CHAOS Theory and Applications in Applied Sciences and Engineering

e-ISSN: 2687-4539
RESEARCH ARTICLE
Vol.6 / No.2 / 2024 / pp.152-158
Special Issue (EDIESCA 2023)
https://doi.org/10.51537/chaos.1376145

Flocking Behavior of Boids Driven by Hyperchaotic MACM System

Medina Galindo Ana C.[α,1], **Cardoza Avendaño Liliana**[α,2], **López Gutiérrez Rosa Martha**[α,3] **and Cruz Hernández Cesar**[*,4]

[α]Engineering, Architecture and Design Faculty, Autonomous University of Baja California (UABC), Ensenada 22860, Mexico, [*]Electronics and Telecommunication Department, Scientific Research and Advanced Studies Center of Ensenada (CICESE), Ensenada 22860, BC, Mexico.

ABSTRACT

In the present work a detailed study is presented, on the design, programming, and investigation of the behavior of flocking (Movement type flock), through the model of BOIDS, for its acronym in English "Bird Oid Object" (Object type bird), which was devised by Craig Reynolds in 1986. This complex flocking behavior that occurs arises from the interaction of simple local rules, in which complexity and sensitivity to initial conditions are present. A measure of chaotic compound will be introduced to the algorithm by means of a new four-dimensional autonomous hyperchaotic system based on the 3D Méndez-Arellano-Cruz-Martínez (MACM) system. The measures proposed herein, therefore, may have the potential to predict, control, and exemplify the behavior of group intelligence study systems that occur in nature, allowing the implementation of these systems in groups of robots through the implementation of hyperchaotic trajectories in the future, to obtain greater speed and efficiency, obstacle and collisions avoidance in their flights.

KEYWORDS

Flocking behavior
Boids
Hyperchaotic
MACM system

INTRODUCTION

The grouping of animals that occurs frequently in nature between different types of species; such as the behavior of bee swarms Karaboga *et al.* (2005), schools of fish (Pourpanah *et al.* 2023), flocks of birds (Duman *et al.* 2012), among others, has been an inspiration for different research groups in recent years, taking this behavior as an approach to solve very complex problems. By studying and simulating how animals behave, scientists hope to create powerful computational models that can solve challenging problems, optimize processes, and make decisions in ways inspired by the efficient and adaptive strategies found in nature.

Multi-agent-based simulation (MBS) is a valuable technique used to model flocking behavior, where collective behavior emerges from individual interactions. It helps understand complex interactions at a larger scale, which are often hard to predict, comprehend, and simulate. This difficulty arises because of the non-linear relationship between micro (individual agent) and macroscopic (overall group) properties. Small changes in an agent's environment or rules can result in vastly different outcomes in the simulation. Due to these complexities, MBS becomes a powerful tool for studying and analyzing emergent phenomena, providing insights into systems where traditional approaches might fall short.

One of the most commonly used methods to simulate the emergent behavior that occurs in different groups of animals in nature, are the so-called boids, devised by Craig Reynolds in 1987 (Reynolds 1987). This model was the first published way of simulating a fairly realistic flock simulation from an algorithm. It was developed into an artificial life program where each individual in the flock is called an agent, which has its own position, speed, and orientation, exhibiting complex flock behavior that arises from the interaction of three simple local rules:

- **Separation**. An agent must avoid collisions with other nearby agents. To avoid collisions, a separation factor is added. An agent will keep a certain distance from all other agents in his neighborhood. If the agent finds another agent too close, it will try to move away from them to avoid collisions.
- **Cohesion**. An agent must stick to the group or flock. To ensure this, a cohesive factor is added. The agent will move towards the average position of the neighboring agents. When

Manuscript received: 15 October 2023,
Revised: 23 February 2024,
Accepted: 9 March 2024.

[1]ana.medina.galindo@uabc.edu.mx
[2]lcardoza@uabc.edu.mx
[3]roslopez@uabc.edu.mx
[4]ccruz@cicese.mx (**Corresponding author**)

other agents are within the neighborhood, the agent will try to move to the midpoint of all the others.

- **Alignment**. The alignment rule is obtained by calculating a force directed at the average of the velocities of the neighbors.

The three rules of separation, alignment, and cohesion will result in different vector forces that must act on the agent to which they belong. Since animals cannot turn instantly in real life, a resultant force is calculated by adding the three effective vectors. The three forces can be applied differently by normalizing the individual vectors and then multiplying them with appropriate weights. Therefore, the specific behavior of each agent could be induced by the slight alteration of an aspect of the so-called flocking.

The resulting force is added to the velocity vector of the agent. Then the speed is normalized or limited to a maximum allowed speed. Finally, the agent's velocity is added to its position vector resulting in motion. The previous velocity can also be used as the base vector for the resultant, giving each agent a flow motion closer to what occurs in nature. Currently there are few works that can be found about the study of boids applied to chaotic systems, however there is great potential to use the dynamics of boids in relation to this type of systems, since adding rules can present behaviors that can be considered emergent, as was presented in previous work (Itoh and Chua 2007).

In this paper, we propose implement a new rules for the behavior of the boids, first with the introduction of a hyper MACM chaotic system (Méndez-Ramírez *et al.* 2021), where the boids are led in this trajectory. Emerging behavior was investigated through simulations implemented in MATLAB. After the analysis of the movement of each boid as well as the chaotic component of their trajectories, finally the conclusions obtained about the control of the boids are presented.

BASIC ANALYSIS AND MODELING OF BOIDS

We consider a network of N identical nodes that will be called *boids*, each node is considered like basic element with behavior depending on the nature of the network, which can be modeled by a set of nonlinear autonomous differential equations, with each boid being an n-dimensional dynamical system. The state equations of the entire network of boids are described as follows,

$$\dot{\mathbf{u}}_i = f_i(\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_n), \quad i = 1, 2, ..., n \tag{1}$$

where $\mathbf{u}_i = (u_1, u_2, ..., u_n)^T \in \mathbb{R}^n$ is a state vector of boid i, and $\mathbf{f}(\mathbf{u}) = (f_1(\mathbf{u}), f_2(\mathbf{u}), ... f_n(\mathbf{u}))$ is a nonlinear vector function of $\mathbf{u}$. Given initial state ${u_i}^{\alpha}(0)$ at $t = 0$, the state ${u_i}^{\alpha}$ of each isolated boid B_α is assumed to envolve for all $t \geq 0$ via state equations,

$$\dot{u}_i^{\alpha} = f_i(u_1, u_2, ..., u_n), \quad i = 1, 2, ..., n. \tag{2}$$

For ease of modeling we will assume that all boids are identical, they are only influenced in their trajectories by those nearby neighboring boids which are located on a sphere described as S_α, if a boid moves in a random trajectory but in its trajectory the boid gets close to another boid they will be coupled as long as they are positioned at any distance inside the sphere S_α with radius ϵ as shown in Figure 1 a),

$$S_\alpha(\epsilon, t) = \left\{ B_\beta : r_{\alpha,\beta} \triangleq \sqrt{\sum_{i=1}^{n} ({u_i}^{\alpha}(t) - {u_i}^{\beta}(t))^2} \leq \epsilon \right\}, \tag{3}$$

at time t, where $r_{\alpha,\beta}$ indicates the distance between the boids B_α and B_β. We will usually delete ϵ and t from $S_\alpha(\epsilon, t)$ and simply write S_α to prevent confusions, see Figure 1 a.

Then, the dynamics of the nonlinear chaotic network of the locally coupled boids defined by

$$\dot{\mathbf{u}}_i^{\alpha} = f_i(u_1^{\alpha}, u_2^{\alpha}, ..., u_n^{\alpha}) + \sum_{B_\beta \in S_\alpha} D_i^{\beta} g_i(u_1^{\beta}, u_2^{\beta}, ..., u_n^{\beta}),$$
$$i = 1, 2, ..., n, \quad \alpha = 1, 2, ..., M \tag{4}$$

where D_i^{β} are coupling coefficients, and $\mathbf{g}(\mathbf{u}) = (g_1(\mathbf{u}), g_2(\mathbf{u}), ..., g_n(\mathbf{u}))$ is a nonlinear vector function of $\mathbf{u}$.

The dynamics of Equation (4) describes networks of boids with nonlinear behaviors, the number of boids belonging to S_α can change continuously as time t increases. Since at first, the boids can be found in random positions and speeds within some area, and as time passes they can get closer to a certain distance within the sphere S_α, continuously changing the number of boids in S_α.

The behaviors that make up the flock model are expressed in terms of "close flockmates". While a boid is in motion, it does not require full knowledge of the position and speed of each boid in the entire herd, it only knows the information of a small subset of it. This subset is composed of what we call the expression "near flockmates", often used in boid modeling of steering behavior, which refers to the awareness each boid has of the bodies of other nearby boids, based on the distance between them. Thus the boid has a range of perception of the world in the shape of the sphere around it, described in Equation (3). When different boids are within a very close distance of each other the boids perception spheres can overlap, influencing each other's behavior depending on their rule parameters.

In this section, we will describe the implementation of the boid rules. Based on the Reynolds model (Reynolds 1987), a model was made a boid swarm model in *MATLAB* software, the boid model has 5 rules: separation, cohesion, alignment, edge avoidance, and hyperchaotic MACM attractor, described as follows.

Cohesion

The cohesion force has the objective of keeping the flock of boids together. This means that this force will drive each agent to move towards the average position of its nearest neighbors which is in the volume of the sphere $S_\alpha(\epsilon, t)$ of each boid, as shown in Figure 1 b). This is expressed mathematically in the Equation (5). Boid cohesion is calculated using two steps.

First, the central position of the nearest neighbors of each agent is calculated by,

$$\overline{u_i^{\alpha}}(t) = \frac{\sum\limits_{\beta \in S_\alpha} u_i^{\beta}(t)}{N_\alpha}, \tag{5}$$

where N_α indicates the number of nearby flockmates. Then the tendency of the boid to sail towards the visible flock center of density $\overline{u_i^{\alpha}}(t)$. Therefore the control dynamics is calculated as shown below,

$$\dot{u}_i^{\alpha} = f_i(u_1^{\alpha}, u_2^{\alpha}, ..., u_n^{\alpha}) + d_i^{\alpha}(\overline{u_i^{\alpha}} - u_i^{\alpha}), \tag{6}$$

where $d_i^{\alpha} > 0$.

There is a special case when there is no one around. The center of the nearby flockmates $\overline{u_i^{\alpha}(t)} = 0$. In this case, Equation (5) is not defined and the cohesion rule does not apply.

This principle encourages boids to stay close to their neighbors, leading to a sense of togetherness within the flock. By gravitating towards the average position of nearby companions, the cohesion rule promotes collective movement, enabling boids to exhibit coordinated behaviors without any centralized leadership. As a result, the flock maintains a cohesive structure.

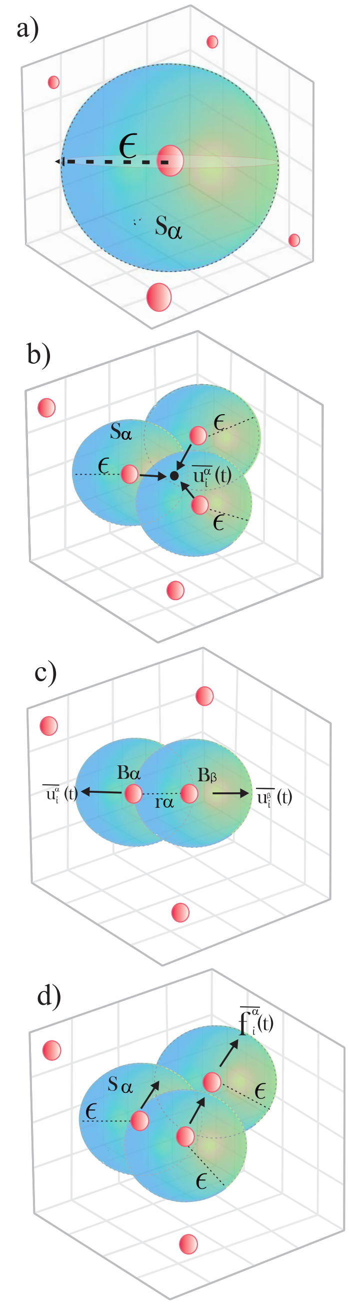

Figure 1 Graphic description of boids: a) Boid inside sphera S_α with radius ϵ. Rules governing the movement of boids, b) Cohesion, c) Separation, d) Alignment.

Separation

Separation force is the complementary force to the cohesion force as shown in Figure 1 c). Each member of a flock tends to avoid collision with its nearby neighbors. This tendency is called separation or collision avoidance. In the case where the distance $r\alpha$, between the boids B_α and B_β becomes less than $\delta > 0$, the boids will tend to disperse from the center of nearby flockmates. This is calculated by the following dynamics:

$$\left.\begin{aligned} \dot{u}_i^\alpha &= f_i(u_1^\alpha, u_2^\alpha, ..., u_n^\alpha) + e_i^\alpha(\overline{u_i^\alpha} - u_i^\alpha), \\ \dot{u}_i^\beta &= f_i(u_1^\beta, u_2^\beta, ..., u_n^\beta) + e_i^\beta(\overline{u_i^\beta} - u_i^\beta), \end{aligned}\right\} \tag{7}$$

where $e_i^\alpha, \quad e_i^\beta > 0$.

The separation rule in boids algorithm is a fundamental principle for simulating flocking behavior. Each boid maintains a minimum distance from its nearby flockmates, preventing collisions and promoting spacing within the group. By avoiding crowding, boids create a sense of personal space, enhancing overall flock stability and preventing individual boids from getting too close to each other.

Alignment

Alignment is the process by which each boid attempts to match its velocity and direction with that of its nearby flockmates as shown in Figure 1 d). It promotes the cohesive and coordinated movement of the group, leading to the emergence of flocking behavior. This rule is essential because it enables the boids to maintain alignment and unity without relying on any centralized control or explicit communication between individuals.

The alignment rule is implemented as follows: each boid examines its surroundings and identifies its nearby neighbors within a certain perception radius $S_\alpha(\epsilon, t)$, dictating how far it can "see" other flockmates. The boid then calculates the average velocity of its neighbors, which represents the average direction in which the neighboring boids are moving. The average velocity of nearby flockmates is defined by

$$\overline{f_i^\alpha} = \frac{\sum\limits_{\beta \in S_\alpha} f_i(u_1^\beta, u_2^\beta, ..., u_n^\beta)}{N_\alpha}. \tag{8}$$

To align with the flock, the boid adjusts its own velocity to match the computed average velocity of its neighbors. However, it doesn't do this instantaneously; instead, it gradually changes its velocity over time to create a smooth and realistic alignment process. This gradual adjustment prevents sudden changes in direction that could disrupt the cohesion of the flock.

IMPLEMENTATION OF THE HYPERCHAOTIC MACM ATTRACTOR TO THE NETWORK OF BOIDS

The behavior of the boids is given by Reynolds (1987), and each of the behavioral rules is expressed as a vector. These rules are sorted by priority and added to an accumulator of the boids. This continues until the sum of the accumulated magnitudes increases the maximum acceleration value. In this work, the value of a new vector given by the new 4D hyperchaotic MACM system in the network of boids is prioritized, so that the boid has the priority to follow the hyperchaotic attractor trajectory, see Figure 2.

The implementation of a new rule consists of placing a new 4D hyperchaotic MACM system (Méndez-Ramírez *et al.* 2021), as a new rule in the behavior of the boids. This MACM attractor is obtained by modifying the 3D MACM system inspired by previous works (Méndez-Ramírez *et al.* 2017). A hyperchaotic system is a mathematical concept that extends the idea of a chaotic system. It has more than one positive Lyapunov exponent, see Table 1, this indicates greater complexity in its dynamic behavior in the projection of the phase space in the plane (Rajagopal *et al.* 2018). To create a hyperchaotic system, k chaotic systems can be coupled, resulting in an attractor with n positive Lyapunov exponents. This

Table 1 **Analysis of stability of equilibrium points for a new hyperchaotic MACM system based on the Lyapunov exponents.**

Point	Eigenvalues	Stability
P_0	$\lambda_1 = -0.5247$	λ_1,λ_2,λ_4< 0, and $\lambda_3 > 0$
	$\lambda_2 = -1$	
	$\lambda_3 = 4.5361$	unstable saddle point
	$\lambda_4 = -6.5113$	
P_{1-4}	$\lambda_1 = -0.4939$	λ_1,$\lambda_4 < 0$, and the real part
	$\lambda_2 = 0.94767 - 3.4506i$	λ_2,$\lambda_3 > 0$
	$\lambda_3 = 0.94767 + 3.4506i$	unstable saddle point
	$\lambda_4 = -4.9014$	
P_{5-6}	$\lambda_1 = -0.4918$	λ_1,$\lambda_4 < 0$, and $\lambda_2, \lambda_3 > 0$
	$\lambda_2 = 1.5384$	
	$\lambda_3 = 2.7915$	unstable saddle point
	$\lambda_4 = -7.3381$	
P_{7-8}	$\lambda_1 = 0$	λ_2< 0, and the real part
	$\lambda_2 = -1$	λ_2,$\lambda_3 < 0$
	$\lambda_3 = 1.25 + 0.9682i$	Spiral stable point
	$\lambda_4 = -1.25 - 0.9682i$	

coupling causes the dimension of the attractor to increase, leading to a transition from chaos to hyperchaos. As this transition occurs, the second Lyapunov exponent increases continuously (Kapitaniak *et al.* 2000), highlighting the greater complexity and richness of the system in its behavior compared to a normal chaotic system.

The dynamics of the hyperchaotic MACM system used is defined as follows (Méndez-Ramírez *et al.* 2021):

$$\begin{aligned} \dot{x} &= -ax - byz, \\ \dot{y} &= -x + cy + cw, \\ \dot{z} &= d - y^2 - z, \\ \dot{w} &= x - w. \end{aligned} \tag{9}$$

The given system in the Equation (9) is a mathematical representation with ten terms, including two quadratic nonlinearities. It also involves four parameters, denoted as a, b, c, and d, which must satisfy certain conditions: $a, b, c, d \in \mathbb{R}^+$ and $c < a + 2$. In this context, b and d are referred to as the bifurcation parameters, which influence the system's behavior. When the specific values $a = 2, b = 2, c = 0.5$, and $d = 14.5$ are used in the MACM system exhibits hyperchaotic behavior.

ALGORITHM IMPLEMENTATION HIERARCHY

The algorithm for the simulation of the boids was realized in MATLAB software.

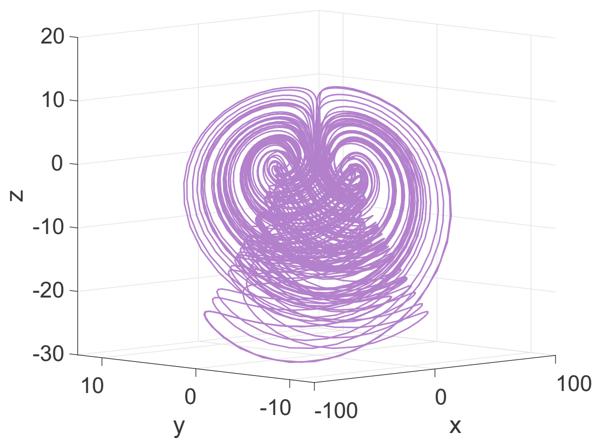

Figure 2 Hyperchaotic attractor of MACM system (9). Phase space x versus y versus z.

A boid can have conflicting requests as long as the matching algorithm is applied. The behavior is simply the result of the interaction of the aforementioned rules. For example, if two boids are moving in such a way that they are getting closer, with different speeds, the cohesion priority could override the spacing rule, since the boid cohesion request is opposite to the spacing rule, and therefore they could overlap, cancel directions, the boid could make only a small turn and crash into another boid. The highest priority should be to avoid collisions between the boids and the cohesion rule. Therefore, the behavior of the boids is modeled using a rule order priority as shown in Table 2.

Table 2 **Rule Order Priority.**

Priority	Order	Rule
High	1	Avoid edge
High	2	Cohesion
Medium	3	Separation
Low	4	Alignment
Low	5	MACM attractor

NUMERICAL SIMULATION RESULTS

This section shows the results obtained from the numerical simulation.

Numerical simulations for two boids

Figures 3 and 4 show the modeling of two boids following the trajectory of the hyperchaotic MACM attractor, Figure 3 show the projections of the phase space in the planes. Figure 4 shows the temporary states x, y, z, and w. Both figures show two boids modeled with the strange attractor of Equation (9) system by using the initial conditions $x(0) = 0.5$, $y(0) = 0$, $z(0) = -5$, $w(0) = 0.51$, the parameter values $a = 2$, $b = 2$, $c = 0.5$, and $d = 14.5$. The value of 80 has been added to each point of the solution of the Equation (9) system, because the attractor is located from -80 to 80 on the x axis, from -13 to 13 on the y axis, -30 to 13 on the z axis approximately.

Computer simulations use the following parameters.

Table 3 MACM´s oscillator.

Rule	Parameter
Flock centering	$d_i^{\alpha} = 5$
Velocity matching	$R_{max} = 2$
Collision avoidance	$\delta = 1.5$

Figures 3 and 4 show the trajectories followed by two boids, their cohesive movement as a flock. The distance between the two boids can be seen in Figure 5, it is shown that for 1000 iterations, the maximum distance does not exceed the value of 8, proving that the cohesion rule holds along the trajectory of the two boids.

The value of 80 has been added to each point of the solution of the hyperchaotic MACM system, because the attractor is located from -80 to 80 on the x axis, from -13 to 13 on the y axis, -30 to 13 on the the z axis approximately, to facilitate the implementation in robot trajectories in the future.

The synchronization behavior of the two boids is presented in Figure 6. However, it is important to note that exact synchronization is not achieved, leading to a certain thickness in the Lissajous figures. A thinner line would indicate perfect timing, but it can also lead to potential collisions.

In Figure 6 in particular, when considering the MACM oscillators, the two boids maintain a close distance of approximately $\delta = 1.5$.

Error analysis in trajectories of 2 boids

The separation measure was obtained in the trajectories of two boids, concerning the desired trajectory of the hyperchaotic MACM attractor employing the Root Mean Square Error, commonly referred to as RMSE, which is a statistical measure used to quantify the average magnitude of the error between predicted values and actual values. It is frequently employed to evaluate the precision of a predictive model. RMSE calculates the mean squared difference between forecasted numbers and subsequently observed numbers. the RMSE is defined by:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{9}$$

where n represents the total number of data points or observations being considered, y_i represents the actual observed values, y represents the predicted values from the model. An RMSE = 7.54 was obtained for boid 1 and an RMSE = 8.56 for boid 2.

It is observed that there is a difference between the trajectory of the hyperchaotic MACM attractor and the trajectory of the boids, this is because of the order of priority shown in Table 2. is implemented so that the boids adjust their speed and position to the rules that define the boids, this implies that the distance between the trajectory of the boids and the trajectory of the MACM chaotic attractor will increase for certain coordinates, despite this, the ability of the boids to drive along the trajectory of the chaotic attractor is observed in Figure 3.

It is worth mentioning that RMSE gives greater weight to larger variations, as it squares off the differences before averaging. Consequently, larger variations between the position of the boids concerning the hyperchaotic MACM attractor trajectory have a greater impact on the RMSE than smaller gaps.

It is observed that the distance maintained between the trajectories of the two boids is less than the distance between the trajectories of the boids and the trajectory of the hyperchaotic attractor, this is because the rules of behavior are imposed, and it is the objective in this studio. If the boids give priority to following the trajectory of the hyperchaotic attractor, collisions could occur between them since the separation rule would be secondary. The scenario could also arise that the distance between them was greater than the parameter implemented in the cohesion rule and the boids would no longer remain together in the trajectory since they would not recognize each other as close neighbors.

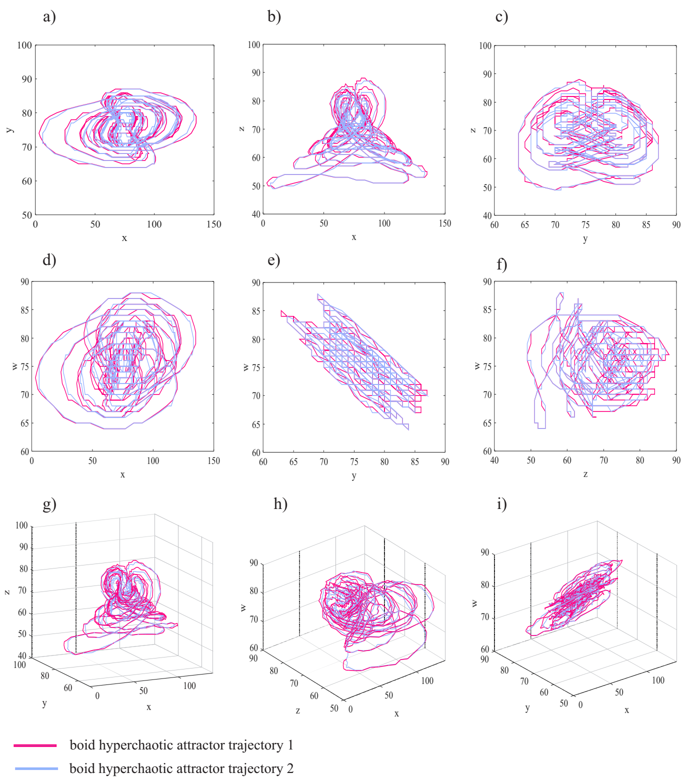

Figure 3 Projections of the phase space of flocking of two boids driven by hyperchaotic MACM attractor. a) (x,y), b) (x,z), c) (y,z), d) (x,w), e) (y,w), f) (z,w), g) (x,y,z), h) (x,z,w), and i) (x,y,w).

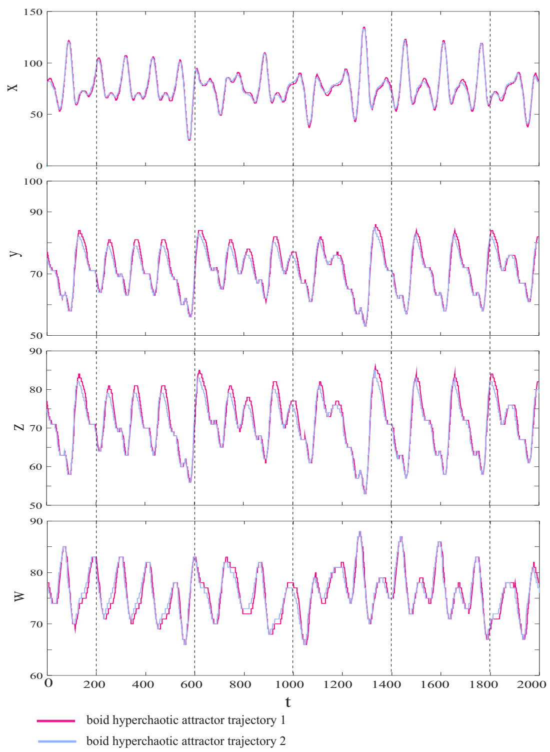

Figure 4 Temporary states x, y, z, and w, of two boids driven by hyperchaotic MACM attractor.

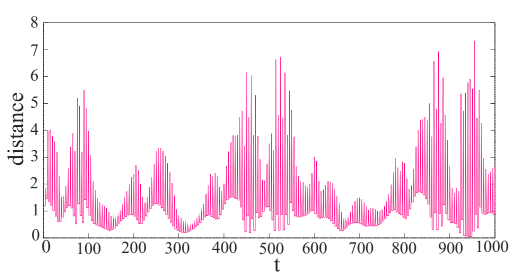

Figure 5 Distance between two boids (MACM's oscillators).

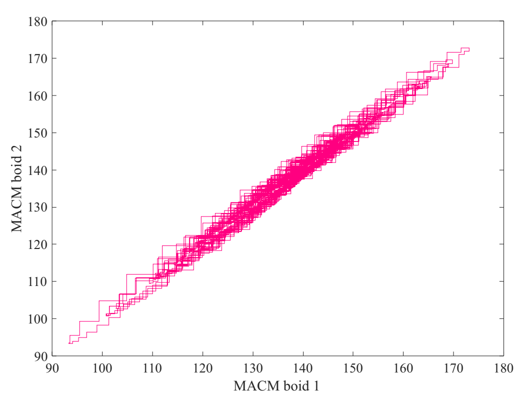

Figure 6 Synchronization of two boids (MACM's oscillators).

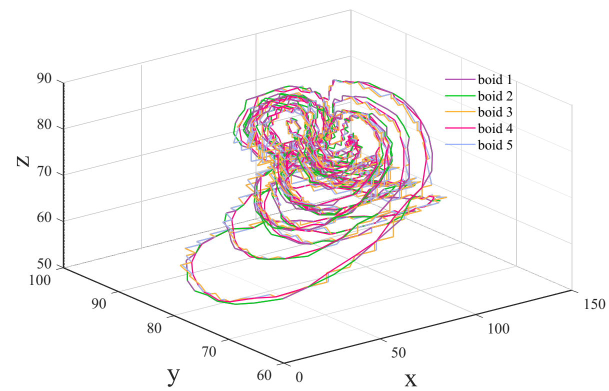

Figure 7 Synchronization of 5 boids (MACM's oscillators for $t \in [0, 50]$. MATLAB overwrites the existing graph in some sections and plots the trajectories in order of purple, green, yellow, orange, red, blue).

Computer simulations of the behavior of 5 boids controlled by the hyperchaotic MACM system is illustrated in Figure 7, it is observed that the flock of 5 boids that two boids are together all the way from the start and as time increases. In Figure 7, it can be seen that the trajectories are intertwined by observing the different colors throughout the entire journey. It is observed that in some sections the trajectories overlap over the same region in space, observing only two or three lines.

CONCLUSION

The investigation of a nonlinear system was carried out through the implementation of an algorithm that describes the behavior of the boids, which are controlled by a hyperchaotic MACM system, the implementation of this system forces the boids to follow their attractor trajectory, induce them to follow it. In this work a network formed by N identical nodes was considered, this behavior can be described using a set of nonlinear autonomous differential equations. The boids maintain the three rules of cohesion, alignment, and separation that define them while maintaining a hyperchaotic trajectory.

It is observed that the boids have characteristics that they share with complex systems with dynamic behavior, and emergent properties that arise from the interactions between the boids were presented.

For future work, tasks remain to be performed described below:

- Implementation of a rule for the introduction of one or more predators in the system, where the prey are the boids.
- Implementation of the rule to avoid obstacles, which prevent continuing with the trajectory, and the boids are forced to surround them to continue with the trajectory of the hyperchaotic attractor.
- Implement a rule to find specific targets.
- Implement a rule to adhere to a leader, the boid who is in front will take the position of leader and the others will follow.
- Implement this study in robots through the generation of hyperchaotic trajectories to directly influence the behavior of the robot's speeds, using the inputs to the system; as are the speeds of the engines. One of the advantages of this method is its simplicity and ease of being implemented in mobile robots

since it is only necessary to know the number of inputs handled by the robot to implement the algorithm. For a potential application in experimentation, we are considering the use of quadcopters. Each quadcopter would function as a boid, interconnected through wireless communication employing a Wi-Fi network, facilitating information exchange through sockets. The positional data for each boid would be acquired either through a Motion Capture System (Mocap, such as OptiTrack) or alternative methods like mounting cameras on each quadcopter, implementing infrared sensors, or utilizing radio frequency triangulation. To attain synchronization within a hyperchaotic MACM system, we propose the inclusion of an additional rule, assigned a lower priority compared to existing rules. This supplementary rule endows each boid with a distinct speed component, individually generated by a hyperchaotic trajectory generator algorithm as reported in (Cetina-Denis *et al.* 2022). This approach aims to provide each boid with a unique characteristic, similar to the individuality observed in a flock of birds, where differences in size, weight or, agility contribute to distinctive behaviors emerging from group interactions.

Acknowledgments

This work was supported by the CONAHCYT Research Project "Synchronization of complex systems and its applications" under grant number A1-S-31628. **In memory of Cornelio Posadas.**

Availability of data and material

Not applicable.

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical standard

The authors have no relevant financial or non-financial interests to disclose.

LITERATURE CITED

Cetina-Denis, J. J., R. M. Lopéz-Gutiérrez, C. Cruz-Hernández, and A. Arellano-Delgado, 2022 Design of a chaotic trajectory generator algorithm for mobile robots. Appl. Sci. **12**: 2857.

Duman, E., M. Uysal, and A. F. Alkaya, 2012 Migrating birds optimization: a new metaheuristic approach and its performance on quadratic assignment problem. Information Sciences **217**: 65–77.

Itoh, M. and L. O. Chua, 2007 Boids control of chaos. International Journal of Bifurcation and chaos **17**: 427–444.

Kapitaniak, T., Y. Maistrenko, and S. Popovych, 2000 Chaos-hyperchaos transition. Physical Review E **62**: 1972.

Karaboga, D. *et al.*, 2005 An idea based on honey bee swarm for numerical optimization. Technical report, Technical report-tr06, Erciyes university, engineering faculty, computer.

Méndez-Ramírez, R., A. Arellano-Delgado, C. Cruz-Hernández, R. Martínez-Clark, *et al.*, 2017 A new simple chaotic Lorenz-type system and its digital realization using a tft touch-screen display embedded system. Complexity **2017**.

Méndez-Ramírez, R. D., A. Arellano-Delgado, M. A. Murillo-Escobar, and C. Cruz-Hernández, 2021 A new 4d hyperchaotic system and its analog and digital implementation. Electronics **10**: 1793.

Pourpanah, F., R. Wang, C. P. Lim, X.-Z. Wang, and D. Yazdani, 2023 A review of artificial fish swarm algorithms: Recent advances and applications. Artificial Intelligence Review **56**: 1867–1903.

Rajagopal, K., S. Vaidyanathan, A. Karthikeyan, and A. Srinivasan, 2018 Complex novel 4d memristor hyperchaotic system and its synchronization using adaptive sliding mode control. Alexandria Engineering Journal **57**: 683–694.

Reynolds, C. W., 1987 Flocks, herds and schools: A distributed behavioral model. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pp. 25–34.

How to cite this article: Ana, C. M. G., Liliana, C.A., Martha, L. G. R., and Cesar, C. H. Flocking Behavior of Boids Driven by Hyperchaotic MACM System. *Chaos Theory and Applications*, 6(2), 152-158, 2024.