Flying Partially Blind: POMDP Navigation with Noisy Sensors

Ganeshprasad Rajashekhar Biradar GBIRADAR@STANFORD.EDU
Haorui Zhang HAORUIZ@STANFORD.EDU
AA228/CS238, Stanford University

1. Goal

Our goal is to build a POMDP-based navigation system for a drone operating in a 2D environment, navigating from point A to point B while avoiding obstacles and dealing with uncertainty. This problem is interesting because many real-world use cases, such as search and rescue operations, autonomous delivery drones, and aerial photography, will continue to have similar challenges of navigation under uncertainty, and we wish to better understand how these problems are solved.

The environment itself will be a simulated, discrete-time, and discrete-space 2D world where the drone moves. Sensors providing information about the environment will have uncertainty, and obstacles will be present in the environment. Success will be measured by the drone's ability to reliably reach the goal while avoiding obstacles, balancing path efficiency with battery constraints, and handling sensor uncertainties robustly.

Haorui Zhang (the 4-credit contribution) will add multi-agent capabilities that will allow multiple drones to communicate with each other and base decision making on nearby drones as well. This will act as a collaborative group of drones looking to all accomplish their goals, either as a Multi-Agent POMDP (Pierre et al., 2023) or as a Decentralized POMDP (Bernstein et al., 2013).

2. Decision Making

The sequential decision-making problem comes from each drone determining how much thrust needs to be applied to navigate to its destination safely and quickly. We know this is a sequential decision-making problem because decisions about the thrust affect the drone's future state (ie., position). With the uncertainties in the environment and partially observable obstacles, it is impossible to calculate the navigation path as a single decision. The system is a POMDP where the drone has its own belief state, impacted by the uncertainties, and chooses its actions based on that belief.

The POMDP can be described as follows:

- **States:** $s = (x, y, v_x, v_y, b, w_x, w_y, O)$ where:
  - Drone position: $(x, y) \in \mathbb{Z}^2$ (coordinate vector, bounded by world size)
  - Drone velocity: $(v_x, v_y) \in \mathbb{Z}^2$ (velocity vector, where $|v_x|, |v_y| \leq V_{max}$)
  - Battery level: $b \in \mathbb{Z}$ and $b \in [0, 100]$ (percentage)
  - Wind velocity: $(w_x, w_y) \in \mathbb{Z}^2$ (velocity vector, unobserved by agent, inferred)
  - Obstacle positions: $O \subseteq \mathbb{Z}^2$ (static, unobserved beyond sensor range)
- **Actions**: Discrete acceleration commands $a = (a_x, a_y)$ where:
  - Drone acceleration: $(a_x, a_y) \in \mathbb{Z}^2$ (acceleration vector, where $|a_x|, |a_y| \leq A_{max}$)
- **Transitions:** Given state $s_t = (x, y, v_x, v_y, b, w_x, w_y, O)$ and action $a_t = (a_x, a_y)$:

- Velocity update: $v_x' = \text{clip}(v_x + a_x + w_x, -V_{max}, V_{max})$, $v_y' = \text{clip}(v_y + a_y + w_y, -V_{max}, V_{max})$
- Position update: $x' = x + v_x'$, $y' = y + v_y'$
- Battery depletion: $b' = b - \alpha \cdot \sqrt{a_x^2 + a_y^2}$ (proportional to acceleration magnitude)
- Wind evolution: static, $(w_x', w_y') = (w_x, w_y)$, agent belief is updated with $w'_{x,inferred} = x'_{obs} - (x + v_x + a_x)$ and $w'_{y,inferred} = y'_{obs} - (y + v_y + a_y)$
- Collision handling: If $(x', y') \in O$ then $v_x' = 0$, $v_y' = 0$ and receives collision penalty
- Obstacles: static, $O' = O$, agent belief is updated with readings from LiDAR

- **Rewards**:
  - $R_{goal}(s) = r_{goal}$ if $(x, y) == (goal_x, goal_y)$ (large positive reward for reaching the goal)
  - $R_{collision}(s) = -r_{collision}$ if $(x, y) \in O$ (large negative penalty for collisions)
  - $R_{progress}(s) = \frac{k}{\epsilon + \sqrt{(goal_x - x)^2 + (goal_y - y)^2}}$ (positive reward for distance to goal)
  - $R_{step}(s) = -r_{step}$ (small negative penalty for time inefficiency)
- **Observations:** $o = (o_{pos}, o_{bat}, o_{lidar})$ where:
  - Position (GPS): $o_{pos} = (x, y) + \mathcal{N}(0, \sigma^2_{GPS} \cdot I_2)$ (noisy reading of true position)
  - Battery: $o_{bat} = b + \mathcal{N}(0, \sigma^2_{bat})$
  - LiDAR: o_{lidar} = ranges to obstacles in k directions (e.g., 8 compass directions) within view radius R, with Gaussian range noise

3. Sources of Uncertainty

We have three main sources of uncertainty: sensor uncertainty (from the GPS, Battery, and LiDAR readings with Gaussian noise), wind uncertainty (no sensor and so it is inferred by the agent), and obstacle uncertainty (the LiDAR has a limited range, limited viewing directions).

We will model wind as vector field that is spatially-varying but temporally static, generated using two Perlin noise maps (magnitude and direction). This ensures that we have spatial coherence that varies smoothly across the environment where the agent must infer through observation.

4. Sketches of Solution

We will establish performance bounds using two baseline methods:

MDP Baseline: Assumes full observability without uncertainties and full perfect knowledge of the environment, providing an upper bound on evaluation metrics.

Greedy Policy: A simple policy that maximizes progress towards goal, without considering wind and uncertainties, providing a lower bound on evaluation metrics.

These approaches will be evaluated using the following metrics: number of successful runs, average reward, path efficiency, and time per action (average and 99th percentile). Each approach will be tested across three difficulty levels (easy, medium, hard) with varying obstacle densities, wind conditions, and sensor noise levels.

We plan on implementing online POMDP using POMCP, Partially Observable Monte Carlo Planning, from the pomdp-py python package (Zheng and Tellex, 2020). We expect this approach to outperform the Greedy lower bound and approach the MDP upper bound.

References

Daniel S Bernstein, Shlomo Zilberstein, and Neil Immerman. The complexity of decentralized control of markov decision processes, Jan 2013. URL https://arxiv.org/abs/1301.3836.

Jean-Elie Pierre, Rafael Fierro, and Xiang Sun. Multi-agent partial observable safe reinforcement learning for counter uncrewed aerial systems — ieee journals & magazine — ieee xplore, Jul 2023. URL https://ieeexplore.ieee.org/abstract/document/10193774.

Kaiyu Zheng and Stefanie Tellex. pomdp py: A framework to build and solve pomdp problems, Apr 2020. URL https://arxiv.org/pdf/2004.10099.pdf.