud for Task Allocation," *2019 5th International Conference on Optimization and Applications (ICOA)*, pp. 1-4, 2019, doi: 10.1109/ICOA.2019.8727618.

[16] H. Zhang, Y. Wu, and Y. Cen, "Multi-robot Dynamic Pursuit Scheme Based on Behavior-Merging and Task Decision-Making Technology," *2009 WRI World Congress on Computer Science and Information Engineering*, vol. 4, 2009, pp. 575-580, doi: 10.1109/CSIE.2009.670.

[17] J. T. Feddema, C. Lewis, and D. A. Schoenwald, "Decentralized control of cooperative robotic vehicles: theory and application," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 852-864, Oct. 2002, doi: 10.1109/TRA.2002.803466.

[18] X. Lin and F. Guo, "A Method of Multi-USV Hunting Based on Extended Kalman Filter" *2020 39th Chinese Control Conference (CCC)*, vol. 2020, 2020, pp. 100-105, doi: 10.23919/CCC50068.2020.9189551.

[19] Y. Wang, G. He, Y. Ma, G. Kong, and J. Gong, "Research on multi-robots self-organizing cooperative pursuit algorithm based on Voronoi graph," *2020 39th Chinese Control Conference (CCC)*, 2020, pp. 3840-3844, doi: 10.23919/CCC50068.2020.9188368.

[20] E. O. Rivas, A. R. Liñan, and L. T. Treviño, "Flock of Robots with Self-Cooperation for Prey-Predator Task," *Journal of Intelligent & Robotic Systems*, vol. 101, no 2, p. 1-16, 2021, doi: 10.1007/s10846-020-01283-0.

[21] C. Yu, Y. Dong, Y. Li, and Y. Chen, "Distributed multi-agent deep reinforcement learning for cooperative multi-robot pursuit," *The Journal of Engineering*, pp. 499–504, 2020, doi: 10.1049/joe.2019.1200.

[22] E. Camci and E. Kayacan "Game of drones: UAV pursuit-evasion game with type-2 fuzzy logic controllers tuned by reinforcementlearning," *IEEE Int. Conf. on Fuzzy Systems*, vol. 2016, pp. 618-625, doi: 10.1109/FUZZIEEE.2016.7737744.

[23] C. Yu *et al.,* "Distributed multiagent coordinated learning for autonomous driving in highways based on dynamic coordination graphs", *IEEE Trans. Intell. Transp. Syst*, vol. 21, no. 2, pp. 735–748, 2019, doi: 10.1109/ TITS.2019.2893683.

[24] X. Cao and L. Guo, “A leader–follower formation control approach for target hunting by multiple autonomous underwater vehicle in three-dimensional underwater environments,” *Int. J. Adv. Robot. Syst*. 2019, vol. 16, no. 04, p. 1729881419870664, doi: 10.1177/1729881419870664.
[25] X. Cao and X. Xu, “Hunting Algorithm for Multi-AUV Based on Dynamic Prediction of Target Trajectory in 3D Underwater Environment,” *IEEE Access*, vol. 8, pp. 138529-138538, 2020, doi: 10.1109/ACCESS.2020.3013032.
[26] P. Agrawal and H. Agrawal, “Adaptive Algorithm Design for Cooperative Hunting in Multi-Robots,” *International Journal of Intelligent Systems and Applications*, vol. 10, no 12, p. 47-55, 2018, doi: 10.5815/ijisa.2018.12.05.
[27] O. Hamed and M. Hamlich, “Improvised multi-robot cooperation strategy for hunting a dynamic target,” *IOT, EAI*, vol. 6, no 24, p. e5, 2021, doi: 10.4108/eai.8-2-2021.168691.
[28] O. Khatib, “Real-time obstacle avoidance for manipulators and mobile robots,” *Proceedings. 1985 IEEE Int Conf on Robotics and Automation*, vol. 02, 1985, pp. 500-505, doi: 10.1109/ROBOT.1985.1087247.
[29] H. Sang, Y. You, X. Sun, Y. Zhou, and F. Liu, “The hybrid path planning algorithm based on improved A* and artificial potential field for unmanned surface vehicle formations,” *Ocean Engineering*, vol. 223, pp. 108709, 2021, doi: 10.1016/j.oceaneng.2021.108709.

BIOGRAPHIES OF AUTHORS

Oussama Hamed is a PhD student in Robotics and Artificial Intelligence at ENSAM Casablanca University Hassan II. His research interests are Robotics, Reinforcement Learning, Electrical Systems and Computer Vision. He can be contacted at email: oussama.hamed-etu@etu.univh2c.ma.

Mohamed Hamlich is a Professor-Researcher at ENSAM (UH2C), Casablanca, MOROCCO since 2014.Currently, he is director of the research laboratory: "Complex Cyber Physical Systems". He is the author of several research papers in the fields of Machine Learning, Robotics and Electrical Energy. He can be contacted at email: moha.hamlich@gmail.com.

Mohamed Ennaji is a Professor at ENSAM Casablanca. In 2015, he obtained his doctoral thesis in Energétiques énergies renouvelables from Hassan II University in Casablanca. His areas of research interests include Modeling and Simulation, Mechatronics, Automobile Engineering. He can be contacted at email: ennaji.moh@gmail.com.