**Path Planning of Small UAV with High Level of Autonomy for a Search and Coverage Mission and Simulation**

**German Gramajo1 and Praveen Shankar2**

California State University, Long Beach

Mechanical and Aerospace Engineering Department

**Abstract**

A path planning algorithm for a search and coverage mission using a small UAV that optimizes the trajectory based on stored energy and maneuverability constraints is presented. The proposed formulation has a high level of autonomy, without requiring an exact choice of optimization parameters. The computed trajectory maximizes spatial coverage while closely satisfying terminal constraints on the position of the vehicle and minimizing the time of flight. 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 use the probability of the location of an object/target in a region, requiring some prior information of the bounded region [7]- [8]. In this form of search the user has to provide the amount of targets in the bounded region and the probability of the distribution of the targets/objects in the bounded region from prior survey information. This time of search method is primarily used in military application in which satellite information can provide the likelihood of the location of objects/targets. The UAV would primarily focus in sections with the greatest probability of identifying object/targets and then searches the regions with the least probability of a target/object.

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 are several shortcomings of the algorithm presented in [9]. The algorithm uses a set mission duration with an 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. Secondly, the algorithm selects the optimal path using a discrete set of turning rates that has to be specified prior to the optimization. Even though, the algorithm can potentially generate the most optimal trajectory, an exhaustive simulation analysis is required to obtain the appropriate set of turning rates resulting in 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. Finally, the algorithm 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. This can result in an overestimation of the actual area covered by the vehicle since it considers a discrete space as fully covered as long as the sensor footprint covers the center of the each discrete. This can be overcome by the varying the amount and size of each discrete space. However, finding the appropriate space discretization is computationally expensive.

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 that will be used in the optimization formulation.

(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 is assumed to have 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 *pentry(x,y)* and is expected to terminate the mission at *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.

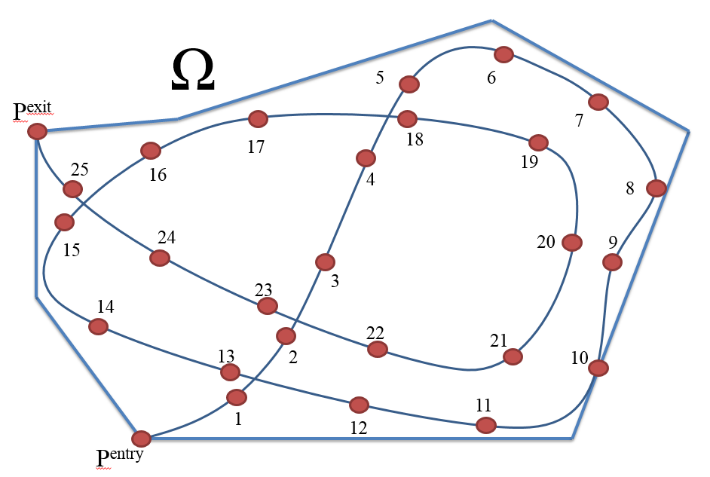


Figure 1: Problem of generating a trajectory that maximizes the area covered

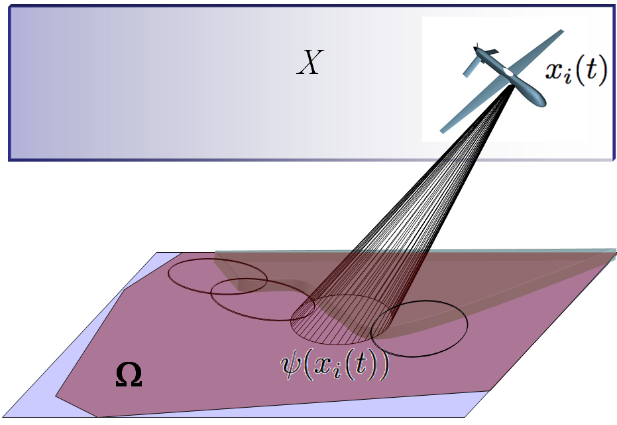


Figure 2: Area covered by sensor footprint for generated trajectory throughout the mission

The procedure 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 (parameterized by turn rate) executed by the vehicle. This enables the accurate determination of the mission duration and enforcing the terminal position requirements 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 avoiding previously covered area as much as possible. The algorithm avoids going over previously covered area as much as possible by redirecting the vehicle portions of the bounded region that have not been previously covered, with an objective function when no new area is covered. The last feature of the proposed formulation of the path optimization is that it does not require choosing the density of the space discretization/decomposition. The density of the space discretization depends on the size of the discretized space and the distance between the center points of the discretized spaces. 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 that 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 with a refresh rate γ, in seconds. The goal is to find a feasible trajectory, defined by , to get maximum coverage of the bounded region in every time interval τ. The time interval τ is the turn duration. The solution requires discretization over time since a continues solution is difficult to generate.

Turning rate range

The proposed formulation optimizes the maximum coverage trajectory for a range of turning rates. Equation (3) provides calculates the maximum turn rate possible based on the maximum load factor for the vehicle [11]- [12].

(3)

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

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, obstacles (already covered area) and boundary conditions. Ω is the area of the bounded region and the variable Ψ represents the sensor footprint. The vector z(t) provides the location and orientation of the vehicle in the xy-plane. The x and y position of the vehicle are provided in discrete form, since it allows for calculation of 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. Based on these assumptions, the path optimization can be formulated as

(4)

S.t.

One of the boundary conditions of the optimization is the physical boundary 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. The second boundary condition is the maximum stored energy available (Etotal) at the start of the mission. The rate of 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. Larger the turn rate, the more power is required by the vehicle to execute the maneuver.

B.C.

(5)

This formulation generates the optimal turn-rate from the allowable range of turning rates to minimize the cost 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

(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.

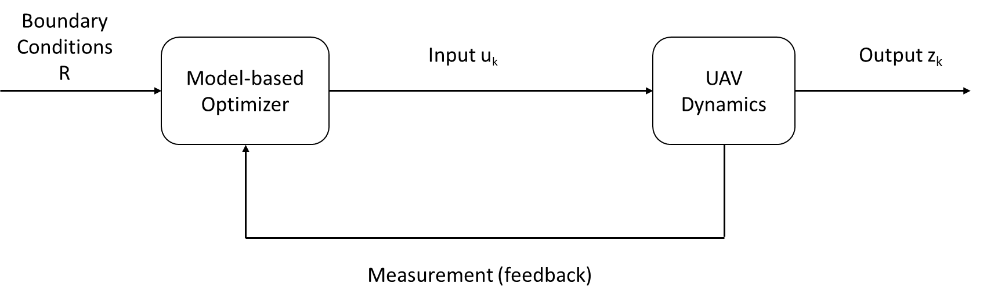


Figure 3: Block diagram of path-planning algorithm execution

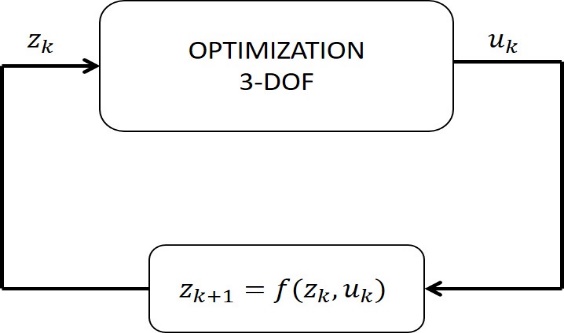


Figure 4: Model-based optimization

1. Priority function, S

The priority function determines the immediate objective of the vehicle during the mission. It controls the priority of the vehicle to continue searching or redirect 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. 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

to the priority function for energy constraint is given by Equation (8). 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 assuming the vehicle is initial heading to the vehicle, 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 travel to 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)

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. We obtain the inverse of the new area covered since the algorithm is a minimizing function, so the more area covers the lower the value of the function. 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 by finding the intersection of the contour of the bounded region and the contour of the area covered. The contours are defined by a matrix of the vertices of each contour. The Boolean operation of not removes the intersecting area of the bounded region and the area covered from the bounded region, generating a matrix of the vertices of the contour.

(9)

where represents the centroid of the uncovered area

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.

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 [11]- [12]. 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. **6-DOF UAV Simulation Platform**

.

The performance of the optimal path planning algorithm is evaluated through its implementation on a high-fidelity 6-DOF nonlinear simulation of a UAV in the Matlab/Simulink environment. The UAV is assumed to be a 1/10th scale model of the Navion general aviation aircraft, allowing for the determination of its geometry and mass properties. The UAV has an elevator, ailerons and rudder to control its motion. Additionally, it has a battery powered motor that produces a constant thrust throughout the mission. The aerodynamic coefficients that are required to determine the forces and moments acting on the UAV to complete the 6-DOF simulation model were obtained using the USAF Datcom software [13] for a given operating condition and imported into Matlab. The 6-DOF equations of motion are written as follows

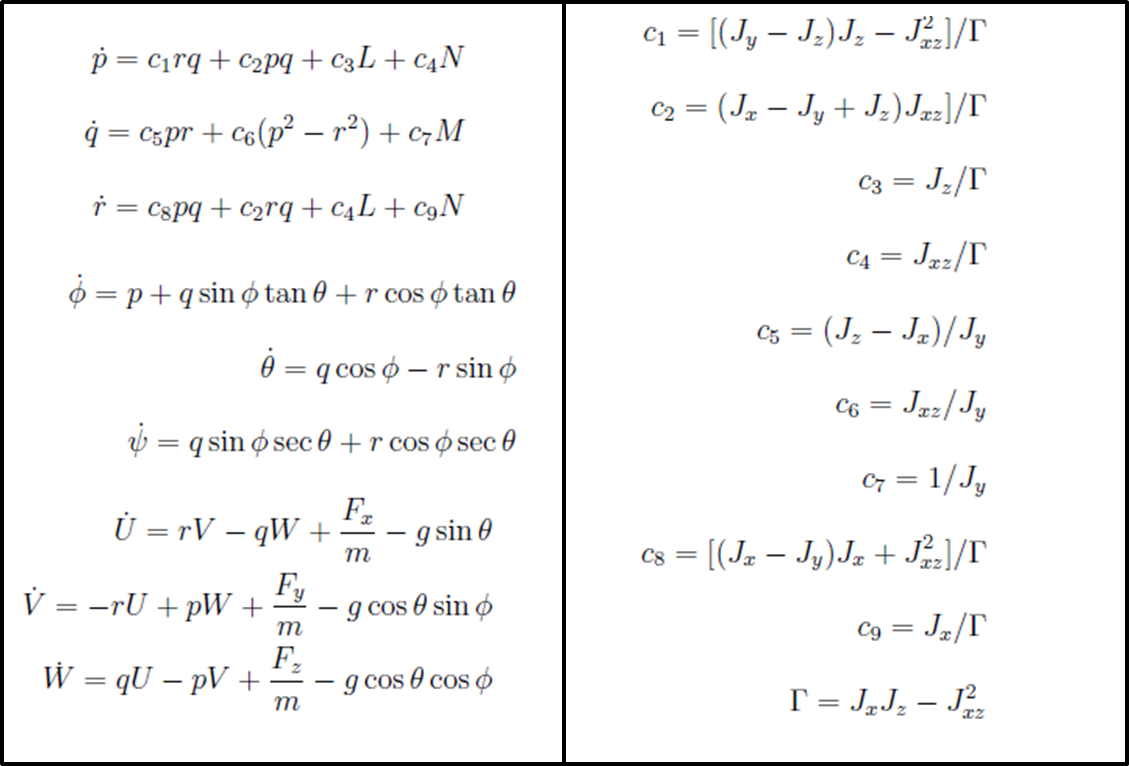


Figure 5: 6-DOF Dynamics of UAV. Fx, Fy, Fz, L, M, N are the aerodynamic forces and moments acting on the aircraft.

The optimization procedure assumes that the aircraft is flying at a constant velocity and fixed altitude. Additionally, it assumes that the turn rate selected by the optimization can be achieved by the aircraft. This requires the UAV simulation to have a velocity-hold, altitude hold and a directional (heading-hold) autopilot. The Simulink model of the Navion general aviation aircraft includes the 6-DOF block and the three aforementioned autopilots. The input for the Simulink model of the Navion general aviation aircraft is the optimized turning rate, which is selected during the path-planning phase based on 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.

The block diagram of simulation model presented in Figure 6. The autopilots are designed so that the motion of the vehicle is as close as possible to the Dubin car model, which requires the vehicle to moves in a plane at constant velocity. The altitude hold autopilot maintains the altitude of the vehicle as it performs a maneuver. The directional hold autopilot allows the UAV to perform a maneuver based on the desired turn rate.

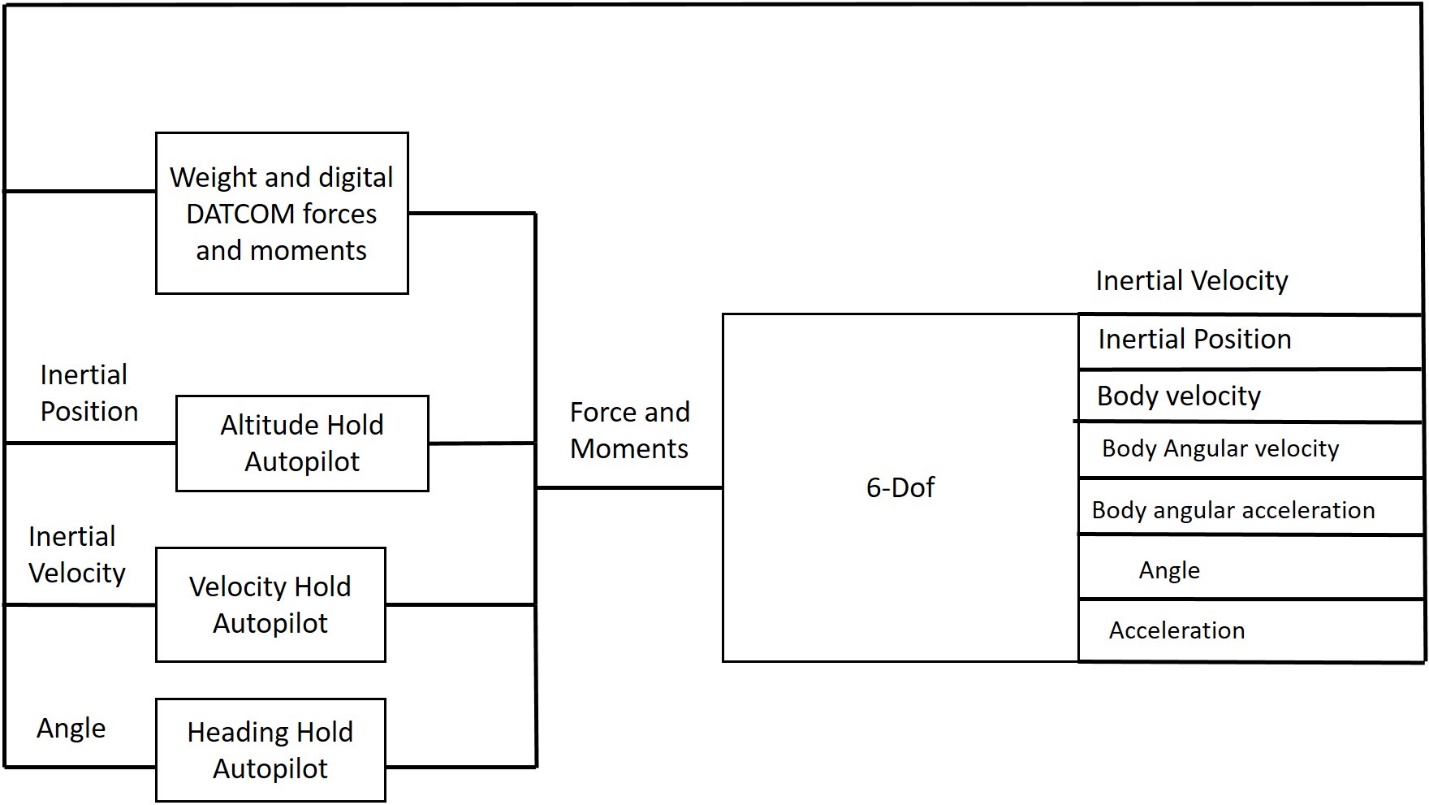


Figure 6. Block Diagram of Simulation Model

The following subsections discuss the autopilots that have been developed to control the motion of the aircraft.

Altitude Hold Autopilot

The altitude hold autopilot ( Figure 7) has a vertical velocity, pitch, and pitch rate feedback. The control law for the altitude hold autopilot is a PID controller operating on the error in altitude. The inner loops that feedback the vertical velocity, pitch and pitch rate are proportional gains that are determined based on the selected turn duration. The varying gains for the feedback of the vertical velocity and pitch are required because the performance of a fixed autopilot degrades for varying turn durations.

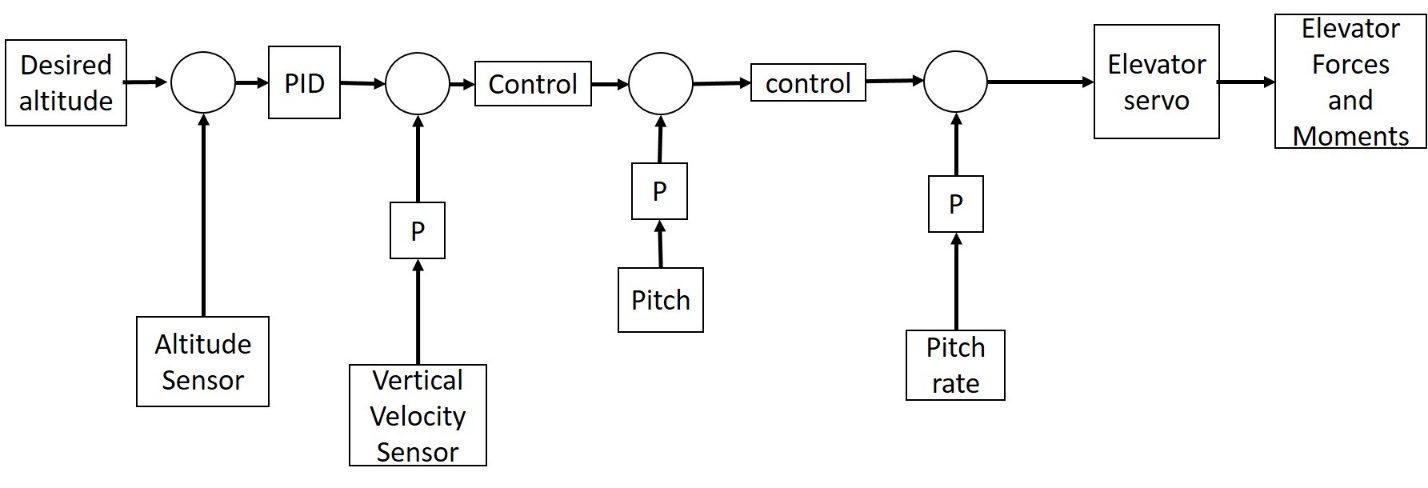


Figure 7. Altitude Hold autopilot

Velocity Hold Autopilot

The velocity hold autopilot is presented in Figure 9. The velocity hold autopilot has a simple PID controller. The velocity hold autopilot has fixed gains since the turn duration does not affect the performance of the autopilot.

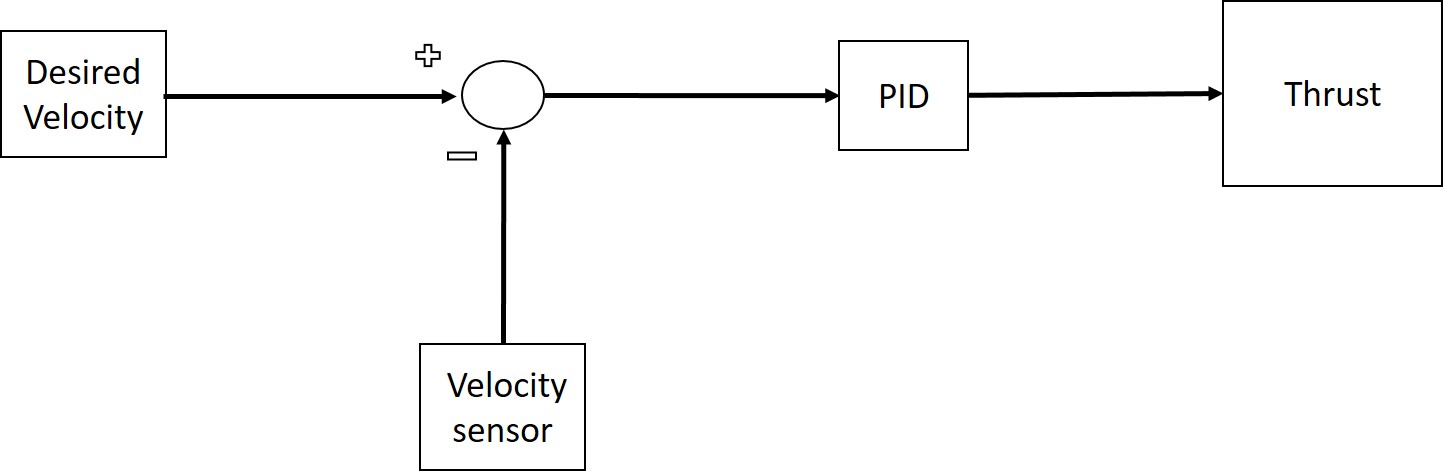


Figure 8. Velocity Hold Autopilot

Heading Hold/Directional Autopilot

Figure 9 provides the block diagram of directional autopilot used to maneuver the vehicle based on the desired turn rate. The autopilot determines the aileron and rudder deflection required to perform the desired maneuver. From the desired turning rate () and length of turn, the required yaw angle () at the end of the turn is determined. This yaw angle is used instead of the turning rate in the design of the autopilot because the turning rate is too noisy making it difficult to design a heading hold autopilot. In addition, from the yaw, one can determine the roll angle required to perform the desired maneuver. The autopilot has a PID that operates on the roll angle error. The controllers that are adaptive in the directional autopilot are the yaw controller and the pitch controller.

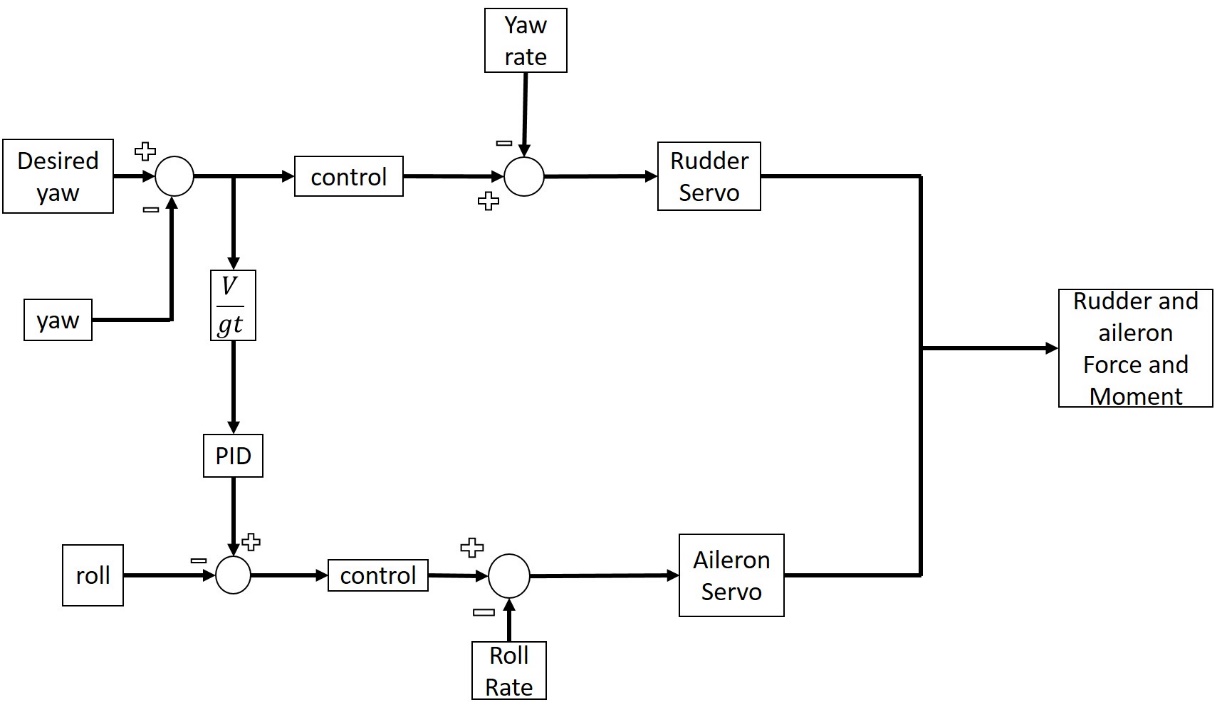


Figure 9. Direction Autopilot

Figure 11 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.

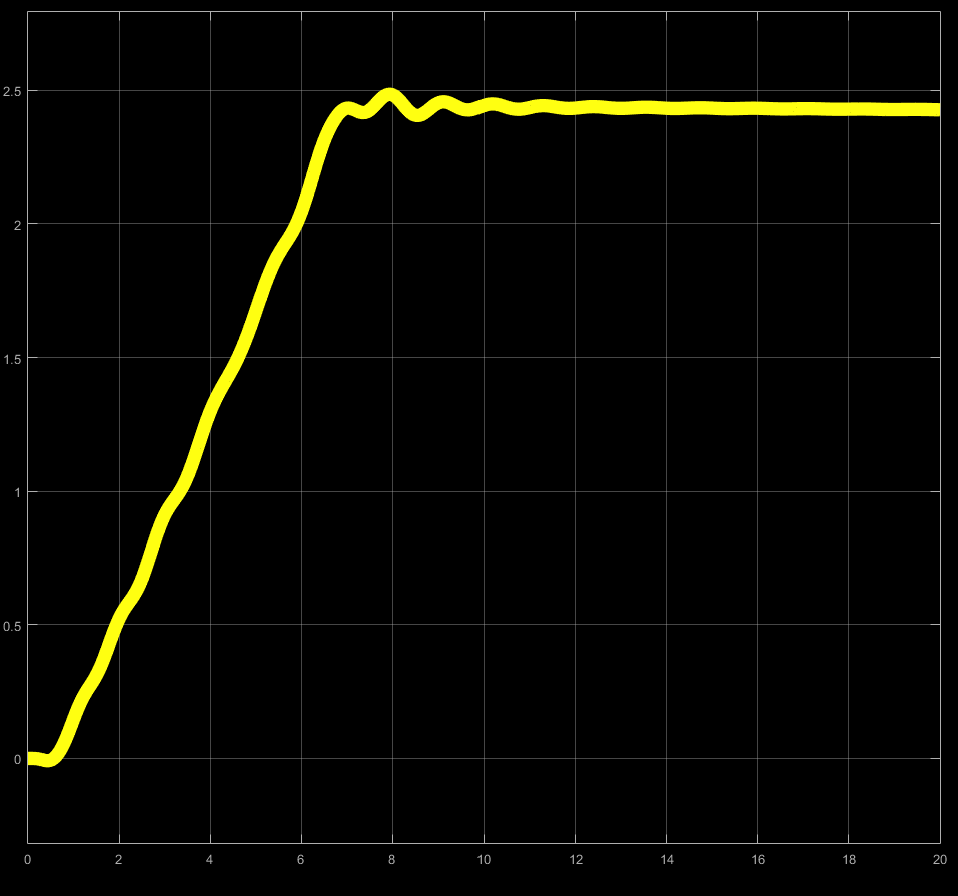


Figure 10. Directional Autopilot Performance

1. **Path Optimization 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 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.

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 |



Figure 11. Performance of planned path and actual path. 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.

Figure 11 plots the actual vehicle path (obtained from 6-DOF simulation) along with the planned path (obtained from the 3-DOF model) from the optimization. It can be seen that the vehicle does not track the planned path accurately but executes a path that is very similar.

Comparative Results

This section discusses the performance of the optimization algorithm utilizing an energy constraints vs one using a time constraint. The performance is measured based on the following factors

1. Percentage of total area covered
2. Distance of the final position of the UAV from the desired exit point

The percent of total area covered is indicative of accomplishing the primary mission while the distance from exit represents the ease of access to the final position of the UAV once the stored energy is spent. The simulation is performed for different values of the duration of the turn as this is an important parameter that affects the total area covered.

Figure 12 provides the percentage of the bounded region that is covered by the vehicle during the mission for different turning durations based on both the energy constraint based optimization and the time constraint based 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, again, for both time constraint and energy constraint path planning optimization. From the data 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 is closer to the desired exit point at the end of the mission for all turn durations. Reaching the desired exit state at the end of the mission is important because it facilitates easy recovery of 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 while 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. Coverage Performance

Figure 15 and Figure 16 provide the coverage performance for a turn duration of 11 s and the distance of the vehicle at the end mission for a turn duration of 11 s. The performance presented in the figures is for the time constraint optimization. The results are for an expected mission of 400 seconds instead of 600. The lower time constraint makes the vehicle end the mission closer to the desired exit state.



Figure 16. Distance from exit performance

1. **Conclusion**

This paper presents a path planning algorithm that maximizes the area covered with energy as a constraint. The algorithm relies on determination of the remaining energy after a maneuver is completed that is consumed 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 that depend on the load factor of the vehicle for a sustained turn. 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 and final position of the UAV for both the energy constraint and time constraint based optimization indicates that the optimization using time constraint calculates it incorrectly. 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.

# **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] | D. P. Raymer, Aircraft Design: A Conceptual Approach Fourth Edition, Reston: American Institute of Aeronautics and Astronautics Inc., 2006. |
| [12] | J. John D. Anderson, Introduction to Flight, New York: McGraw-Hill, 2005. |
| [13] | *DATCOM+ Pro Version 3.2,* Orlando: Holy Cows , INC, 2014. |
| [14] | A. D. J. Caves, "Human-Automation collaborative RRt For UAV Mission Path Planning," Massachusetts Institute of Technology, Boston, 2010. |
| [15] | H. Choset and P. Pigno, "Coverage Path Planning: The Boustrophedon Decomposition," in *International Conference on Field And Service Robotics*, 1997. |
| [16] | C. Coleman, "Design of an Autonomous Platform for Search and Rescue UAV Network," Worcester Polytechnic Institute , 2012. |
| [17] | P. Eaungpulswat, "Area Coverage Algorithms for Multiagent Surveillance Task," Technische Universitat Hamburg-Harburg, Hamburg, 2012. |
| [18] | 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. |
| [19] | J. W. Langelaan, "Gust Energy Extraction for Mini- and Micro- Uninhabited Aerial Vehicles," in *46th Aerosciences Conference*, Reno, 2008. |
| [20] | J. W. Langelaan, "Long Distance/Duration Trajectory Optimization for Small UAVs," in *Guidance, Navigation, and Control Conference* , Hilton Head, 2007. |
| [21] | S. M. LaValle, Planning Algorithms, New York : Cambridge University Press, 2006. |
| [22] | I. Maza and A. Ollero , "Multiple UAV Cooperative Searching Operation Using Polygon Area Decomposition and Efficient Coverage Algorithms," University of Seville, Seville. |
| [23] | 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. |
| [24] | N. Nourani-Vatani, "Environment, Coverage Algorithms for Under-actuated Car-like Vehicle in an Uncertain," Technical University of Denmark, Lyngby, 2006. |
| [25] | M. M. Polycarpous, Y. Yang and K. M. Passino , "Cooperative Control of Distributed Multi-Agent Systems," *IEEE Control Systems Magazine,* p. 27, 2001. |
| [26] | P. Rocha and M. A. Gomez, "A Decomposition Approach for the Complete Coverege Path Planning Problem," Universidade Do Porto, Porto. |
| [27] | A. Ahmadzadeh, J. Keller , G. J. Pappas, A. Jadbabaie and V. Kumar , "An optimized-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. |
| [28] | Matlab, version 8.6.0 (R2015b), Natick: The Mathworsk Inc. , 2015. |