Optimal Policies in Complex Large-scale UAS Traffic Management

1st David Sacharny
School of Computing
University of Utah
Salt Lake City, USA
sacharny@cs.utah.edu

2nd Thomas C. Henderson
School of Computing
University of Utah
Salt Lake City, USA
tch@cs.utah.edu

***Abstract*—There is currently a worldwide effort to develop UAS Traffic Management (UTM) systems that help ensure safe and reliable operation of Unmanned Autonomous Systems (UAS) in urban environments. A large number of factors must be considered in planning such flights, including GIS (roads, topography, etc.), weather (temperature, wind, precipitation), localization and navigation (GPS, V2X communication), infrastructure obstacles (buildings, towers, etc.), excluded zones of operation, etc. We have developed a cloud-based geospatial intelligence system, BRECCIA, which brokers such information among a set of intelligent agents. In this work, an extended version of BRECCIA is proposed as a universal-UTM (U-UTM) which allows the specification of urban airways constrained to be above roadways. In addition, we develop a reinforcement learning approach for the determination of optimal flight policies through such airways, where these policies can take into account a variety of factors (wind, precipitation, communication, etc.) which impact UAV path following capabilities. A novel context-based probabilistic state transition function is introduced. Simulation experiments are performed to demonstrate the performance of the approach.**

***Index Terms*—UAS Traffic Management, Reinforcement Learning**

I. Introduction

The state of Utah through the Utah Department of Transportation is, like many other locales around the world, developing plans to exploit the Class G uncontrolled airspace for commercial and other applications to enable *Urban Air Mobility*. To that end, the state is working to develop an adequate infrastructure to support such operations by providing real-time road-weather information (RWIS), micro-radar sensors, differential GPS, and dedicated short-range communications radios (DSRC). In addition, the state has decided that UAVs airways will be located directly above existing ground road systems. Figure 1 shows a set of airways defined by our Universal-UAS Traffic Management (U-UTM) system, based on **BRECCIA**, a geospatial intelligence system that we previously developed in the framework of the Air Force Office of Scientific Research (AFOSR) DDDAS (Dynamic Data Driven Application Systems) research program [4], [8], [9].

As specified by the FAA and NASA, a valid UTM must provide a framework which brokers information among UAS operators and UAS System Services. A number of UTMs exist, including one by AirMap [3], and Low's Modular UTM Framework (see [5]). The latter is to be deployed in Singapore and provide urban airspace management, risk and flight management, as well as interfaces between the UAV and operators, and logistical support as well (e.g., authentication, registration, etc.). AirMap's goal is to provide USS services globally, including user interfaces, flight and traffic information, geographic and registry capabilities.

Fig. 1. U-UTM Defined Airways for a Portion of Salt Lake City, UT

Others have used reinforcement learning to study various aspects of the UAS path planning problem. Wulfe [10] considers the problem of UAV collision avoidance, and shows that the Deep Q-Network [6] outperforms value iteration in terms of safer and more efficient policies, and is computationally less complex as well. Brittain and Wei [1] have described the use of Deep Reinforcement Learning to provide "air traffic control sequencing and separation." They use the NASA Sector 33 app to simulate flights and demonstrate that the required separation can be met efficiently. Their goal is to address increased air traffic in traditional controlled airspace with respect to VTOL operations in the low-altitude air taxi and air mobility space. Finally, for a broad review of the use of Deep learning methods in UAV applications, see Carrio et al. [2].

This research supported in part by Dynamic Data Driven Application Systems AFOSR grant FA9550-17-1-0077.

II. Reinforcement Learning Approach

The goal here is to determine an optimal action selection policy for a UAV with a given destination goal and a set of specific environmental conditions which defines the state space. The agent must operate successfully in this environment by learning a utility function on the state of the world and from those utilities determine an optimal action policy for each state. We consider an agent in a fully observable environment. Once a policy, π, is learned to maximize utility, $U(s)$, then it deterministically specifies the action for each state, and the agent will always choose action $\pi(s)$ for state s. The goal is to maximize the expected utility (see [7] for details). The utility for each state is defined by the Bellman equation:

$$U(s) = R(s) + \gamma max_{a \in A(s)} \sum P(s' \mid s, a) U(s')$$

where $U(s)$ is the utility of state s, a is an action, $A(s)$ is the set of actions possible for state s, and P is the probability of state s' given state s and action a. We use *value iteration* to solve for the state utilities; i.e., the above equation is iterated, updating the utility of each state until convergence is achieved. Once the utilities are known, the optimal policy at each state corresponds to the action which maximizes the expected utility from the action:

$$\pi^*(s) = argmax_{a \in A(s)} \sum_{s'} P(s' \mid s, a) U(s')$$

A. State Representation

In order to solve for the optimal policy for UAV control, we define the state space as:

$$S = \mathcal{Z}^3 \times \Re^3 \times \Re^+ \times \Re$$

where the space is composed of three integer grid coordinates, a real-valued 3D wind vector (whose magnitude is the wind speed), a precipitation value, and a temperature value. Note that although the wind, precipitation and temperature values have difference dimensions, their values are represented by indexes that designate intervals in the appropriate range. For the study here, the grid consists of a 4x4x4 set of voxels (representing airspace volumes where the specific dimensions of the air volumes are determined by the problem under consideration; here we assume reasonably large volumes), 2 values are used to indicate the wind (none, high), precipitation is binary (raining or not), and 3 values for temperature (cold, normal, hot). Thus, the state vector is a 6-tuple, where the number of values for each element is [4,4,4,2,2,3], resulting in a total of 768 distinct states.

B. Possible Actions

The action set for this problem is simply the selection of a neighboring air volume; in particular, one of the 6 orthogonal direction cells. These actions will be labeled {*X*, *-X*, *Y*, *-Y* , *Z* , *-Z*}, so that these actions align with a standard frame in the center of the cell (see Figure 2).

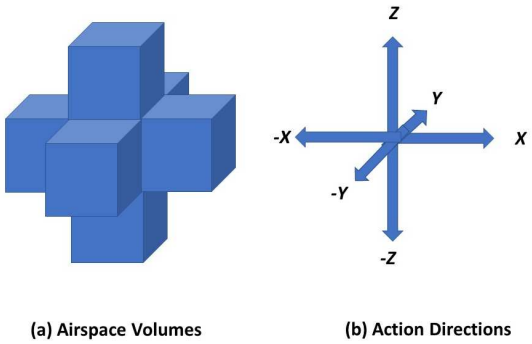

Fig. 2. (a) Airway Volumes; (b) Action Directions.

C. Probabilistic State Transition Function

Next, a probabilistic state transition function, $P(s' \mid s, a)$, is defined which provides the probability that state s' results from choosing action a while in state s. The particular function used here is based on the physics of the state transitions, accounting for the impact of motion based on wind, temperature and precipitation.

D. Reward Function

Finally, a reward function is defined as follows:

$$R(s) = \begin{cases} -0.04 & s \neq \text{goal,excluded state} \\ -1 & \text{excluded state} \\ +1 & \text{goal state} \end{cases}$$

The goal state has grid location [4,4,4] and reward value 1, while excluded cells have a value of -1.

III. Experimental Results

In order to better understand the method, a specific example will be considered here. First, a 4x4x4 grid of 64 air volumes as shown in Figure 3 will be indexed by either their grid coordinates or by a simple index. E.g., cell [3,2,4] will also be identified by the index 31. In addition to the grid location, each volume also has temperature, wind and precipitation information. This latter aspect will be considered below.

The actions available are directly tied to moving to one of the neighboring (closest) six-neighbors. Figure 2 shows the semantics of the actions. Note that when a cell is on the boundary, then the UAV is not allowed to exit the 4x4x4 grid, and so if that direction is chosen, the UAV will remain in the same cell with some probability.

In order to solve the value iteration problem, the neighbors of each cell in each action direction is first determined. A few of these are as follows:

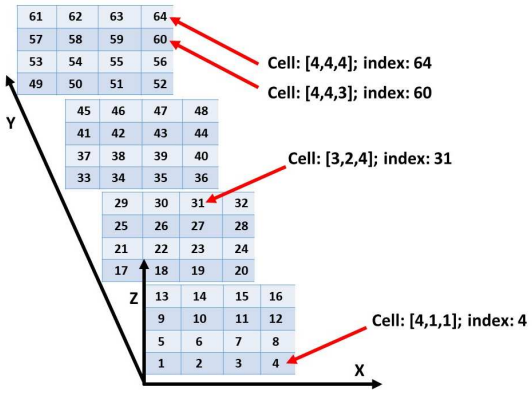

Fig. 3. States for a 4x4x4 grid.

State Index	X	-X	Y	-Y	Z	-Z
1	2	1	17	1	5	1
2	3	1	18	2	6	2
3	4	2	19	3	7	3
4	4	3	20	4	8	4
5	6	5	21	5	9	1
6	7	5	22	6	10	2
7	8	6	23	7	11	3
8	8	7	24	8	12	4
9	10	9	25	9	13	5
10	11	9	26	10	14	6
...						
61	62	61	61	45	61	57
62	63	61	62	46	62	58
63	64	62	63	47	63	59
64	64	63	64	48	64	60

Table 1. Examples of Neighbors for Some Selected States.

Next it is necessary to define the probability of moving into each neighboring cell given the desired action. This is provided in terms of Table 2 as shown here:

Action	X	-X	Y	-Y	Z	-Z
1	0.60	0.00	0.10	0.10	0.05	0.15
2	0.00	0.60	0.10	0.10	0.05	0.15
3	0.10	0.10	0.60	0.00	0.05	0.15
4	0.10	0.10	0.00	0.60	0.05	0.15
5	0.15	0.15	0.15	0.15	0.40	0.00
6	0.05	0.05	0.05	0.05	0.40	0.80

Table 2. Probabilities Used for Transitions for Actions given Normal Temperature, No Wind and No Precipitation.

Note that we assume it is more likely that motions in the X-Y plane will have a certain probability, and that moving up is more uncertain than moving down. Although we have assigned likely values here, these are also parameters that may be learned over time.

Given this information, it is possible to run the value iteration algorithm and find the utilities for the states. Figure 4 shows the utilities produced at each state as well as the path through the highest probability sequence of states. Note that this may not match the optimal actions selected since it does not take into account the maximal expected utility of the action. It does show however, that the UAV will most likely move up and then over in such a way as to avoid the excluded air volume (index 60, cell [4,4,3]). Figure 5 shows how the utility values converge for this problem.

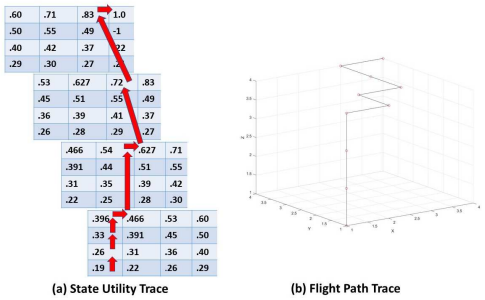

Fig. 4. The state Utilities and Path through the Highest Utility States.

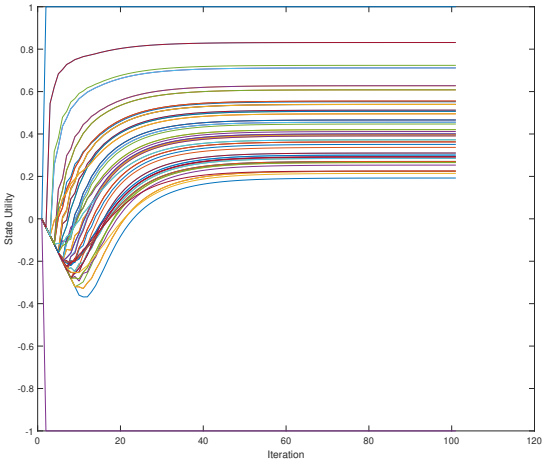

Fig. 5. The Convergence of the state Utilities.

An optimal policy can then be determined for each state and produces the result shown in Table 3.

State	Policy	State	Policy	State	Policy
1	5	23	5	45	1
2	5	24	5	46	1
3	5	25	5	47	3
4	5	26	5	48	3
5	5	27	5	49	5
6	5	28	5	50	5
7	5	29	3	51	5
8	5	30	3	52	2
9	5	31	3	53	5
10	5	32	3	54	5
11	5	33	5	55	5
12	5	34	5	56	2
13	3	35	5	57	5
14	1	36	5	58	5
15	3	37	5	59	2
16	3	38	5	60	-
17	5	39	5	61	1
18	5	40	5	62	1
19	5	41	5	63	1
20	5	42	5	64	-
21	5	43	5		
22	5	44	4		

Table 3. Optimal Policies for the states.

These results are also shown in Figure 6. Note that the red arrows indicate a move in the Y direction.

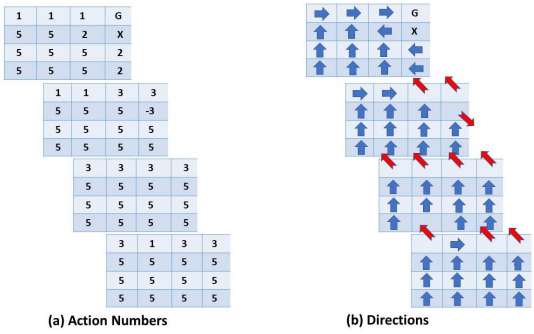

Fig. 6. Optimal Policies for the States.

Of course, the actions are not deterministic, and in order to better understand the impact of this policy, 1000 trials were run with start location cell [1,1,1], index 1, and with goal location cell [4,4,4], index 64. One cell is excluded, cell [4,4,3], index 60; if the UAV enters that cell it must land and terminates the mission. Figure 7 shows number of times each cell in the airspace was traversed by the 1000 trials. As can be seen, much information may be gleaned from these results as to the probability that a UAV will be in the assigned air volume (we have not included temporal aspects, but that is readily available, if desired). Also, note that a half a percent of the trials resulted in failure.

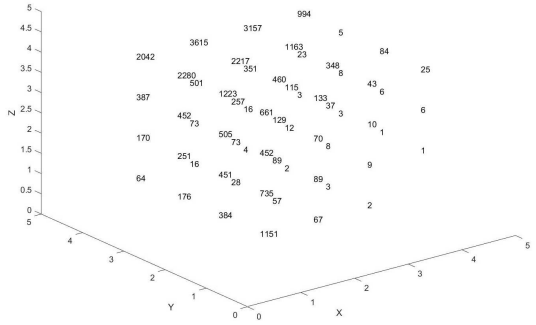

Fig. 7. The Number of Times Each Cell is Traversed in 1000 Trials.

A. *State with Wind Effect*

Next, consider the impact of a stiff wind blowing in the Y axis direction. This information is easily added to the model by simply providing the state transition probability for motions impacted by the wind. This is called context-based probabilistic state transition. In the case of a strong wind in the Y axis direction, say produced by afternoon canyon winds in Salt lake City, a set of state-action probabilities are provided (either by learning over time, or by physics-based simulation). Figure 8 shows the values used here.

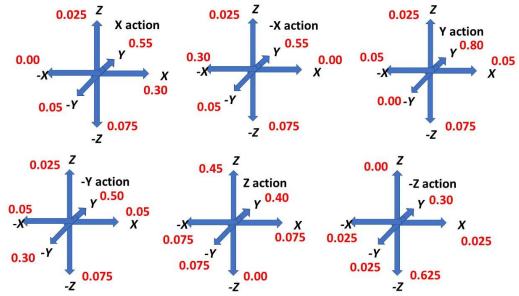

Fig. 8. The Context-based Probabilistic State Transition Probabilities for the Case of a Strong Wind in the Y Axis Direction.

Running value iteration with these probabilities gives rise to the convergence values shown in Figure 9.

The policy determined for these utilities is shown in Figure 10.

Some interesting observations may be made. For example, the policy never chooses a Y-axis action. Figure 11 shows the number of times each air space volume is traversed over 1000 trials. Note that there are a significantly higher number of failures (39) due to the strong wind. Of course, this policy is the result of a certain level of tolerance for failure vs energy expenditure. It is possible to vary these parameters and produce policies less likely to fail by entering excluded zones. Finally, it is important to note that the method can be considered having a learning part and an application part.

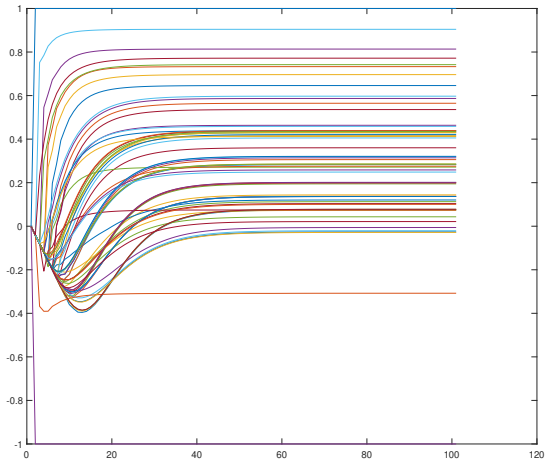

Fig. 9. The Convergence of Value Iteration with the Context-based Method.

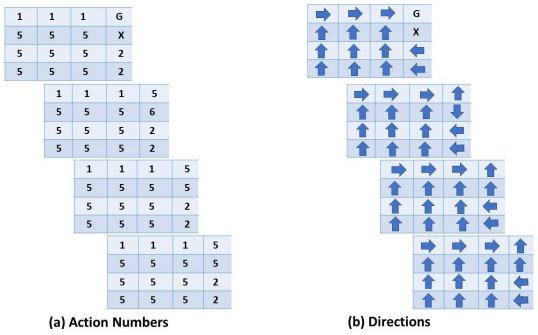

Fig. 10. The Policy Produced by the Context-based State Transition Method.

Once the policies are learned, they can be applied directly in real-time applications.

IV. Conclusions and Future Work

A context-based method for state transition probabilities is incorporated into a reinforcement learning paradigm, and this is used to determine an optimal action policy for UAV flights. This has been demonstrated under ideal and wind-impacted scenarios which extends readily to account for other factors like temperature, precipitation, and GPS or communication degradation.

Future work includes:

- the method will be applied to larger scale spatial dimensions as well as to include multiple context parameters.
- A physics simulation will be developed to take advantage of GPU computational abilities. A colleague (Prof. C. Yuksel) has already demonstrated the ability to simulate 10's of thousands of UAVs flying in the airspace over Salt lake City (see Figure 12).
- real-time sensor data and flight plan fusion will play an important role in exploiting the methods described here.

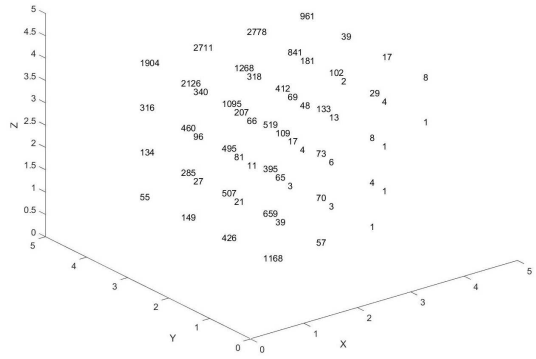

Fig. 11. The Number of Times Each Space Volume is Traversed over 1000 Trials.

Figure 13 shows the sensor sets that are being deployed in Utah to support this effort.

- validation is also of interest once large-scale simulation has proven effective. This will be performed at the Deseret UAS Test Facility (see Figure 14. Experiments will take place at the Deseret UAS test facility in Utah (Deseret UAS partners include: the Governor's Office (Utah), Metron Aviation, Airbus North America, Precision Hawk, Loveland Innovations, McGuireWoods Consulting, Logistics Specialties, SkyLark Drone Research Robo-Economics, ASSUREuas, and the C-UAS University Coalition). The facility includes sites that encompass an expansive, climatically diverse area across Tooele Army Depot South (TEAD-S), and neighboring sites in Box Elder County. TEAD-s is covered by 26 mi^2 of airspace designated as a National Security Area, due to US Army munitions elimination work. Deseret UAS offers one of the largest UTM testing arenas in the world and provides a secure environment surrounded by a sparsely populated area with an ideal climate, making it safe and reliable for testing. UAS operation in urban conditions is critical for developing the data to support services such as package delivery. All Meteorological Terminal Aviation Routine (METAR) conditions for the Salt Lake International Airport are similar to 27 of the 35 Operation Evolution Plan (OEP) airports that are co-located with major urban populations, and conditions at SLC Airport are surrogates for the test site. The diversity of weather conditions at this site enables users to test under atmospheric conditions commonly experienced in urban centers throughout the US. In addition, the space includes two fabricated villages built inside secure areas for testing in a mock urban setting. Finally, Deseret UAS plans a complete operating command and control center with software adapted for UAS along with tracking and communication systems, including radar and ADS-B. The command and control center employs industry elite resources to oversee air traffic operations, and offers

management of: (1) intra-organizational UTM operations (private sector entities manage own fleet), and (2) inter-organizational UTM (Deseret UAS oversees all operations). Deseret UAS traffic flow experts will provide air traffic management to ensure aircraft safety while gathering critical test data related to communication and instrumentation through advanced sensing and control systems.

Fig. 12. Current work includes development of a physics-based simulation capable of handing 1000's of UAV's; here is shown 1000 UAV's flying over Salt Lake City.

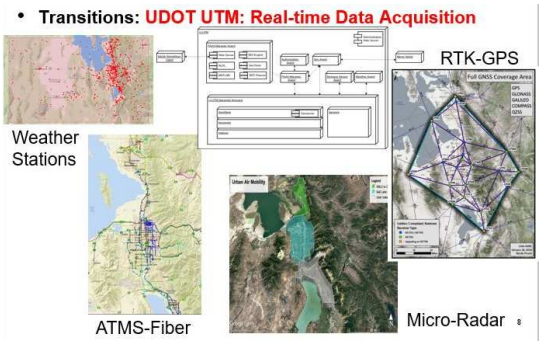

Fig. 13. The development of a UAV tracking system is underway which will exploit a set of real-time sensor platforms deployed by the Utah Department of Transportation.

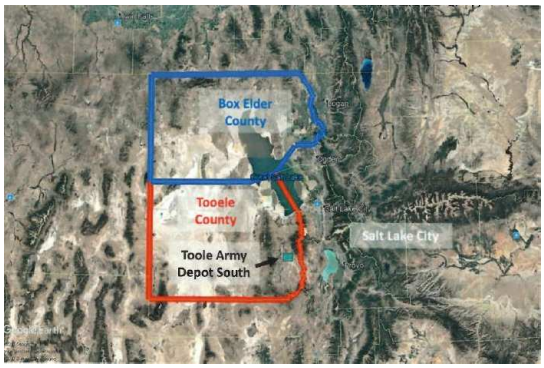

Fig. 14. Large-scale testing of UAV traffic management strategies is planned at the Deseret UAS Test Facility Site in Utah.

References

[1] M. Brittain and P. Wei. Autonomous Aircraft Sequencing and Separation with Hierarchical Deep Reinforcement Learning. In *International Conference on Research in Air Transportation*, Barcelona, Spain, June 2018.

[2] A. Carrio, C. Sampedro, A. Rodriguez-Ramos, and P. Campoy. A Review of Deep Learning methods and Applications for Unmanned Aerial Vehicles. *Journal of Sensors*, 2017(2):1–13, August 2017.

[3] AirMap Company. Five Critical Enablers for Safe, Efficient, and Viable UAS Traffic Management (UTM). In *Whitepaper*, Santa Monica, CA, 2018.

[4] D. Sacharny and T.C. Henderson and A. Mitiche and R. Simmons and T. Welker and X. Fan. BRECCIA: Unified Probabilisitic Dynamic Geospatial Intelligence. In *IEEE Conference on Intelligent Robots and Systems (IROS 2017 Late Breaking Paper)*, Vancouver, Canada, September 2017.

[5] K. Low. Framework for Urban Traffic Management of Unmanned Aircraft System. In *Proceedings of the ICAO Conference*, Montreal, Canada, 2017. ICAO.

[6] V. Minh, K. Kavukcuoglu, D. Silver, A.A. Rusu, J. Veness, M.G. Bellemare, A. Graves, M. Riedmiller, A.K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Leg, and D. Hassabis. Human-level Control through Deep Reinforcement Learning. *Nature*, 518:529–533, February 2015.

[7] S.J. Russell and P. Norvig. *Artificial Intelligence*. Prentice-Hall, Upper Saddle River, NJ, 2010.

[8] D. Sacharny, T.C. Henderson, A. Mitiche, R. Simmons, T. Welker, and X. Fan. BRECCIA: A Multi-Agent Data Fusion and Decision Support System for Dynamic Mission Planning. In *2nd Conference on Dynamic Data Driven Application Systems*, Cambridge, MA, August 2017.

[9] D. Sacharny, T.C. Henderson, R. Simmons, A. Mitiche, T. Welker, and X. Fan. BRECCIA: A Novel Multi-source Fusion Framework for Dynamic Geospatial Data Analysis. In *IEEE Conference on Multisensor Fusion and Integration*, Daegu, S. Korea, September 2017.

[10] Blake Wulfe. UAV Collision Avoidance Policy Optimization with Deep Reinforcement Learning. https://wulfebw.github.io/assets/CS238_Final_Paper.pdf, December 2018.