

Makes drones in circle Experiments Report

Zhang Jinrui*
jerryzhang40@gmail.com

20250720

Abstract

In this article, I tried to do apply a simple policy to each drone to make them fly in circle

1 recite of the problem & assumptions

There are 10 drones and fly on the sky obeys Newton's second law of motion. which is

$$\begin{aligned}\vec{F} &= m\vec{a} \\ \vec{a} &= \frac{d\vec{v}}{dt} = \frac{d^2\vec{x}}{dt^2}\end{aligned}$$

And I mean the policy by, we need a function of force depending on some communication between drones to decide the \vec{F}

$$\vec{F} = f(\text{thecurrentinformation})$$

And then we want the following dynamic system

$$\begin{bmatrix} \frac{d\vec{x}}{dt} \\ \frac{d\vec{v}}{dt} \end{bmatrix} = \begin{bmatrix} \vec{v} \\ \vec{a} = f/m \end{bmatrix}$$

has some Self-organized emergent phenomena, to automatically emergent a circle rounding pattern.

2 a simple prompt

To be more clear of the notations we use, we have $i \in \{1, 2, \dots, 10\} = N$ And the drones are ignored of its flying height, which the position vector can be a 2d

*alternative email: zhangjr1022@mails.jlu.edu.cn

vector note it as \vec{d}_i And so the velocity and acceleration we denote as $\vec{v}_i = \frac{d\vec{d}_i}{dt}$ and $\vec{a}_i = \frac{d\vec{v}_i}{dt}$ I want to prompt a f so that it can form a circle.

$$\vec{f}_i = m_i \left(\sum_{\forall k \neq i, \|\vec{d}_i - \vec{d}_k\| \leq R} \left(\frac{\vec{d}_i - \vec{d}_k}{\|\vec{d}_i - \vec{d}_k\|^3} \right) + \left(\frac{\vec{d}_{t(i)} - \vec{d}_i}{\|\vec{d}_{t(i)} - \vec{d}_i\|} - v_i \right) \right)$$

This model is easy to explain, the first term is just a inverse square propell force, the second term is make the velocity quickly approach a set direction the $t(i)$ is just a randomly choosed target drone other than i that is $t(i) \in N, t(i) \neq i$

This formula can be rewrite without physical term as follow.

$$\vec{a}_i = \sum_{\forall k \neq i, \|\vec{d}_i - \vec{d}_k\| \leq R} \left(\frac{\vec{d}_i - \vec{d}_k}{\|\vec{d}_i - \vec{d}_k\|^3} \right) + \left(\frac{\vec{d}_{t(i)} - \vec{d}_i}{\|\vec{d}_{t(i)} - \vec{d}_i\|} - v_i \right)$$

separately view this is combined by two independent force

$$(\vec{a}_i)_{target} = \left(\frac{\vec{d}_{t(i)} - \vec{d}_i}{\|\vec{d}_{t(i)} - \vec{d}_i\|} - v_i \right)$$

$$(\vec{a}_i)_{propell} = \sum_{\forall k \neq i, \|\vec{d}_i - \vec{d}_k\| \leq R} \left(\frac{\vec{d}_i - \vec{d}_k}{\|\vec{d}_i - \vec{d}_k\|^3} \right)$$

3 a simple prompt:simulation

3.1 Four drone case

3.1.1 derivation

This case just choose $N = \{1, 2, 3, 4\}$ and don't allow $t(i) = i$ which definitely form a three element loop and a dangling drone.

We have a (4,2)-tensor \vec{d}_i and two other (4,2)-tensor \vec{v}_i and \vec{a}_i The initial points are randomly choosed in Uniformly $[0, 1] \times [0, 1]$

choose a time increment dt and the simulation update formula is simple to write

just as follow

$$\begin{bmatrix} \vec{d}_{n+1,i} \\ \vec{v}_{n+1,i} \end{bmatrix} = \begin{bmatrix} \vec{d}_{n,i} + \vec{v}_{n,i} dt \\ \vec{v}_{n,i} + \left(\sum_{\forall k \neq i, \|\vec{d}_{n,i} - \vec{d}_{n,k}\| \leq R} \left(\frac{\vec{d}_{n,i} - \vec{d}_{n,k}}{\|\vec{d}_{n,i} - \vec{d}_{n,k}\|^3} \right) + \left(\frac{\vec{d}_{n,t(i)} - \vec{d}_{n,i}}{\|\vec{d}_{n,t(i)} - \vec{d}_{n,i}\|} - v_{n,i} \right) \right) dt \end{bmatrix}$$

simple Euler method.

3.1.2 code & result

the computational code are [1, FourDroneCase] . the results are shown by the following pictues which generated by the code. The video are [2, sample1-video] and more other in the same folder on github.

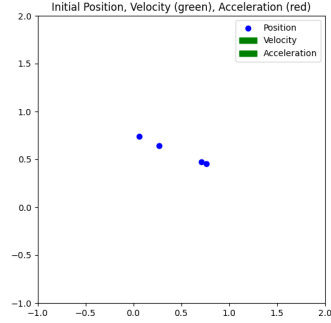


Figure 1: sample1 randomly initial position

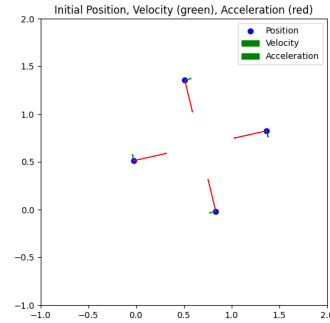


Figure 2: sample1 after a period of time

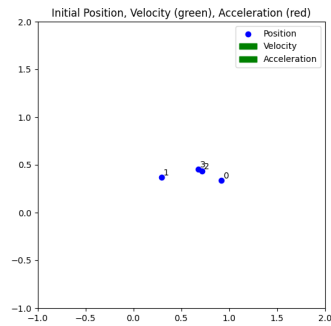


Figure 3: sample1 randomly initial position

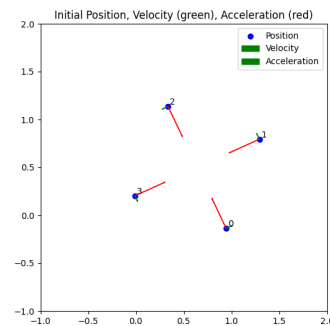


Figure 4: sample2 after a period of time

3.2 Ten drone case

undergoing

3.2.1 derivation

undergoing

4 some analysis why it will have a stability property

4.1 the terminate radius R

undergoing

4.2 the terminate center O

undergoing

4.3 graph theory part

The $t(i)$ forms a graph which have n points and n oriented edges, this forms a tree with a extra edges, and this case It will obviously form a Unicyclic Graph.

Which is a tree if we treat all the point on the loop as the same point.

References

- [1] Zhang Jinrui. Fourdronecase.py. https://github.com/jerzha40/2025_exchange_at_universityofalberta/blob/main/DroneInCircleEXP/FourDroneCase.py. Accessed: 2025-07-20.
- [2] Zhang Jinrui. sample1-video. https://github.com/jerzha40/2025_exchange_at_universityofalberta/blob/main/DroneInCircle/fig/sample1/0001-18255.mp4. Accessed: 2025-07-20.