**Path-Planning Optimization for an Unmanned Aerial Vehicle with Energy Constraint in a Search and Coverage Mission**

**German Gramajo and Praveen Shankar**

California State University, Long Beach

Mechanical and Aerospace Engineering Department

**Abstract**

This paper describes the path planning algorithm for a search and coverage mission using a small UAV that optimizes the trajectory based on energy and maneuverability constraints. The proposed formulation has a high level of autonomy, without relying of the user to make the appropriate choice of optimization parameters. The computed trajectory maximizes spatial coverage while satisfying constraints such as the original flight plan of reaching a desired end state from the initial state for the initial available energy and the maneuverability limitations of the vehicle. Comparisons of this formulation to a path planning algorithm based on time constraint optimization show equivalent coverage performance but improvement in prediction of overall mission duration and terminal position of the vehicle.

1. **Introduction**

The increased interest in UAVs has seen their implementation in military and civilian operations. Small inexpensive autonomous aerial vehicles are of great interest in search and coverage, surveillance, border patrol, and mapping missions [1]- [2]. These missions require repetitive aerial maneuvers in order to locate objects/targets as soon as possible in a region, or to generate a collage of a specified region. Due to the repetitive nature of the mission, small autonomous aerial vehicles are the clear choice to perform these tedious missions. The repetitive nature allows for the automation of the mission that reduces cost of the mission and allow for a faster and more effective coverage of an area [1]- [3].

The primary challenge in implementing small autonomous aerial vehicles for a search and coverage mission is planning the path of the vehicle that will effectively cover the specified region. This requires the development of an algorithm that will always generate trajectories to maximize the spatial coverage for any specified conditions. Different approaches exist to deal with the search and coverage problem. These fall broadly into two categories - standard search patterns and nonstandard search patterns. Standard patterns include those such as spiral and serpentine/grid (boustrophedon motion) [4]- [5]. Even though, standard search patterns have proven useful to search an area, they are not ideal since they do not suit situations where multiple vehicles are cooperating to accomplish the task. On the other hand, nonstandard search patterns that have a random trajectory are easier to program in cooperative search and coverage scenario. Some algorithms used to generate the nonstandard search patterns are the A\* and traveling salesmen, which are heuristic techniques [6]. A heuristic technique optimizes the trajectory based on the cost to reach the current state and the cost to reach the goal from the current state. Other forms of search techniques used are probabilistic, which compute the probability of the location of an object/target in a region [7] [8].

The foundation for the primary contribution in this paper is the algorithm defined in [9]- [10] for generating a trajectory that maximizes spatial-temporal coverage based on a preset of turning rates and preset mission duration. It is a heuristic algorithm and a modified version of A\* that quantifies the amount area covered by each of the possible paths from the current state and the cost to reach the desired exit state from the end of the current path being calculated. The path selected is the one with lowest value of the sum of the two costs. The algorithm utilizes a receding horizon control (RHC) formulation to generate the optimal trajectory which includes a feedback to account for any disturbance that may deviate the vehicle from its predicted path. There several shortcomings of the algorithm presented in [9]. One of the problems is that the algorithm uses a set mission duration related to the assumption of constant power consumption by the vehicle during the mission. In reality, the power consumption of the vehicle is not constant since it varies according to the maneuver and turn rate. Assuming a set mission duration and constant power consumption is a significant shortcoming of the algorithm since the varying power consumption will result in the vehicle running out of energy before covering the most amount of area and reaching the desired exit point. In addition, the algorithm selects the optimal path using a discrete set of turning rates that is user specified, and hence may not even be the optimal set of turning rates. Even though, the algorithm can potentially generate the most optimal trajectory, it is computationally exhaustive to obtain the optimal discrete set of turning rates because it requires multiple iterations. Moreover, the optimal discrete set of turning rates may only be optimal for particular set of boundary conditions. This requires the user to iterate the computation of the most optimal discrete set of turning rates for each possible specified condition. The last shortcoming of the algorithm is that it requires the discretization of the search region in order to compute the amount of the area covered by the vehicle as it performs the mission. The issue with space discretization is the algorithm can over calculate the actual amount of area covered by the vehicle in its path. The algorithm considers a discrete space as fully covered as long as the sensor footprint covers the center of the each discrete space even if the sensor did not cover the entire discrete space. One way of solving this problem is by the varying the amount and size of each discrete space. However, finding the appropriate space discretization is computationally exhaustive.

This paper proposes an optimization formulation for the path planning of a single UAV that maximizes the spatial coverage of an area under the constraints of limited energy and power consumption that is dependent on the maneuver. Additionally, the method operates over a range of turning rates rather than a discrete set. Finally, the formulation utilizes a Boolean operation based polygon area calculation to update the area covered by the UAV over a particular time interval to improve accuracy and overcome the limitation of choosing an appropriate discretization of the space by the user.

The rest of the paper is organized as follows: Section II provides a brief background of the path planning problem. Section III discusses the optimization formulation that is the primary contribution of the paper. Section IV provides the simulation results and discussion for optimization based on a time constraint and energy constraint. This is followed by conclusions in Section V.

1. **Path Planning for Unmanned Aerial Vehicles**

Consider the UAV modeled as a non-holonomic point mass moving in a two-dimensional plane at a constant velocity [9]. The following equations represent the vehicle’s dynamics.

(1)

where (x,y) are the coordinates of the vehicle in a two dimensional space and is the turn angle of the vehicle. The vehicle has a constant velocity *v*. The state of the vehicle is defined by . The vehicle is expected to search the bounded region defined by and the vehicle has a sensor footprint defined by . The vehicle starts the mission at point *pentry(x,y)* and is expected to end the mission at point *pexit(x,y)*. Etotal is the amount of energy available at the start of the mission. The exact solution of the problem satisfies

(2)

The trajectory maximizes the sensor footprint coverage of the search region. Figure 1 demonstrates a visual representation of the problem of generating a trajectory that maximizes the area covered and satisfies the exit stated for energy available. Figure 2 demonstrates the generated trajectory along with the sensor footprint area throughout the mission.

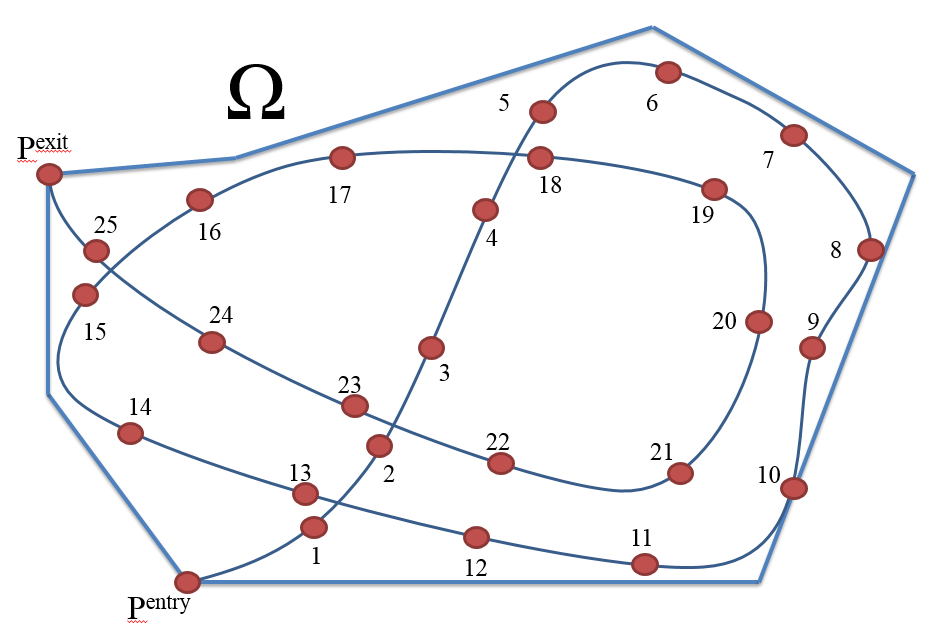


Figure 1: Problem of generating a trajectory that maximizes the area covered

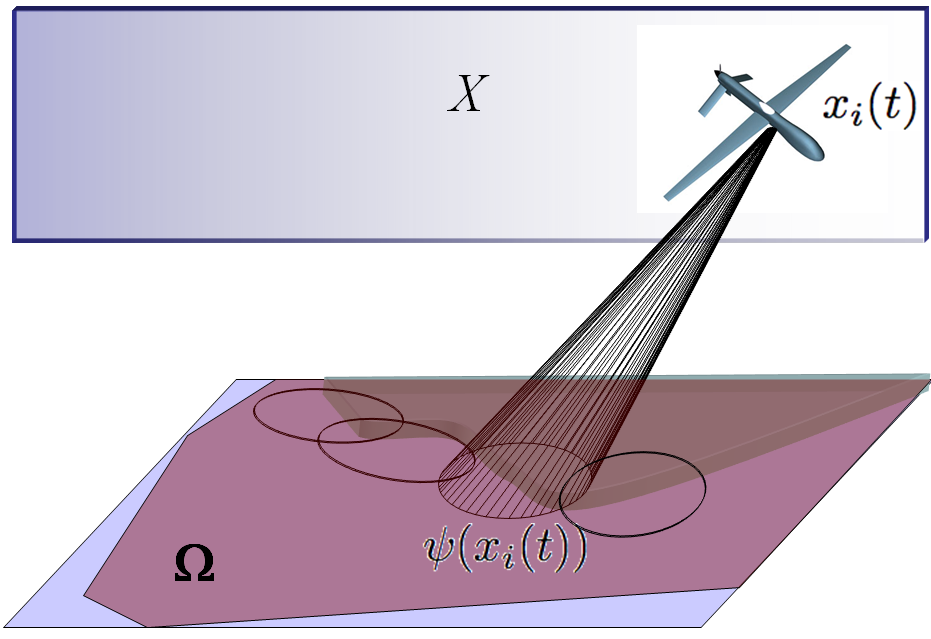


Figure 2: Area covered by sensor footprint for generated trajectory throughout the mission

The heuristic algorithm proposed in this paper to determine a path that maximizes coverage area depends on optimization across not a discrete set of turn rates as in [10], but a range of turning rates. In addition, the algorithm considers realistic power consumption of the vehicle, which is not constant and varies with the type of maneuver executed by the vehicle. This enables the accurate calculation of the actual mission duration and final position of the vehicle for the energy available at the start of the mission. The high-level of autonomy ensures that the algorithm also generates the most optimal trajectory for any conditions by not going over previously covered area. The last feature of the proposed formulation of the path optimization is that it does not require choosing the parameters for the space discretization. Without the need to evaluate the parameters for optimal space discretization, the computation process is faster while still providing an accurate calculation of the covered area.

1. **Range of Turning Rates and Available Energy Optimization**

The problem of interest is of an unmanned aerial vehicle operating at a fixed altitude in a closed, bounded region. The vehicle searches the bounded region equipped with a non-fixed camera and a refresh rate of τ, which can also be termed as the turn duration and the length of the execution horizon. The goal is to find a feasible trajectory, defined by , to get maximum coverage of the bounded region in each time interval. The problem requires time discretization because a continuous solution is computationally exhaustive. The algorithm considers portion of the trajectory defined by τ, the turn duration, which is also the execution horizon.

1. Turning rate range

The proposed formulation optimizes the maximum coverage trajectory for a range of turning rates. Equation (3) provides the equation to calculate the range of turning rates by calculate the maximum turning rate that the vehicle can achieve [12] [13].

(3)

The maximum load factor, defined by nmax, determines the vehicle’s turn capability. The range of turning rates is therefore defined as .

1. Optimization Formulation

The formulation of the search and coverage algorithm presented in this paper is minimizing the sum of the amount of the area of the bounded region not covered by the UAV and the terminal cost function. The function is subject to the equation of motions of the vehicle moving on a two-dimensional plane, sensor footprint, range of control inputs, obstacle, boundary conditions provided below. The variable Ω is the area of the bounded region and the variable Ψ represents the sensor footprint. The vector z provides the location of the vehicle in the xy-plane, which is a function of velocity, control input, and time. The x and y position of the vehicle are provided in discrete form, since discrete form enables for an easier way of calculating the area covered by the vehicle as the mission progresses. The area of the bounded region that remains uncovered by the vehicle at the end of the current time step is provided by . The uncovered area depends on the uncovered area of the previous time step minus the new area covered by the sensor footprint at the end of the time step. In order to ensure that the vehicle covers the most area of the bounded region the formulation has a no-fly zone/obstacle from the area previously covered, provided by the union of the regions . As the mission progresses, the region that vehicle has to avoid increases since the area of bounded regions covered by the vehicle increases.

(4)

S.t.

The boundary conditions of the optimization of the formulation are the boundaries of the search region. The vehicle should stay within the boundaries and if the vehicle does move out of the boundary region then the vehicle should return inside of the boundary region as soon as possible. In addition, the boundary condition contains the energy constraint of the maximum energy at the start of the mission. The energy consumption of the vehicle during the mission depends on the velocity, the control input, and time. The turn rate of the maneuver affects the amount of power required by the vehicle, sharper the turn the more power is required by the vehicle.

B.C.

(5)

The algorithm generates the optimal trajectory for a range of turning rates by minimizing the sum of the cost function and the product of the priority function and area function. Equation 6 details the cost function that is used to generate the optimal trajectory of the vehicle for the time interval [(k-1)τ, kτ] for , where τ is the execution horizon (turn duration) and *k* is the number of times the optimum maneuver is executed. In Equation 6 *S* is the priority function, *A* is the area function, and *C* is the terminal function. The optimization of Equation 6 is done in the MATLAB programming environment using the ***fminbnd*** function [14].

(6)

The algorithm in Equation 6 operates in the model-based optimizer block in Figure 3. The model-based optimizer simulates the motion of the vehicle for a planning horizon for kth step. It’s input includes the boundary conditions such as exit and entry point, velocity of the vehicle, entry heading angle and available energy together defines as *R*. The output of the model-based optimizer is the optimal turning rate sequence *u(t)* for the planning horizon of the predicted optimal trajectory. The UAV only executes the first maneuver of the optimal turning rate sequence determined by the model-based optimizer. This is repeated after updating the state of the UAV, *z(t)*. This continues until the boundary conditions are met. The algorithm only applies the first optimal turning rates and feeds back the actual position of the vehicle at the end of the execution horizon to account for disturbances that may alter the position of the vehicle from calculated optimal path for the planning horizon. Figure 4 provides the block diagram of the model-based optimizer. As previously stated, the model-based optimizer plans the path for a particular planning horizon that covers the most area of the search region not previously covered. The model-based optimizer calculates the ideal optimal turning rates of the trajectory for the planning horizon, since it does not account for disturbances that can disturb the vehicle from the calculated path.

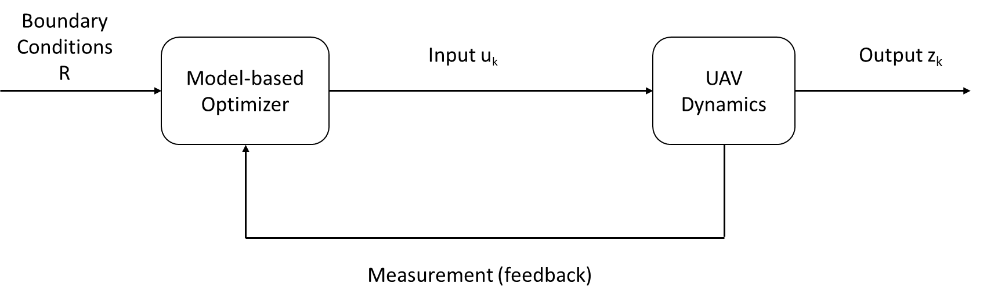


Figure 3: Block diagram of path-planning algorithm execution

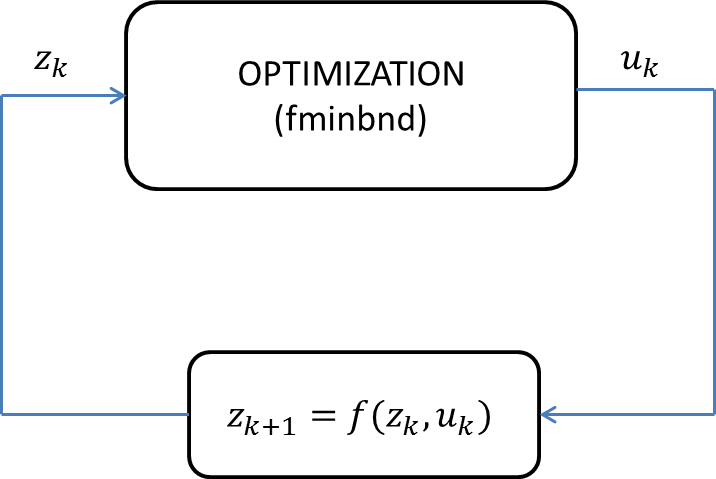


Figure 4: Model-based optimization

1. Priority function

The priority function determines the immediate objective of the vehicle during the mission. The priority function either determines if the priority of the vehicle to is to continue searching or head to the desired exit point. The function can be based on the time constraint as in [9] or can be based on the energy constraint as in this study.

* 1. Time constraint

Equation 7 provides the priority function for the time constraint. The priority determines if the vehicle should continue to search area or head to the desired exit point. If the remaining mission duration at the end of each interval is great than then it prioritizes the vehicle to continue to attempt to the cover the most area possible. However, if the remaining mission duration is less than then the priority function instructs the vehicle to try to reach the desired exit state as soon as possible. The variable is the amount of time required by the vehicle to reach the desired state, using a direct path.

(7)

* 1. Energy Constraint with non-constant power consumption

Equation 7 is similar to Equation 8 but has an energy constraint instead of a time constraint. If the energy remaining in the vehicle at the end of the interval is greater than the energy required to reach the desired exit state, using a direct path, then the priority function directs the vehicle to continue search as much area as possible. However, if the amount of energy remaining at the end of the interval is less than the energy required to reach the desired exit, using a direct path, then the priority function directs the vehicle to reach the exit immediately. Moreover, the priority function takes into consideration that the power consumption of the vehicle is not constant. The energy remaining calculated in the priority function accounts for the varying energy consumption of the vehicle due to the maneuver of the vehicle.

(8)

1. Area Function – A(z, τ)

The area function determines the amount of area covered in each time interval by the possible paths. If the sensor covers area that was not previously covered then the area function is equal to the inverse of the new area covered. However, if the sensor covers area that was previously covered without covering any new area then the area function is equal to the distance from the current state to the centroid of uncovered area. The second condition in the area function ensures that the algorithm always searches for area not previously covered, in order to maximize the amount of area covered at the end of the mission. The amount of uncovered area is obtained using *polybool* – a Boolean operation on polygons in MATLAB.

(9)

where represents the centroid of the uncovered area

1. Terminal Function

The cost function of a receding horizon optimization problem estimates the cost-to-go from a selected terminal state to the final goal. Again, this terminal function can be based on time constraint [9] or the energy constraint as in this study.

* 1. Time constraint – *C(T)*

Equation 10 provides the terminal function for the time constraint. If the remaining time at the end of the interval is greater than the time required to directly head to the desired exit state then the cost is zero, which means that the vehicle’s objective is to cover as much area as possible. If the remaining mission duration at the end of the interval is less than the time-required to head directly to the desired exit state, then the cost function is equal to the inverse of the remaining mission duration. However, if the remaining mission time is less than zero then the cost function is equal to infinity.

(10)

* 1. Energy constraint with non-constant power consumption – *C(E)*

Equation 11 provides the terminal function for the energy constraint. If the remaining energy at the end of the interval is greater than the energy required to directly head to the desired exit state, then the cost is zero. A cost of zero means that the vehicle’s objective is to cover as much area as possible. If the remaining mission duration at the end of the interval is less than the energy required to head directly to the desired exit state, then the cost function is equal to the inverse of the remaining energy required. However, if the remaining energy is less than zero then the cost function is equal to infinity. Moreover, the terminal function with the energy constraint assumes that the energy consumption is not constant since the amount of energy consumption varies due the vehicle’s maneuver. The calculated energy remaining accounts for the energy required by the vehicle for the turning rate used since energy consumption is not constant for different turning rates.

(11)

1. Power required

In order to optimize the trajectory that satisfies the energy constraint requires calculating the power requirements of the vehicle during a maneuver and the power required to reach the desired exit states, using a direct path [12] [13]. Equation 12 calculates the power required by the vehicle to reach the exit state using a direct path. Equation 13 calculates the powers required by the vehicle for particular maneuver. The power required by a maneuver is a function of the load factor of the maneuver. Equation 14 calculates the load factor of the maneuver, which in turn is a function of the turn rate, .

(12)

(13)

(14)

1. **Simulation Results**

The simulation performed considers a single UAV navigating a specified region. The region is defined from the maximum area that the vehicle can observe, based on the vehicle specifications, assuming ideal conditions. The simulation plans the path using a three degree of freedom model and applies the control input, the turning rate obtained in the three degree of freedom model, to a six degree of freedom model of the small UAV. The simulation performed on Simulink is for a Navion general aviation aircraft that is one tenth in scale.

The maximum area that the vehicle can cover assuming straight and steady level flight, is 429,460 m2 that will provide a square region of approximately 655 m by 655 m. Table 1 provides the power specifications sensor footprint of the vehicle in the simulation. The camera is assumed to always pointing straight to the ground during the maneuvers .i.e. it is gimbaled such that the focal plane of the lens is parallel to the ground. In addition, the simulation considers that the vehicle is operating at altitude of 121.92 meters, which is the federal operating limit for model airplanes, on a standard day. The intended exit point

Table 1: Properties of the Vehicle for Simulation

|  |  |
| --- | --- |
| **Property** | **Value** |
| ***Vehicle specification*** |  |
| Oswald efficiency | 1 |
| Motor efficiency | .9 |
| Max load factor | 1.5 |
| ***Battery specification*** |  |
| Electric charge (mAh) | 2200 |
| Voltage (V) | 11.1 |
| ***Sensor specification*** |  |
| Camera radius (m) | 50 |

Moreover, the Simulink model of the Navion general aviation aircraft incorporates three autopilots. The model includes a velocity, altitude, and heading hold autopilots. Figure 5 demonstrates the Simulink model used in the simulation of the vehicle performing the mission. As previously mentioned, the Simulink model of the Navion general aviation aircraft includes the 6-DOF block. The input for the Simulink model of the Navion general aviation aircraft is the turning rate, which is selected during the path-planning phase using a 3-DOF model of the vehicle. From the turning rate and the turn duration the heading of the vehicle at the end of the turn duration is determined, which is then provided to the heading autopilot.



Figure 5. Simulink model of Navion General Aviation Aircraft

A simplified diagram of the Simulink simulation model presented in Figure 5 is provided in Figure 6. As previously stated, the simulation includes three autopilots that were designed specifically for this mission. In will further elaborate each autopilot later in the paper. The autopilots are designed so that the motion of the vehicle is as close as possible to the dubin car model, in which the vehicle moves on a plane. The altitude hold autopilot try to maintain the altitude of the vehicle as it performs a maneuver. The directional hold autopilot performs a maneuver from the control input, which is the desired turn rate.

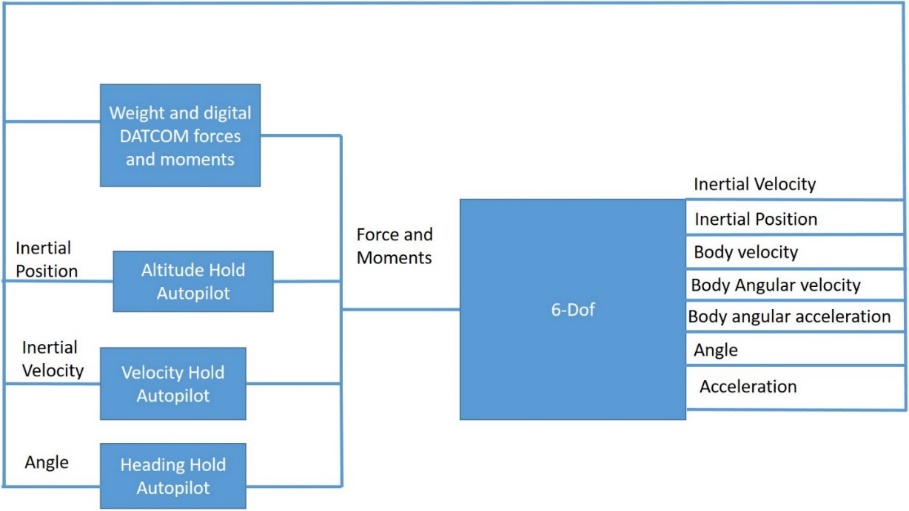


Figure 6. Diagram of simulation model

The altitude hold autopilot is presented in Figure 7. The altitude hold autopilot has a vertical velocity, pitch, and pitch rate feedback to improve the performance of the altitude hold. The altitude controller in the altitude hold autopilot is a PID controller. The vertical velocity controller and pitch controller in the altitude hold autopilot vary for the selected turn duration. The varying velocity controller and pitch controller make the altitude hold autopilot an adaptive autopilot. An adaptive autopilot is required because the performance of a fixed autopilot degrades for larger turn durations.

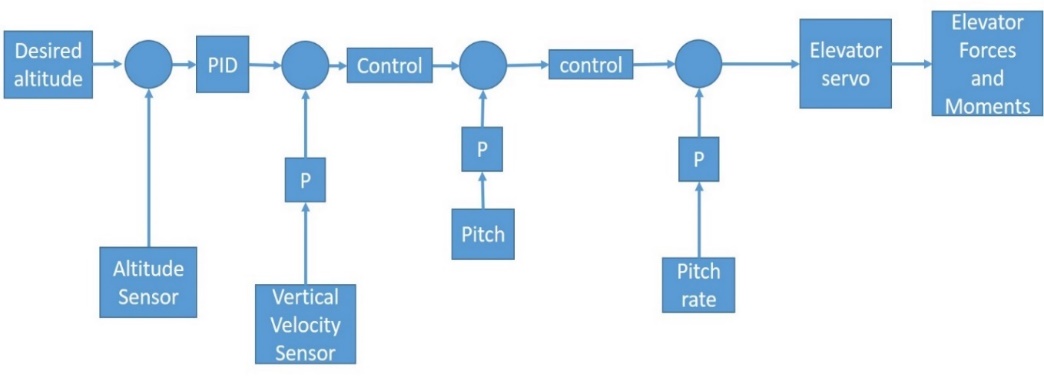


Figure 7. Altitude Hold autopilot

The velocity hold autopilot is presented in Figure 8. The velocity hold autopilot has a simple PID controller. The velocity hold autopilot is not an adaptive autopilot since the turn duration does not affect the performance of the autopilot. The PID controller is tuned using the Simulink control tuner.

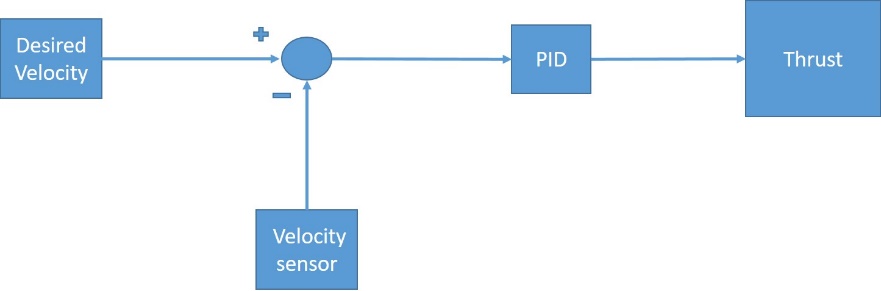


Figure 8. Velocity Hold Autopilot

Figure 9 provides the directional autopilot used to control the vehicle. The autopilot determines the amount of aileron and rudder deflection is required to obtain the desired maneuver. The directional autopilot is an adaptive autopilot as the altitude hold autopilot. From the desired turning rate we determine the amount of yaw that the vehicle requires by the end of the turn duration. The amount of yaw is used instead of the turning rate in the design of the autopilot because the turning rate is too noise making it difficult to design an autopilot for the turning rate. In addition, from the yaw we can calculate the amount of roll required to perform the desired maneuver. The autopilot has a PID controller after calculating the amount of roll required by the vehicle from the yaw error. The controllers that are adaptive in the directional autopilot are the yaw controller and the pitch controller. The adaptive controller are inversely proportional to the turn duration. The PID controller in the directional autopilot is tuned using Simulink controller tuner.

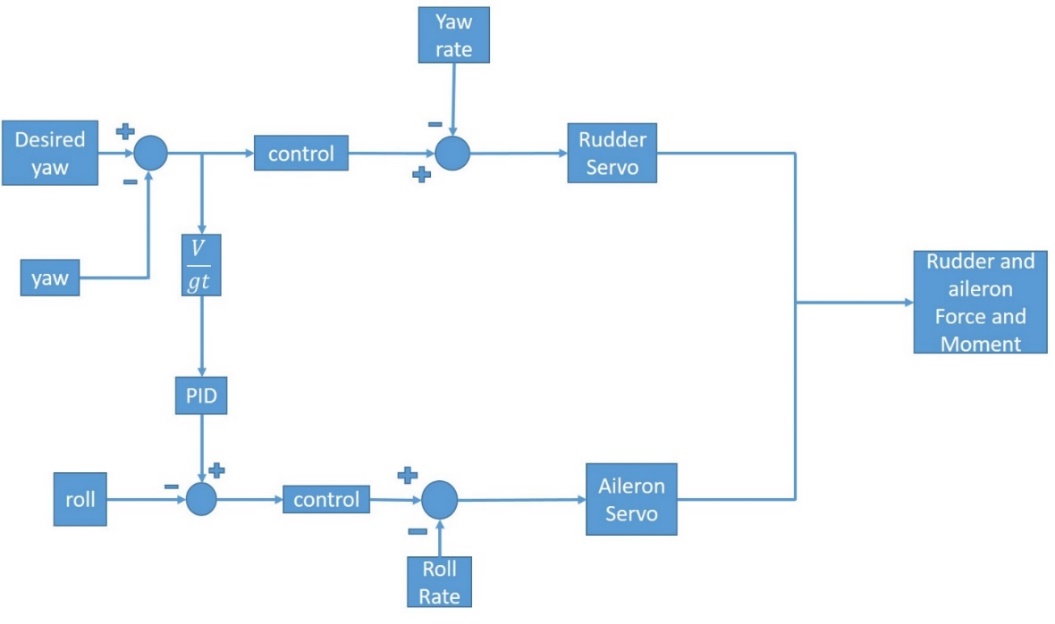


Figure 9. Direction Autopilot

Figure 10 provides the performance of the directional hold autopilot designed for this mission purpose. The autopilots designed for this mission are adaptive in which controller in the directional and altitude hold autopilots change for different turn durations. The performance presented is for a turn duration of 5 seconds and desired turning rate of .39 rad/s, which is the maximum turning rate of the vehicle for the load limit. The response of the directional autopilot has the same profile, presented in Figure 10, for different turning rates.



Figure 10. Directional Autopilot Performance

Figure 11 provides the performance of the vehicle motion with the planned motion from the optimization. The black path is the actual path the vehicle takes and the red path is the planned path. We see that the vehicle does not track the planned path accurately but executes a path that is very similar. The actual path presented is the path from the 6-dof simulation generated from desired turning rate. The planned path is a path generated using the dubin car model. The magenta out line is the area covered by the sensor for the path presented in the figure.



Figure 11. Performance of planned path and actual path

Figure 12 provides the percentage of the bounded region that is covered by the vehicle during the mission for different turning durations. The figure provides the percent of area covered by the vehicle with a path planning implementing the energy constraint optimization. In addition, the figure includes the percentage of the bounded region covered by the vehicle with a path planning using the time constraint optimization, while considering the varying power consumption throughout the mission. It is evident that the coverage of the bounded region by the sensor footprint of the vehicle is not significantly degraded. Both path planning optimizations achieve almost the same percentage of coverage of the bounded region.



Figure 12. percent of area covered

Figure 13 provides the distance of the vehicle from the desired exit point at the end of the mission. The figure provides the distance of the vehicle at the end of the mission for both time constraint and energy constraint path planning optimization. From the figure it is evident that the path planning with an energy constraint optimization performs better than the path planning with the time constraint optimization. The energy optimization ensures that the vehicle approaches the desired exit point at the end of the mission as much as possible. The vehicle ends the mission closer to the desired exit state when using the energy constraint path planning optimization. Reaching the desired exit state at the end of the mission is important because it facilitates recovering the vehicle. The time constraint path planning optimization makes it difficult to recover the vehicle since the predicted end point at the end of the mission will not be correct location. The vehicle will consume all the available energy at the start of the mission before the predicted mission duration since the time constraint path planning optimization assumes constant power consumption but the vehicle is actually consuming power at a varying rate, due the varying turning rate of the maneuvers during the mission.



Figure 13. Distance from desired exit state

Figure 14 provides the durations of the mission for different turn durations. The time provided in the figure consider the varying power consumption by the vehicle as it performs the mission. Form the energy capacity of the battery, assuming constant power consumption, the determined mission duration is 600 seconds. Since the vehicle is not consuming power at a constant rate, due to the maneuvers, the actual mission duration is less than the expected mission duration.



Figure 14. Mission duration

Figure 15 provides the coverage performance of the vehicle for both energy and time constraint like Figure 12. The difference between figure 15 and figure 12 is that the coverage performance for the time constraint is for a generated path with a mission duration of 400 seconds instead of the 600 seconds previously stated. For this optimization we limit the time constraint/mission duration to 400 seconds because Figure 14 demonstrates that the maximum mission duration for the energy available at the start of the mission is 400 seconds. By decreasing the expected mission duration, the percent of area covered decreases and improves the end location of the vehicle at the end of the mission from the desired exit state. From Figure 16 it is evident that decreasing the expected mission duration decreases the distance of the vehicle at the end of the mission and the desired end state. Even though, decreasing the expected mission duration decreases the distance at the end of the mission, the energy optimization still performs better overall. Moreover, the realistic mission duration provided for the time constraint requires the energy constraint in order to determine the mission duration for a varying power consumption.



Figure 15. Coverage Performance



Figure 16. Distance from exit performance

1. **Conclusion**

This paper presents a path planning algorithm for that utilizes available energy as a constraint and updates the remaining energy after a maneuver based on the power consumed. Further, the maneuvers which are defined by rates of turn are not restricted to a discrete set but chosen from a range of possible values. Simulation results show that the novel formulation of the optimization problem does not degrade the area covered as compared to the typical optimization using a time constraint. Evaluation of the overall mission duration assuming that the power required is not constant but dependent on the maneuver (turn rate) indicates that the optimization using time constraint calculates it incorrectly. This further causes the vehicle to complete its position at a location that is not at the one predicted by the algorithm. Contrary to that, the algorithm proposed in this paper that uses energy as a constraint and varies the power required based on the maneuver is more accurate in the prediction of the mission duration and final position. Further, direct comparison of the final position of the vehicle when comparing the two optimization formulation shows that the energy constraint allows the vehicle to be recovered from a location closer to the desired exit point. Potential future work includes understanding the effect of disturbances and model uncertainties on the performance of the algorithm and implementation using multiple cooperative vehicles.

**References**

|  |  |
| --- | --- |
| [1] | A. Ryan , M. Zennaro , A. Howell , R. Sengupta and J. K. Hendrikc, "An Overview of Emerging Results in Cooperative UAV Control," University of California, Berkeley, Berkeley. |
| [2] | A. Ryan and J. K. Hendrick , "A mode-switching path planner for UAV-assisted search and rescue," University of California Berkeley, Berkeley. |
| [3] | S. Waharte and N. Trigoni , "Supporting Search and Rescue Operations with UAVs," in *International Conference on Emerging Security Technologies*, Canterbury, 2010. |
| [4] | J. S. Kim and B. K. Kim , "Minimum-Time Grid Coverage Trajectory Planning Algorithm for Mobile Robots with Battery Voltage Constraints," in *International Conference on Control, Automation and Systems*, Gyeonggi-do, 2010. |
| [5] | D. P. Gillen, "Cooperative Behavior Schemes for Improving the Effectiveness of Autonomous Wide Area Search Munitions," Air Force Institute of Technology, Ohio, 2001. |
| [6] | M. Balakrishnan, "Coverage Path Planning and Control for Autonomous Mobile Robots," University of Central Florida, Orlando, 2005. |
| [7] | S. Waharte , A. Symington and N. Trigoni , "Probabilistic Search with Agile UAVs," in *2010 IEEE International Conference on Robotics and Automation*, Anchorage, 2010. |
| [8] | S. Y. a. Z. Zhou, "Cooperative Control for Target search, Classification and Attack for AUAVs (Attack Uninhabiitted Air Vehicle," in *26th Chinese Control Conference*, Hunan, 2007. |
| [9] | V. Kumar, "Cooperative Control of UAVs for Search and Coverage," Philadelphia, 2006. |
| [10] | A. Ahmadzadeh, A. Jadbabaie, V. Kumar and G. J. Pappas, "Multi-UAV Cooperative Surveillance with Spatio-Temporal Specifications," in *45th IEEE Conference on Decision and Contorl* , San Diego , 2006. |
| [11] | A. Ahmadzadeh, J. Keller , G. J. Pappas, A. Jadbabaie and V. Kumar , "An optimization based Approach to Time Critical Cooperative Surveillance and Coverage with Unmanned Aerial Vehicles," in *Experimental Robotics: The 10th International Symposium on Experimental Robotics* , vol. 39, Springer Berlin Heidelberg, 2008. |
| [12] | D. P. Raymer, Aircraft Design: A Conceptual Approach Fourth Edition, Reston: American Institute of Aeronautics and Astronautics Inc., 2006. |
| [13] | J. John D. Anderson, Introduction to Flight, New York: McGraw-Hill, 2005. |
| [14] | Matlab, version 8.6.0 (R2015b), Natick: The Mathworsk Inc. , 2015. |
| [15] | A. D. J. Caves, "Human-Automation collaborative RRt For UAV Mission Path Planning," Massachusetts Institute of Technology, Boston, 2010. |
| [16] | H. Choset and P. Pigno, "Coverage Path Planning: The Boustrophedon Decomposition," in *International Conference on Field And Service Robotics*, 1997. |
| [17] | C. Coleman, "Design of an Autonomous Platform for Search and Rescue UAV Network," Worcester Polytechnic Institute , 2012. |
| [18] | P. Eaungpulswat, "Area Coverage Algorithms for Multiagent Surveillance Task," Technische Universitat Hamburg-Harburg, Hamburg, 2012. |
| [19] | S. P. F. a. J. S. B. M. Esther M. Arking, "Approximation Algorithms for Lawn Mowing and Milling," *Computational Geometry ,* vol. 17, no. 1-2, pp. 25-50, 2000. |
| [20] | J. W. Langelaan, "Gust Energy Extraction for Mini- and Micro- Uninhabited Aerial Vehicles," in *46th Aerosciences Conference*, Reno, 2008. |
| [21] | J. W. Langelaan, "Long Distance/Duration Trajectory Optimization for Small UAVs," in *Guidance, Navigation, and Control Conference* , Hilton Head, 2007. |
| [22] | S. M. LaValle, Planning Algorithms, New York : Cambridge University Press, 2006. |
| [23] | I. Maza and A. Ollero , "Multiple UAV Cooperative Searching Operation Using Polygon Area Decomposition and Efficient Coverage Algorithms," University of Seville, Seville. |
| [24] | N. Nourani-Vatani, "Coverage Algorithms for Under-Actuated Car-like Vehicle in an Uncertain Environment," in *IEEE International Conference on Robotics and Auomation* , Roma, 2007. |
| [25] | N. Nourani-Vatani, "Environment, Coverage Algorithms for Under-actuated Car-like Vehicle in an Uncertain," Technical University of Denmark, Lyngby, 2006. |
| [26] | M. M. Polycarpous, Y. Yang and K. M. Passino , "Cooperative Control of Distributed Multi-Agent Systems," *IEEE Control Systems Magazine,* p. 27, 2001. |
| [27] | P. Rocha and M. A. Gomez, "A Decomposition Approach for the Complete Coverege Path Planning Problem," Universidade Do Porto, Porto. |