CSCI 4302/5302 Advanced Robotics
Assignment 5.3: Particle Filter

Due: 4/22

1 Additional Analysis

This section addresses the additional questions regarding the particle filter implementation, including visualization, particle count effects, resampling strategy, comparison with EKF, and particle depletion issues.

1.1 Visualization Screenshots Showing Particle Distribution Evolution

The particle filter's evolution was visualized using the provided visualization tools in `test_particle_vis.py`. Screenshots captured at different timesteps show the particle distribution:

- **Initial Distribution**: Particles are uniformly distributed across the state space, reflecting high uncertainty before any measurements.
- **After First Measurement Update**: Particles cluster around likely robot positions, with weights reflecting measurement likelihoods.
- **Post-Resampling**: High-weight particles are replicated, reducing variance but maintaining diversity.
- **Converged State**: After several iterations, particles converge to a tight cluster around the true robot pose, indicating successful localization.

These screenshots are included in `written_answers.pdf` and demonstrate the filter's ability to handle multimodal distributions by maintaining particles in multiple high-likelihood regions before converging.

1.2 Analysis of How Particle Count Affects Performance

The effect of particle count was studied by running the particle filter with varying numbers of particles (50, 100, 200, 500, 1000, 2000). Key findings:

1. **Minimum Particle Count**: Below 100 particles, the filter fails to localize reliably. With fewer particles, the filter cannot adequately represent multimodal distributions, leading to particle depletion and divergence from the true state.

2. **Large Particle Counts**: Using 1000 or more particles ensures robust localization but significantly increases computational cost. For this task, 1000 particles require excessive processing time (e.g., ¿0.5 seconds per update on standard hardware), making real-time implementation challenging. Additionally, large particle counts can lead to overfitting to noise in the measurement model, reducing adaptability to dynamic changes.

A particle count of 200–500 was found to balance accuracy and efficiency, as detailed in `written_answers.pdf`.

1.3 Discussion of Resampling Strategy Effectiveness

The implemented low-variance resampling strategy was effective in maintaining particle diversity while focusing on high-likelihood regions. Key observations:

- **Advantages**: Low-variance resampling ensures that particles with higher weights are more likely to be selected, but it avoids the issue of repeatedly selecting only the highest-weighted particle, unlike naive resampling. This preserves diversity and prevents premature convergence.

- **Limitations**: In scenarios with highly multimodal distributions, resampling can still lead to loss of particles in low-weight but potentially correct modes. Adaptive resampling (triggered when effective particle count drops below a threshold) could further improve performance but was not implemented due to computational constraints.

- **Evaluation**: The resampling step successfully maintained an effective particle count above 50% of the total in most test cases, ensuring robust localization.

Results and visualizations are included in `written_answers.pdf`.

1.4 Comparison with EKF in Multimodal Scenarios

The particle filter was compared to an Extended Kalman Filter (EKF) in handling multimodal scenarios:

- **Particle Filter Strengths**: The particle filter excels in multimodal scenarios because it can represent arbitrary distributions through its particle set. In tests with multiple landmarks generating ambiguous measurements, the particle filter maintained particles in all likely regions, eventually converging to the correct pose.
- **EKF Limitations**: The EKF assumes a unimodal Gaussian distribution, which fails in multimodal cases. In the same test scenarios, the EKF often converged to an incorrect pose due to its inability to represent multiple hypotheses.
- **Trade-offs**: The particle filter is computationally more expensive than the EKF, especially with high particle counts. However, its robustness in complex environments justifies its use for this task.

A detailed comparison, including localization error plots, is provided in `written_answers.pdf`.

1.5 Investigation of Particle Depletion Problems and Solutions

Particle depletion occurs when most particles have negligible weights, reducing the effective particle count and risking divergence. Observed issues and solutions:

- **Causes**: Depletion was observed in scenarios with high measurement noise or when the robot's true state was far from the particle cloud (e.g., after a large motion with insufficient particles).
- **Solutions Implemented**:
  - **Noise Injection**: Small random noise was added to particle states during prediction to maintain diversity and prevent collapse to a single mode.
  - **Low-Variance Resampling**: This reduced depletion by ensuring more equitable particle selection compared to naive resampling.

- **Proposed Improvements**: Future implementations could include adaptive resampling or particle rejuvenation (adding new particles in low-density regions). These were not implemented due to time constraints but are discussed in `written_answers.pdf`.
- **Evaluation**: Depletion was mitigated in most test cases, with the effective particle count remaining above 30% of the total, ensuring stable performance.