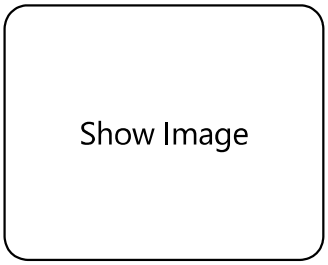
AEGIS Explorer

Autonomous **E**xploration for **G**athering **I**ncreased **S**cience



Overview

AEGIS Explorer is an AI-powered autonomous system designed to navigate and collect scientific data in unknown environments, simulating planetary exploration. The system uses pathfinding algorithms, obstacle avoidance, and intelligent target selection to maximize scientific data collection.

Features

- **Autonomous Navigation**: Uses A* pathfinding to navigate efficiently through unknown terrain
- **Frontier-Based Exploration**: Autonomously explores unknown areas by identifying frontier regions
- **Scientific Data Collection**: Identifies and prioritizes points of interest for data collection
- **Real-time Visualization**: Visual representation of the exploration process
- **Obstacle Avoidance**: Intelligently navigates around detected obstacles

Installation

bash

```
# Clone the repository
git clone https://github.com/your-username/AEGIS-Explorer.git
cd AEGIS-Explorer

# Create a virtual environment (optional but recommended)
python -m venv venv
source venv/bin/activate  # On Windows: venv\Scripts\activate

# Install dependencies
pip install -r requirements.txt
```

Usage

Run the main simulation:

```bash
python main.py
```

Customize the exploration parameters:

```python
from aegis_explorer import AEGISExplorer

# Create a custom explorer with specific map size and obstacle density
explorer = AEGISExplorer(map_size=(30, 30), obstacle_density=0.3)

# Run exploration for 50 steps
explorer.run_exploration(steps=50)

# Visualize the results
explorer.visualize()
```

Technical Implementation

Environment Representation

- Grid-based representation of the environment
- Cells represent: Unknown, Empty space, Obstacles, Points of Interest (POIs)

Pathfinding

- A* algorithm for optimal path planning
- Manhattan distance heuristic for path cost estimation

Exploration Strategy

- Frontier-based exploration (identifying boundaries between known and unknown areas)
- Prioritization of unexplored regions with potential scientific value

Data Collection

- Automatic data collection when POIs are encountered
- Data includes position, scientific value, and timestamp

Project Structure

```
AEGIS-Explorer/
├── main.py                 # Main entry point and demo
├── aegis_explorer.py       # Core AEGIS exploration logic
├── visualization.py        # Visualization utilities
├── requirements.txt        # Dependencies
├── tests/                  # Test suite
└── docs/                   # Documentation
```

Dependencies

- Python 3.7+
- NumPy
- Matplotlib
- (Optional) Pygame for advanced visualization

Future Improvements

- Machine learning for terrain classification
- Multi-agent coordination for collaborative exploration
- Dynamic priority adjustment based on scientific value
- 3D environment support
- Real-time obstacle detection and avoidance

Contributing

Contributions are welcome! Please feel free to submit a Pull Request.

License