Capstone Project: Robotics and Artificial Intelligence

Title

Swarm Robotics: Boids-Based Cleaning Robots Simulation

1. Project Overview

The goal of this project is to demonstrate how a group of simple autonomous robots can collectively achieve a complex task using swarm intelligence. Instead of a single vacuum robot moving randomly, multiple agents collaborate to clean an environment filled with dust and obstacles.

The project is inspired by the Boids model proposed by Reynolds (1987), which describes flocking behavior in birds. By applying similar rules—separation, alignment, and cohesion—to cleaning robots, we can observe how local interactions between agents lead to emergent global behavior.

This project relates to course topics in robotics fundamentals, AI algorithms, simulation, and autonomous systems, since it integrates local decision-making, group coordination, and performance evaluation within a simulated robotic system.

2. Technical Approach

Environment

A 60×60 grid is used to represent the environment.

Each cell can be dirty, clean, or an obstacle.

Robots move continuously in 2D space, but cleaning occurs when a robot passes over a dirty cell.

Agents (Robots)

Each robot is represented as a point agent with position and velocity. The robots follow simple local rules without centralized control.

Swarm Behavior Rules

Separation: Avoid collisions with nearby robots.

Alignment: Adjust velocity to match the average heading of neighbors.

Cohesion: Move toward the center of mass of neighboring robots.

Extensions

Obstacle avoidance: Robots are repelled by nearby obstacles.

Dust bias: Robots are slightly attracted to regions with higher concentrations of dirt.

Coverage metric: The percentage of cleaned cells relative to all accessible (non-obstacle) cells is computed over time.

This approach demonstrates how local rules + simple AI heuristics lead to global cooperation without centralized control.

3. Implementation Details

Programming Language: Python 3.10

Libraries: numpy, matplotlib

Visualization:

Left panel shows the environment:

Red circles = robots

Black = obstacles

White = dirty cells

Gray = cleaned cells

Trails = robot recent paths

Right panel shows the coverage percentage over time.

Interactivity:

Left-click = toggle dust on a cell

Right-click = toggle obstacle on a cell

Simulation Parameters:

Number of robots: 18

Max steps: 4000

Neighbor radius: 6 units

Separation radius: 2 units

Results
The swarm successfully cleaned nearly 100% of all accessible cells within ~4000 steps.

Coverage increased steadily, showing that swarm behavior prevented robots from clustering in a single region.
Compared to a single random-walk cleaner, the swarm achieved faster and more complete coverage.

(Here you should insert 1–2 screenshots: one of the environment with robots, one of the coverage graph.)
4. Results and Discussion
The results highlight the effectiveness of swarm robotics for distributed problem-solving. Even though each robot has limited perception and follows very simple rules, the collective group is able to achieve high performance on the cleaning task.

This emergent behavior is significant because it scales well: adding more robots improves coverage speed without requiring central coordination. Such principles are applicable in autonomous drones, search-and-rescue teams, and distributed cleaning systems.

Limitations:

The robots rely on local sensing only, which may cause temporary clustering.

No global optimization is performed; paths are emergent rather than optimal.

Performance depends on tuning of parameters (e.g., neighbor radius, weights of alignment/separation).

5. References

Reynolds, C. W. (1987). Flocks, herds, and schools: A distributed behavioral model. ACM SIGGRAPH Computer Graphics.

Brambilla, M., Ferrante, E., Birattari, M., & Dorigo, M. (2013). Swarm robotics: a review from the swarm engineering perspective. Swarm Intelligence, 7(1), 1–41.

Boids Algorithm – Wikipedia

Matplotlib Animation Documentation