Report on R&D
Cooperative Trajectory Planning Of UAVs For Area Coverage

Poulastya Mukherjee*

B-IT Master Studies in Autonomous Systems

Bonn-Rhein-Sieg University of Applied Sciences

Fraunhofer FKIE

Advisors: Prof.Dr. Paul.G.Ploeger**
Dr. Marek Schikora††

January 15, 2016

*poulastya.mukherjee@smail.inf.h-brs.de
**Paul.Ploeger@h-brs.de
††marek.schikora@fkie.fraunhofer.de

Declaration

I, **Poulastya Mukherjee**, hereby declare that this work has not previously been submitted to this or any other university, and that unless otherwise stated, the content is entirely my own work.

Date

Poulastya Mukherjee

Abstract

Traditional approach to coverage planning has always focused on developing new techniques for area decomposition and applying search based techniques to navigate an Unmanned Aerial Vehicle (UAV) through the decomposed areas. A key feature ignored in these approaches is the nature of the motion followed by the UAV. Usually the actual motion path of an UAV, using these traditional techniques, is characterized by sharp turns and quite a few start-stop motions. This results in higher time requirement for completing coverage and depletion of excess battery power (as more work needs to be done to move the UAV). Considering the fact that UAVs have limited computational capability and power supply, these two factors are equally important as the percentage of area covered.

In this work we propose algorithms based on geometric functions which have smoother trajectory compared to the lawnmower approach. By conducting experiments using our proposed methods, we were able to prove that algorithms with smoother trajectories take 22.22% lesser time and consume 30% lesser energy compared to lawnmower approach - while achieving atleast 98% coverage.

1 Acknowledgment

I would like to express gratitude to both the supervisors, Prof. Paul G. Ploeger and Dr. Marek Schikora for their continuous guidance and invaluable inputs remarks and engagement through the learning process of this Research and Development project. Furthermore I would like to thank the entire team of Sensor Data Fusion (SDF) from Fraunhofer FKIE for their continued support and the help they provided whenever I faced any difficulty. Last but not the least I would like to thank all my dear friends in Masters in Autonomous Systems (MAS) for supporting me throughout the course of this project.

Contents

List of Figures

2 Introduction

2.1 Motivation

The late 20th century marked the beginning of a shift from electronics age to information age. Be it health-care, entertainment, defense, industries, having thorough information about every aspect of operation is considered critical. Having requisite amount of information not only helps in devising detailed plans but also provides insights about how to deal with a situation in which a plan has failed.
For situations like post disaster damage mitigation, formulating defense reconnaissance plans, or conducting geological survey of a terrain, gathering information is the most important aspect. A common solution for all these three situations is to cover a targeted area with a sensor's footprint so as to gather information about them. The technical term for solution to these classes of problems is called area coverage [1].
Area coverage is usually done by attaching a sensor(e.g. camera, laser scanner) to a robot and moving the robot in the target area in such a way, that information about the entire area is captured by the robot's sensor. Like most robotic solutions, ground robots have been traditionally used for area coverage as well because:

- they are easier to control
- they are robust
- they can carry higher loads compared to aerial vehicles which means they can carry better battery,greater array of sensors, better computers etc.

However certain disadvantages of using ground robots in area coverage scenario are:

- they are terrain dependent (although all terrain robots exists, they are slow and might fail if the terrain is extremely uneven)
- they might not be able to overcome/bypass all obstacles

Herein lies the key advantage of Unmanned Aerial Vehicles or UAVs. Due to the fact that UAVs have an extra degree of motion, they can avoid obstacles much easier compared to ground robots. For disaster mitigation scenarios and defense reconnaissance missions, only gathering information is not sufficient, gathering information in the shortest possible time is also a key requirement. Being airborne, enables the UAVs to take the shortest distance to a goal - which might not be always possible for a ground robot - thus saving a lot of time.

Recent advancements in electronics have led to significant increase in performance of computers while both their size and weight have drastically fallen. This has led to the development of UAVs with better on-board computers and sensors, which have increased their scope of usage especially in area coverage problems. However despite the technological advancements, UAVs still

- have lower computational ability
- and poor batteries

compared to even basic ground robots. Poor batteries directly translate to the fact that UAVs cannot fly for a long time at a stretch. While limited computational abilities restricts it from running advanced algorithms (e.g. search algorithms) on its on-board computer. The combination of these two factors pose a serious challenge to UAVs being used for area coverage problems, which require a robot to move over a large area for long duration of time while performing tasks like recording data and avoiding obstacles. Thus developing efficient techniques for doing area coverage using UAVs are a need of the hour. A possible solution to the problem mentioned above as proposed in [1] is, using multiple UAVs to do area coverage. It has the following advantages,

- reduction in the time required for each UAV to cover an area as now each UAV will have to cover a smaller area
- faster completion of task.

In this work we propose alternate algorithms, which generate a smooth motion trajectory like spirals, Lissajous Curves, hybrid approach. We also use two UAVs for implementing our proposed methods, which will ensure reduced time requirement to achieve goal while not making the process of controlling the overall multi-UAV, system too complex.

2.2 Objective

The aim of this work is to propose alternate methods - spirals, Lissajous Curves, a hybrid approach- to the lawnmower approach [1] and randomized approaches [9] and experimentally determine whether the proposed methods are better or not. For implementing our proposed methods we will use Robot Operating System(ROS) [10]. For simulating and visualizing our results we will use Gazebo [11]

2.3 Structure

- Section 2 descibes the formulated problem and the task description.

- Section 3 describes the state of the art work in the field of area coverage using single UAVs, multiple cooperative UAVs and their deployment using libraries and existing simulation platforms.
- Section 4 describes the approach taken to solve the formulated problem.
- Section 5 describes the experimental setup and the results of the experiment.
- Section 6 describes a use case in which the algorithm was implemented and subsequent results visualized.

3 Problem formulation and Task description

3.1 Problem formulation

Recent works in the field of area coverage using UAVs, is focused on two key aspects for improving area coverage algorithms

- efficient methods of area decomposition [5, 12, 3]
- application of existing search based algorithms to generate coverage paths [13, 14, 4, 9]

A key aspect missing from the above mentioned works is that these works do not focus on the actual motion path followed by the UAVs, which in most cases is simple back and forth motion or a lawnmower type approach, as proposed in [1]. Though the proposed methods in [5, 12, 3, 13, 14, 4, 9] ensure complete coverage, they do not specify any means to reduce the time of coverage. Reducing the time of area coverage is extremely important, to ensure the UAV is able to complete the area coverage task within the time constraints imposed by its limited power supply. Another problem in lawnmower approach is that, during the course of the motion a UAV has to accelerate and decelerate multiple times 1 which increases the time of coverage and also leads to higher power consumption.

3.2 Task Description

In order to compare Like any standard scientific work, we will carry out a state of the art evaluation of area coverage, focusing on area coverage using UAVs in section: 4. Apart from area coverage using UAVs, we will also focus on multi UAV cooperative systems and deployed libraries and existing platforms.
We will then describe our approach in section: 5 where we will focus on formulating mathematical models for proposed algorithms. The following methods are proposed

- Spiral method
- Lissajous Curves based method
- Hybrid approach

For our implementation we will write a ROS package in C++ language which will contain implementation of the proposed algorithms along with some other key functionality for incorporating multi-agent behavior among the two UAVs. We will also implement the lawnmower approach and Rapidly Exploring Random Tree(RRT) approach. In the experiment section we will run each of these algorithms while varying various parameters like size and shape of the area to be covered, number of commanded waypoints. We will then formulate a cost function to determine which method performs the best.

4 State of the Art

Conventional start-goal and map based path planning algorithms do not address problems such as exploration, reconnaissance, [1]. This type of problems require the agent (in our case an UAV) to cover all points in the search space. The problem of coverage has received lot of attention lately due to development of advanced UAVs and increase in their applicability in search and rescue, surveillance etc. [5, 15, 12, 3, 14]. We will look into some existing work in the field of area coverage using UAV. We will also look into path planning of multiple UAVs and finally coverage planning using multiple UAVs. In the last section we discuss an existing platform for simulating UAVs and existing ROS libraries which have been used in this work to implement area coverage algorithms.

4.1 Area Coverage using UAVs

According to author in [1] coverage path planning is defined as a type of algorithm which emphasizes space swept by a robot's sensor. Coverage is useful for certain tasks like floor cleaning, lawn mowing, harvesting, painting etc which are not addressed by conventional start stop algorithms.
A key consideration of area coverage according to the author in [1] is, the choice between complete coverage with higher computational and sensory power requirement or more randomized approach which ensures better cost per quality coverage. Another issue pointed out by the paper is lowering time to completion while ensuring complete coverage. A couple of approaches stated by the paper for reducing time to completion are, using multiple robots instead of a single robot and reducing the number of turns executed by the robot during its movement. Some possible approaches for solving area coverage,are presented in [1], they are classified according to categories, namely **(i)** Heuristic and Randomized **(ii)** Approximate and Exact Cellular Decomposition **(iii)** Multi-Robot Coverage.
Among the heuristic approaches discussed in [1], one approach employs building simple behaviors in the robots. A hierarchical architecture constructed from these simple behaviors result in complex behaviors like exploration. The author also mentions some heuristics for better coverage performance in [1] . In one approach as mentioned in [1] the UAVs are programmed to be repelled from other robots, which ensure the robots are spread over a large area and cover the area uniformly. In this method the paths are not planned and the robots move in random directions until they encounter target or goal positions. The main disadvantage of randomized approach as stated by [1] is that it does not guarantee complete coverage, however few advantages of this approach are non-requirement of costly localization sensors, low computational costs, thus bringing the

overall cost down.
According to [1] in approximate cell decomposition the coverage space is divided into small grids whose size in approximately similar to the footprint of the UAV sensor. There might be some overlap between a pair of cells. A cell is considered to be covered when the robot visits that cell or grid. In case of exact cellular decomposition, the divided cells do not intersect with each other. The method used to cover the area inside each cell is simple back and forth motion as illustrated by figure 1

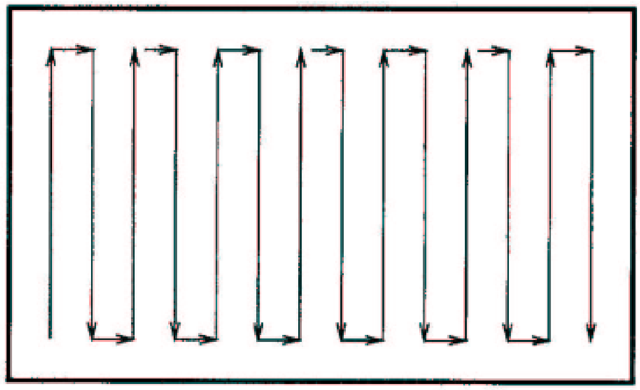
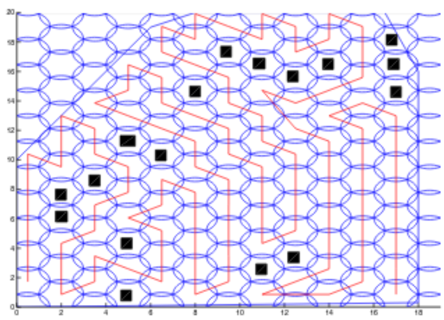


Figure 1: Area coverage inside each grid cell using simple back and forth or lawnmower technique proposed as in [1].

In multi-robot decentralized approach[1] the agents share a memory. The movement of the agents are guided by an adjacency graph. A critical requirement of this approach is that the generated graph should always remain connected in order for the agents to move from one state to another. Two more approaches presented by the author in [1]- Mark and Coverage(MAC) and Probabilistic Coverage(PC) - are two contradictory methods, the former being deterministic and the latter being random- for addressing multi-robot coverage problem. MAC's time complexity was proven to be $O(A/a)$ where A is the total area and a is the step size of the robot's movement. The time complexity for PC was calculated to be $O(n\rho logn)$. A drawback of MAC approach is that it is prone to sensor noise and disturbances, while PC does not guarantee complete coverage. As a way to compensate for the drawbacks of these two methods the author suggests a hybrid approach in [1], combining these two approaches with a balance between performance and tolerance for mission of continuous covering.

The authors in [12] proposes a visual coverage method using a camera and a laser scanner. They posits a two step approach to this problem. The first half deals with a visual coverage descriptor. Visual coverage detector measures coverage of each voxel of 3D map. This map is then fed to 3D reconstruction algorithm. The measure of coverage is the amount of ray penetration in a voxel volume towards the 3D point inside the camera field of view. However the part of this work we will focus on is the coverage approach proposed by the authors. The authors create an utility function that takes into account the amount of visual information the robot's camera takes in for a particular position and orientation. The points in which the camera captures maximum visual features are selected as the next waypoints for the robot to move to. However complexity of a 3D environments makes it impossible to compute closed form solution as it will increase computational cost significantly. Hence the computation is restricted to a square local grid or n voxels per side around the current robot position. A cost function then sums the distance traveled by the UAV. A utility function is then used to maximize the visual representation and minimizes the cost function. The visual representation function is weighted, which signifies the preference of gaining new information over minimizing cost.

In [2] a neural network based approach for area coverage is provided which also includes collision avoidance with both stationary and moving objects. Initially a bounded region is filled with disks whose radius is same as the field of view radius of the sensors(LIDAR or camera) as illustrated in figure 2. Then the objective of the coverage algorithm is to ensure the robot travels to each of these disks thus ensuring complete coverage. The authors generate a dynamic landscape on which the neural network can propagate in such a way that the obstacles repel the robot while the unexplored areas attract it. The robot path is generated from the activity of the neural network and previous locations of the robot. After the coverage paths are generated the authors move on to generate smooth trajectories which ensures a smoother control.



Complete coverage paths, the dark rectangles are stationary obstacles

Figure 2: Disk shaped Area decomposition and complete area coverage proposed in [2].

4.2 Multi UAV Cooperative System

4.2.1 Planning

In this section we will look into some work related to both path and trajectory planning of multi-UAV systems.

In [3], the authors propose a coverage method in which either a hemisphere or cylinder is constructed around the object to be covered. The UAV then follows this trajectory with its sensor - either a laser scanner or a camera pointing towards the center of the hemisphere or the cylinder. This approach is useful for generalizing coverage of objects. In case of the hemispherical trajectory - see figure 3 - the radius is made as large as possible while in the case of cylindrical trajectory - see figure 4 - the radius is the shortest distance from the center without touching the object.

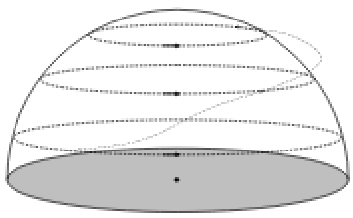


Figure 3: Hemispherical trajectory based coverage method as proposed in [3].

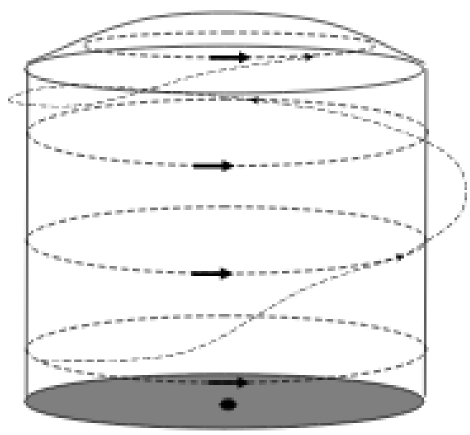


Figure 4: Cylindrical trajectory based coverage method as proposed in [3].

In [13] and [14], the authors propose algorithms for area coverage using UAVs which are based on Particle Swarm Optimization. The authors in [13] aims to minimize the distance traveled by the UAVs for doing coverage of an area. The PSO algorithm optimizes both distance traveled by each particle and sum of distances traveled by all particles. In this case the authors only consider the sum of distances traveled by all particles and tries to minimize it. One drawback of this method is the amount of time taken to compute a solution is quite high, thus it may not be suitable for situations that require fast response times. A geometric approach based path planning algorithm for multiple UAVs with optimal resource allocation has been proposed in [14] . The authors posits that though Dubin's path ("In geometry, the term Dubins path typically refers to the shortest curve that connects two points in the two-dimensional Euclidean plane (i.e. x-y plane) with a constraint on the curvature of the path and with prescribed initial and terminal tangents to the path" - [16]) is a well known planner for finding shortest path and being a composite curve of both lines and circles, easy to produce, it lacks a smooth curvature. This lack of smooth curvature results in variation in the acceleration rate of the UAV and thus makes its maneuver difficult. The authors thus come up with a smoother curve based on Pythagorean Hodographs technique. A common trend is visible in the discussed papers, i.e. need for the UAV's trajectory to have a smooth curvature in order to reduce abrupt accelerations or deceleration and to have a better control on the UAV's behavior. The second half of [14] deals with resource allocation for multiple UAVs. The authors combine Discrete Particle Swarm Optimization technique with Evolutionary Game Theory as a solution to the resource allocation problem. The discrete particle swarm optimization technique maximizes the exploration and global efficiency of the algorithm to find a solution, while the function of the evolutionary game theory algorithm is to ensure a faster convergence to solution.

A large section of research on coverage also deals with randomized approaches. In [4] the authors propose a randomized approach to multi UAV path planning based on the well known RRT (Rapidly-exploring random tree) algorithm. The authors state that due to incapability of the traditional RRT-connect - see figure 5 - algorithm to be applied to the problem of the UAV path planning, a variable probability based bidirectional RRT (VPB-RRT) - see figure 6 - algorithm is proposed. The authors point out that in RRT algorithm a low coverage occurs when the path of the expanding tree from the start node encounters obstacles. On the contrary a large coverage may not necessarily mean that the coverage algorithm will reach completion. In the proposed variable probability based bidirectional RRT algorithm, the authors take into account the areas which have not been covered by the tree and increase the probability of generating random nodes in that area. Again if the coverage is high, the authors vary the probability function to generate more nodes around the goal node to ensure convergence. A key consideration that the authors mention while generating nodes is that the previous and the follow up nodes satisfy a maneuverable trajectory for the UAVs.

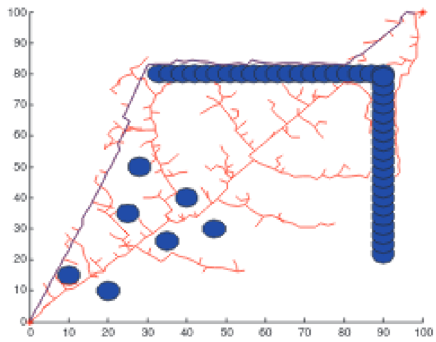


Figure 5: Performance of RRT-connect algorithm with an obstacle in coverage space as proposed in [4]

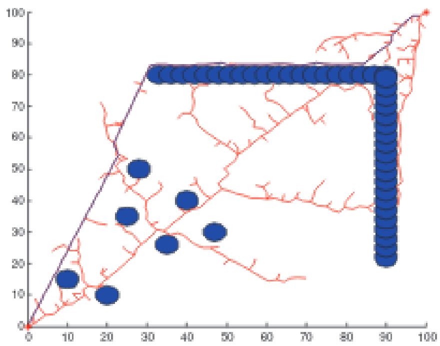


Figure 6: Performance of VPB-RRT algorithm with an obstacle in coverage space [4]

4.2.2 Coverage

In the previous sections we have looked into state of the art research in the fields of area coverage and multi-UAV planning. In this section we will delve on the work done in the field of area coverage using multiple UAVs. More precisely this section will focus on the various optimization which have been proposed by researchers to reduce the three critical considerations - as proposed in [1] - of any coverage problem namely (i) Percentage of Area actually covered by the UAV's sensor's field of detection (ii) Time required by the UAV to finish coverage of an area (iii) Computational cost of coverage algorithms. With more technological innovations in the field of computing, the third criteria may lose its relevance, however for this work we consider it an important metric for evaluating performance of a coverage algorithm.

The authors in [9] treats the problem of multiple UAV cooperative reconnaissance as a task allocation and resource scheduling problem. A key point to be noted in this paper is that it takes into consideration various constraints and performance planning target. Constraint include number of surveillance targets, reconnaissance time and performance requirements of the UAVs. The paper then goes on to propose a mathematical model which takes into consideration all these criteria. The mathematical model does two things - (i) Calculates the minimum number of UAVs required to cover the area (ii) Formulate constraints of this model. After generating solutions the authors use genetic algorithm to find out an optimal solution.

In [5] the authors consider a scenario of area coverage for a non-convex region. For addressing this problem the authors approach a two step approach - (i) Division of the area into smaller sub areas, this approach ensures that no two robot share the same space and thus does not need special obstacle avoidance capabilities. (ii) Straight lines lanes with lesser number of turns are designed to minimize acceleration and deceleration of the UAVs. The division of the area is done along the longest diagonal of the polygon. A technique called area sweeping is used as illustrated in figure 7.

In [15] the authors propose a model based approach for cooperative area reconnaissance using multiple UAVs. The authors use the Model Predictive Control (MPC) technique to for online computation of paths, however determining the optimal path is not possible using this method. The authors propose their own modification of the Particle Swarm Optimization (PSO) algorithm, Simulated Annealing Particle Swarm Optimization (SAPSO). The three key advantages of SAPSO as stated by the authors are simplicity, fast convergence and fewer parameters requiring adjustment. Another key feature proposed by this paper is a simple collision avoidance method. When an UAV encounters an obstacle it adjusts it height to avoid it. The advantage of this method is that it reduces computation

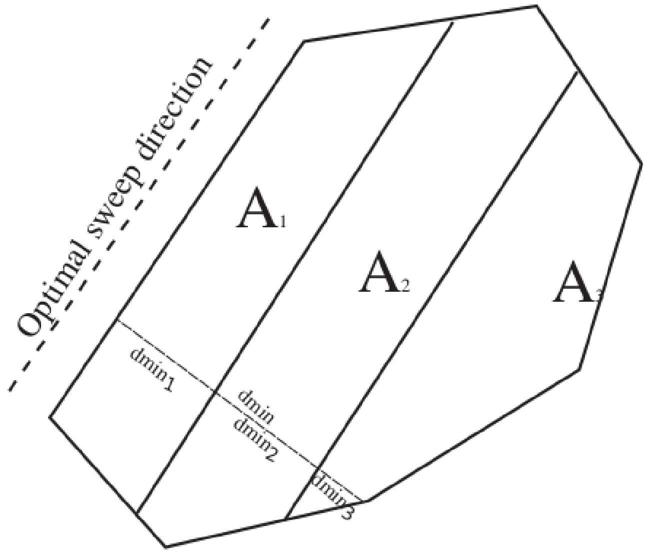


Figure 7: Area decomposition using sweeping technique proposed in [5]. Here the optimal sweep direction is perpendicular to the dotted line.

complexity while being highly efficient.

Using this algorithm 90 percent coverage was achieved and the computation time of the algorithm was found to be 1.3892 sec only.

4.3 Deployed Libraries and Existing Platforms

In this section we will discuss about some existing UAV simulation platforms and libraries using which we can implement coverage algorithms and study their performance.

As stated by[6], UAVs are being used for a multitude of functions both in research and real-world applications. However due to an UAV's complex control mechanism, a small error in an algorithm can cause the UAV to crash. Therefore it is imperative, a simulation environment, which not only models the UAV's dynamics but also models various sensors like SONAR, laser scanner, camera is required. Before [6], there has been few simulation environments for UAVs, however they were deficit in realistic flight dynamics and sensor models. Hector Quadrotor package was created to address this specific deficit. It is deployed as a standard ROS package and can be used like any other ROS packages. It includes its own Gazebo environments. There is only one way to command the UAVs i.e. by giving it velocity commands, however with a little change in package parameters it is also possible to give position commands. Flight dynamics were created using Matlab. Inertial Measurement Unit(IMU), Ultrasonic Sensor, GPS are among the various sensors that are included in the package. The control part of the package was created using the OROCOS [17] tool chain.

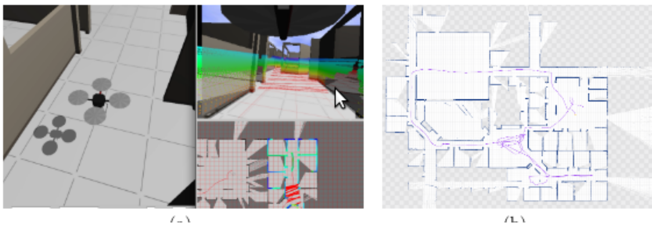


Figure 8: Hector Quadrotor package for simulation of UAVs in Gazebo as proposed in[6]

5 Approach

In this section we will briefly describe the approach taken to propose a solution model for the problem formulation. Then we go on to propose an experimental setup which will validate the proposed method.

5.1 Overview

A scientific approach to any problem, requires an extensive research on the state of the art that exists for this particular problem, which has already been done in the previous section. The first step is to determine on what factors does the performance of the back and forth or lawnmower method -suggested in [1, 5, 15, 14, 9]- depend on, and record those parameters:

- Time of coverage
- Trace of the UAV's path
- Velocity - both linear and angular

The next step is to determine algorithms which will minimize these parameters. From [3] we learned that using simple trajectories like hemispherical - see figure 3 - and cylindrical - see figure 4 - we can do coverage of any area. After implementing the algorithms which will be formulated in 5.2, the next step will be to record the above mentioned parameters and compare with the lawnmower approach parameters.

5.2 Formulating mathematical model of the algorithms

In [3] the authors propose hemispherical - see figure 3 - and cylindrical trajectories - see figure 4 - for 3D coverage, likewise for 2D coverage we can use smooth curves like spirals, splines and Lissajous Curves to generate trajectory path for the UAVs. In this section we will describe the mathematical models of the above mentioned curves.

5.2.1 Spiral

Spiral - According to the The American Heritage Dictionary of the English[1] a spiral is defined as a curve which winds around a fixed center point at continuously increasing or

[1]The American Heritage Dictionary of the English Language, Houghton Mifflin Company, Fourth Edition, 2009.

decreasing radius. For generating way-points we need to use the parametric equation of a spiral, given by[2]

$$x(t) = atcos(t) \tag{1}$$

$$y(t) = atsin(t) \tag{2}$$

In 1, 2 x and y are the coordinates of the waypoints, a is a constant value and t is the varying radius. The value of a will depend on the dimensions of the maximum value of x and y. The parameter t can be adjusted to increase or decrease the width between two successive curves of the spiral. This value will be adjusted according to the field of view radius of the UAV sensor.

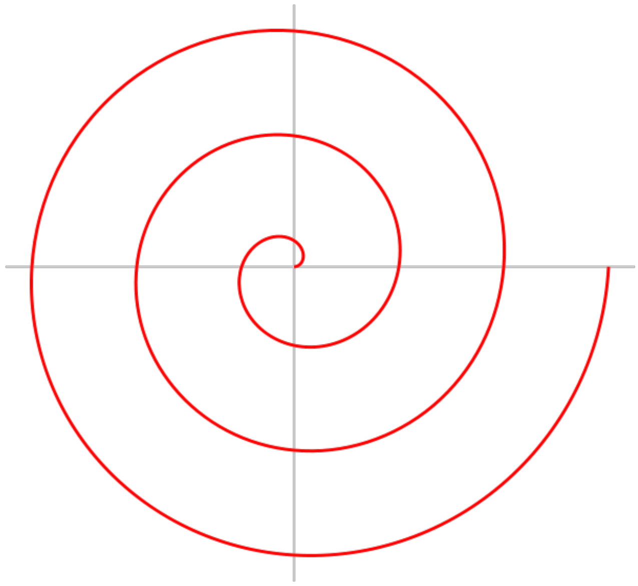


Figure 9: Archimedean Spiral [2]

5.2.2 Lissajous Curves

Lissajous Curves - From the work in [7] we get the parametric for of Lissajous curves.

$$x(t) = Acos(\omega_x t - \delta_x) \tag{3}$$

$$y(t) = Bcos(\omega_y t - \delta_y) \tag{4}$$

$$x(t) = Asin(\omega' * t + \delta) \tag{5}$$

[1]http://www.mathematische-basteleien.de/spiral.htm

$$y(t) = Bsin(\omega * t) \tag{6}$$

In equations 3, 4, 5, 6 x and y are the coordinates of the waypoints, $\omega_x t$ and $\omega_y t$ are the amplitude of the *cosine* curve, ω' and ω are the amplitudes of the *sine* curve and δ, δ_x ,δ_y are the necessary phase difference to create a Lissajous pattern and t is a variable. If the ratio of $\frac{\omega'}{\omega} = 5/4$, the following Lissajous pattern can be generated - see figure: 10 - which can act as a space filling curve, thus in our case act as an algorithm for coverage.

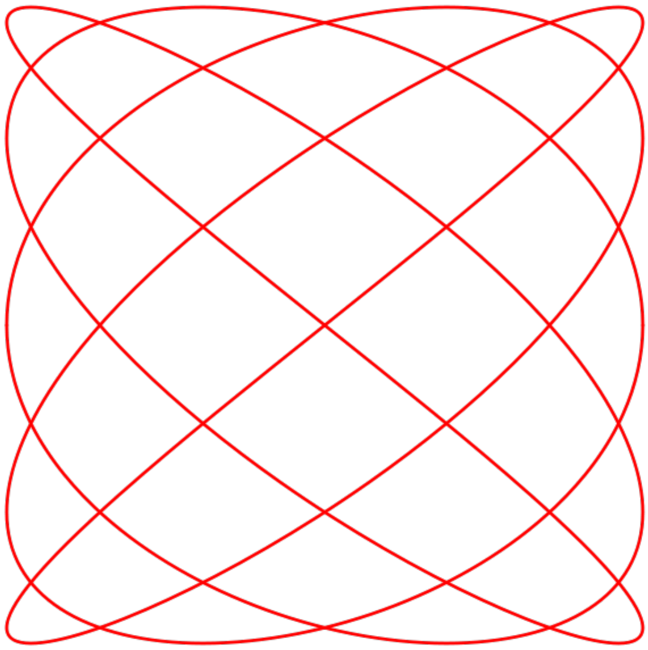


Figure 10: Lissajous Curve with $\frac{\omega'}{\omega} = 5 : 4$ [7]

5.2.3 Hybrid of Spiral and Lissajous Curves

To generate this Lissajous Curve 11, the ratio of $\frac{\omega'}{\omega}$ was set to $1 : 2$, while varying the value of t(in this case increasing) with each iteration we create the following coverage pattern. This pattern can be used in case of doing coverage while persistently tracking any target object. The center of the curve where the spirals intersect can be fixed on a target object and the UAV can cover an area while persistently tracking the target object.

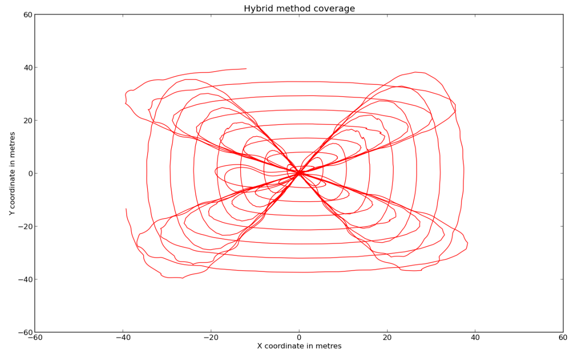


Figure 11: Hybrid approach based on Spiral and Lissajous Curves method [7]

5.2.4 Rapidly Exploring Random Trees

Along with the above mentioned geometrical approaches we also implement a randomized approach in order to have a better comparison of our proposed method. Unlike probabilistic approaches which require point to point convergence to progress [8], we require an approach which can randomly expand and fill a coverage space. Rapidly Exploring Random Trees are thus ideal for our problem[8]. The only change we make here is that we do not consider a shortest path between start and goal node, instead we try to generate a large number of nodes in the space to be covered and use these generated nodes as waypoints for the UAVs to follow. We try to order the waypoints generated in ascending order of Euclidean distance from the starting point.

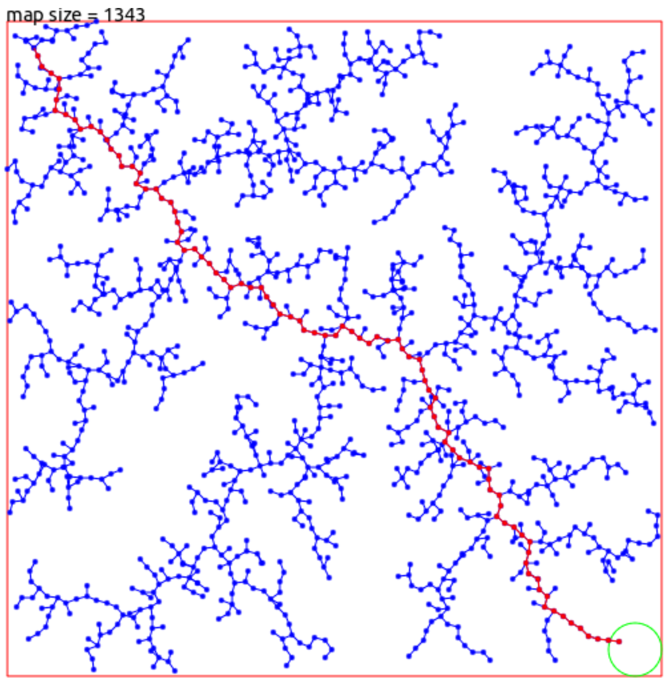


Figure 12: Rapidly Exploring Random Tree [8]

5.3 Performance Metric Function

An effective means of comparing methods, which are dependent on multiple parameters is to formulate a performance metric function with certain weights assigned to each parameter. The performance metric function will take into account percentage of area covered, percentage of overlap, percentage of area it exceeds from the commanded area, time of coverage and energy consumed during run (which depends on the rate of change of velocity during its motion). The highest weight-age will be assigned to percentage of area coverage, time required to completion, and energy cost. The parameters percentage of area overlap and percentage of area exceeded will be assigned lower weight-age value as these parameters have lower effect on the performance of the proposed algorithms.

5.4 Experimentation

For carrying out our experiment we will adopt the rolling landscape scenario from Gazebo simulation environment. We will consider a coverage space of certain size and vary the following parameters for each of the aforementioned algorithms -

- Number of waypoints
- Size and shape of the area to be covered

After implementing the above mentioned algorithms using ROS and Hector Quadrotor package, we will record the ground truth information of the UAVs.

5.5 Evaluation

From the experiments that will be carried out in section: 7.3 we will record the following values

- Time
- X,Y,X poses
- Linear X velocity, Linear Y velocity, Linear Z velocity
- Angular X velocity, Angular Y velocity, Angular Z velocity

Next we will evaluate the implemented methods based on these parameters and determine which method is best suited for coverage in a certain space. The evaluation criteria will be explained in details in the next section.

6 Use Cases

This section gives a description about a scenario in Gazebo simulation environment which is tasked to be covered using two UAVs. The shape of the area to be covered is assumed to be square.

6.1 Scenario:Gazebo Rolling Landscape

The Gazebo rolling landscape simulates a world with elevations and plane ground. The two UAVs start on a plane ground. The image of the scenario is given below.

6.1.1 Task Description

The task of this scenario is to use both the UAVs to cover the entire area using their laser scanners. The UAV also must avoid any obstacles it might encounter in the course of its flight including the other UAV. Besides area coverage the other goal is to do it in as shortest time as possible.

6.1.2 Task Analysis

The action- covering the rolling landscape scenario would require the system to have following capabilities:

- **Reducing the search space**
  Generating geometry based trajectory plans without using area decomposition algorithm would reduce the computation cost.
- **Adopting different strategies**
  The trajectory generation algorithm needs to have different geometrical strategies and should know when which strategy has to be adopted. For example depending on the area to be covered, spiral trajectories may work best while in some other case Lissajous curves based trajectories might be a better suit.
- **Dealing with obstacles and other UAVs**
  Since this work is ultimately aimed to be implemented in real system which will function in real world, having an obstacle avoidance system is extremely crucial. Moreover the system being implemented here is based upon multiple UAVs, thus

each UAV must know the position of other UAVs during flight and maintain a safe distance from them. This can be achieved using GPS and uploading the UAVs knowledge base in real time. One other advantage of using multiple UAVs is that each UAV can also act as a beacon for localization and thus give better results in situations where the GPS module may malfunction [15, 14, 18] .

7 Experimentation and Results

In the previous section we described some methods based on the method suggested by [3] which can be used instead of the normally used back and forth motion as used by [5, 15, 12, 9]. However merely implementing an algorithm doesn't validate whether it is better than existing algorithms. A proper evaluation of any algorithm requires certain metrics against which its performance can be measured. In this section we formulate certain criteria for evaluating the proposed algorithms and further present results of carried out experiments. In the last sub section we will evaluate the results of the experiments against the determined criteria to determine which algorithm is better.

7.1 Formulating Criteria for Evaluation of Algorithms

In [1], the author defines three criteria for evaluating a coverage algorithm,

- percentage of area covered
- time taken by the UAVs to finish coverage of a given area
- computational and time complexity of the algorithm.

Along with these we add three more criteria:

- energy cost during its motion - which depends on the rate of change of velocity
- percentage of area overlapped by sensor's footprint.
- percentage of area exceeded from the commanded area

7.1.1 Percentage of Area covered

In [1, 3] percentage of area covered is defined as the actual area covered by the footprint of the robot's sensor to the total goal area that was to be covered. For calculating this, we record the pose of the UAV during its run. Next we get the value of field of view of the robot's sensor from the code. The field of view radius is calculated using the height of the robot and the aperture angle of the sensor's field of view. In the figure: 13 we illustrate an example of a robot's traced path and the corresponding trace path plus field of view radius for the UAV.

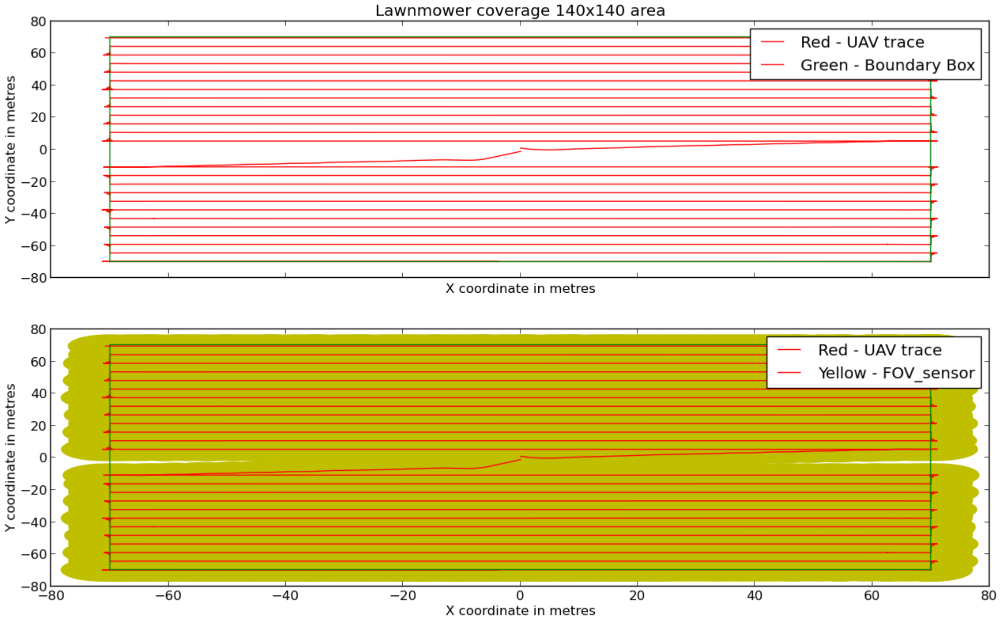


Figure 13: Two UAVs doing coverage of an area of 140x140 m^2 using lawnmower approach

7.1.2 Time taken by the UAVs to complete coverage of a given area

We see that using either traditional search algorithms like A* [4] and randomized approach like RRT algorithm [19],time complexity of the algorithms are low. However these do not take into consideration the physical constraints of an UAV. For example lets take the case of RRT algorithm, since the approach is based on random generation of points, two consecutive points may be farther from each other while the point after that may be near the initial point. This will cause the UAV to move randomly and will end up having a high execution time.

7.1.3 Computational and time complexity of the algorithm

From the equations of splines, Lissajous curves, hybrid curves, described in the approach section we see that these methods can be implemented using a simple for loop which has a time complexity of $O(n)$ [20], where n is the maximum number of points to be generated.

7.1.4 Energy costs of UAV's motion

Since we are doing 2D coverage, the energy cost of the UAV flying from ground to the required height is same for all the methods mentioned above thus for calculating the energy costs we will be considering only Kinetic Energy.
From Newton's 2nd law of motion we know that

$$\vec{F} = m.\vec{a} \tag{7}$$

$$\vec{F} = m.\frac{\mathrm{d}\vec{v}}{\mathrm{d}t} \tag{8}$$

We also know that work done W is given by

$$W = \Delta KE....eq(7) \tag{9}$$

where KE is Kinetic Energy. Therefore to minimize the change in Kinetic Energy it is sufficient to show that the work done is minimum. Now,

$$W = \vec{F}.\vec{x} \tag{10}$$

replacing from earlier equation of force

$$W = m.\frac{\mathrm{d}\vec{v}}{\mathrm{d}t}.\vec{x} \tag{11}$$

considering m (mass) and $\vec{x}$ to be constant for a UAV and for a fixed area covered, the only variable that can affect work done is the $\frac{\mathrm{d}\vec{v}}{\mathrm{d}t}$ factor.

7.1.5 Percentage of area overlap by sensor's footprint

A certain percentage of area overlap might be a desired criteria in order to verify data gathered by the UAV's sensor, however unless it is specifically requested by the user, it is undesirable as it causes a repetition of data by making an UAV fly over the same region already covered by another UAV. Thus this causes an increase in total energy cost.

7.1.6 Percentage of area exceeded as regards to the commanded area

During the course of its flight, if an UAV goes out of the commanded area then it is considered as a negative trait. Though if the area exceeded is small, its effect on the total energy cost wouldn't be significant, however if the exceeded area is large then it can seriously increase the energy cost.

7.2 Formulating the Performance Metric Function

Based on the parameters established in 7.1, we will define the performance metric function.

7.2.1 Assigning weights to the parameters

Based on the importance of each parameter we assign the following weights:

- Percentage of area coverage ($w1$) - This is a very important criteria for measuring performance of an area coverage problem, hence we assign it a value of $w1 = 2$
- Time taken by the UAV to complete coverage of a given area ($w2$) - This is also a very important criteria for measuring performance of an area coverage problem. We assign it a value of $w2 = 2$ as it is desired to have low time of completion.
- Energy cost of UAV during coverage ($w3$) - Considering the fact that UAVs have limited battery supply, keeping this parameter low is of utmost importance, hence we also assign it a value of $w3 = 2$.
- Percentage of area overlap ($w4$) - In some cases overlap might be a desired criteria while in some cases not. Since this criteria does not play a big role in affecting area coverage we assign it a value of $w4 = 0.1$ considering the case we want to keep it as low as possible.
- Percentage of area exceeding commanded area of coverage ($w5$) - Exceeding area of coverage is an undesirable factor and we want to keep it low. However considering the overall performance, this factor is not as important as either of percentage of area coverage, time to completion or energy cost, hence we assign a value of $w5 = 0.5$

7.2.2 Formulating the final performance metric function

We have seen in that for the lawnmower approach the percentage of area coverage is 100%, percentage of area overlap is 0% and percentage of area exceeded is also 0%, so we take the values of the lawnmower approach as the standard for evaluating the other approaches.

- For percentage of area coverage (denoted by P), 100% coverage is assigned a value of 1. 90% coverage is assigned a value of 0.9 and likewise.

- We normalize time taken (denoted by t) to complete coverage for all approaches using the time for lawnmower approach.
- Similarly we normalize the energy cost (denoted by EC) of completing coverage for all approaches using the energy cost for lawnmower approach.
- For percentage overlap (denoted by O) we consider no overlap to have a value of 0 and 100% overlap to have a value of 1.
- Similarly for percentage of area exceeded (denoted by E) we consider no area exceeded to have a value of 0 and 100% exceeded area to have a value of 1

The final function is defined as

$$f_p(method) = w1*(P_{method})+w2*(1-\frac{t_{method}}{t_{lawnmower}})+w3*(1-\frac{EC_{method}}{EC_{lawnmower}})+w4*(1-O_{method})+w5*(1-E \tag{12}$$

In equation 12, the variable P_{method} is the percentage of area covered for a particular method. The variable $t_{l}awnmower$ is the time taken by the lawnmower approach to complete coverage, $t_{m}ethod$ is the time for taken by the method (lawnmower, spiral, Lissajous Curves, hybrid approach, RRT approach) for which we are calculating the performance metric. $EC_{l}awnmower$ is the energy cost taken by the lawnmower approach to complete coverage, $EC_{m}ethod$ is the energy cost for taken by the method for which we are calculating the performance metric. O_{method} is the percentage of area overlap for a particular method and E_{method} is percentage of area exceeded while doing coverage for a particular method.

The value of the performance metric function is directly related to the performance of method, hence a larger the value better is the performance of the method. Since time, energy cost, percentage of overlap and percentage of excess affect the performance of a method negatively, we subtract the normalized value from 1 and then add it.

7.3 Conducting Experiments

We will conduct our experiments in a simulated environment. The simulation platform is Gazebo while the code is written using Robot Operating System(ROS) and C++. Using the ROS package Hector Quadrotor we spawn two UAVs inside the rolling landscape world of Gazebo. Then we run our coverage quadrotor nodes (one node for each UAV) and record run time, poses, velocity information from the ground state topic. For each algorithm - Lawnmower, Spiral, Lissajous curves, Hybrid approach, RRT - we vary the the area to be covered. The results from the experiments are displayed in 7.3.2 Description for each graph and what they represent are mentioned in the corresponding graphs. We present the result of coverage done by the UAV using the following method

- Lawnmower Method
- Spiral Method
- Lissajous Curves Method
- Hybrid Approach Method
- RRT Approach

7.3.1 Image of the Simulation Environment and UAVs

In this section we present recorded images of the experiment. The Gazebo environment displayed is the rolling landscape scenario.

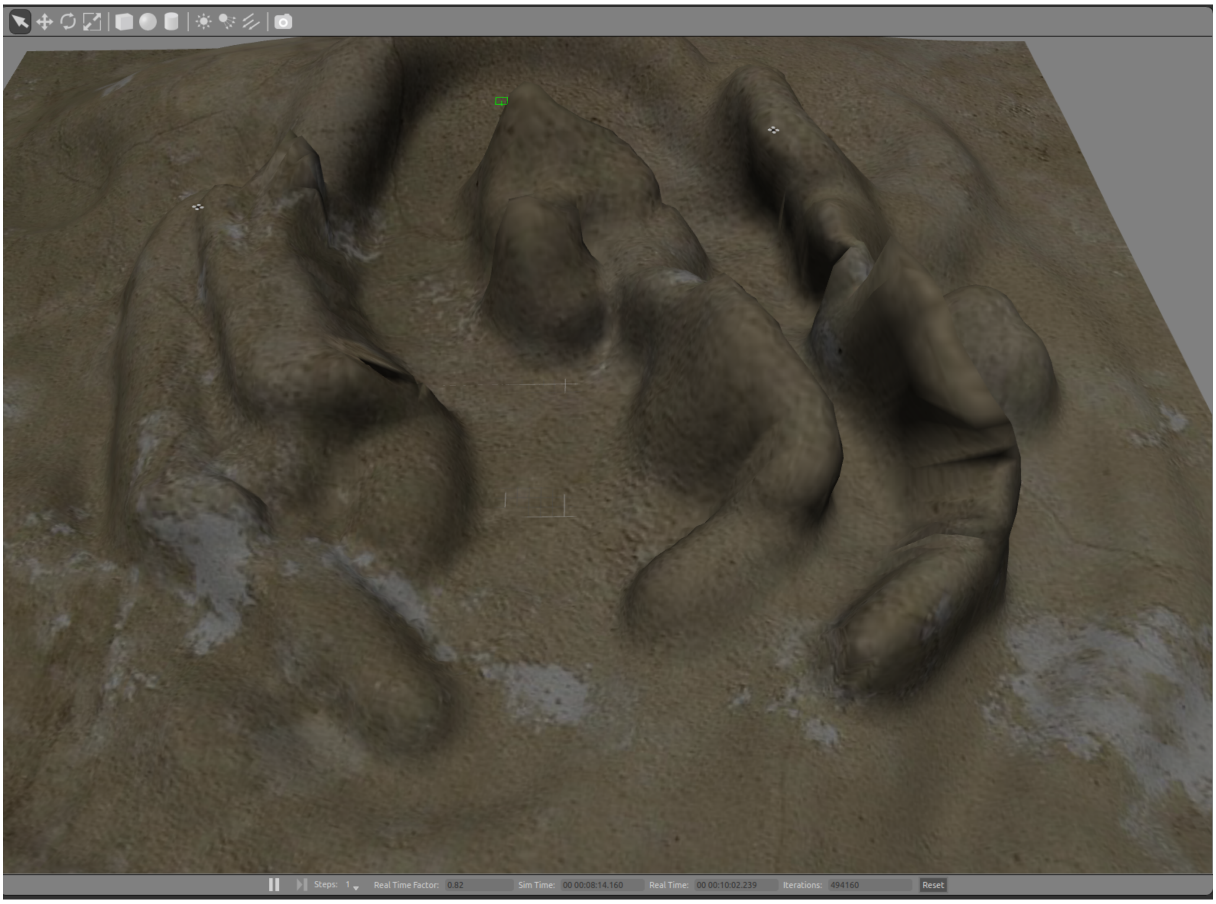


Figure 14: Two UAVs doing coverage of an area of 80x80 m^2 in Gazebo rolling landscape scenario

7.3.2 Images of UAVs' trace and sensor footprint with varying size of target area

In this section we present images of the UAVs' trace and their respective sensor footprint. The area to be covered is a 80x80 m^2 square shaped area. We vary the number of waypoints while keeping the area fixed. For the following results, the number of waypoints are 300. The green box represents the area to be covered. The red line represents each UAV's trace and the yellow region represents the area covered by the UAV's sensor's footprint.

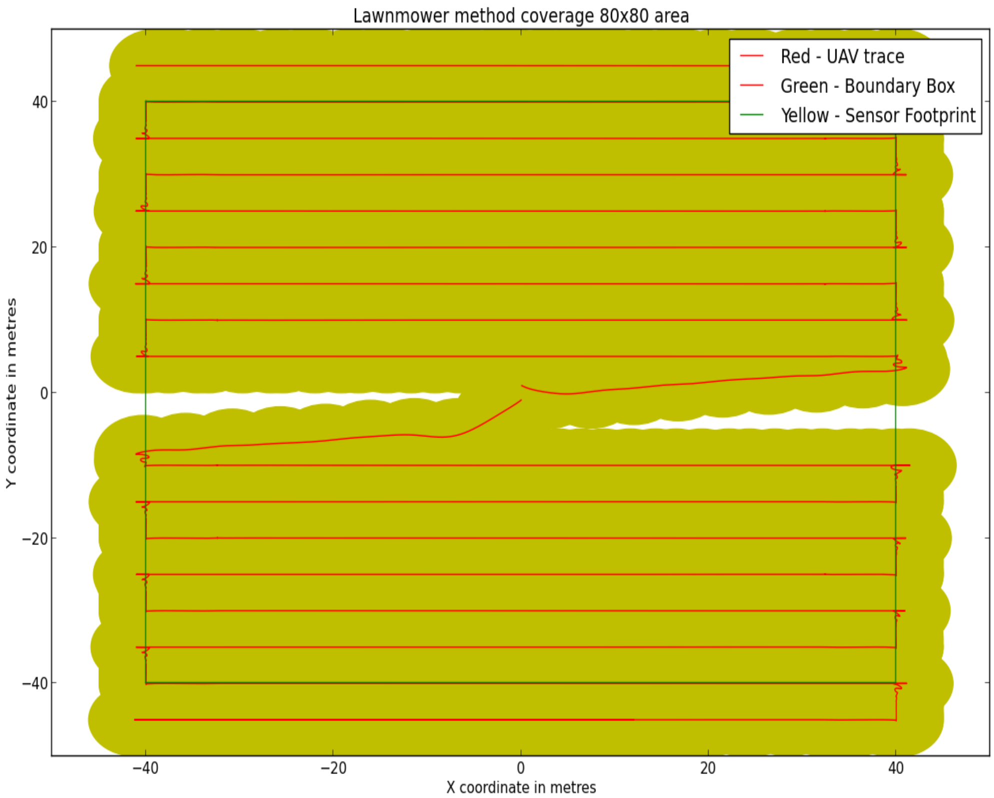


Figure 15: UAV traces and sensor footprint - Lawnmower Method

From the above figures we can see that lawnmower (see figure: 15) and Lissajous curve (see figure: 17) methods ensure almost hundred percent coverage. For the spiral (see figure: 16) and the hybrid methods (see figure: 18) the coverage is also more then ninety-five percent, while the coverage for the randomized approach (see figure: 19) is quite poor and does not even cover fifty percent of the target area. Details about percentage of area coverage for each method is analyzed in the subsequent section.

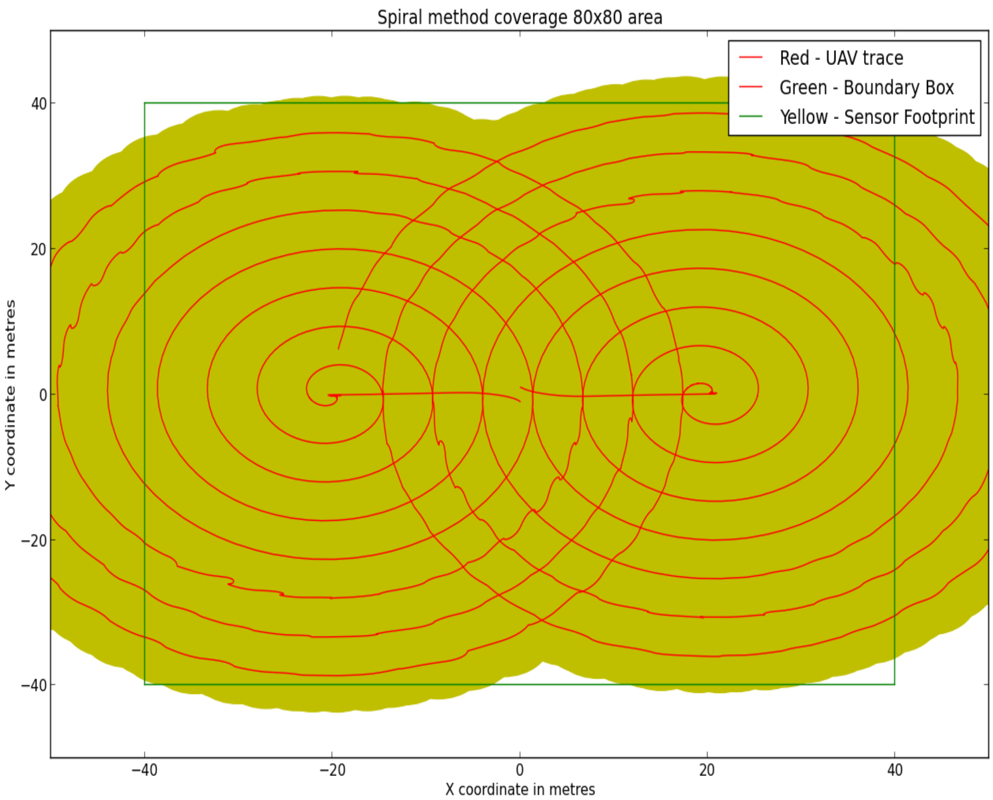


Figure 16: UAV traces and sensor footprint - Spiral Method

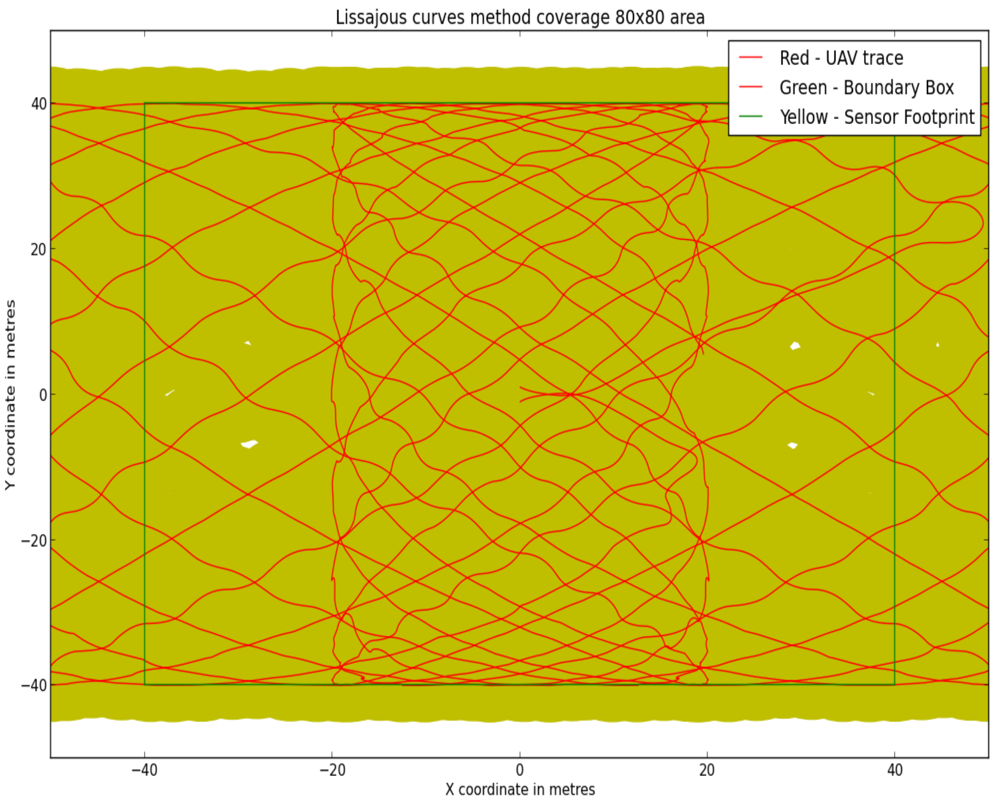


Figure 17: Two UAVs doing coverage of an area in Gazebo - Lissajous Curves Method

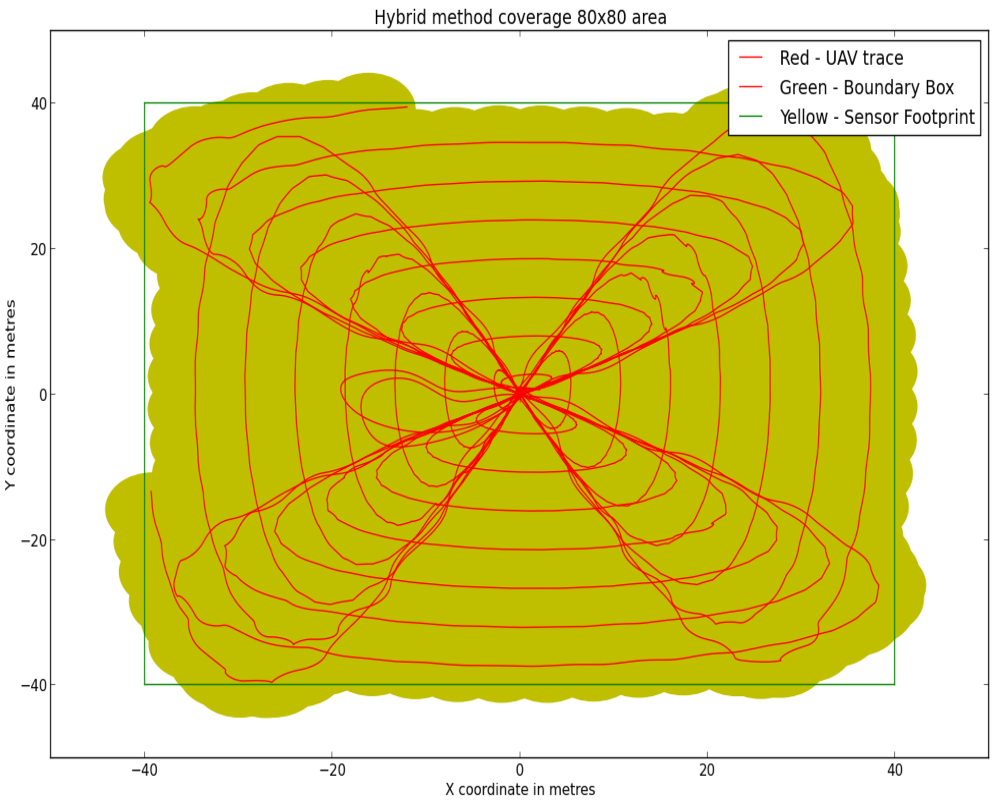


Figure 18: Two UAVs doing coverage of an area in Gazebo - Hybrid Approach Method

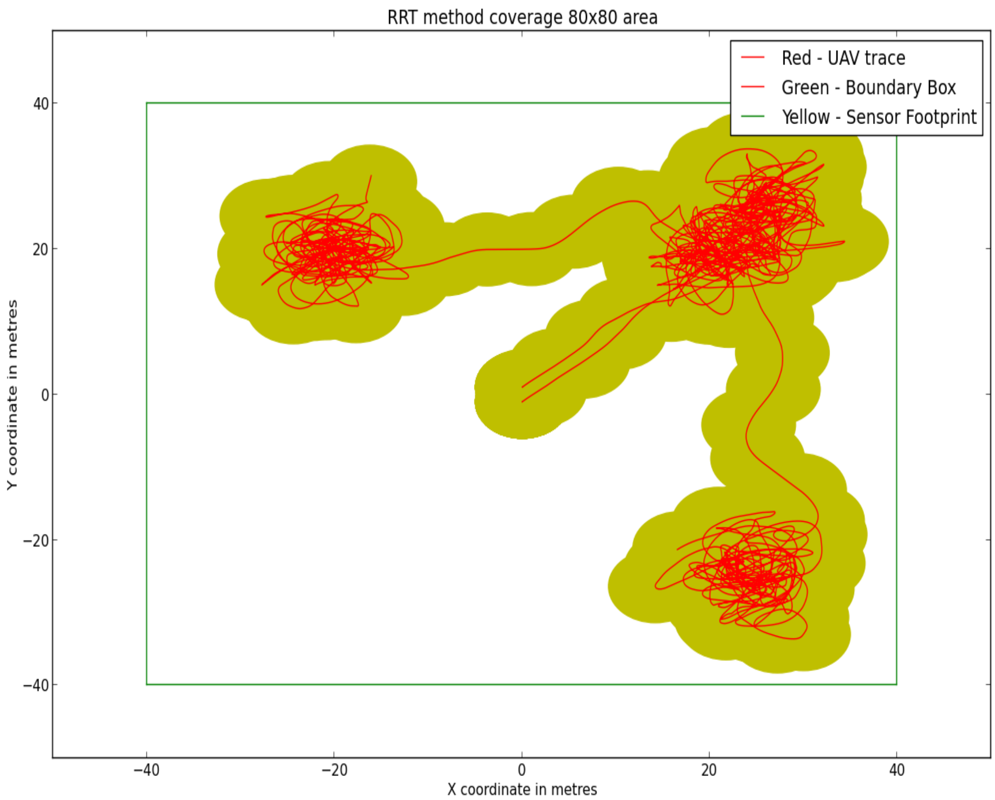


Figure 19: Two UAVs doing coverage of an area in Gazebo - RRT Method

For the following images, the area is a square bounded region if 140x140 m^2.

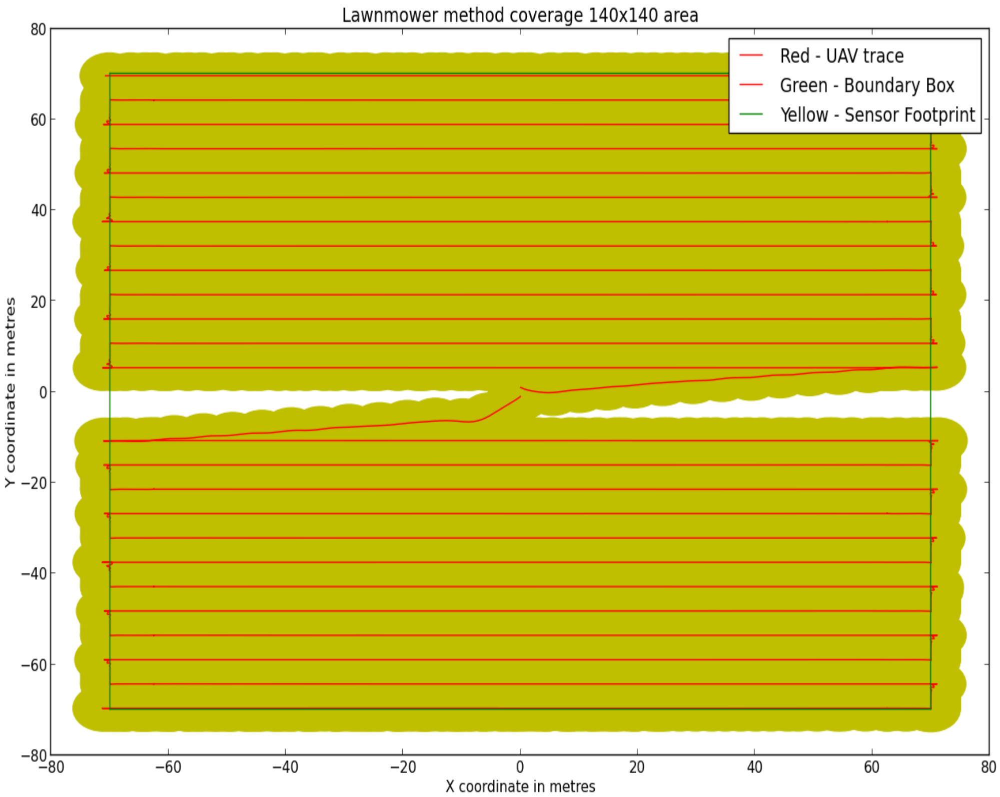


Figure 20: UAV traces and sensor footprint - Lawnmower Method

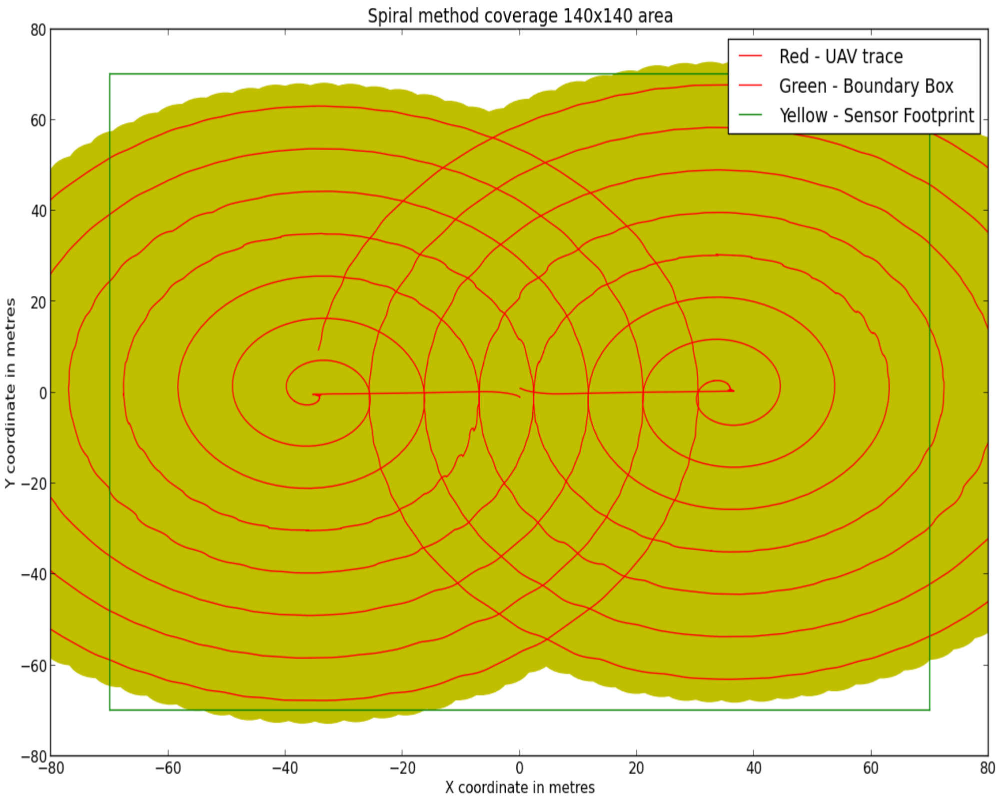


Figure 21: UAV traces and sensor footprint - Spiral Method

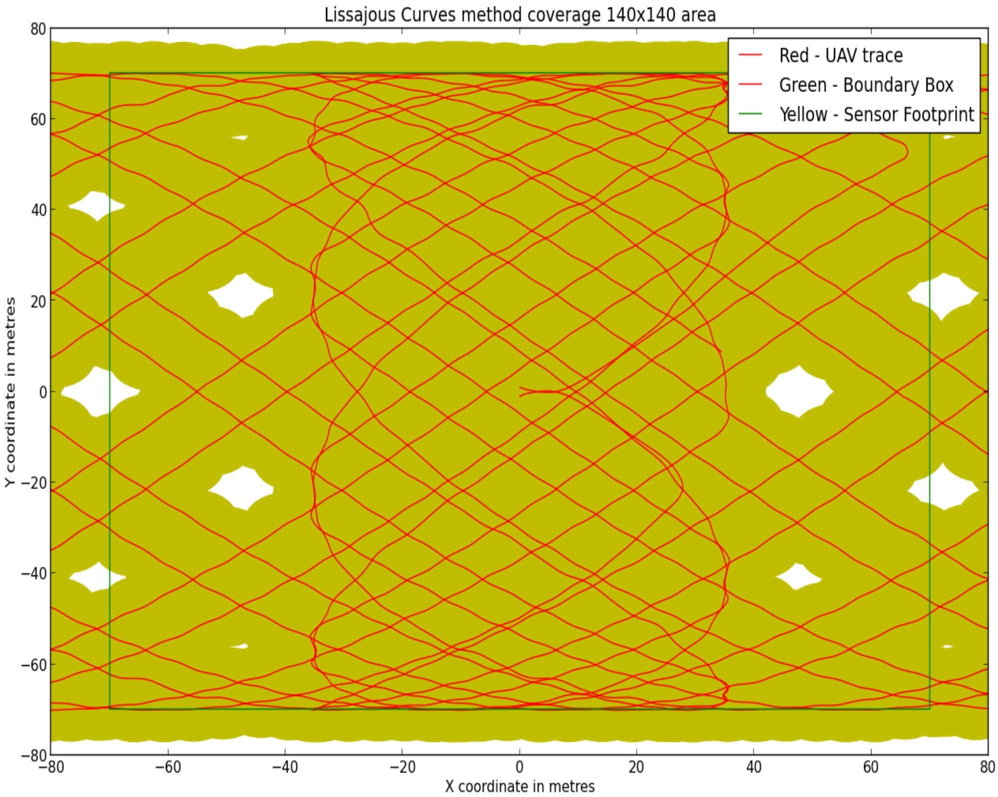


Figure 22: Two UAVs doing coverage of an area in Gazebo - Lissajous Curves Method

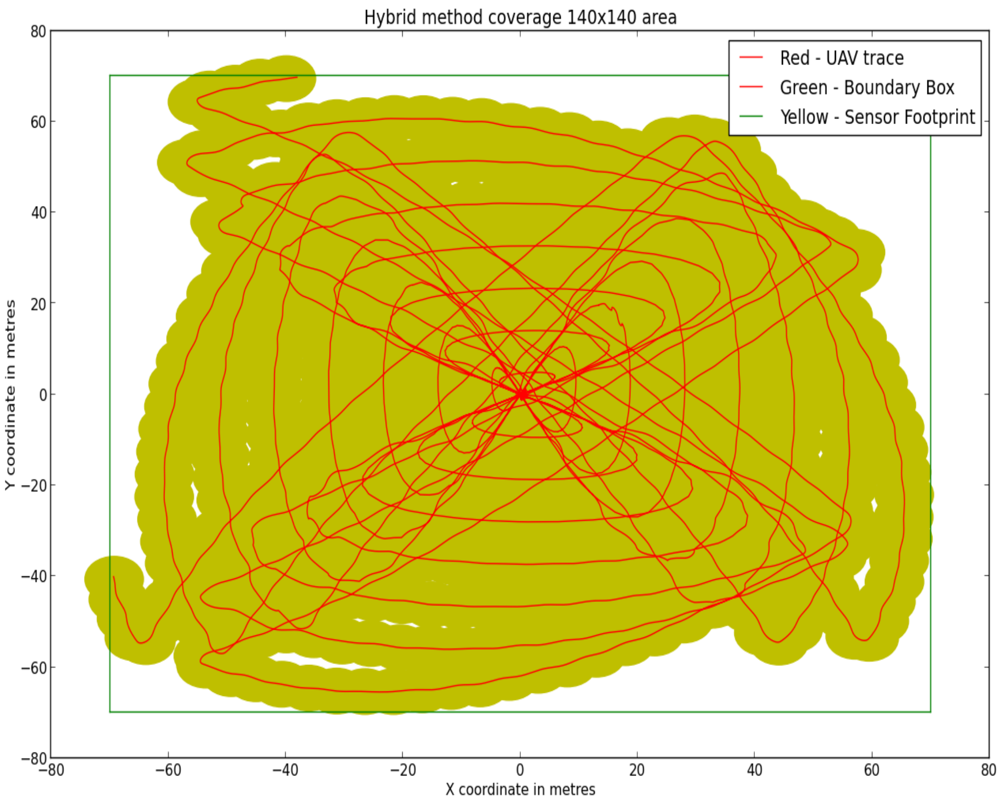


Figure 23: Two UAVs doing coverage of an area in Gazebo - Hybrid Approach Method

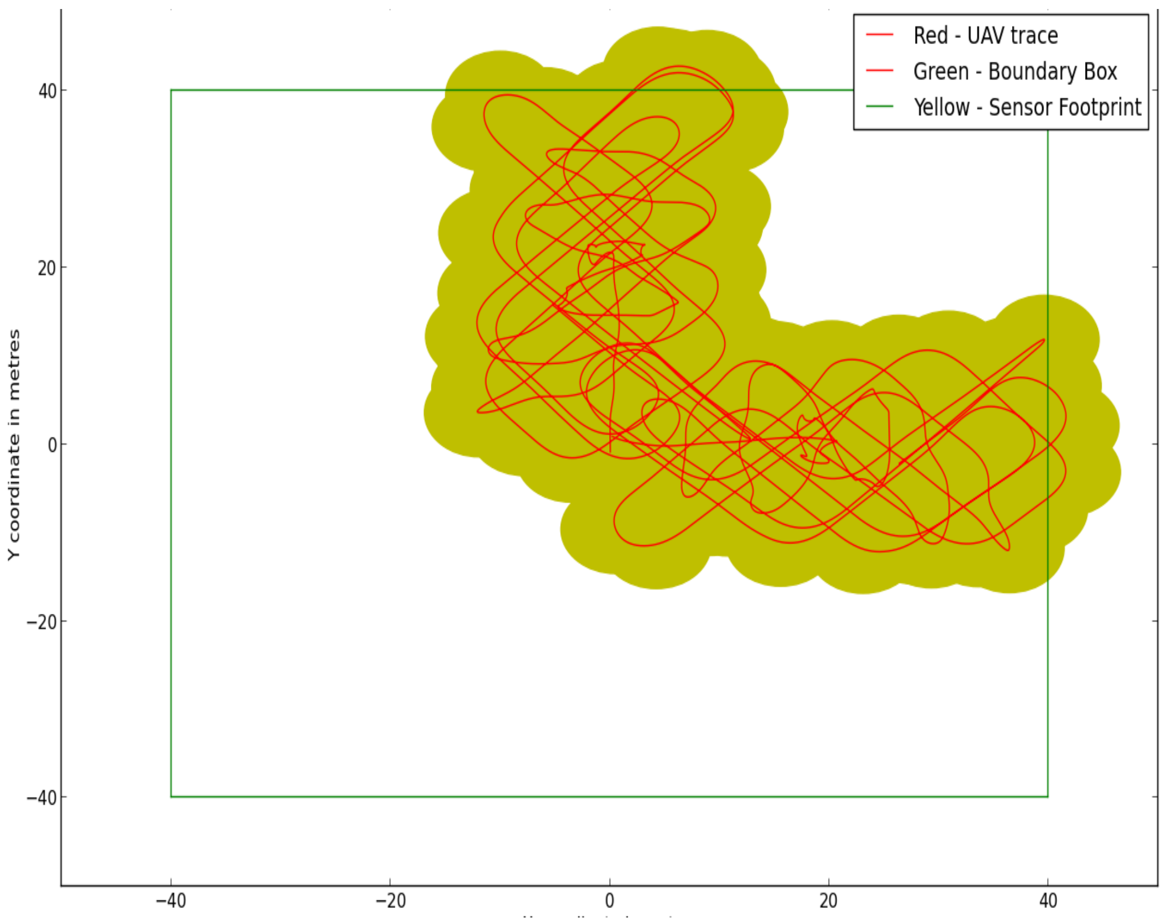


Figure 24: Two UAVs doing coverage of an area in Gazebo - RRT Method

7.4 Results of the Experiments

In this section we will analyze the results of the experiment which were presented in the previous section.

7.4.1 Percentage of area coverage and parameters which affect it

From the images presented in section: 7.3.2 , we saw that there is a relationship between the commanded area to be covered and the percentage of area actually covered by the UAV's sensor. We present a table which lists percentage of area coverage to the size of the total area to be covered.

Table 1: Area vs Percentage Area Coverage

Method	80x80	100x100	160x120	140x140	140x340
Lawnmower	100%	100%	100%	100%	100%
Spiral	93%	96%	97%	98%	99%
Lissajous Curves	100%	99%	96%	94%	92%
Hybrid	99%	99%	99%	99%	99%
RRT	40%	37%	31%	28%	25%

From the results in table: 1 we see that the proposed algorithms - Spiral, Hybrid achieves almost complete coverage and their performance remains mostly unchanged with the size of the area to be covered, while the performance of Lissajous curves method degrades with larger areas. The main reason for the degradation of the performance of the Lissajous curves method is due to the non-uniform gap between the generated paths (see figure: 25).

Another conclusion that can be drawn from the above data is that,the performance of the spiral method improves as the size of the area to be covered increases, even though the improvement is quite nominal.

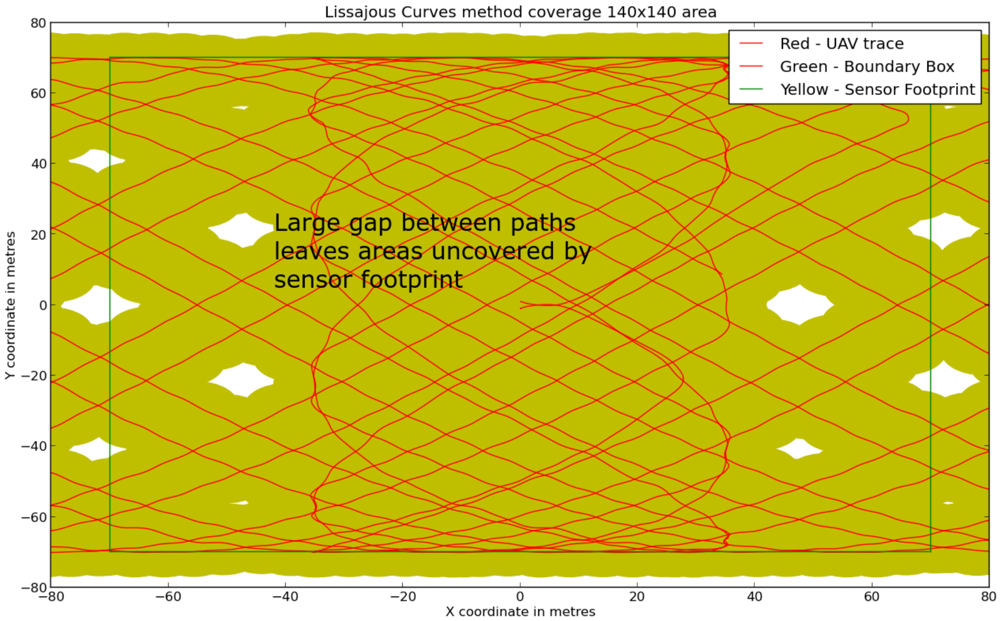


Figure 25: Incomplete coverage of the Lissajous curves method due to non-uniform gap between generated paths for area of 140x140 m^2

7.4.2 Completion time of the proposed methods

In this section we will analyze the time performance of the algorithms compared to the size of the area to be covered.

Table 2: Table of Area of coverage in m^2 to Time taken in *seconds* for all methods

Method	80x80 m^2	100x100 m^2	160x120 m^2	140x140 m^2	140x340 m^2
Lawnmower	437 secs	440 secs	864 secs	918 secs	1703 secs
Spiral	420 secs	428 secs	674 secs	706 secs	1043 secs
Lissajous Curves	420 secs	426 secs	672 secs	705 secs	1042 secs
Hybrid	421 secs	427 secs	675 secs	706 secs	1041 secs
RRT	372 secs	368 secs	568 secs	602 secs	997 secs

From the data in table 2 it is quite evident that as the size of the area to be covered increases, the time to required to cover it using either of Spiral, Lissajous curves, or Hybrid method is significantly lower than that of Lawnmower method. Even though the proposed method may cover slightly lesser area compared to the lawnmower method, the significant reduction in time makes these methods more efficient compared to the Lawnmower approach.

7.4.3 Comment on the smoothness of the generated trajectories

In this section we will compare the rate of change of velocities (acceleration/deceleration) of the proposed methods to lawnmower method. During the course of motion of any UAV too much changes in acceleration or deceleration gives rise to jerk, which can make the UAV unstable. Hence it is always desired to have a smoother trajectory. From the following data we will try to analyze which method among Lawnmower, Spiral, Lissajous Curves, Hybrid, RRT produces the least changes in its rate of change in velocity and hence generate a smooth trajectory with less jerk.

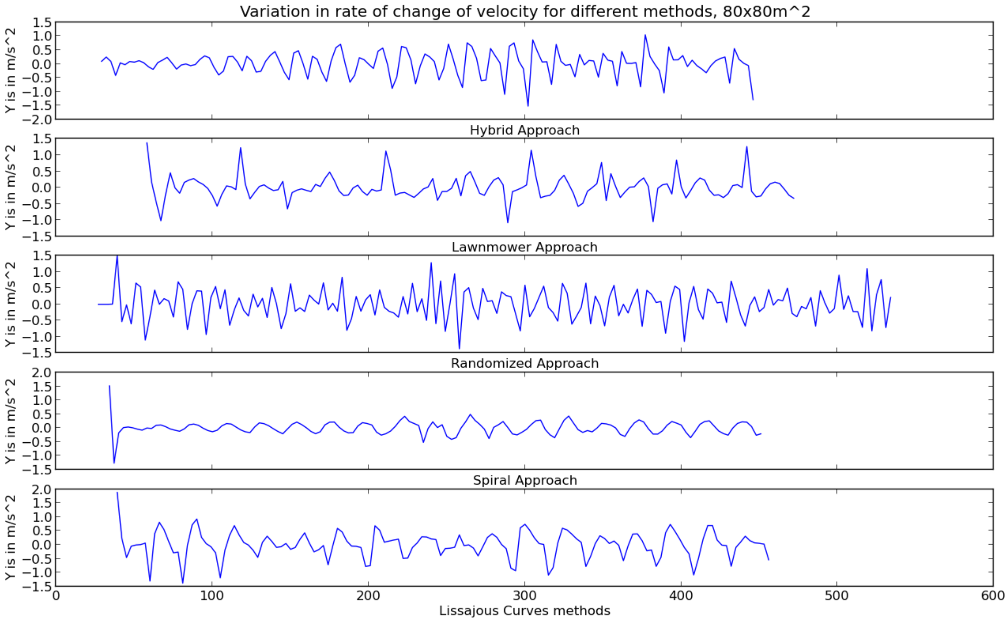


Figure 26: Rate of change of velocity vs time for an area of 80x80 m^2

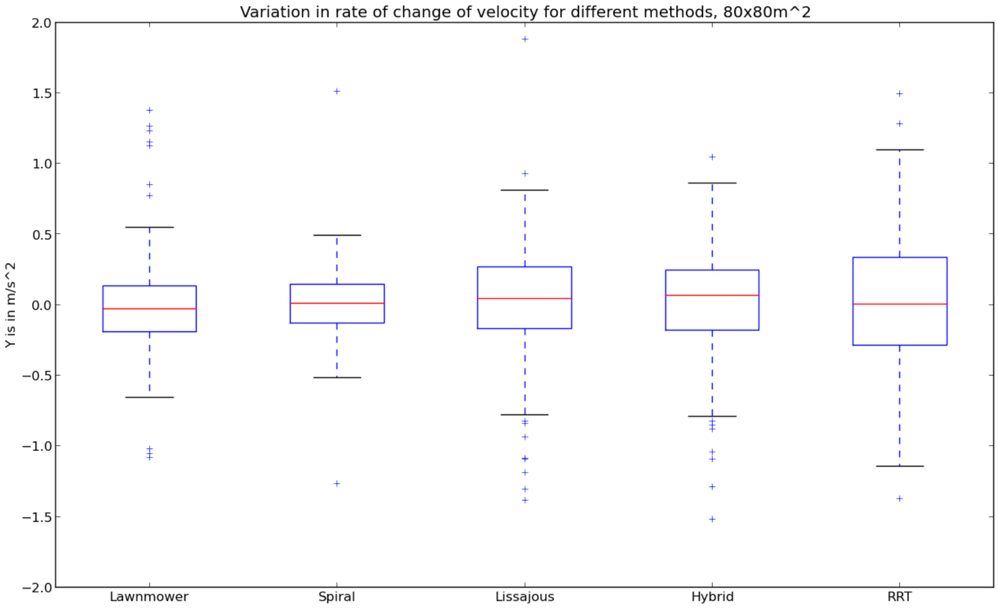


Figure 27: Box Plot representing velocity vs time for an area of 80x80 m^2

From the figure: 26 we can see that the lowest changes in rate of change of velocity can be observed in the spiral approach while the lawnmower and the hybrid approach shows quite large number of spikes and uneven changes. It is also to be noted that the range of values for these two methods are also higher compared to the Spiral method. It is also quite obvious that the randomized approach produces the most uneven changes and highest range of values.

The figure: 27 displays that for the spiral method not only does most of its value lie near the median, its overall range of values is also lower than the other method and it has the least number of outliers, only 2. The range of values for the lawnmower approach is also quite low compared to Lissajous Curves method and Hybrid approach.

7.4.4 Energy Consumed calculation for each method

In this section we will calculate the energy consumed for areas of **(i)**80x80 m^2 , **(iii)** 160x120 m^2 and **(iii)** 140x140 m^2 We assume the weight of an UAV to be $m = 10$ kg.

Using the formula in section: 7.1.4, we calculate the energy costs. The results are presented in table 3

Table 3: Table for energy cost for each method.

Method	80x80 m^2	160x120 m^2	140x140 m^2
Lawnmower	291.01 kJ	1155.71 kJ	1280.52 kJ
Spiral	227.65 kJ	577.12 kJ	639.91 kJ
Lissajous Curves	917.65 kJ	1464.11 kJ	1869.93 kJ
Hybrid	603.93 kJ	951.43 kJ	1122.45 kJ
RRT	158.11 kJ	363.55 kJ	423.55 kJ

From the data in table 3 we see that Spiral method has the least energy cost and as the size of the area to be covered increases, its energy consumption compared to other methods become even lower. Though the least energy consumption is by the RRT method, it does not do coverage of the area completely and thus we are not considering it for our final evaluation.

7.4.5 Percentage of area overlap and percentage of area coverage exceeded

In this section we will present two tables detailing the percentage of area coverage and the percentage of area exceeded.

Table 4: Percentage of area overlap

Method	80x80 m^2	160x120 m^2	140x140 m^2
Lawnmower	0 %	0 %	0 %
Spiral	24 %	20 %	19 %
Lissajous Curves	31 %	32%	34%
Hybrid	25 %	25 %	25 %
RRT	44 %	43 %	42 %

Table 5: Percentage of area exceeded

Method	80x80 m^2	160x120 m^2	140x140 m^2
Lawnmower	0 %	0 %	0 %
Spiral	9 %	8 %	8 %
Lissajous Curves	21 %	22%	24%
Hybrid	5 %	6 %	6 %
RRT	0 %	0 %	0 %

7.5 Calculating the Performance Metric

In section: 7.4 the results of all the parameters were presented. Now using those parameters we will calculate the values of the performance metric function for each method for (i)80x80 m^2,(ii)160x120 m^2 and (iii) 140x140 m^2.

Table 6: Performance Metric of methods based on area

Method	80x80 m^2	160x120 m^2	140x140 m^2
Lawnmower	2.6	2.6	2.6
Spiral	2.9	3.94	3.98
Lissajous Curves	0.26	2.24	1.83
Hybrid	0.45	3.23	3.32
RRT	0.01	0.78	1.45

From table 6, it is quite evident that the spiral method performs best compared to all other methods and its performance increases as the size of the area to be covered increases. The next best method is the hybrid method, whose value also increases as the size of the area to be covered increases, but it is also to be noted that for smaller areas, its performance is very poor. The RRT method fails to cover area completely, hence its performance is worst compared to all other methods.

8 Conclusion

8.1 Contributions

In this work we proposed a geometry based approach to area coverage problem using UAVs. We proposed that since UAVs have limited computational capabilities, using algorithms which require very little computational power can be helpful in creating better solutions. Based on the experiments carried out we came to the conclusion that for doing coverage of small areas lawnmower approach is more suited as it ensures complete coverage in minimal time, however from table 6, we can clearly conclude that spiral method is the best method overall as it ensures atleast 98 % coverage while having minimum energy cost and time to completion. Along with the spiral method, the hybrid method also shows some promise, but its performance in small areas is really poor. The RRT method does not guarantee complete coverage.
Regarding the fact that a smooth motion of the UAVs during its course of flight is a requirement not only for capturing information uniformly but also ensuring a high level of flight stability, the results published in 7.4.3 clearly show that the spiral method gives the best performance as it ensures its acceleration and deceleration are within a certain range thus ensuring greater flight stability

8.2 Possible Improvements and Future Directions

A very important future work is to implement the proposed algorithms on a real system and determine whether their performance is in accordance with the findings of this work. In terms of improving the performance of the algorithm, introducing machine learning to make the UAVs autonomously decide which algorithm to use for which scenario will be a big improvement. The primary requirement for creating such a system would be to run multiple number of test to gather enough data.

Apart from the above mentioned directions, changing the orientation and position of the sensor on the UAV and studying how it affect coverage performance is also a possible extension of this work.

References

[1] Howie Choset. Coverage For Robotics - A Survey Of Recent Results. *Annals of Mathematics and Artificial Intelligence*, 31:113 – 126, 2001.

[2] Yi Guo and Mohanakrishnan Balakrishnan. Complete Coverage Control for Nonholonomic Mobile Robots in Dynamic Environments. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 1704–1709, Orlando, USA, 2006. IEEE.

[3] Peng Cheng, James Keller, and Vijay Kumar. Time-optimal UAV Trajectory Planning for 3D Urban Structure Coverage. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008. IROS 2008.*, pages 2750–2757, Nice, France, 2008. IEEE.

[4] Wu Xinggang, Guo Cong, and Li Yibo. Variable Probability Based Bidirectional RRT algorithm for UAV Path Planning. In *Control and Decision Conference (2014 CCDC), The 26th Chinese*, pages 2217–2222, China, 2014. IEEE.

[5] JF Araujo, PB Sujit, and JB Sousa. Multiple UAV Area Decomposition and Coverage. In *2013 IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA)*, pages 30–37, Singapore, 2013. IEEE.

[6] Johannes Meyer, Alexander Sendobry, Stefan Kohlbrecher, Uwe Klingauf, and Oskar von Stryk. Comprehensive Simulation of Quadrotor UAVs Using ROS and Gazebo. In *Simulation, Modeling, and Programming for Autonomous Robots*, pages 400–411. Springer, Dramstadt, 2012.

[7] H Cundy and A Rollett. Lissajous's Figures. *Mathematical Models*, pages 242–244, 1989.

[8] Steven M. LaValle. Rapidly Exploring Random Trees A New Tool for Path Planning. In *Proceedings of the Mediterranean Conference on Control and Automation*, Sardinia, Italy, 1998.

[9] Wei-Long Yang, Luo Lei, and Jing-Sheng Deng. Optimization and Improvement for Multi-UAV Cooperative Reconnaissance Mission Planning Problem. In *International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), 2014 11th*, pages 10–15, Chengdu, China, 2014. IEEE.

[10] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. ROS: an open-source Robot Operating System. In *ICRA workshop on open source software*, volume 3, page 5, Kobe, Japan, 2009.

[11] Nathan Koenig and Andrew Howard. Design and Use Paradigms for Gazebo, an Open-Source Multi-robot Simulator. In *International Conference on Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ*, volume 3, pages 2149–2154, Sendai, Japan, 2004. IEEE.

[12] Alessio Del Bue, M Tamassia, F Signorini, Vittorio Murino, and A Farinelli. Visual Coverage Using Autonomous Mobile Robots for Search and Rescue Applications. In *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 1–8, Linköping, Sweden, 2013. IEEE.

[13] Yong Bao, Xiaowei Fu, and Xiaoguang Gao. Path Planning for Reconnaissance UAV Based on Particle Swarm Optimization. In *Second International Conference on Computational Intelligence and Natural Computing Proceedings (CINC), 2010*, volume 2, pages 28–32, Wuhan, China, 2010. IEEE.

[14] Hyo-Sang Shin, Cédric Leboucher, and Antonios Tsourdos. Resource Allocation with Cooperative Path Planning for Multiple UAVs. In *UKACC International Conference on Control (CONTROL), 2012*, pages 298–303, Coventry University,UK, 2012. IEEE.

[15] Jie Chen, Wenzhong Zha, Zhihong Peng, and Jian Zhang. Cooperative Area Reconnaissance for Multi-UAV in Dynamic Environment. In *Control Conference (ASCC), 2013 9th Asian*, pages 1–6, Istanbul, 2013. IEEE.

[16] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79(3):497–516, 1957.

[17] Herman Bruyninckx. Open robot control software: the orocos project. In *IEEE International Conference on Robotics and Automation, 2001. Proceedings 2001 ICRA.*, volume 3, pages 2523–2528, Seoul, Korea, 2001. IEEE.

[18] Han J Wang C Yi G. Cooperative control of UAV Based on Multi-Agent System. In *Proceedings of the 2013 IEEE 8th Conference on Industrial Electronics and Applications, ICIEA 2013*, Melbourne, Australia, 2013.

[19] Zhu W Li L Teng L Chenglong Y Jiaxun K. Enhanced Sparse A * Search for UAV Path Planning Using Dubins Path Estimation. In *Proceedings of the 33rd Chinese Control Conference*. China, 2014.

[20] Michael Sipser. *Introduction to the Theory of Computation*. Thomson Course Technology, Boston, 2006.