Makes drones in cirle 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 cirle

1 recite of the problem & assumptions

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

$$\vec{F} = m\vec{a}$$

$$\vec{a} = \frac{d\vec{v}}{dt} = \frac{d^2\vec{x}}{dt^2}$$

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

$$\vec{F} = f(the current information)$$

And then we want the following dynamic system

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

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 $i \in \{1, 2, ..., 10\} = N$ 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 cirle.

$$\vec{f_i} = m_i(\sum_{\forall k \neq i, \|\vec{d_i} - \vec{d_k}\| \leq R} (\frac{\vec{d_i} - \vec{d_k}}{\|\vec{d_i} - \vec{d_k}\|^3}) + (\frac{\vec{d_{t(i)}} - \vec{d_i}}{\|\vec{d_{t(i)}} - \vec{d_i}\|} - v_i))$$

This model is easy to explain, the first term is just a inverse square propell force, the second term is make the velocity quickly approch 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.

$$\vec{a_i} = \sum_{\forall k \neq i, \|\vec{d_i} - \vec{d_k}\| \leq R} (\frac{\vec{d_i} - \vec{d_k}}{\|\vec{d_i} - \vec{d_k}\|^3}) + (\frac{\vec{d_{t(i)}} - \vec{d_i}}{\|\vec{d_{t(i)}} - \vec{d_i}\|} - v_i)$$

separately view this is combined by two independent force

$$(\vec{a_i})_{target} = (\frac{\vec{d_{t(i)}} - \vec{d_i}}{\|\vec{d_{t(i)}} - \vec{d_i}\|} - v_i)$$

$$(\vec{a_i})_{propell} = \sum_{\forall k \neq i, \|\vec{d_i} - \vec{d_k}\| \leq R} (\frac{\vec{d_i} - \vec{d_k}}{\|\vec{d_i} - \vec{d_k}\|^3})$$

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 $t(t(i)) = i$ 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

$$\begin{bmatrix} \vec{d_{n+1}}_i \\ \vec{v_{n+1}}_i \end{bmatrix} = \begin{bmatrix} \vec{d_n}_i + \vec{v_n}_i dt \\ \vec{v_n}_i + (\sum_{\forall k \neq i, \|\vec{d_n}_i - \vec{d_n}_k\| \leq R} (\frac{\vec{d_n}_i - \vec{d_n}_k}{\|\vec{d_n}_i - \vec{d_n}_k\|^3}) + (\frac{\vec{d_n}_{t(i)} - \vec{d_n}_i}{\|\vec{d_n}_{t(i)} - \vec{d_n}_i\|} - v_{n\,i}))dt \end{bmatrix}$$

simple Euler method.

3.1.2 code & result

the computational code are [1, FourDroneCase] . the results are shown by the following pictrues which generated by the code. The video are [2, sample1-video] and [3, sample2-video] and more other in the same folder on github.

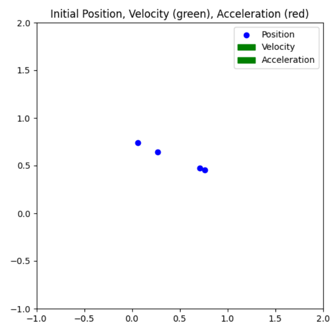

Figure 1: sample1 randomly initial position

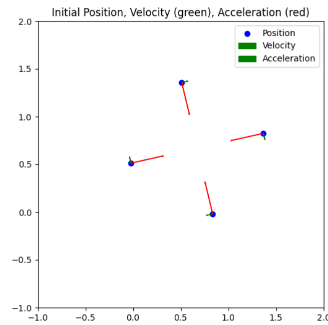

Figure 2: sample1 after a period of time

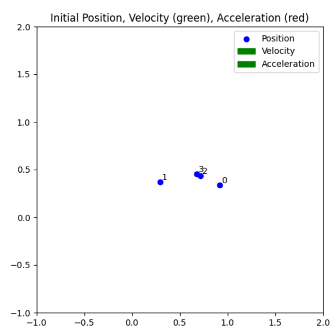

Figure 3: sample1 randomly initial position

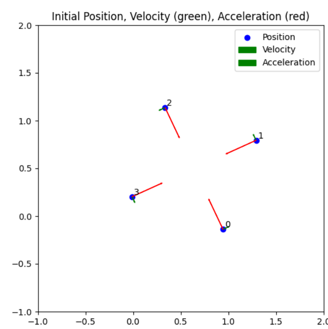

Figure 4: sample2 after a period of time

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.

[3] Zhang Jinrui. sample2-video. https://github.com/jerzha40/2025_exchange_at_universityofalberta/blob/main/DroneInCircle/fig/sample2/0001-18994.mp4. Accessed: 2025-07-20.