Multi-Objective Design and Path Planning Optimization of Unmanned Aerial Vehicles (UAVs)

Eliot Rudnick-Cohen[1,3] Shapour Azarm[1] Jeffrey W. Herrmann[1,2]

[1] Department of Mechanical Engineering
[2] Institute for Systems Research
[1, 2] University of Maryland, College Park, MD 20742
[3] Corresponding Author: erudnick@umd.edu

This paper presents the topic of combined UAV design and path planning using bi-objective optimization. Two optimization models, integrated and decomposed, are developed and solved for path planning and design optimization of UAVs. Both models minimize a weighted sum of risk and time. The risk is based on a simulation of a UAV's crash after it loses power and is estimated by computing an expected number of on-the-ground human population fatalities which are located at or near the UAV crash site. The flight time is computed for the UAV as it travels from a start to an end point. The integrated model solves the design and path planning optimization problem all-at-once. The decomposed model has two subproblems: one for the design optimization of UAV and another for its path planning optimization. A new decomposed method for solving the path planning subproblem while considering the design optimization subproblem is presented, which solves the path planning subproblem for a discrete graph. Results for this new method are compared to an integrated all-at-once approach that does not treat the design and path planning subproblems separately, which can indicate the method proposed is less vulnerable to local minima in the path planning optimization subproblem.

I. Introduction

There are several potential benefits to combining UAV's design and path planning optimization problems. Optimizing a path for a fixed design or optimizing a design for a fixed path can yield solutions that are inferior to those found by optimizing the design and the path together. Additionally, the ability to adjust the design as well as the path during optimization can allow for a wider variety of solutions to multi-objective UAV optimization problems as changes to a design can allow for a much wider possible range of values for objective functions than would be possible with changing only the path.

Previous works have considered the issue of path planning for the purposes of minimizing risk to third parties or other safety related objectives. Medeiros and Da Silva [1] constructed a visibility graph around highly populated areas and then used discrete path planning algorithms such as Dijkstra's algorithm to generate paths by solving the shortest path problem. Narayan *et al.* [2] considered a trajectory optimization approach for the problem of optimizing multiple objectives using weighted sum methods and dynamic programming. Rudnick-Cohen et al. [3] compared multiple methods for risk-based path planning for a given UAV. They used Monte Carlo simulations of a flight dynamics model to determine a crash location distribution, which was used to estimate the expected fatalities of a flight over inhabited areas. Their results showed that a network approach, combined with a greedy algorithm to improve the best path on the network, generated high-quality solutions more quickly than other approaches tested. For a more in depth survey of methods for UAV path planning in general, consult [4].

Although there has been little work looking into the specific problem of combined UAV design and path planning optimization, there have been several works that have considered issues related to this problem. Kallrath [5] discussed mixed integer programming solutions to production planning and network design problems in the process industry, which have some similarities to design and path planning problems. There are some similarities also to the work done on design and control optimization [6], in that both the design of a system and how the system moves were both optimized. Nigam and Kroo [7] studied the optimization of both the design and mission of a UAV (or multiple UAVs) for surveillance tasks using an approach based around coordinating two optimizers to solve two subproblems that decompose the primary problem being considered with a third coordinating "system-level" optimizer. The approach described by Rastegar *et al*. [5] avoids having the two optimizers directly interact; instead, the third optimizer mediates any conflicts between them. In this paper we also consider a similar decomposition scheme using two separate subproblems. That is, instead of solving the problems independently, we will present a new algorithm which takes advantage of the coupled nature of the subproblems in order to solve the path planning subproblem while keeping it independent from the aspects of the design optimization subproblem that would normally compromise the optimality of typical methods for path planning. We will also discuss what these aspects of the design optimization subproblem are and the situations where they can cause a conventional path planning method to arrive at suboptimal solution for the combined path planning and design optimization problem.

The organization of the paper is as follows: Section II defines the problem, including the objectives considered and the specific scenario under consideration. Section III discusses the two models that are considered and compared. Section IV details the approach used to implement these models. This includes a description of the algorithm proposed for solving the path planning subproblem while the UAV design is considered and also a description of the issues the can arise in attempting to use a conventional path planning method. Section V presents the scenario we used for testing the two models, results from applying each of the models to that scenario and a discussion of the results. Section VI presents some concluding remarks and directions for future research.

II. Problem Definition

The problem specifically considered in this work is to find the optimal design and path for a UAV travelling from College Park Airport in College Park, Maryland, to Virginia Tech Executive Airport in Blacksburg, Virginia. The UAV design consists of the speed that the UAV will be flying along the path and the wing reference area of the UAV. Two objectives were optimized using a weighted sum method to convert the optimization problem into a single objective optimization form in order to generate a Pareto frontier. The two objectives considered were the time needed to traverse the path from starting point to the destination and the risk to the human population on the ground associated with that path. The time objective was based off the assumption that speed of the UAV would remain constant throughout the flight, thus defining it as simply being the length of the path divided by the speed of the vehicle. The risk objective was defined using the risk metric defined in [3]. The crash location probability distribution was parametrized in terms of the design variables under considerations, which was used to optimize the risk objective. The crash location distributions used in the model were generated by running Monte Carlo simulations of an unpowered UAV crashing such as the ones done in [3]. Multiple distributions were generated for different design variable combinations, the parametrization used these distributions to estimate the distribution for arbitrary design variable combinations within the bounds of the design variables using Delaunay triangulation. For the exact formulae for the objectives used see [3].

III. Proposed Model

We solved and compared results from two different models, which are described in Sections A and B. In both models we considered the bi-objective problem of minimizing the risk associated with a path and a design and the time needed to traverse the path.

A. Integrated Model

The first model considers design and path planning optimization problem in an all-at-once (AAO) model. This model defines the flight path as a path that travels through a finite number of waypoints and then treat the coordinates of those waypoints as design variables. This model also considers the design of the UAV. This approach allows for the use of standard optimization techniques, making this a fairly conventional approach to attempt to solve the problem as a single (integrated) optimization problem.

B. Decomposed Model

The second model entails separating the problem into two subproblems, a design subproblem and a path planning subproblem, both of which have the same objective functions. There are two sets of design variables: X_D, for the UAV design variables, and X_P, for the UAV path design variables (i.e., coordinate of waypoints) located between the start and end points. The objective function is $f\left(X_D, X_P\right)$. In this model, the two subproblems interact by solving the design optimization subproblem for partial and complete solutions to the path planning subproblem and then feeding those solutions back into the algorithm we have developed for solving problems of this type. The structure of this decomposed model is depicted in Figure 1.

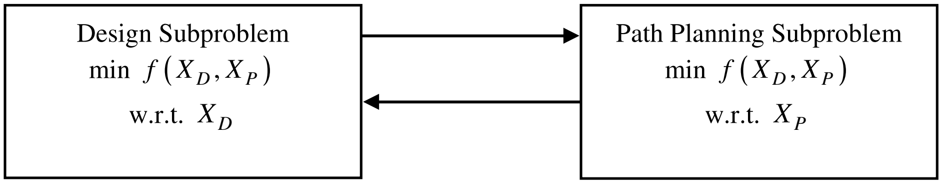


Figure 1: The two subproblems considered for the decomposed model

IV. Approach

With the proposed decomposed approach, it is not possible to use a typical graph-based method such as Dijkstra's algorithm [8] for solving the path planning sub-problem. This is because it is possible that the optimal design and path for reaching a location on the optimal path from the starting point to the goal will not be part of the optimal path to reach the goal. This situation can occur when the optimal design to reach the goal may need to compensate for some part of the optimal path to reach the goal that is not part of the optimal path to a location on the optimal path to the goal. Consequently the optimal path to reach that location on the optimal path may not be a section of the optimal path to reach the goal.

To get around this issue we developed a new algorithm for handling the path planning subproblem that was capable of accounting for the issues introduced by the capability of the design to change. We can provide a brief high level description of this algorithm as follows. The algorithm works by first using Dijkstra's algorithm, with each edge cost being determined for the optimal path to that edge as determined by solving the design optimization problem. Once a path is found that connects the start node to the goal node, alternate paths from the start to the goal are generated by merging the paths that Dijkstra's algorithm has found with a path from the start to the goal and solving a design optimization subproblem for the resulting path from the start to the goal. As this is done for nodes, they are removed from the part of the graph being considered by the Dijkstra's algorithm portion of this approach. This causes the algorithm to search alternate paths to nodes, which accounts for the issue of past edges becoming sub-optimal choices due to edges located further ahead in the path.

To describe the specific algorithm used for the decomposed approach the following definitions and step by step process are provided:

Node: Represents a location that a path can travel through, keeps track of the best partial or complete solution that involves a path traveling through it.
Solutions Lists: Lists of nodes that depend on other nodes in the solution list for their best solutions, the algorithm will not consider edges that connect to nodes in the same solution list when generating candidate solutions
Edge: Represents that it is possible to travel from the start node of the edge to the destination node of the edge.
Partial Solution: A solution that consists of an incomplete solution to the path planning problem (in the form of an incomplete path that starts from either the start or end node and does not reach the other node) and a solution to the design problem.
Complete Solution: A solution that consists of a complete solution to the path planning problem (a path that travels from the start node to the end node) and a solution to the design optimization problem.

Steps:

1. Initialize two partial solution lists, 1 for the start node and 1 for the goal node, each list should only consist of a single node at this point.
2. For each edge of each node in each solution list that does not go to another node in that solution list, compute the partial or complete optimal solution by solving the design problem for that edge. For complete solutions, determine the path by taking the start nodes path up until that node occurs in its path and then go to the destination node's path starting from where the destination node is in its path. Add solutions that are better than a node's current best known solution to this list of candidate solutions.
3. Select the best solution from the list of candidate solutions to add to the solution lists, if no solutions remain in the list, terminate.
4. With the selected best solution so far two different actions can be taken depending on whether the solution is a complete solution or not. Add partial solutions to the partial solution list of the start node of the edge associated with the selected solution. For complete solutions, if the destination node of the edge associated

with the selected solution is the last node in its solution list, add it to the edge's start node's solution list, otherwise add the complete solution to a new solution list.

5. Set all edges leading to and from the edge associated with the selected solution destination node to be recomputed and remove any nodes in the partial solution list that may have depended on the node associated with the solution added if that node was originally in a partial solution list. Go back to step 2.

V. Results

A version of the AAO model was implemented for a small number of waypoints. In the model, risk is determined by determining a distribution of possible UAV crash locations and evaluating the risk level, as defined in [9]. For the population density data needed for the model, data from the US census was used. The risk was evaluated along with path segments, which are straight lines between waypoints. To evaluate the risk along the path being followed, the risk was calculated at multiple points along each segment of the flight path in order to approximate the value of the risk along the entire segment. Flight time was calculated by dividing the length of the path by the airspeed.

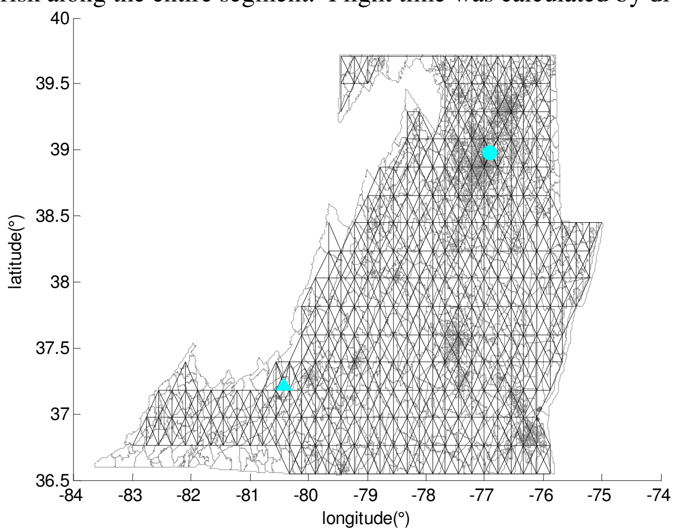


Figure 2: The grid used by the decomposed approach for the region under consideration. The cyan circle indicates the start point at the College Park, Maryland airport and the cyan triangle indicates the end point at Virginia Tech Executive Airport in Blacksburg, Virginia [3]

For the integrated model, three waypoints were considered along with the airspeed (V) and the wing reference area (S) of the vehicle. The start point was the College Park, Maryland airport and the destination was Virginia Tech Executive Airport in Blacksburg, Virginia. The locations of the two points relative to the region under consideration can be seen in Figure 2. Figure 2 also depicts the grid that was used for the decomposed approach. The upper and lower bounds were set to keep the waypoints within a box enclosing the region being considered and to keep design variables within the range of values used for Monte Carlo simulation. The initial conditions used for optimization were a straight line path for the waypoint variables, an airspeed of 50 m/s, and a wing reference area of 16.17 m^2. Optimization was done using MATLAB's fmincon [10] function for the integrated model.

For the decomposed approach a 40×16 node grid was created and then pruned to remove any nodes outside the states that were being considered in the grid. The initial solution for the design problem was the same as the one that was used for the integrated approach.

For both models, the integrated and decomposed, 11 equally spaced weights between a 100% and 0% weighting on the risk objective and the time objective were used to generate a range of solutions for different objectives. The results from this are provided in Figure 3.

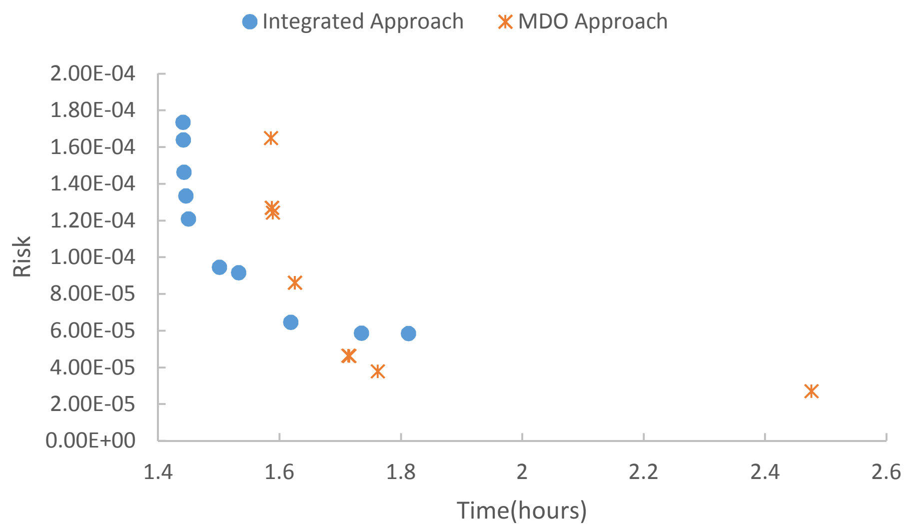


Figure 3: Pareto front of risk vs. time for both approaches

Design variables		Waypoint coordinates						Objectives	
V(m/s)	S(m)^2	x1(deg)	x2(deg)	x3(deg)	y1(deg)	y2(deg)	y3(deg)	t(hours)	risk
70.00	16.13	-76.79	-77.60	-79.64	39.12	39.31	37.18	1.81	5.86×10^{-5}
70.00	16.48	-77.15	-78.36	-79.53	39.20	39.25	38.71	1.74	5.88×10^{-5}
62.62	16.28	-77.14	-77.91	-79.13	39.28	38.82	37.80	1.62	6.48×10^{-5}
66.43	16.34	-76.88	-77.63	-79.09	38.77	38.48	38.18	1.53	9.18×10^{-5}
70.00	16.10	-76.91	-76.90	-79.41	38.76	38.77	37.75	1.50	9.46×10^{-5}
70.00	16.18	-77.10	-77.52	-79.83	38.80	38.61	37.57	1.45	1.21×10^{-4}
70.00	16.66	-76.95	-77.09	-78.43	38.94	38.85	38.29	1.45	1.34×10^{-4}
70.00	16.15	-77.03	-78.07	-78.69	38.96	38.42	38.13	1.44	1.46×10^{-4}
70.00	16.14	-77.22	-78.25	-79.62	38.85	38.30	37.64	1.44	1.64×10^{-4}
70.00	16.17	-78.34	-79.06	-79.49	38.28	37.91	37.69	1.44	1.74×10^{-4}

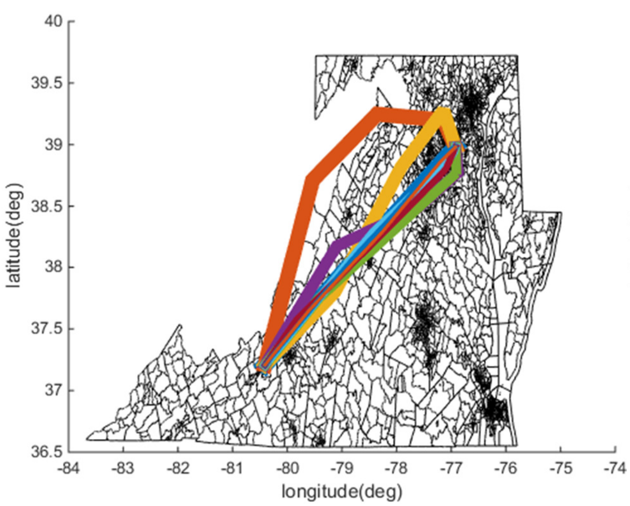
Table 1: Solutions from Figure 3 for integrated approach

Table 1 depicts the design variable values for the solutions shown in Figure 3 for the integrated approach. The variables x1, x2, x3 and y1, y2, y3 describe the 3 waypoints in terms of their longitudes (x) and latitudes (y). As expected most solutions are close to the upper bound for speed (70 m/s) as currently no constraints are being enforced that can limit the speed based off any of the other design variables and the objective function used for risk can be reduced by reducing the time needed to traverse a path, which is minimized by maximizing the speed of the UAV. Several solutions did deviate from the maximum speed allowed, indicating that the optimal solution is not necessarily always to travel at the maximum possible speed. Variations in the optimal wing reference area indicate that while there may not exist a specific trend, it was possible to reduce the risk for a given path by adjusting the wing reference area. The lack of a trend here is expected, as adjusting the wing reference area will change the shape of the crash location distribution, but the optimal shape of the distribution will depend on the population distribution along the path taken, meaning that the optimal wing reference area for a specific path could be significantly different depending on what sort of distribution is best for that path. These results demonstrate the capability for this type of optimization for generating a variety of solutions for multiobjective problems as the variation in the wing reference area in conjunction with the variation of the waypoint locations create a clear trade-off between time and risk.

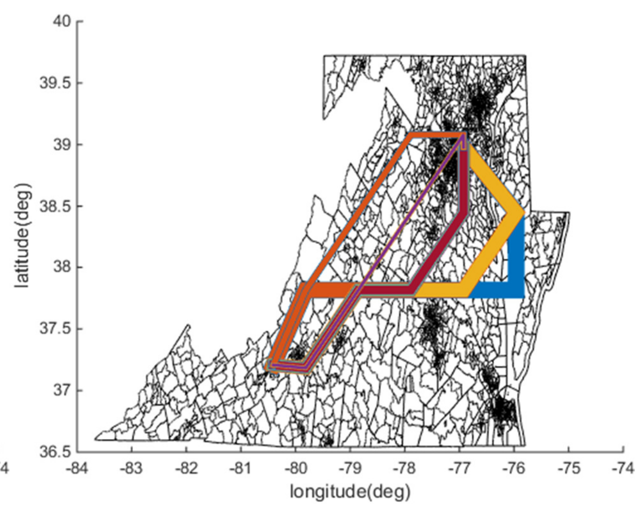
Design variables		Objectives	
V(m/s)	S(m^2)	t(hours)	risk
70.00	16.17	1.59	1.65×10^{-4}
70.00	16.17	1.59	1.27×10^{-4}
70.00	16.17	1.59	1.27×10^{-4}
70.00	16.17	1.59	1.27×10^{-4}
70.00	16.17	1.59	1.24×10^{-4}
70.00	16.17	1.63	8.63×10^{-5}
70.00	16.17	1.63	8.63×10^{-5}
70.00	16.17	1.71	4.65×10^{-5}
70.00	16.18	1.71	4.64×10^{-5}
70.00	16.18	1.76	3.80×10^{-5}
70.00	16.17	2.48	2.73×10^{-5}

Table 2: Solutions from Figure 3 for decomposed approach

Table 2 depicts the design variable values for the solutions shown in Figure 3 for the decomposed approach. The decomposed approach is capable of generating superior solutions in terms of the risk objective as unlike the integrated approach the decomposed approach will not get stuck and local optima and will thus actually reach globally optimal solutions for both objectives. However, the solutions generated are restricted to remaining on the grid, which prevents the solutions from taking the most direct possible routes to the goal, consequently the integrated approach is able to generate solutions that take more direct paths to the goal and thus solutions that are more optimal with respect to the time objective. This can be seen in Figure 4, as the solutions for the decomposed approach are forced to make small detours instead of following a direct straight line path. The results from Table 2 also indicate different trends in terms of the optimal design variables in that the results for the decomposed approach indicate the existence of an ideal point for the design optimization problem. The optimal wing reference areas all tended to be near the initial condition chosen for that variable, which is likely due to the presence of a local optimal located there. A possible cause for this may be that the optimal wing reference area may be very sensitive to the locations of the waypoints that make up the path, since the decomposed approach does not have the ability to make small adjustments to where it places waypoints since its path is defined between elements located on a grid.



(a)



(b)

Figure 4: Plots of solution paths for: (a) integrated approach, (b) decomposed approach, with thickness of a path corresponding to the amount of risk for that path, the thicker the line path the lower the risk

Figure 4 also provides a comparison of the paths generated by the different approaches. As can be seen, the more globally optimal solutions that the decomposed approach generates for the cases with a heavier weight on risk follow fundamentally different paths than the solutions generated by the integrated approach for those weights. As discussed before, the cause for this is due to the integrated approach getting stuck at local optima, which keeps it from finding more globally optimal solutions such as the ones found by the decomposed approach. A major difference between the paths generated by the two methods is that the paths generated by the integrated approach cross over into West Virginia, while the paths generated by the decomposed approach remain inside the 2 states (Maryland, Virginia) plus Washington D.C. under consideration for this scenario. This is due to the fact that the integrated approach has to use a bound constraint to limit the region where waypoints can be placed, which leads to a section of West Virginia being present in the feasible region for waypoint placement. However, with the decomposed approach, it is possible to always restrict the paths under consideration to stay within the 3 states being considered as by not placing nodes outside those states the paths found will never be able to go outside them.

VI. Concluding Remarks

A new algorithm was presented for solving path planning subproblem within a decomposed approach for solving path planning and design optimization subproblems. The decomposed approach was able to find more globally optimal solutions than an integrated approach, though it did suffer some restrictions that resulted from the discrete nature of how it determined the optimal path. Future work on this topic will likely include the development of approaches that can handle constraints that relate the design variables of both subproblems, methods for refining the size of the grid used for the path planning subproblem inside the algorithm to reduce computational cost and consideration of additional design variables relating the physical design of a UAV and its operating parameters.

VII. Acknowledgement

The authors acknowledge the assistance of Dr. David Burke and Dr. John Tritschler and the financial support of the Naval Air Warfare Center under cooperative agreement N00421132M006. Such support does not constitute an endorsement by the funding agency of the opinions expressed in the paper.

VIII. References

[1] F. L. L. Medeiros and J. D. S. Da Silva, "Computational modeling for automatic path planning based on evaluations of the effects of impacts of uavs on the ground," *Journal of Intelligent & Robotic Systems*, vol. 61, no. 1-4, pp. 181–202, 2011.

[2] P. Narayan, D. Campbell, and R. Walker, "Computationally adaptive multi-objective trajectory optimization for uas with variable planning deadlines," *IEEEAC*, 2009.

[3] E. Rudnick-Cohen, J. Herrmann, and S. Azarm, "Risk-based path planning optimization methods for uavs over inhabited area," in *To appear in Proceedings of the ASME 2015 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference IDETC/CIE 2015*, 2015.

[4] C. Goerzen, Z. Kong, and B. Mettler, "A survey of motion planning algorithms from the perspective of autonomous uav guidance," *Journal of Intelligent and Robotic Systems*, vol. 57, no. 1-4, pp. 65–100, 2010.

[5] J. Kallrath, "Solving planning and design problems in the process industry using mixed integer and global optimization," *Special Edition of Annals of Operations Research*, 2004.

[6] J. S. Rastegar, L. Liu, and D. Yin, "Task-specific optimal simultaneous kinematic, dynamic, and control design of high-performance robotic systems," *Mechatronics, IEEE/ASME Transactions on*, vol. 4, no. 4, pp. 387–395, 1999.

[7] N. Nigam and I. Kroo, "Control and design of multiple unmanned air vehicles for a persistent surveillance task," in *12th AIAA/ISSMO multidisciplinary analysis and optimization conference. Victoria, British Columbia, AIAA-2008-5913*, 2008.

[8] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[9] D. Burke, "System level airworthiness tool: A comprehensive approach to small unmanned aircraft system airworthiness," Ph.D. dissertation, North Carolina State University, 2010.

[10] "Matlab and optimization toolbox release 2014b," The MathWorks Inc., Tech. Rep., 2014.