Cp2

Murmani Akhaladze

TarnNished

Modeling the Dynamics of a Swarm of Drones: A Comparison of Implicit Methods for Solving ODEs

Introduction

Swarm dynamics, particularly in the context of autonomous drones, is an area of increasing interest for both research and practical applications, including surveillance, delivery systems, and environmental monitoring. These systems are often governed by a set of complex ordinary differential equations (ODEs) that describe the interactions between individual agents (drones) within the swarm. In this work, we model the dynamics of a swarm of drones experiencing both attractive and repulsive forces, where the drones are affected by each other's position in the environment. The goal is to explore the effectiveness of two implicit numerical methods for solving the system of ODEs governing the dynamics of the drone swarm, and to compare them to a standard explicit method.

Model Description

The dynamics of the drone swarm are described by a system of ODEs in which each drone is subjected to pairwise attractive and repulsive forces from other drones. The forces on a given drone are modeled as a sum of attractive forces (which pull drones towards each other) and repulsive forces (which push drones away to prevent collisions). The total force on each drone is computed by summing the contributions from all other drones in the swarm. The system of ODEs governing the motion of the drones is given by:

$$\frac{d\mathbf{r}_i}{dt} = \mathbf{v}_i, \quad \frac{d\mathbf{v}_i}{dt} = \frac{\mathbf{F}_i}{m}$$

Where:

- $\mathbf{r}_i$ is the position vector of drone i,
- $\mathbf{v}_i$ is the velocity vector of drone i,
- $\mathbf{F}_i$ is the total force acting on drone i,
- m is the mass of the drone.

The attractive force $\mathbf{F}_{\text{attr}}$ and the repulsive force $\mathbf{F}_{\text{rep}}$ are given by:

$$\mathbf{F}_{\text{attr}} = -k_{\text{attr}}(\mathbf{r}_i - \mathbf{r}_j)$$

$$\mathbf{F}_{\text{rep}} = k_{\text{rep}}\frac{\mathbf{r}_i - \mathbf{r}_j}{|\mathbf{r}_i - \mathbf{r}_j|^2}$$

Where k_{attr} and k_{rep} are the constants governing the strength of the attractive and repulsive forces, respectively.

Numerical Methods

To solve the system of ODEs numerically, three methods were considered: the explicit Euler method, implicit backward Euler with fixed-point iteration, and implicit backward Euler with Newton-Gauss-Seidel.

Explicit Euler Method

This is a straightforward, explicit method used as a baseline for comparison. In this method, the position and velocity of each drone are updated at each time step based on the current forces:

$$\mathbf{r}_i(t + dt) = \mathbf{r}_i(t) + \mathbf{v}_i(t) \cdot dt$$
$$\mathbf{v}_i(t + dt) = \mathbf{v}_i(t) + \frac{\mathbf{F}_i(t)}{m} \cdot dt$$

Implicit Backward Euler with Fixed-Point Iteration

This is a more stable method, where the position and velocity are implicitly updated using a fixed-point iteration scheme. The method starts with an explicit Euler guess and iteratively refines the position guess until convergence is achieved. The iterative procedure is stopped when the difference between successive position guesses falls below a specified tolerance.

Implicit Backward Euler with Newton-Gauss-Seidel

In this method, the backward Euler update is combined with a Newton-Gauss-Seidel approach, which uses a sequential update strategy for each drone's position. This method involves solving the system of equations using a Jacobian matrix and updating each drone's position iteratively.

Experiment Setup

The experiment was set up with 5 drones, each initialized with random positions within a 10x10 unit square and zero initial velocities. The drones experience attractive and repulsive forces, with constants $k_{\text{attr}} = 1.0$, $k_{\text{rep}} = 0.5$, and each drone has a mass of $m = 1.0$. The total simulation time was set to 10 seconds, with a time step of 0.05 seconds, leading to a total of 200 simulation steps.

Three simulations were run using the three methods described above. The performance of each method was compared in terms of computational efficiency, accuracy, energy conservation, and convergence behavior.

Results and Discussion

The computational efficiency of the three methods was evaluated by measuring the total runtime of each simulation. The results showed that the Newton-Gauss-Seidel method was the slowest, taking more time per step compared to the fixed-point iteration method. However, the difference in runtime between the two implicit methods was not very large.

In terms of accuracy, the implicit methods (both fixed-point iteration and Newton-Gauss-Seidel) produced more stable results compared to the explicit Euler method. The average pairwise distance between drones at the final time step was smallest for the implicit methods, indicating that they produced more compact formations. The explicit Euler method, on the other hand, resulted in a larger spread between the drones, which demonstrates its lack of stability in the presence of large time steps or complex interactions.

Energy conservation was another key aspect of the comparison. The implicit methods showed better energy conservation, with smaller energy drift over time compared to the explicit Euler method. This is expected, as implicit methods are known for their better stability and ability to handle stiff systems.

The convergence behavior of the fixed-point iteration and Newton-Gauss-Seidel methods was also compared. Both methods showed rapid convergence, with the Newton-Gauss-Seidel method requiring slightly more iterations per step. However, the difference in accuracy between the two methods was negligible, suggesting that the fixed-point iteration method might be a more efficient choice for this particular problem.

Conclusion

This study compared two implicit methods for solving the system of ODEs describing the dynamics of a drone swarm. The implicit backward Euler method with fixed-point iteration proved to be the most computationally efficient while maintaining accuracy and stability. The Newton-Gauss-Seidel method, although slightly slower, provided similar results with more robust convergence properties. Both implicit methods outperformed the explicit Euler method in terms of stability, energy conservation, and swarm formation accuracy.

Based on the results, it is recommended to use the implicit fixed-point iteration method for solving similar swarm dynamics problems, as it offers a good balance between computational efficiency and accuracy. However, for problems requiring higher robustness, the Newton-Gauss-Seidel method may be preferable. The findings also emphasize the importance of implicit methods in handling stiff systems and ensuring stable simulations in complex multi-agent environments.