Urban Navigation Decision-Making and Localization for a Small, Highly-Maneuverable Aircraft

1st Tim Behrer
Dept. of Aerospace Engineering
University of Colorado Boulder
Boulder, United States
tibe0349@colorado.edu

2nd Christopher Kong
Dept. of Aerospace Engineering
University of Colorado Boulder
Boulder, United States
chko2537@colorado.edu

3rd Tsuening Lee
Dept. of Aerospace Engineering
University of Colorado Boulder
Boulder, United States
tsle8661@colorado.edu

***Abstract*—The recent improvement in Unmanned Aerial Systems (UAS), more specifically drones, capabilities provides the opportunity for the introduction of UAS into various important applications. With this opportunity, the problem of decision-making in unknown environments with uncertain sensor data acquisition, commonly referred to as POMDP, is an active area of growth and development. The proposed system architecture formulates decision-making processes to enact navigation through an unknown urban environment for a small and highly maneuverable rotor-craft. A custom-built state space to simulate an unknown, urban environment is used to test both online and offline solvers. The reward structure for these solvers focuses on rapid localization and safety of possible bystanders. Success is demonstrated through means of small-size state space testing and a high quantity of reward rollouts to reduce uncertainty in POMDP solver capabilities.**

I. Introduction

In recent years, drones or Unmanned Aerial Vehicles (UAVs) have started to be widely used in real-life operations. From inspections, disaster relief, rescue operations, camera and cinema work, and package delivery, UAVs have entered both the industrial and commercial space. UAV operations are reliably successful in clear airspace and GPS-supported areas in which a ground station can waypoint commands for the vehicle to follow; however, they can become quite dangerous in situations where this is not the case. This is because UAVs rely on the use of Global Navigation Satellite Systems (GNSS) to determine their current and future states accurately. Nonetheless, tall buildings and obstructions may obstruct GPS lock, adding uncertainty to their desired flight paths. Furthermore, this type of environment not only impedes communications but may reflect the transmission and reception of signals, causing signal occlusion and multipath effects. In this event, a drone without any alternative navigation capabilities may have an unreliable position estimate, which critically degrades its localization and trajectory execution accuracy. In the worst-case scenario, a fatal collision or crash may be a likely outcome.

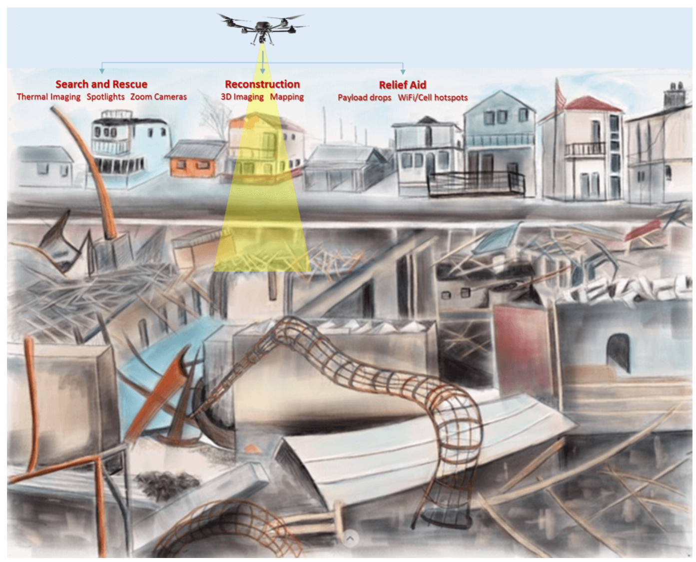

Fig. 1. Example Depiction of Aerial Assist from Unmanned Autonomous Vehicles. [1]

This raises the valid topic of an effective way to handle uncertain environments and GPS-denied functionality. For applications in which autonomy is used, but decisions are sequential and absolute, Markov Decision-Making Processes (MDPs) are known to be quite effective. However, due to the high levels of uncertainty in the prior defined environments, a more mathematically complex form called POMDPs or Partially Observable Markov Decision Process can be implemented. POMDP solution methods are a new and growing area of research, and the ability to handle large environments with many observations improves consistently. That being said, POMDP solution methods and problem definition can theoretically be implemented to handle real-world navigation in uncertain environments.

Many robotics problems involving uncertainty have been through POMDP methods. Through time, POMDPs were solved using classical algorithms and solvers that often come with expensive computation time and/or limitations in environment operations, such as POMDP solvers' lack of ability to properly handle abundantly large continuous spaces rather than smaller discrete spaces. Thus, for a small and highly maneuverable aircraft with an expectedly large navigation state space, to be able to determine its actions under uncertainty is an interesting and complex challenge.

To cope with such scenarios where navigation accuracy may be degraded, causing fatal collisions or crashes, this paper addresses the problem of safe path planning for UAV urban operation. Motivated by safe decision-making for small aircraft, this project strives to develop a decision-making algorithm to safely and efficiently navigate an urban environment in a partially observable environment defined by a pre-determined sensor package. The drone must be able to locate a target in an unknown position and also localize itself without GPS. Formulated as a POMDP problem, the Julia coding language is used to stage a pre-existing/pre-defined environment. A target's destination is defined, therefore, there is a terminal path for the drone to take. To complete the mission, the simulated drone must localize its position concerning nearby obstacles while simultaneously finding the target's unknown position efficiently and safely.

II. Background and Related Work

The motivation to implement POMDP solutions into uncertain, cluttered, and/or GPS-denied environments is such that it would allow for widespread implementation of safe, reliable, and rapid use of multi-rotors in adverse environments. Theorize that a hiker is lost in the woods, miles from cell service, and injured. What would usually take weeks and a search party of hundreds of people could be accomplished in days with a handful of autonomy-capable drones. Instead of organizing a search pattern comprised of an abundance of people, a few officials could deploy drones to explore quadrants spanning miles, capable of pinpointing the exact location and relaying that information to a singular search and rescue team. Subsequently, this could reduce the risk that said hiker is lost for long periods, thus reducing the chance that a person is injured further or killed. Consider another possibility, an earthquake strikes a major city, completely altering the once-known environment and making navigation slow and methodical. A drone capable of autonomously making decisions through a POMDP solver can rapidly navigate this unknown environment, recording and relaying changes to the environment, potential victims, and areas of critical conditions (Structural instability, populated areas, etc.). This can be completed without risk to a human observer, reducing the risk of further injury. This is the driving cause behind the development of autonomous capabilities in adverse environments.

Juan Sandino [2] demonstrates the existence of uncertainty in drone sensor readings in Figure 2, highlighting the common realization that POMDP drone localization is a potentially life-saving area of research and development.

The POMDP environment is suitable for defining a state space in which a robotic vehicle can adaptively determine optimal policies to reach optimal rewards effectively. This

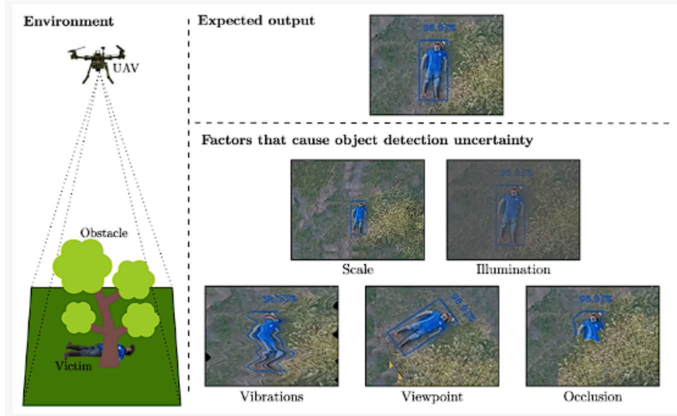

Fig. 2. Demonstration of Uncertainty in UAS Sensor Measurements. Credit: [2]

implementation has been realized in prior experiments and is an active area of research. A similar consideration by Fernando Vanegas involves the implementation of POMDP solvers in cluttered and GPS-denied simulations and found that "Experimental results show that the system is robust enough to overcome uncertainties that are present during a flight mission in a GPS-denied and cluttered environment" [3]. The prior research along with Ory Walker's paper on Multi-Agent UAV Exploration and Target-Finding introduces the use of Adaptive Belief Trees (ABT), defined as "a recent online POMDP solving algorithm... capable of solving complex POMDP problems with significant state-spaces despite potential changes in the problem space by efficiently adapting previously generated policies during run-time to incorporate new information and produce near-optimal solutions" [4]. ABT is an extremely similar application to POMCP methods, however, ABT methods are optimized for C++ languages while POMCP algorithms are optimized for languages such as Julia or Python. Therefore, the use of POMCPOW solution methods, POMCP with weighted particle mixture beliefs: a more intricate solution approach but similar.

III. Problem Formulation

A Markov Decision Process (MDP) is a mathematical framework that models sequential decision-making problems affected by uncertainties. However, in the case of unmanned aircraft vehicles, taking in information about the environment and deciding what series of actions to take brings up realistic uncertainties. The drones have limitations in sensors and perceptions, causing the state to not be completely accurate. These cases of insufficient information lead to deviations from the real state and therefore encourage the use of Partially Observable Markov Decision-Making Processes. POMDP was used throughout the work of this project as a means to incorporate the uncertainties in the sensors and location coupled with the partial observability of the UAV, target, and obstacles throughout the environment. A POMDP's formal definition consists of the following elements:

$$DronePOMDP = (\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \mathcal{Z}, \mathcal{O}, \gamma)$$

$\mathcal{S}$ represents the set of states in the environment. $\mathcal{A}$ represents the set of actions the UAV drone can execute. $\mathcal{T}$ is the transition function for the state after taking an action. $\mathcal{R}$ stands for the set of rewards for every given state and action. $\mathcal{O}$ is the set of observations that return from the UAV drone sensors. $\mathcal{Z}$ is the distribution function that describes the probability of an observation, o returning from a given state s after taking an action a. Lastly, γ is the discount factor. Each will be discussed further below.

A. State Space

The state space environment is a three-dimensional grid world of size $[5, 5, 5] = [\vec{x}_{max}, \vec{y}_{max}, \vec{z}_{max}]$, this size is equivalent to 125 possible state locations. Upon initialization, the drone, target, and bystander occupy a random location within the state space. Also within the state space, 20 obstacles are initialized, which are meant to model buildings within an urban environment. These buildings have various heights, as expected in various urban environments. An example of such an environment can be seen below in 3 where gray cubes are the obstacles, the green pyramid is the target, the red pyramid is the bystander, and the blue sphere is the controlled drone.

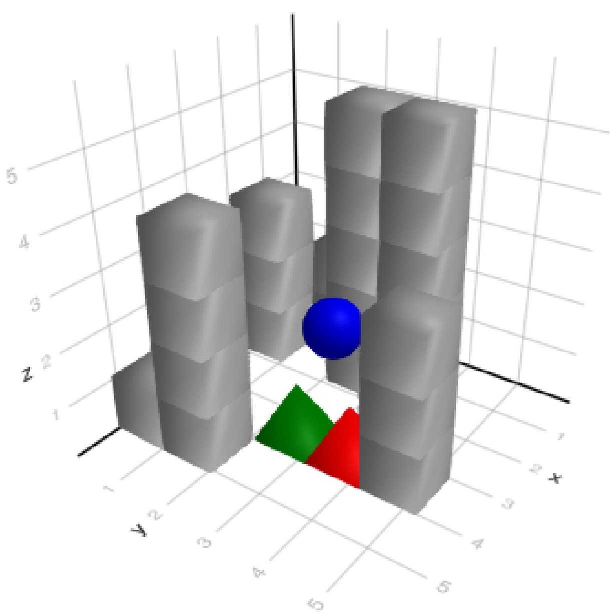

Fig. 3. Example Initial State Space - ***Blue: Drone, Green: Target, Red: Bystander, Gray: Obstacles. Makie: [5]***

The state space encapsulates the positions of three main subjects. The subjects include the drone, the target, and the bystander. Each position is defined by a vector consisting of the X, Y, and Z positions. The drone is the subject to which the POMDP solver will be applied, the target and the bystander both move randomly and cannot leave the ground. For example, the position of the drone in any state is defined as follows:

$$\text{Drone Position} = \{X, Y, Z\}$$

For simplification purposes, the state space is defined in a local inertial frame in which $\{0, 0, 0\}$ is the {*backward, left, up*} corner of the state space. Therefore, {*forward, right, down*} is defined as a positive change within the local inertial frame, as diagrammed in Figure 4. Finally, the terminal state occurs when the drone is at the X and Y position of the target.

B. Action Space

The action space describes the six possible directions the drone can move across the three axes. The three-axis is a common right-hand coordinate system for an aircraft.

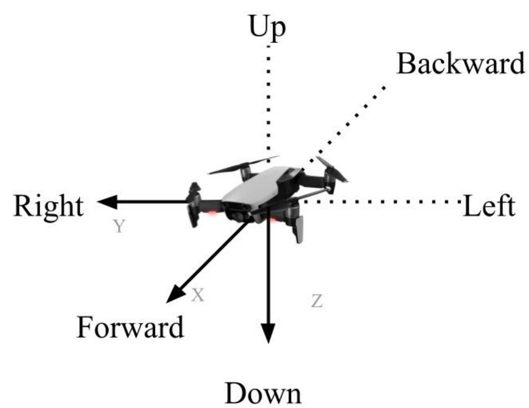

Fig. 4. 3-Axis Drone Diagram

Here, the drone moves in the X direction by the *forward* and *backward* actions. The drone moves in the Y direction by the *right* and *left* actions. Lastly, the drone moves *down* and *up* along the Z axis. In the case of movement, since the environment was modeled by a grid world, the action to move will increment the position by one grid unit. The last action is the *measure* action, this performs an observation and will be discussed in a later section. Thus, the action space is defined as follows:

$$\mathcal{A} = \{forward,\ backward,\ left,\ right,\ up,\ down,\ measure\}$$

C. Transition

It was decided that the transitions were to be deterministic for the drone. In application, drones tend to be fairly reliable when it comes to moving between points, whether that be horizontally or vertically. As a result, any uncertainty when transitioning between positions was removed. Additionally, the drone was able to move in the X, Y, and Z directions. On the other hand, the target as well as the bystander were limited to the lowest Z level and could only move in the X and Y directions.

D. Rewards

There are five rewards for the Drone POMDP. If the drone reaches the target, the reward is +200. The drone has achieved its mission objective and therefore will receive the biggest reward. If the drone reaches and flies in the same space as a bystander, the reward is -5. Here, the bystander acts as an obstacle that the drone does not want to fly over as it risks a safety hazard. If the drone measures and observes its surroundings, the reward is -2. Otherwise, moving the drone uses up valuable resources such as battery power and data space and thus returns a reward of -1. Lastly, once the POMDP reaches a terminal state, the return is 0. This can be summarized below:

$$\mathcal{R}(s,a) = \begin{cases} 0 \text{ if } s = \text{terminal} \\ +200.0 \text{ if } s = \textit{target} \\ -5.0 \text{ if } s = \textit{bystander} \\ -2.0 \text{ if } a = \textit{measure} \\ -1.0 \text{ otherwise} \end{cases} \tag{1}$$

Furthermore, the drone and target only need to have the same states in the X and Y state vectors and not in the Z direction. This is because the drone will assume to have localized and found its target when it is directly above the target with a down-facing camera. This shared X and Y state vector was also applicable to the drone and the bystander to discourage flying overhead of the bystander and possibly inducing a safety hazard.

E. Observations

Observations are performed by the drone at discrete times. The actual observation action is defined by a *sensorbounce* function. This function takes the distance between the current state and the nearest object or boundary in all six directions, shown in Figure 5 with dashed red lines, and outputs that range measurement with some uncertainty. It then returns a *SparseCat* object of these observations, equivalently returning a categorical distribution with uncertainty. This distribution is returned and weighed by the solver to determine how to handle the uncertainty. The drone can decide to perform the *measure* action as mentioned in section III-B, this will always return an observation to the drone. However, there is also a ten percent chance that the drone gathers an observation without performing the measurement action. Allowing for some observation to be made on "movement" actions such as *forward* and the others defined in section III-B.

F. Discount Factor

A discount factor γ of 0.95 was selected based on prior papers and published work. In Ref 3, the discount factor of 0.97 was selected based on numerous simulations and flight tests and taking into account the distance traveled by the UAV at every step. Thus for this small scale test, the discount factor of 0.95 was reasonable.

IV. Solution Approach

A. QMDP

A custom-constructed QMDP solver that follows the typical construction of a QMDP solver as commonly implemented and supported in [6] and [7]. QMDP solution algorithms are well-developed and understood solution approaches. Essentially, core QMDP solvers are offline algorithms that compute the alpha-vectors for a given POMDP state space. The key approximation of a QMDP solver is that the "future" environment is fully observable. There exist key implications of these assumptions: firstly, QMDP solvers do not tend to reach an optimal policy should "costly" or critical information gathering occurs within the state space. Secondly, as a direct result of the fully observable future assumption, state spaces with long-term uncertainty also tend not to be efficiently solved by QMDP algorithms. With this in mind, for the described state space environment above, it is expected that the QMDP algorithm will perform fairly well with the current observation, transition probabilities, and rewards.

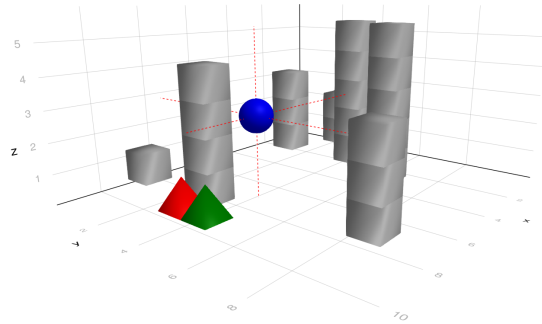

Fig. 5. Render of Observation/"Measure" Action - ***Red Dashed Lines: Range Observations, Blue: Drone, Green: Target, Red: Bystander, Gray: Obstacles. Makie: [5]***

However, as this solution method is an offline solution method and thus computational time is quite large, the feasibility of implementing this algorithm within real flight software is suboptimal. Extensive assumptions of sparsity and/or applications of alpha vector pruning would need to be implemented before utilization in rapid implications. For these reasons, other feasible solution algorithms should be considered for desired applications.

B. POMCPOW

The Julia package POMCPOW.jl was utilized as an off-the-shelf solution. As the name Partially Observable Monte Carlo Planning with Observation Widening suggests, POMCPOW is a modification of the POMCP algorithm and seeks to limit the number of branches in the POMCPOW tree. This is done through the use of double progressive widening which sees a gradual expansion of state and action nodes along with weighted particle filtering. Thus, POMCPOW is a more robust algorithm than POMCP regarding adverse state space POMDP's. Therefore, by applying POMCPOW algorithmic solvers the benefits of ABT, (see Figure 6), previously mentioned in section II are capitalized upon.

There are various expected benefits of applying a POMCPOW solver. The first is such that POMCPOW is an online POMDP method. This allows the developed solver to be feasibly implemented in real-time on drones with large state

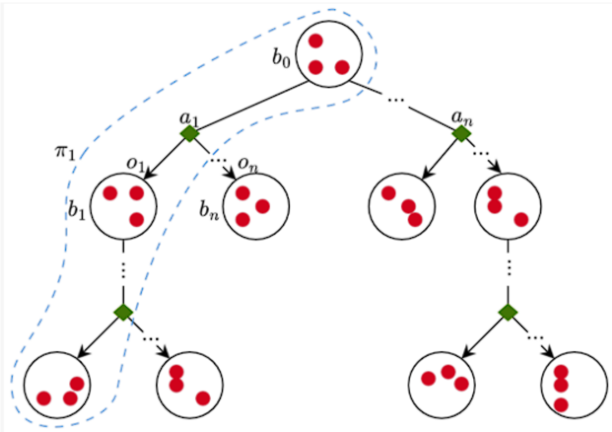

Fig. 6. Diagram Representing the ABT/POMCP Tree Commonly Used in Drone Localization State Models. [2]

spaces, as expected in true flight. The POMCPOW solution method is also applicable to continuous observation spaces, where POMCP is not as readily applicable. Subsequently, as UAS rescue/localization efforts typically have continuous sensor measurements: Bluetooth, Lidar, Computer Vision, etc., POMCPOW is likely a more practical real-world applicable solution method than POMCP.

V. Results

To quantify the success of the QMDP and POMCPOW solvers, the respective policies were rolled out 300 times, and the cumulative reward was recorded at the end of each trial. Mean is the quantification of how well the described solvers will do on average The results of this rollout function are displayed in Table I:

TABLE I
Solution Approach Results

Method	**Mean Reward**	**Standard Deviation**	**Standard Error of the Mean**
QMDP	65.07	78.45	4.53
POMCPOW	70.91	71.90	4.15

From the generated policies, a higher mean reward would correlate with a more successful policy as it would mean the drone was reaching the target reward faster. On a similar note, the standard deviation informs how consistent the solver was in reaching the target, with a smaller number meaning a more consistent policy. It can help show how often the reward was actually reached and how fast that tended to occur across all the rollouts. In order to explore some uncertainty, the standard error of the mean was also calculated. This parameter shows the bounds of the true mean parameter which is the mean +/- the standard error of the mean. Therefore, more conclusions can be drawn about the mean as its associated error is noted and understood.

The results are also visualized in Figure 7:

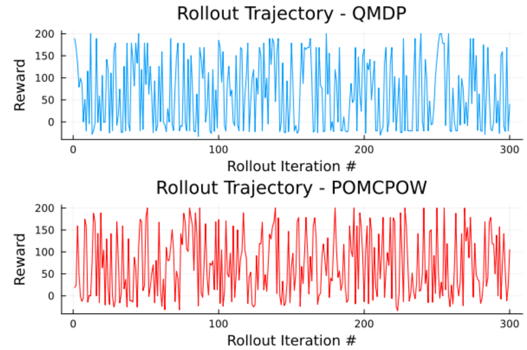

Fig. 7. Reward Trajectories Throughout 300 Length Rollout

Lastly, Table II, is used to highlight the difference between the overall computational time of QMDP and POMCPOW. This was found with a more generalized time keeping method.

TABLE II
Policy and Rollout Computational Time on Desktop with *32GB Ram and AMD Ryzen 5 5600 6-Core Processor*

Method	**Policy Generation Runtime [minutes]**	**Rollout Runtime [minutes]**
QMDP	32	16
POMCPOW	1	13

A. Discussion

From the solution approach results in Table I, it is evident that the POMCPOW solution is expected to be an improvement compared to the QMDP solution, based on 300 rollouts. This 8.97% increase in mean performance highlights the QMDP solver's inherent limitations, but also further demonstrates the effectiveness of POMCPOW in managing complex, information-rich environments. This increase also demonstrated that POMCPOW's adaptive decision-making framework is better equipped to dynamically integrate new information and adjust its strategies accordingly. This capability is particularly beneficial in scenarios where rapid and accurate decision-making is critical under uncertainty, as applied within this state space.

Also shown in Table I, the standard deviation is lower for the POMCPOW solver, this lower variance in rollout performance suggests that the POMCPOW solver is slightly more reliable at this scale. This reliability can likely be attributed to the robustness of POMPCPOW solvers to uncertainty and noise, which is due to observational widening. This likely moderately lowered the standard deviation in the more novice state space described in III-A compared to QMDP. However, in real drone applications, it is expected that the adaptive planning, full consideration of a partially observable state space, and robustness would have a much more substantial effect. Therefore, it is determined that the POMCPOW solver greatly outperforms QMDP in terms of robustness and variance as measured by standard deviation.

The standard error of the mean (SEM) values, 4.53 for the QMDP solver and 4.15 for the POMCPOW solver suggest that the estimates of the average performance for QMDP exhibit slightly greater variability compared to POMCPOW. This higher SEM in QMDP might indicate less consistent performance or a smaller sample size used in the evaluation, this could be attributed to the known inability of QMDP assumptions to handle critical information gathering or long-term uncertainty. Whereas POMCPOW's lower SEM suggests more stable performance outcomes across different scenarios or trials. It is believed that the POMCPOW performed better in the described state space because

In terms of computational power analyzed for the theoretical application in real-time drones, POMCPOW greatly outperformed the QMDP solver. The runtime of the QMDP and POMCPOW method on a desktop computer as shown in Table II as $\sim$16 and $\sim$13 minutes respectfully confirmed that the POMCPOW solution method is more capable in large state spaces in terms of computational requirements. This, confirms the expectation that for applications of real drones with extremely large state spaces, QMDP would not be feasibly done in real-time, meaning that POMCPOW is the only feasible option based on computational methods.

VI. Conclusion

This project demonstrates the application of POMDPs to model unmanned aerial vehicles in cluttered urban environments. The system evaluated within a Julia 3D grid world sandbox created obstacles, bystanders, and targets for the UAV to explore and travel around. Tasked with finding the target in an outdoor environment coupled with uncertainty and partial observability, the UAV traversed a $[5x5x5]$ space with six degrees of freedom given an arbitrary departure position. Then two proposed POMDP-based planning and solutions algorithms, QMDP and POMCPOW, were evaluated over three hundred simulations to reach a cumulative reward mean of 65.07 and 70.91 respectively. Given two distinct solvers, the POMCPOW resulted in an improvement in rewards compared to the QMDP solution. With this in mind, POMCPOW presents itself as a more optimal solution, as its smaller standard deviation is reliable for drone use. Lastly, the POMCPOW online method of solving would prove less of a computational burden for in-the-field drone path planning compared to the offline method of QMDP. Although the simulations run in a simplified grid world, this project proves that a virtual POMDP-based navigation solution can be generalized to safe target tracking and localization in more complex, large-scale, potential GPS-denied operations.

A. Future Work

The environment space was limited to a three-dimensional grid world of size $[5x5x5]$ due to a lack of computing power. There are a variety of improvements and changes that could be employed to further explore the problem. For starters, simply increasing the size of the three-dimensional space could generate some interesting results. In addition to increasing the size of the grid world, the number of bystanders and obstacles could also be increased to further reflect a more realistic scenario. An area of possible experimentation would also be the testing of the performance of the small state-space solver on larger state spaces, this would demonstrate the capability to scale the small-scale developed policies for various solvers. Second, adding critical observations by making collisions with obstacles terminal would increase the complexity of the problem. As it stands now, the drone simply bounces off obstacles. Another area for growth is altering the existing observation space. Many current sensors are continuous and report a "confidence value" to demonstrate their reported uncertainty in their measurements. Therefore, simulation and analysis of this behavior would provide a relative understanding of the capability of various solvers to handle this observation space. Also, as currently modeled, the simulated "measure" action simultaneously takes measurements in all six directions, an area of improvement to resemble real drone flight would be the introduction of attitude changes and fewer sensors. This would better model real environment UAS flight modeling. Finally, the ultimate change would involve making the POMDP continuous, this would require a variety of new solution approaches as QMDP would no longer be a viable choice.

VII. Contribution

Timothy Behrer led the development and enhancement of the grid-world environment for the drone localization process. He also assisted with the development and enhancement of the POMDP algorithm implementation. Finally, he helped collect and present data.

Christopher Kong led the collection and presentation of data. He also assisted with the development and enhancement of the grid-world environment for the drone localization process. Finally, he also assisted with the development and enhancement of the POMDP algorithm implementation.

Tsuening Lee led the development and enhancement of the POMDP algorithm implementation. He also assisted in the development and enhancement of the grid-world environment for the drone localization process. Finally, he helped collect and present data.

A. Release

The authors grant permission for this report to be posted publicly.

Acknowledgment

References

[1] G. D. Conference, "How drones are used in disaster management," Dec 2022.

[2] J. Sandino, F. Maire, P. Caccetta, C. Sanderson, and F. Gonzalez, "Drone-based autonomous motion planning system for outdoor environments under object detection uncertainty," *Remote Sensing*, vol. 13, no. 21, 2021.

[3] F. Vanegas and F. Gonzalez, "Enabling uav navigation with sensor and environmental uncertainty in cluttered and gps-denied environments," *Sensors*, vol. 16, no. 5, 2016.

[4] O. Walker, F. Vanegas, and F. Gonzalez, "A framework for multi-agent uav exploration and target-finding in gps-denied and partially observable environments," *Sensors*, vol. 20, no. 17, 2020.

[5] S. Danisch and J. Krumbiegel, "Makie.jl: Flexible high-performance data visualization for Julia," *Journal of Open Source Software*, vol. 6, no. 65, p. 3349, 2021.

[6] JuliaPOMDP, "Juliapomdp/pomdps.jl: Mdps and pomdps in julia - an interface for defining, solving, and simulating fully and partially observable markov decision processes on discrete and continuous spaces.."

[7] Z. Sunberg, "Cu-adcl/dmustudent.jl: Julia package for students in decision making under uncertainty."

[8] C. Wang, J. Wang, Y. Shen, and X. Zhang, "Autonomous navigation of uavs in large-scale complex environments: A deep reinforcement learning approach," *IEEE Transactions on Vehicular Technology*, vol. 68, pp. 2124–2136, March 2019.

[9] F. Vanegas, K. J. Gaston, J. Roberts, and F. Gonzalez, "A framework for uav navigation and exploration in gps-denied environments," in *2019 IEEE Aerospace Conference*, pp. 1–6, 2019.

[10] J.-A. Delamer, Y. Watanabe, and C. P. Chanel, "Safe path planning for uav urban operation under gnss signal occlusion risk," *Robotics and Autonomous Systems*, vol. 142, p. 103800, 2021.

[11] C. Yun and S. Choi, "Visual localization and pomdp for autonomous indoor navigation," 2014.

[12] H. Kurniawati and V. Yadav *An online POMDP solver for uncertainty planning in Dynamic Environment*.

[13] Z. Sunberg and M. Kochenderfer, "Online algorithms for pomdps with continuous state, action, and observation spaces," 2018.