报告

张津睿*

zhangjr1022@mails.jlu.edu.cn

2025 年 9 月 26 日

摘要

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 $\vec{F}$

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

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.

*alternative email:jerryzhang40@gmail.com

2 Jinrui Zhang's prompt

2.1 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 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.

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

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

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

separately view this is combined by two independent force

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

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

2.2 a simple prompt:simulation

2.2.1 Four drone case: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

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

simple Euler method.

2.2.2 Four drone case:code & result

the computational code are [2, FourDroneCase] . the results are shown by the following pictrues which generated by the code. The video are [3, sample1-video] and [4, sample2-video] and more other in the same folder on github.

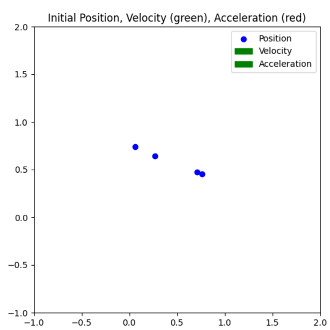

图 1: sample1 randomly initial position

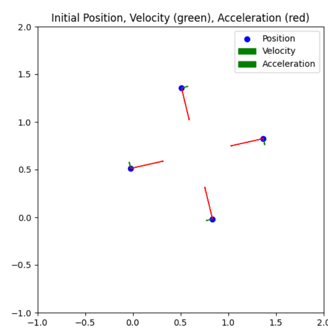

图 2: sample1 after a period of time

2.2.3 Ten drone case:derivation

undergoing

2.2.4 Ten drone case:result

undergoing

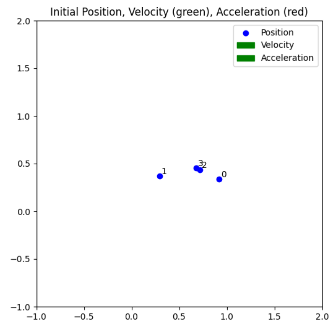

图 3: sample1 randomly initial position

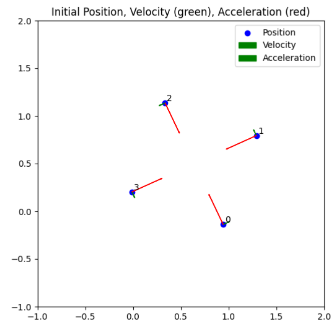

图 4: sample2 after a period of time

2.3 some analysis why it will have a stability property

2.3.1 the terminate radius R

undergoing

2.3.2 the terminate center O

undergoing

2.3.3 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.

2.4 target distance method

undergoing

2.5 target distance method:simulation

undergoing

3 Zinan Su's approach

3.1 notations & equations

safe collide radius is d_s

$$\sigma = 2d_s$$

NUM is the total number of the drones. And then we want the following dynamic system

$$\begin{bmatrix} \frac{d\vec{p_i}}{dt} \\ \frac{d\vec{v_i}}{dt} \end{bmatrix} = \begin{bmatrix} \vec{v_i} \\ \vec{a_i} \end{bmatrix}$$

circle origin is a function

$$c = \frac{1}{NUM} \sum_{k=1}^{NUM} p_k$$

Four constants.

$$k_p =$$

$$k_d =$$

$$k_v =$$

$$k_r =$$

$$R^* = \frac{1}{NUM} \sum_{k=1}^{NUM} p_k(0) - c(0)$$

$$v_d = \frac{1}{NUM} \sum_{k=1}^{NUM} \|v_k(0)\|$$

and

$$r_i = p_i - c$$

$$d_i = \|r_i\|$$

$$\hat{r_i} = \frac{r_i}{d_i}$$

$$\hat{\theta}_i = \mathbb{M}_\theta r_i$$

$$\mathbb{M}_\theta = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

$$v_{i\|} = \hat{r_i} \cdot v_i$$

$$v_{i\perp} = \hat{\theta}_i \cdot v_i$$

$$U(r) = k_r e^{-\frac{r}{2\sigma^2}}$$

$$U_{ij} = U(\|p_i - p_j\|)$$

$$\vec{u_{i1}} = [-k_p(d_i - R^*) - k_d v_{i\|}]\hat{r}_i$$

$$\vec{u_{i2}} = [-k_v(v_{i\perp} - v_d)]\hat{\theta}_i$$

$$\vec{u_{i3}} = \sum_{\forall k \neq i} (-\nabla_{p_i} U_{ij})$$

$$\vec{u_i} = \vec{u_{i1}} + \vec{u_{i2}} + \vec{u_{i3}}$$

3.2 analysis

考虑带阻力的动力方程

$$m\frac{\mathrm{d}^2 \boldsymbol{p}_i}{\mathrm{d}t^2} + c\left\|\frac{\mathrm{d}\boldsymbol{p}_i}{\mathrm{d}t}\right\|\frac{\mathrm{d}\boldsymbol{p}_i}{\mathrm{d}t} = \boldsymbol{u}_i,$$

其中 m 为质量, c 为阻力系数, 即

$$\frac{\mathrm{d}\boldsymbol{X}}{\mathrm{d}t} = \boldsymbol{F}(\boldsymbol{X}), \tag{1}$$

其中

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{p} \\ \boldsymbol{v} \end{pmatrix} \in \mathbb{R}^{40},\ \boldsymbol{p} = \begin{pmatrix} \boldsymbol{p}_1 \\ \vdots \\ \boldsymbol{p}_{10} \end{pmatrix},\ \boldsymbol{v} = \begin{pmatrix} \boldsymbol{v}_1 \\ \vdots \\ \boldsymbol{v}_{10} \end{pmatrix},\ \boldsymbol{F}(\boldsymbol{X}) = \begin{pmatrix} \boldsymbol{v} \\ m^{-1}(\boldsymbol{u} - c\|\boldsymbol{v}\|\boldsymbol{v}) \end{pmatrix}.$$

易知质心 $\boldsymbol{c}(\boldsymbol{X}), \|\boldsymbol{r}_i\|, \widehat{\boldsymbol{r}}_i, \mathit{\Phi}$ 均为 Lipschitz 连续函数, 且初值条件满足

$$\min_{i \neq j} \|\boldsymbol{p}_i(0) - \boldsymbol{p}_j(0)\| > d_s > 0.$$

由 Picard 定理可知方程 (1) 的解存在且唯一, 解为

$$\boldsymbol{X}(t) = \boldsymbol{X}(0) + \int_0^t \boldsymbol{F}(\boldsymbol{X}(s))\mathrm{d}s,$$

即

$$\begin{aligned} \boldsymbol{v}_i(t) &= \boldsymbol{v}_i(0) + \int_0^t \left[m^{-1}\boldsymbol{u}_i(\boldsymbol{X}(s)) - \frac{c}{m}\|\boldsymbol{v}_i(s)\|\boldsymbol{v}_i(s)\right]\mathrm{d}s, \\ \boldsymbol{p}_i(t) &= \boldsymbol{p}_i(0) + \int_0^t \boldsymbol{v}_i(s)\mathrm{d}s. \end{aligned}$$

下面证明防撞性. 定义

$$\begin{aligned}
&\Psi(d)=k_r\exp\left\{-\frac{(d-d_s)^2}{2\sigma^2}\right\},\\
&\Phi(d)=-\frac{\mathrm{d}\Psi}{\mathrm{d}d}=\frac{k_r}{\sigma^2}\exp\left\{-\frac{(d-d_s)^2}{2\sigma^2}\right\}(d-d_s),\\
&E_{ij}(t)=\frac{1}{2}\left(\frac{\mathrm{d}d_{ij}}{\mathrm{d}t}\right)^2+\psi(d_{ij}(t)),
\end{aligned}$$

其中 $d_{ij}(t)=\|\boldsymbol{p}_i(t)-\boldsymbol{p}_j(t)\|$. 则系统能量为

$$\mathcal{E}(t)=\sum_{1\leqslant i<j\leqslant N}E_{ij}(t).$$

而

$$\frac{\mathrm{d}E_{ij}}{\mathrm{d}t}=\frac{\mathrm{d}d_{ij}}{\mathrm{d}t}\cdot\frac{\mathrm{d}^2d_{ij}}{\mathrm{d}t^2}+\Phi(d_{ij})\frac{\mathrm{d}d_{ij}}{\mathrm{d}t},$$

$$\frac{\mathrm{d}^2d_{ij}}{\mathrm{d}t^2}=\frac{1}{d_{ij}}\left[\|\boldsymbol{v}_i-\boldsymbol{v}_j\|^2+(\boldsymbol{p}_i-\boldsymbol{p}_j)\cdot(\boldsymbol{u}_i-\boldsymbol{u}_j)-\left(\frac{\mathrm{d}d_{ij}}{\mathrm{d}t}\right)^2\right],\qquad(2)$$

其中 $\boldsymbol{u}_i=\dot{\boldsymbol{v}}_i$. 对于 $\boldsymbol{u}_i$, 有

$$\boldsymbol{u}_i=\frac{1}{m}[-k_p(d_i-R^*)\widehat{\boldsymbol{r}}_i-k_dv_{r,i}\widehat{\boldsymbol{r}}_i-k_v(v_{\theta,i}-v_d)\hat{\boldsymbol{\theta}}_{\boldsymbol{i}}]+\frac{1}{m}\sum_{k\neq i}\Phi(d_{ik})(\boldsymbol{p}_i-\boldsymbol{p}_k)=:\boldsymbol{u}_{i_1}+\boldsymbol{u}_{i_2}.$$

设 $\|\boldsymbol{u}_{i_1}-\boldsymbol{u}_{j_1}\|\leqslant L$. 取

$$k_r>L\sigma^2\mathrm{e}^{\frac{1}{2}}\max\left\{\frac{1}{d_s},\frac{1}{\min\limits_{k\neq l}d_{kl}(0)}\right\},$$

则当 $d_{ij}\leqslant d_s+\sigma$ 时, 有

$$\|\boldsymbol{u}_{i_2}-\boldsymbol{u}_{j_2}\|>2L.$$

于是

$$(\boldsymbol{p}_i-\boldsymbol{p}_j)\cdot(\boldsymbol{u}_i-\boldsymbol{u}_j)\geqslant\|\boldsymbol{u}_{i_2}-\boldsymbol{u}_{j_2}\|d_{ij}-Ld_{ij}>Ld_{ij}.$$

代入 (2) 式有

$$\frac{\mathrm{d}E_{ij}}{\mathrm{d}t}\geqslant\frac{\mathrm{d}d_{ij}}{\mathrm{d}t}(L+\Phi(d_{ij}))>0.$$

由此即知

$$\frac{\mathrm{d}\mathcal{E}}{\mathrm{d}t}\geqslant-\kappa\mathcal{E}(t),$$

其中 $\kappa > 0$ 为常数. 而

$$\begin{aligned}
&\mathcal{E}(0) \geqslant \sum_{i<j} \Psi(d_{ij}(0)) > \psi(d_s + \sigma) \cdot \binom{N}{2}, \\
&\mathcal{E}(t) \geqslant \mathcal{E}(0)\mathrm{e}^{-\kappa t} > 0, \\
&\Psi(d_{ij}(t)) \leqslant E_{ij}(t) \leqslant \mathcal{E}(t).
\end{aligned}$$

由 Ψ 严格单调递减可知

$$d_{ij}(t) \geqslant \Psi^{-1}(\mathcal{E}(t)) > \Psi^{-1}(\mathcal{E}(0)\mathrm{e}^{-\kappa t}).$$

令 $t \to \infty$, 有

$$\varliminf_{t\to\infty} d_{ij}(t) \geqslant \Psi^{-1}(0) = d_s.$$

故总是不会相撞.

下面讨论收敛性, 即讨论系统收敛至

$$\mathcal{S} = \{\|\boldsymbol{r}_i\| = R, \boldsymbol{v}_i \cdot \widehat{\boldsymbol{r}}_i = 0, \|\boldsymbol{v}_i\| = v_d\}.$$

构造 Lyapunov 函数

$$V = \frac{1}{2}\sum_{i=1}^{N}[k_p(d_i - R)^2 + \|\boldsymbol{v}_i - v_d\widehat{\boldsymbol{\theta}}_i\|^2] + \sum_{i<j}\Psi(d_{ij}),$$

则

$$\dot{V} = -\sum_i k_d v_{r,i}^2 - \sum_i k_v(v_{\theta,i} - v_d)^2 - \sum_i c\|\boldsymbol{v}_i\|^3 \leqslant 0,$$

故方程渐进收敛至 $\mathcal{S}$.

3.3 some constants calculation

undergoing c is air resistance constant.

$$m\frac{d^2\vec{p_i}}{dt^2} + c\|\frac{d\vec{p_i}}{dt}\|\frac{d\vec{p_i}}{dt} = u_i$$

4 An approach inspired by Reynolds

4.1 background

By wangdering around I have found an article [5, ReynoldsMethod] and a video explain it [1].

In this article it gives three rule to have a self regulation flock of birds.

First, to avoid collision, they give every individual a propell force just as i did in 2.1.

Second, it has done a similar propose as Professor Li says, to adjust their direction according to their neighbours. That is, match its own derection to align with its surroundings.

Third, to make a flock stay together it make a similar following target as i has done in 2.1. Which is calculate the flock geometric center and make that center as a target.

4.2 quantize and formulas

To simplify the realizaiton, we need to precalculate two constant for every frame. That is v_c and d_c, for every individual it can only see within radius R, that is a neighbours as follow

$$\mathcal{N}_i = \{\, k \;\big|\; k \neq i,\ \|\vec{d_i} - \vec{d_k}\| \leq R \,\}.$$

where

$$d_i^c = \frac{1}{|\mathcal{N}_i|} \sum_{k \in \mathcal{N}_i} d_k$$

$$v_i^c = \frac{1}{|\mathcal{N}_i|} \sum_{k \in \mathcal{N}_i} v_k$$

$$\vec{a_i}^{propell} = \mu^{propell} \sum_{k \in \mathcal{N}_i} \frac{\vec{d_i} - \vec{d_k}}{\|\vec{d_i} - \vec{d_k}\|^3}$$

$$\vec{a_i}^{aligning} = \mu^{aligning} (v_i^c - v_i)$$

$$\vec{a_i}^{centering} = \mu^{centering} (\lambda^{centering} \frac{\vec{d_i^c} - \vec{d_i}}{\|\vec{d_i^c} - \vec{d_i}\|} - v_i)$$

so the final formula is

$$\vec{a}_i = \vec{a_i}^{centering} + \vec{a_i}^{aligning} + \vec{a_i}^{propell}$$

4.3 numerical simulation

basically for N individual birds. First use space spliting tree to maintain its neighbours $\mathcal{N}_i$ in a time complexity of $\mathcal{O}(n \log n)$

then update the acceleration then use euler method to calculate velocity and position.

here is the code reference.

here is the result pictrues. video are in reference.

参考文献

[1] https://www.youtube.com/@xpinman. In order to create animation special effects, he accidentally created an algorithm that affects the world. what are these three rules, and how do they change the world? [bad review]. `https://www.youtube.com/watch?v=glPLK4Kh6ds&t=45s`. Accessed: 2025-01-27.

[2] Zhang Jinrui. Fourdronecase.py. `https://github.com/jerzha40/2025_exchange_at_universityofalberta/blob/main/DroneInCircleEXP/FourDroneCase.py`. Accessed: 2025-07-20.

[3] Zhang Jinrui. sample1-video. `https://github.com/jerzha40/2025_exchange_at_universityofalberta/blob/main/DroneInCircle/fig/sample1/0001-18255.mp4`. Accessed: 2025-07-20.

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

[5] Craig W. Reynolds. Flocks, herds and schools: A distributed behavioral model. *SIGGRAPH Comput. Graph.*, 21(4):25–34, August 1987.