KUTAISI INTERNATIONAL UNIVERSITY

Numerical Programming

Ramaz Botchorishvili

Kutaisi International University

Final Project - Iluminated Drone Show Simulation

Problem statement

Problem 0.1

Simulate illuminated drone show using numerical modeling of a synchronized drone swarm with shape preservation. Three sub-problems must be solved:

1. Static Formation on a Handwritten Input, 5 points.
    - Input: image of your handwritten name with at least 8 characters, number of drones and their initial position.
    - Task: drone swarm must move from a still initial position to the handwritten name. number of drones must be sufficient to display the handwritten name.
    - Output: trajectory of each drone, visualization. The input for the visualization is trajectory.
2. Transition to New Year Greeting, 5 points
    - Input: drone swarm at your handwritten name, and the greeting "Happy New Year!"
    - Task: drone swarm must move from handwritten name position to the the holiday greeting.
    - Output: trajectories of each drone, visualization. The input for the visualization is trajectory.
3. Dynamic Tracking and Shape Preservation, 5 points
    - Input: drone swarm at new year greeting, and a video of your choice.
    - Task: drone swarm must move from the new year greeting position to the moving object in the video, and dinamicaly repeat motion of an object with shape preservation.
    - Output: trajectories of each drone, visualization. The input for the visualization is trajectory.

Notes on project implementation

Important remarks

To guarantee a successful implementation of this computational project, the student must make a series of interdependent decisions concerning the mathematical model, numerical methods, input data (including pre-processing), validation and verification, as well as output interpretation and communication.

- The student must develop or select a mathematical model that describes collision-free motion of a drone swarm. Several candidate models are presented on the slides below; choosing and refining the appropriate model is the student's responsibility.
- The initial configuration of the swarm may be positioned along a line, within a square, or inside a cube.
- The student must determine the number of drones required for each sub-problem. One may use a fixed count for all sub-problems, or vary the number per sub-problem, provided that drones return to or originate from their initial positions.
- The student may work in two- or three-dimensional space. Three dimensions afford greater freedom in trajectory selection, whereas a two-dimensional motion is governed by four equations instead of six in three dimensions.
- Visualization: the swarm must be rendered with illumination so that it remains visible at all times, including during motion. Any built-in library may be employed for visualization.

General requirements, common tasks for sub-problems

Tasks

- For each sub-problem formulate suitable BVP or IVP.
- Clearly describe all inputs to the selected equations including initial conditions, boundary conditions, and user defined parameters.
- Formulate numerical methods, algorithms, and their properties, explain and justify your approach in written.
- Develop test cases and demonstrate validity of your results.
- Upload all necessary files, including
  1. Presentation file
  2. Code
  3. Test data and their description
  4. Visualisation
  5. Test cases for which your approach does work well, and does not work, with explanations of limitations.

Initial Value Problem (IVP)

IVP

- The state of drone i is governed by its position $\mathbf{x}_i \in \mathbb{R}^3$ and velocity $\mathbf{v}_i \in \mathbb{R}^3$, $i = 1, \ldots, N$
- Governing equations with velocity saturation:

$$\dot{\mathbf{x}}_i = \mathbf{v}_i \cdot \min\left(1, \frac{v_{max}}{\|\mathbf{v}_i\|}\right)$$

$$\dot{\mathbf{v}}_i(t) = \frac{1}{m}\left[k_p\left(\mathbf{T}_i(t) - \mathbf{x}_i(t)\right) + \sum_{j \neq i} \mathbf{f}_{\text{rep}}(\mathbf{x}_i(t), \mathbf{x}_j(t)) - k_d \mathbf{v}_i(t)\right],$$

- Initial conditions:

$$\mathbf{x}_i(0) = \mathbf{x}_{i,0}, \quad \mathbf{v}_i(0) = \mathbf{v}_{i,0},$$

  - **Target Tracking:** $\mathbf{T}_i(t)$ represents the time-varying waypoint for drone i corresponding to the desired figure.
  - **Collision Avoidance:** $\mathbf{f}_{rep}$ is a repulsive force (e.g., inverse-square law) that activates when drones are within a safety radius R_{safe}.
  - **Damping:** k_d ensures the system reaches a steady state without infinite oscillation.
  - More on notations with explanation is given below.

Boundary Value Problem (BVP)

BVP

- The state of drone i is governed by its position $\mathbf{x}_i \in \mathbb{R}^3$ and velocity $\mathbf{v}_i \in \mathbb{R}^3$, $i = 1, \ldots, N$
- Governing equations with velocity saturation:

$$\dot{\mathbf{x}}_i = \mathbf{v}_i \cdot \min\left(1, \frac{v_{max}}{\|\mathbf{v}_i\|}\right)$$

$$\dot{\mathbf{v}}_i(t) = \frac{1}{m}\left[k_p\left(\mathbf{T}_i(t) - \mathbf{x}_i(t)\right) + \sum_{j \neq i} \mathbf{f}_{\text{rep}}(\mathbf{x}_i(t), \mathbf{x}_j(t)) - k_d \mathbf{v}_i(t)\right],$$

- Boundary conditions:

$$\mathbf{x}_i(0) = \mathbf{x}_{i,0}, \quad \mathbf{x}_i(T) = \mathbf{x}_{i,T},$$

  - **Target Tracking:** $\mathbf{T}_i(t)$ represents the time-varying waypoint for drone i corresponding to the desired figure. You can consider several sub-models where T_i is time independent, $k_d = 0$ and (or) $k_p = 0$.
  - **Collision Avoidance:** $\mathbf{f}_{rep}$ is a repulsive force (e.g., inverse-square law) that activates when drones are within a safety radius R_{safe}.
  - More on notations with explanation is given below.

Initial Value Problem with Velocity Tracking (IVP VT)

IVP VT

- The state of drone i is governed by its position $\mathbf{x}_i \in \mathbb{R}^3$ and velocity $\mathbf{v}_i \in \mathbb{R}^3$, $i = 1, \ldots, N$.
-
$$\mathbf{V}_{\text{sat}}(\mathbf{x}, t) = \mathbf{V}(\mathbf{x}, t) \cdot \min\left(1, \frac{v_{\max}}{\|\mathbf{V}(\mathbf{x}, t)\|}\right) \quad \text{if } \|\mathbf{V}(\mathbf{x}, t)\| > 0, \quad \mathbf{0} \quad \text{otherwise.}$$
- Governing equations with velocity saturation:
$$\dot{\mathbf{x}}_i = \mathbf{v}_i \cdot \min\left(1, \frac{v_{max}}{\|\mathbf{v}_i\|}\right)$$
$$\dot{\mathbf{v}}_i(t) = \frac{1}{m}\left[k_v\left(\mathbf{V}_{\text{sat}}(\mathbf{x}_i(t), t) - \mathbf{v}_i(t)\right) + \sum_{j \neq i} \mathbf{f}_{\text{rep}}(\mathbf{x}_i(t), \mathbf{x}_j(t)) - k_d \mathbf{v}_i(t)\right],$$
- Initial conditions:
$$\mathbf{x}_i(0) = \mathbf{x}_{i,0}, \quad \mathbf{v}_i(0) = \mathbf{v}_{i,0},$$

Parameters and variables

- saturated velocity field:

$$\mathbf{V}_{\mathrm{sat}}(\mathbf{x}, t) = \mathbf{V}(\mathbf{x}, t) \cdot \min\left(1, \frac{v_{\max}}{\|\mathbf{V}(\mathbf{x}, t)\|}\right) \quad \text{if } \|\mathbf{V}(\mathbf{x}, t)\| > 0, \quad \mathbf{0} \quad \text{otherwise.} \tag{1}$$

- $\mathbf{x}_i \in \mathbb{R}^d$: Position of drone i.
- $\mathbf{v}_i \in \mathbb{R}^d$: Velocity of drone i.
- $m > 0$: Mass of each drone (constant, assumed uniform).
- $\mathbf{V}(\mathbf{x}, t) \in \mathbb{R}^d$: Velocity field extracted from the reference video.
- $\mathbf{V}_{\mathrm{sat}}(\mathbf{x}, t) \in \mathbb{R}^d$: Saturated velocity field with $\|\mathbf{V}_{\mathrm{sat}}\| \leq v_{\max}$.
- $v_{\max} > 0$: Maximum allowable velocity magnitude (scalar, hardware-dependent).
- $k_v > 0$: Velocity-matching gain.
- $k_d > 0$: Damping coefficient.
- $k_{\mathrm{rep}} > 0$: Repulsion gain.
- $R_{\mathrm{safe}} > 0$: Safety radius.
- $\mathbf{f}_{\mathrm{rep}}(\mathbf{x}_i, \mathbf{x}_j) \in \mathbb{R}^d$: Pairwise repulsive force.

$$\mathbf{f}_{\mathrm{rep}}(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} k_{\mathrm{rep}} \dfrac{\mathbf{x}_i - \mathbf{x}_j}{\|\mathbf{x}_i - \mathbf{x}_j\|^3} & \text{if } \|\mathbf{x}_i - \mathbf{x}_j\| < R_{\mathrm{safe}}, \\ \mathbf{0} & \text{otherwise.} \end{cases} \tag{2}$$

- N: Number of drones.
- d: Spatial dimension (typically 2 or 3).

Final Remarks

Important Notice

- For successfully solving the problem student must know the following: derivatives and their approximations, edge detection, splines, IVP for ODEs, BVP for ODEs, RK methods and Butcher's table, truncation error, A-stability, shooting methods, numerical solution of linear and nonlinear systems of equations, optical flow.
- Student must explicitly mention using of AI and provide all related necessary details.
- The project is assigned 0 points if:
  - Any of the requested and/or necessary file is missing.
  - Submitted results are not reproducible.
  - Student cannot apply his own code for the input data provided by TA or instructor.
- Submission deadline: will be aligned with the schedule of final exams. See corresponding assignment.