Waypoint-to-waypoint Energy-efficient Path Planning for Multi-copters

Nirmal Kumbhare
Department of Electrical and Computer Engineering,
University of Arizona, Tucson, AZ 85721, USA
nirmalk@email.arizona.edu

Aakarsh Rao
Department of Electrical and Computer Engineering,
University of Arizona, Tucson, AZ 85721, USA
aakarshrao7@email.arizona.edu

Chris Gniady
Department of Computer Science,
University of Arizona, Tucson, AZ 85721, USA
gniady@cs.arizona.edu

Wolfgang Fink
Department of Electrical and Computer Engineering,
University of Arizona, Tucson, AZ 85721, USA
wfink@email.arizona.edu

Jerzy Rozenblit,
Department of Electrical and Computer Engineering,
University of Arizona, Tucson, AZ 85721, USA
jerzyr@email.arizona.edu

Abstract—**Flight time is a key aspect for ubiquitous deployment of unmanned multi-copters. The duration of flight depends on on-board component efficiency along with the battery capacity of the multi-copter. The propulsion system (motor and propeller) of the multi-copter consumes more than ~60% of the battery energy. In this paper, we present a novel algorithm for finding an energy-efficient path between two waypoints in three dimensional space. The algorithm takes into consideration the mechanical and electrical properties of its propulsion system along with the dimensions of the obstacle space to determine an energy efficient flying path. Through detailed simulations of real world flying scenarios we demonstrate that our algorithm provides ~30 % energy savings for mid-range flights as compared to traditional line-of-sight approaches.**

TABLE OF CONTENTS

1. INTRODUCTION

In recent times, Unmanned Aerial Vehicles (UAVs) are gaining popularity in research, commercial use, and surveillance. In the commercial sector, multi-copters have found significant importance in autonomous delivery systems and personal aerial photography. This has been made possible with the advances in electronic systems that allow low-cost production of flight controllers, accelerometers, global positioning systems, and cameras. Software environments have also improved with the emergence of simulation frameworks, communication protocols, ground control stations, and development environments; whereas, the core components of multi-copters like motors, propellers and batteries have not seen drastic changes in technologies.

Average flight time is the most crucial aspect to consider during the deployment of multi-copters, especially in autonomous delivery systems. Since, UAVs carry their energy source on-board it becomes even more essential to consider energy efficiency of on-board components, which directly relates to total flight time. Propulsion systems (motors and propellers) are identified as the most energy consuming systems of a multi-copter that account for over 60% of energy consumption [1-3]. Hence, improving energy consumption of the propulsion system is a critical aspect while considering multi-copter deployment to extend flight time.

Energy consumption of the propulsion system is related to many factors, such as number of obstacles in the path, size of obstacles, weight and dimensions of the UAV, combination of motor and propeller, and wind conditions. While there are numerous factors, in this paper we are focusing on finding an energy efficient flight path for a given combination of motor, propeller, multi-copter weight, obstacles dimension and obstacles layout. We ignore wind conditions in this work and plan to pursue its incorporation in our future work. Traditional way of flying a multi-copter is to first set up the GPS coordinates for the source and destination waypoint. Then raise the altitude of the multi-copter to a level (permissible by Federal Aviation Administration (FAA)), such that no, or minimum obstacles are in the way of flight between the waypoints. This approach simplifies offline computation of the path from source to destination based on a 'line-of-sight' approach [4]. We refer here to 'line-of-sight' as the offline path planning approach of traveling in straight lines in open spaces clear of all obstacles. In case of obstacles, a pilot would maneuver a multi-copter around the obstacle to avoid collision following the preplanned paths. The decision of taking the path around the obstacle is left to the pilot and these decisions may not be energy efficient. While this is not a

978-1-5090-1613-6/17/$31.00 ©2017 IEEE

significant problem in rural or suburban areas that do not have many obstacles in form of tall buildings, the energy efficiency can be significantly degraded in metropolitan areas with many high-risers. Subsequently, we show that a simple line-of-sight approach with manual maneuvering around obstacles may not turn out to be an energy efficient choice. In addition, a shortest path approach may not always result in an energy efficient path due to the involvement of additional non-linear factors such as motor and propeller characteristics that vary significantly with maneuverability and altitude changes.

In this paper, we propose an energy efficient offline path planning algorithm which takes electrical characteristics of motor, mechanical characteristics of propellers, and physical dimensions of obstacles as inputs. Our approach addresses two main problems: (1) correlating mechanical characteristics of propeller with the electrical characteristics of motor to derive energy efficient multi-copter velocity (airspeed) between two points, and (2) planning energy efficient path from source waypoint to destination waypoint in the obstacle space. We also consider the constraints set by the FAA [6] and obstacle dimensions for our path planning approach. We develop a simulation setup consisting of real-world urban scenarios to experimentally validate our approach and compare our energy efficient path planning approach with the traditional line-of-sight method.

The rest of paper is organized as follows: Section 2 describes related work and how our approach differs from existing state of the art solutions. Section 3 details the motor propeller characteristics. Section 4 discusses the algorithm used for deriving energy efficient airspeed from one point to another point. Section 5 elaborates on the path planning algorithm with energy as the cost function. Section 6 provides details of our experimental and simulation setup, and we conclude with future work in Section 7.

2. RELATED WORK

Energy efficient path planning has been an active area of research especially in the realm of mobile land robots/Unmanned Ground Vehicles [7-12]. Complementary work involves finding precomputed paths based on efficiently harvesting energy from the environmental sources (e.g., sun, wind) [7, 8]. Our focus however is to consider the on-board battery as the only source of energy for the multi-copter.

A wide range of work involves planning paths while reducing energy consumption. A real-time energy efficient path planning approach is discussed in [9], that uses Bayesian estimation to predict energy efficient path alternatives in real-time with *a priori* knowledge of the mission. Variants of Traveling Salesman Problem and A* algorithm have also been used to guarantee land-robot task completion with minimum energy [10-12]. In [13], energy consumption of locomotion of mobile robots is minimized by determining the relationship between motor speeds and energy consumption with fixed path strategies. But, multi-copter path planning is significantly different as it involves motion in 3D space, and involvement of propellers along with motors.

Related works in Unmanned Aerial Vehicles (UAVs) have adopted similar approaches and algorithms as mobile land robots [14-16]. An analogous work discusses an energy consumption model of UAVs by modeling the aerodynamic properties of the propellers [17]. However, they do not consider the energy capacity of the battery and motor characteristics with its implication for path planning. A related work in vehicle routing problems for multi-copter can be found in [18], where the authors optimize energy and time considering weight and reusability of the multi-copter. However, they do not consider the characteristics of the propulsion system.

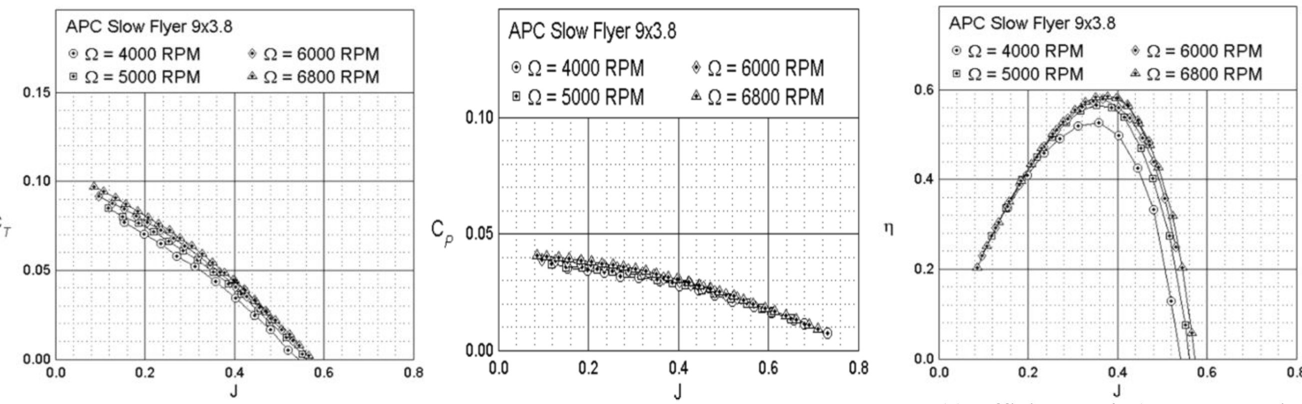
Our work differs from the existing works [13, 17, 18] mainly due to the following: (I) motor-propeller characteristics have been consider in a multi-copter because an energy efficient operating point for a propeller might not be an energy efficient operating point for a motor and *vice versa*, (II) minimum energy flying airspeed is derived based on physical characteristics of the multi-copter components, and (III) obstacle dimensions based on a real-world urban scenario are considered for multi-copter path planning. Our implementation can be easily added onto existing multi-copter ground control station software that utilizes the line-of-sight approach.

3. MOTOR-PROPELLER CHARACTERISTICS

Propellers and motors are responsible for more than ~60% of energy consumption [1-3]. Hence, it is critical to consider the physical characteristics of the propeller and electrical characteristics of the motor while deducing least energy flying conditions. The motor-propeller characteristics of the multi-copter are the basis for deriving a relationship between multi-copter airspeed and power demand from motor-propellers. The non-linear characteristics of the propulsion system pose a challenge in finding an energy efficient airspeed for a particular flight between any two points in 3D space.

Propeller Characteristics

Fixed-wing propellers are commonly used in small UAVs and multi-copters for generating required thrust to take flight. Small size of these propellers causes them to operate at very low Reynolds number (< 100,000) [20, 21], which increase the drag on the propeller. The decrease in lift and increase in drag is more pronounced as the Reynolds number goes below 100,000. In general, performance of a propeller is described using normalized coefficients viz.: advance ratio (J), thrust coefficient (C_T), power coefficient (C_P), and torque coefficient (C_Q). In the remainder of the paper, the term "Performance coefficient" refers to these coefficients. Equations for these coefficients are given as:



(a) Thrust coefficient v/s Advance Ratio **(b) Propeller coefficient v/s Advance Ratio** **(c) Efficiency v/s Advance Ratio**

Figure 1. Propeller Characteristics Curves [19].

$$J = \frac{u}{nD}\ ,\ C_T = \frac{T}{n^2 D^4 \rho}\ , C_P = \frac{2\pi Q}{\rho n^2 D^5} \text{ and } C_Q = \frac{C_P}{2\pi} \quad (1)$$

where, u is air velocity (m/s) through propeller, n is propeller speed (rev/s or RPM), D is propeller diameter (m), T is thrust (N) generated by propeller, ρ is air density (kg/m36), Q is torque (N/m) needed and P is power. The prediction of performance for small scale propellers is difficult because of low Reynolds number [20, 21]. UIUC propeller database [19] consist of these performance coefficients measured during wind tunnel tests for large number of small scale propellers for wide range of propellers' RPM along with performance coefficient plots for a range of advance ratio (J) and propeller speed (n). An illustration of the propeller characteristic curves for the propeller APC Slow Flyer 9x3.8 is shown in Figure 1. As it can be seen in Figure 1, as J ($\propto u$) increases the variation in performance coefficients is noticeable in a non-linear fashion. The values of performance coefficients also vary with the propeller speed n, as in Equation 1.

Motor Characteristics

Brushless DC (BLDC) motors are commonly used in multi-copters [22]. BLDC motors are known for their high efficiency with noiseless operation, high speed range, long life, and better speed v/s torque characteristics curves. Varying voltage input to BLDC motor is one of the common technique for changing altitude and speed of multi-copters. All the commercial BLDC motors are provided with specifications of rated voltage, power, speed, torque, efficiency at rated speed, torque constant and back EMF constant. These specifications are required to derive the characteristics of a motor that are represented by torque v/s speed and efficiency v/s speed curves, as shown in Figure 2. The electric motor characteristics are not linear and shift right as the motor input voltage increases (Figure 2). BLDC motor electrical characteristics are defined using following equations:

$$V_T = K_e n_m + IR, \quad (2)$$

$$T_m = K_T I \quad (3)$$

where, V_T is terminal voltage, n_m is motor angular velocity, I is motor current, R is motor internal resistance, T_m is torque, K_e is back-EMF constant, and K_T is torque constant.

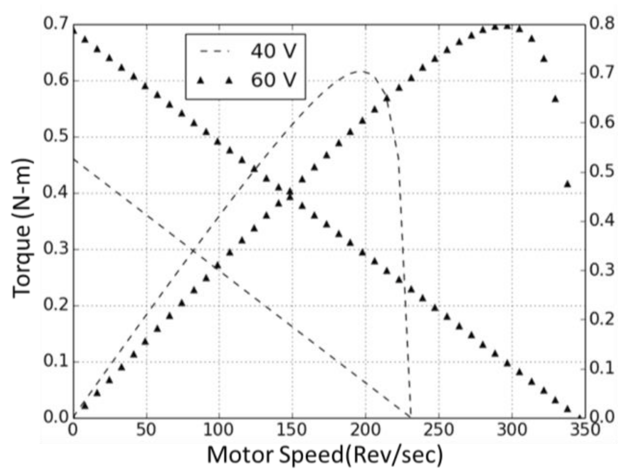


Figure 2. BLDC Motor Characteristics.

4. ENERGY EFFICIENT AIRSPEED CALCULATION

Selecting an appropriate motor and propeller combination is crucial in designing an energy efficient multi-copter. As the use of multi-copters is becoming more pronounced in commercial applications (e.g., package delivery), just a one-time selection of a motor-propeller combination will not suffice in maintaining energy efficiency. Multi-copter weight can vary from one trip to another, due to changes in the package weight it is carrying, which leads to variations in the operating region of propeller and motor efficiency curves. These variations bring uncertainty in the total flight time that a battery can deliver. In our work, we primarily focus on deducing an energy efficient airspeed for a given combination of motor and propeller. Selection of an appropriate combination of motor and propeller is beyond the scope of the paper.

In this section we briefly describe the iterative algorithm (Algorithm 1) used to find an energy efficient airspeed (flight velocity) between two points for a given combination of motor, propeller, and weight of a multi-copter.

Algorithm 1:
Initialization:
Intialize() {
RPM_arr = *1-D array of RPM values*
airspeed_arr = *1-D array of air speed values*
T_arr = *2-D array for "propeller-thrust vs Air speed" for all available RPMs thrust coefficients.*
Q_arr = *2-D array for "propeller-torque vs Air speed" for all available RPMs thrust coefficients.*

return **RPM_arr**, **airspeed_Arr**, **T_arr**, **Q-arr**
}

Execution:
*find_optimal_airspeed (***weight***,* **src***,* **dst***){*
1) **min_energy** = *MAX_value_possible*
2) **min_airspeed** = *0*
3) for **i** in *(***airspeed_arr***):*
   a) *Find* thrust-per-propeller $\mathbf{T_h}$ *and* **multi-copter tilt** *needed to achieve air speed* **airspeed_arr[i]** *for* **weight** *from* **src** *to* **dst**
   b) *Search for index* **k, j** *in* **T-arr** *which delivers value closest to requested* $\mathbf{T_h}$ *and* **airspeed_arr[i]**. **k** = *RPM index,* **j** = *air speed index in* **T-arr**.
   c) **required_torque** = **Q_arr[k][j]**
   d) **V** = *BLDC motor input voltage to deliver torque of* **Q_arr[k][j]** *at RPM* **RPM_arr[k].**
   e) **I** = *find motor armature current.*
   f) **t** = *(distance between* **src-dst***) /* **airspeed_arr[i]**
   g) **input_energy** = **V * I * t**
   h) **min_energy** = *minimum (***input_energy***,* **min_energy***)*
   i) if **min_energy** *updated* then **min_airspeed** = **i**,
4) return **min_energy**, **min_airspeed**
}

Algorithm 1: Iterative approach to deduce energy efficient airspeed.

The algorithm as shown in Algorithm 1, is divided into two phases: 1) Initialization phase, and 2) Execution Phase. Initialization phase creates a one-time look up tables using arrays to reduce run time of execution phase. Assuming propeller coefficients are known for a wide range of RPMs and air speeds in advance (from [19]), **RPM_arr** and **airspeed_arr** are used to store these values respectively. Additional propeller coefficients values are calculated by interpolation if certain ranges of values are not provided. **T_arr** and **Q_arr** are 2-D arrays for thrust and torque respectively, where each row index corresponds to RPM at the same index in **RPM_arr**, each column index corresponds to air speed at same index in **airspeed_arr**. Each element of these arrays represent value of thrust and torque (calculated using Equation (1)), for a particular value of RPM and airspeed. Following the initialization phase, execution phase invokes *find_optimal_airspeed()* to determine energy efficient air speed for a multi-copter of given **weight** from source (**src**) to destination (**dst**) in 3-D space. An iterative approach is utilized to find the airspeed that consumes the least energy. *Step 3* traverses through all the air speed values present in **airspeed_arr**. *Step 3.a* finds the thrust and tilt required by the propeller to generate required multi-copter speed in flight. *Step 3.b* search for the required thrust and air-speed in **T_arr** and returns the RPM required by the propeller to generate the same. *Step 3.c* looks up for the required torque corresponding to the RPM and airspeed in array **Q_arr**.

Using Equation (2) and Equation (3), *step 3.d* determines the input terminal voltage required by the motor to generate torque and RPM calculated in *step 3.c*. *Step 3.e* finds the armature current of the motor using Equation (2). *Step 3.f* finds the time required to go from **src** to **dst** at air speed **airspeed_arr[i].** *Step 3.g* calculates the input energy consumed during the flight from **src** to **dst**. *Step 3.h* keeps the track of the minimum energy required to make the flight in **min_energy** and *Step 3.i* keeps the track of the multi-copter air speed in **min_airspeed** corresponding to minimum energy. **min_energy** and **min_airspeed** are updated only if the calculated operating point of BLDC motor is in the permissible range (defined by motor ratings). At the end of all iterations, values for the minimum airspeed and the corresponding minimum energy consumption are returned.

5. Energy Efficient Path Planning

To determine energy efficient path planning for a multi-copter to make a flight from source waypoint to destination waypoint, we employ the following algorithm (Algorithm 2):

1. *Define boundaries of air space in between two waypoints.*
2. *Identify all the obstacles present in the airspace.*
3. *Relax the dimensions of each obstacle to maneuver multi-copter around the edges of the obstacle, without any collision.*
4. *Create empty graph G.*
5. *Create uniformly distributed multiple node points (n_p) on the relaxed boundaries of each obstacle.*
6. *Connect all the node points which are in line sight with each other and do not collide with obstacles (**e**). Construct G(n_p,**e**).*

7. *Add source s_{np} and destination d_{np} nodes to G and connect them to all line of sight nodes.*

8. *Execute Initialization Phase (Intialize()) to create arrays for propeller characteristics.*

9. *For each connected pair of points in G, execute find_optimal_airspeed(weight, n_{src}, n_{dst}) to find* **min_energy** *between src and dst. Update the cost of edge with* **min_energy**.

10. *Run Dijkstra's algorithm on G(n_p,e) with* **min_energy** *from source and destination node as cost to find minimum energy path E_{min}. Return total energy consumed.*

The above is now explained in more detail in the following: *Step 1* first defines the boundaries for the flying airspace in between the source and destination waypoints. These boundaries represent the air passage in 3-D space in which multi-copter can fly. This space includes the flying space and obstacle space. The boundaries are represented in terms of maximum allowable altitude, minimum allowable altitude, and maximum allowable lateral deviation from hypothetical line of sight from source to destination. *Step 2, 3, and 4* are self-explanatory in the algorithm. S*tep 5* creates multiple nodes on the edges of the faces, facing towards source and destination waypoints. *Step 6* creates a mesh of nodes G(n_p, $\boldsymbol{e}$) by connecting only those nodes which are in line of sight with each other to avoid collisions. Nodes are connected to form a directed graph such that all the edges of the graph are directed towards destination waypoint. Source n_{src} and destination node n_{dst} are suitably chosen and appended to G as described in step 6. Following Algorithm 1, step 8 initialized the required data structures for the propeller characteristics. *Step 9* starts traversing from n_{src} to n_{dst} in G(n_p,$\boldsymbol{e}$) while calculating minimum energy **min_energy** required on each edge of the graph. For each edge of the graph it uses *find_optimal_airspeed(weight, n_{src}, n_{dst})* function (defined in Algorithm 1) to calculate the minimum energy and required airspeed. For each edge, its minimum energy is attributed as its traversal cost in the graph. Following cost calculation, *step 10* runs Dijkstra's algorithm on graph G(n_p,$\boldsymbol{e}$) to find minimum energy path from source node to destination node and reports the least energy consuming path along with total energy E_{min} and airspeed for all the edges.

The output of this algorithm generates a list of real world nodes that needs to be traversed from the source to the destination waypoint. For each edge the minimum airspeed can differ depending on the elevation angle of the edge in 3D world. The algorithm is an approximate algorithm for minimum energy estimation between waypoints. By varying the node counts on the obstacles, the output of the algorithm can be fine-tuned if needed.

6. Experiments and Simulations

We implemented our simulation environment to perform our experiments using Python. For our simulation we assumed multi-copter to be a four rotor quad-copter. Subsection "Component Selection" first gives an overview of the quad-copter specifications used in simulation environment. Subsection "Efficient Airspeed Estimation" describes the Algorithm 1 implementation using our experimental scenario. Subsection "Simulation Environment" provides the details of our simulation environment and our real-world experimental scenario, and finally, subsection "Experimental Results" gives the comparative results of energy consumption between the traditional line-of-sight approach and our proposed algorithms.

Component Selection

For our experiments, an off the shelf propeller from the UIUC propeller database is chosen. It is APC Slow Flyer propeller with a 9 inch diameter and 3.8 inch/revolution pitch. The performance coefficients of this propeller can be found in database [19]. Figure 1 shows the propeller performance coefficients used in our simulation testbed. For this propeller, the UIUC database provides performance coefficients for RPM ranging from 4000 to 7000. In order to widen our search space for determining energy efficient airspeed, we used extrapolation and interpolation technique to create performance coefficients for RPMs ranging from 4000 to 15000 in steps of 400. For simulating motor characteristics, we used the specifications (Table 1) of BN23-MG13 BLDC motor. The electrical characteristics of BLDC motor used for simulations are shown in Figure 2.

Table 1. BN23-MG13 BLDC Motor Specifications.

Parameters	Value
Rated Voltage	36 V
No load speed	12500 RPM
Rated Torque	0.114 N-m
Rated Current	4.3 A
Torque-constant	0.027 N-m/A
Back EMF constant	5.95 Rev/(V-sec)
Weight	0.238 Kg

We choose the quad-copter design as our multi-copter for our simulations. We have following assumptions in our simulation environment:

1. Quad-copter is in the shape of a cylinder (radius 0.5 m and height 0.2 m).

2. Majority of Quadcopter drag comes from its main body but not from the four arms holding the motor and propeller.

3. The weight of the quadcopter is 2 Kg and drag coefficient is assumed to be 0.55.

4. Air density is constant at 1.225 Kg /m^3 throughout the flight

Efficient Airspeed Estimation

We utilize Algorithm 1 for determining energy efficient airspeed. For our experiment we assume that the source waypoint is on the ground and the destination waypoint is 1000 m above the source waypoint. Both the waypoints are in line of sight with each other and quad-copter only has to make a vertical ascend from source to destination. In order to make this vertical flight, Algorithm 1 scans over a range of airspeeds from 1 m/sec to 14 m/sec.

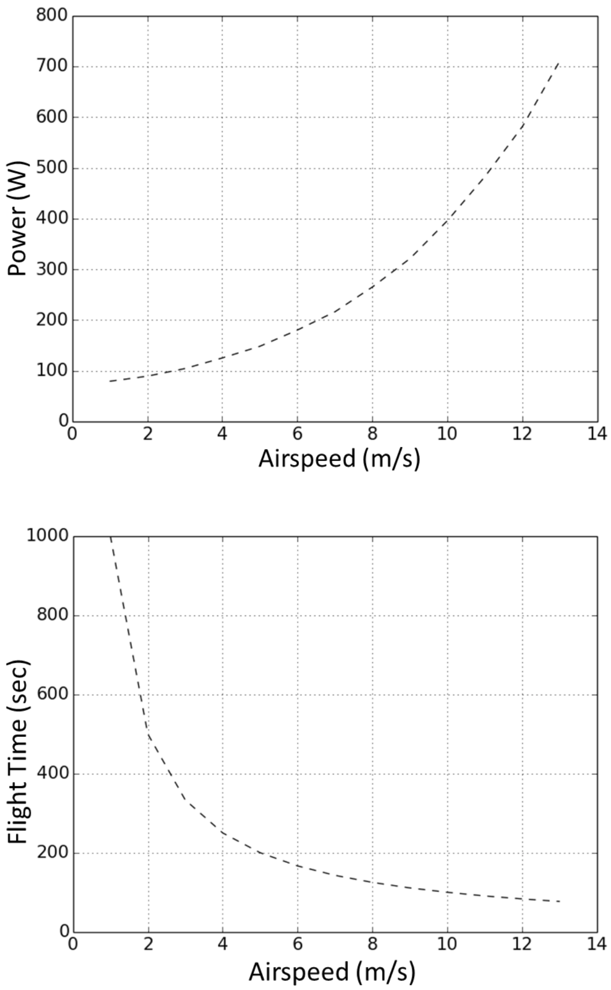


Figure 3. Input Power v/s Airspeed (top). Flight time v/s Airspeed (bottom).

For this range of airspeed, Figure 3(top) represents the plot for the input power that needs to be supplied to each motor. The input power requirement increases exponentially, which can be attributed to the fact that, as the propeller airspeed increases the drag force on the quadcopter body increases, and performance coefficient decreases significantly (shown in Figure 1). In order to maintain constant higher airspeed, propeller has to rotate at a higher RPM to oppose the increased drag force. Further, as airspeed increases, the increase in drag force and decrease in performance coefficient dominates. This leads to an increase in RPM and torque requirements that pushes the motor to operate in a low efficiency region. Figure 3(bottom) shows the airspeed – time curve. For a fix distance of 1000 m, the airspeed is inversely proportional to flight time. Figure 4 shows the plot for the input energy w.r.t airspeed. Input energy is given as *(input power) * (Flight time)*. Product of Figure 3(top) and Figure 3(bottom) gives Figure 4. Due to exponential drop in flight time, and exponential increase in input power, the input energy plot creates a minimum point. The airspeed at this minimal point can be concluded as the energy efficient airspeed. In our experiment, we observed that minimal energy airspeed changes with the direction (different flight angle w.r.t ground) of flight. So it becomes important to determine energy efficient airspeed every time the flight angle changes to save energy.

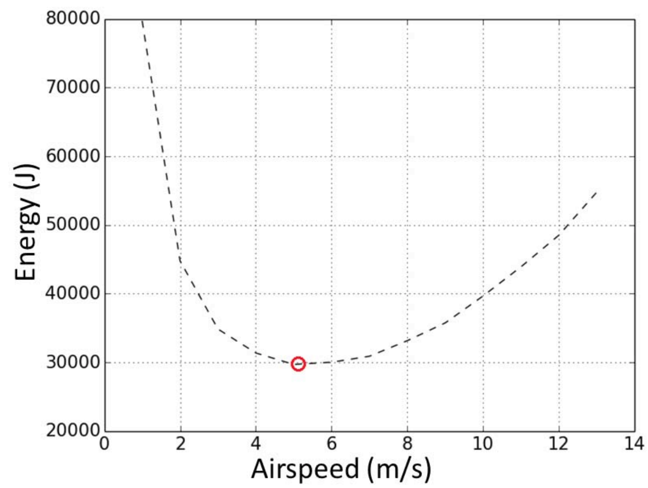


Figure 4. Energy v/s velocity curve – least energy deduction.

Simulation Environment

We used the following range of parameters for simulating multiple real world scenarios:

1) Maximum allowed flying altitude 400 ft. (approved by FAA).

2) Minimum allowed flying altitude 150 ft.

3) Random selection of height for the building in the range of 150 ft. to 450 ft.

4) Random selection of length of the building in the range of 50 ft to 80 ft.

5) Random selection of width of the building in the range of 50 ft to 80 ft.

6) Distance between consecutive buildings is set to be 40 ft.

7) Lateral air passage width is defined as 100 ft.

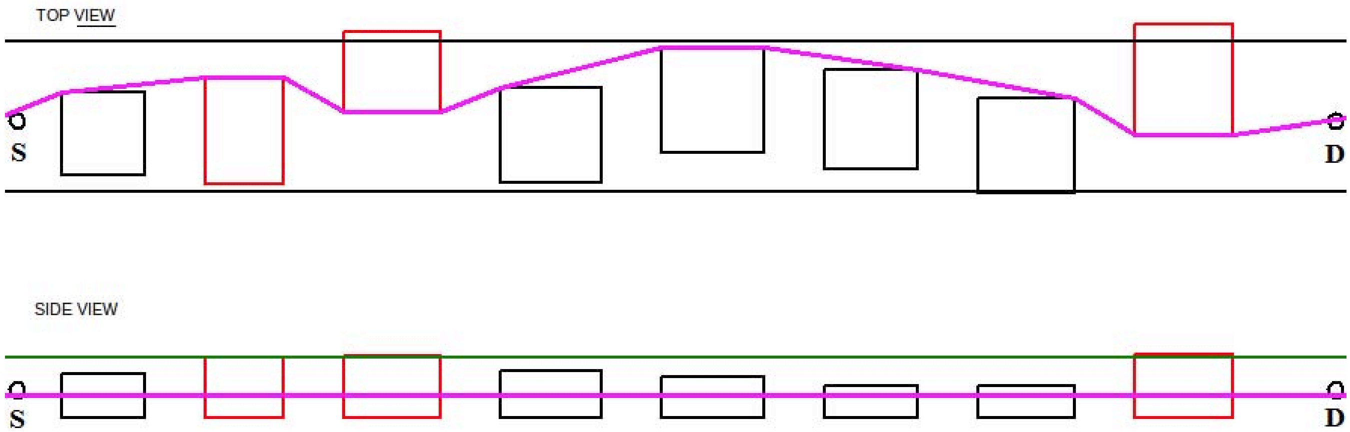


Figure 5. Simulation environment for a real-world urban scenario with energy efficient path.

In order to simulate a real world urban scenario, we consider multiple cuboidal obstacles with random dimensions, and place them randomly in between the source and destination waypoint (Figure 5). These cuboids represent buildings in a real world. The simulation environment is fully parametrized such that dimensions of the buildings, maximum–minimum flying altitude distance between source and destination waypoint, distance between consecutive buildings, and width of lateral air passage can be varied for simulating wide range of real world conditions. Figure 5 represents one of the simulation scenarios for a distance of 800 feet from source to destination waypoint. In this figure, 'black' cuboids represent buildings of height ranging from 150 – 400ft and 'red' represents buildings of height greater than 400ft.

Experimental Results

We perform a comparative study of energy consumption between line-of-sight approach and our proposed approach. Different simulation environments are generated by varying the random seed value of the simulator. To highlight the benefits of our algorithm we fix the distance between source and destination waypoint for preliminary experiments. The average energy saving will vary depending on the distance between source and destination. A minimum flying altitude for the quad-copter is defined for our simulation environment to make sure that the quad-copter is always at a safe altitude. We do not account for the energy consumed in raising the quadcopter to this minimum altitude as it will be similar for both the approaches.

In general, for a line-of-sight approach the obstacle count is minimal, hence it is valid to assume that for every simulation scenario, starting at the source waypoint, the quadcopter makes a vertical ascend from 150ft altitude to an altitude equal to either maximum height of the building or to the maximum allowable altitude (if obstacles are higher than permissible altitude). It then flies straight towards the destination waypoint, and descends down to 150ft. Airspeed for all the flights are calculated using Algorithm 1 to keep the comparison fair. In case obstacles are encountered while flying towards the destination, the quadcopter makes an arbitrary selection to get around the building.

The proposed path planning algorithm follows the Algorithm 2 flow for determining an energy efficient path from source to destination waypoints. We ran these experiments on multiple different scenarios and we observe that an alternative path is always chosen as compared to the line-of-sight approach. This implies that our proposed algorithm choses an energy efficient alternative path.

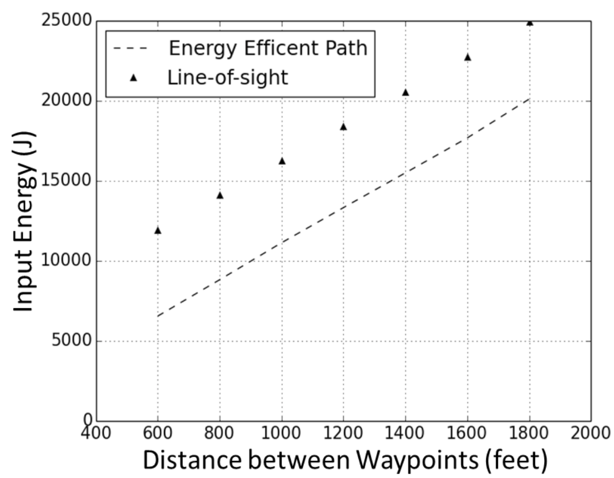


Figure 6. Input energy v/s Waypoint distance.

Figure 5 shows the energy efficient path selected by Algorithm 2 in 'magenta' color for a real-world urban scenario for a distance of 800ft. between source and destination waypoints. As we can see in Figure 5, the energy efficient path selected by Algorithm 2 prefers to get around the buildings instead of raising the altitude of the quad-copter to fly towards the destination (line-of-sight). The energy consumed by our algorithm for this scenario is 8668.2J and for a line-of-sight approach for the same scenario is 12,901J. This directly translates to energy savings of 32.6%. We average out the energy savings across multiple scenarios for 800ft and it results in an average savings of 30.3%. Further, to show the validity of our algorithm we compare its energy consumption with line-of-sight approach for a range of distances between source to destination waypoint. As we can see in Figure 6, our algorithm always performs better than the line of sight approach. Using the proposed algorithms for midrange distances, quad-copter rarely flies above tall buildings and saves major amount of energy by maneuvering around the building to reach the destination.

7. Conclusion and Future Work

In this paper, we have devised and implemented an energy efficient path planning algorithm for multi-copters. Flight time is an essential feature of a multi-copter which directly depends on the battery capacity. With motors consuming most of the battery energy it is essential to save energy of the propulsion system. We have linked motor and propeller characteristics in determining energy efficient path. The path is determined offline. We proposed two algorithms: 1) an iterative algorithm to determine an energy efficient airspeed between two line of sight points and 2) an energy efficient path planning algorithm between two waypoints in a real-world flying scenario. Even in the absence of obstacles, significant amount of energy can be saved by our algorithm as it selects an optimal flying airspeed for a multi-copter. Our simulation and experimental results show that the multi-copter chooses alternative energy efficient paths as compared to the traditional line-of-sight based path planning

In the future, simulator will be enhanced to integrate more environmental effects like wind and temperature effects. Simulator can be modified for comparing the energy consumption of the shortest path as oppose to energy efficient path. Detail study on the effects of distance between source and destination waypoints on energy savings will be done. The simulator can be also used for proper selection of motor and propeller combination. Accuracy of the simulator can be enhanced by integrating aerodynamic characteristics of a real quad-copter. We would also like to compare the simulation studies with an actual quad-copter hardware.

References

[1] Mulgaonkar, Yash, et al. "Power and weight considerations in small, agile quadrotors.", *SPIE Defense+Security*. International Society for Optics and Photonics, 2014

[2] Bershadsky, Dmitry, Stephen Haviland, and Eric N. Johnson. "Electric Multirotor Propulsion System Sizing for Performance Prediction and Design Optimization.", *American Institute of Aeronautics and Astronautics SciTech*, 2016.

[3] Gur, Ohad, and Aviv Rosen. "Optimizing electric propulsion systems for unmanned aerial vehicles.", *Journal of aircraft* 46.4 (2009): 1340-1353.

[4] Peot, Mark A., et al. "Planning sensing actions for UAVs in urban domains.", *European Symposium on Optics and Photonics for Defence and Security*. International Society for Optics and Photonics, 2005.

[5] Dijkstra, Edsger W. "A note on two problems in connexion with graphs.", *Numerische mathematik* 1.1 (1959): 269-271.

[6] Federal Aviation Administration, "Summary Of Small Unmanned Aircraft Rule (Part 107)", 2016. Available: https://www.faa.gov/uas/media/Part_107_Summary.pdf [Accessed: 21- June- 2016].

[7] Plonski, Patrick A., Pratap Tokekar, and Volkan Isler. "Energy- efficient Path Planning for Solar- powered Mobile Robots." *Journal of Field Robotics* 30.4 (2013): 583-601.

[8] A. Kaplan, P. Uhing, N. Kingry, and R. D. Adam, "Integrated path planning and power management for solar-powered unmanned ground vehicles", *IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 982–987.

[9] Sadrpour, Amir, Jionghua Jin, and A. Galip Ulsoy. "Real-time energy-efficient path planning for unmanned ground vehicles using mission prior knowledge.", *International Journal of Vehicle Autonomous Systems* 12.3 (2014): 221-246.

[10] Wang, T., Wang, B., Wei, H., Cao, Y., Wang, M., and Shao. Z, "Staying-alive and energy-efficient path planning for mobile robots", *IEEE American Control Conference*, pp. 868-873, 2008.

[11] Borgstrom, P., Singh, A., Jordan, B., Sukhatme, G., Batalin, M. and Kaiser, W., "Energy based path planning for a novel cabled robotic system", *International Conference on Intelligent Robots and Systems (IROS)*, pp.1745–1751, 2008.

[12] S. Liu and D. Sun, "Minimizing Energy Consumption of Wheeled Mobile Robots via Optimal Motion Planning", *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 2, pp. 401-411, April 2014.

[13] Y. Mei, Y.-H. Lu, Y. C. Hu, and C. S. G. Lee. "Energy-Efficient Motion Planning for Mobile Robots", *IEEE International Conference on Robotics and Automation (ICRA'04)*, Vol. 5, pp. 4344–4349, 2004.

[14] N. Mathew, S. L. Smith and S. L. Waslander, "Planning Paths for Package Delivery in Heterogeneous Multirobot Teams", *IEEE Transactions on Automation Science and Engineering*, Vol. 12, no. 4, pp. 1298-1308, Oct. 2015.

[15] P. Tokekar, J. Vander Hook, D. Mulla and V. Isler, "Sensor planning for a symbiotic UAV and UGV system for precision agriculture", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5321-5326, 2013.

[16] Al-Sabban, W. H., Gonzalez, L. F., and Smith, R. N., "Wind-energy based path planning for unmanned aerial vehicles using Markov decision processes", *IEEE International Conference on Robotics and Automation (ICRA),* pp. 784-789, 2013.

[17] Phung, D.K and Morin, P., "Modeling and energy evaluation of small convertible UAVs." *IFAC Proceedings Volumes,* Vol. 46, no. 30, pp. 212-219, 2013.

[18] K. Dorling; J. Heinrichs; G. G. Messier; S. Magierowski, "Vehicle Routing Problems for Drone Delivery," *IEEE Transactions on Systems, Man, and Cybernetics: Systems* , Vol.PP, no.99, pp.1-16, 2016.

[19] J.B. Brandt, R.W. Deters, G.K. Ananda, and M.S. Selig (20th April 2016), *UIUC Propeller Database*, University of Illinois at Urbana-Champaign, retrieved from http://m-selig.ae.illinois.edu/props/propDB.html

[20] J. B. Brandt and M. S. Selig, "Propeller Performance Data at Low Reynolds Numbers", *49th AIAA Aerospace Sciences Meeting*, AIAA Paper 2011-1255, 2011.

[21] Deters, R.W., Ananda, G.K., and Selig, M.S., "Reynolds Number Effects on the Performance of Small-Scale Propellers," AIAA Aviation and Aeronautics Forum and Exposition (Aviation 2014), AIAA Paper 2014-2151, 2014.

[22] Fuleky A., "Driving Systems of Unmanned Air Vehicles", *Academic and Applied Research in Public Management Science*, Vol.3, No.5, pp.665-688, 2004.

[23] "Flight thrust, power, and energy relations," 2009.

BIOGRAPHY

Nirmal Kumbhare *is currently pursuing his PhD at University of Arizona, Tucson, USA with ECE department. His research interest involves Reconfigurable systems, cloud computing, and modelling of simulation environments. Prior to joining PhD program he worked with Intel, India for 3 years as an Emulation and Validation Engineer. He received his M.Tech in Electronic systems from Indian Institute of Technology (IITB), Mumbai, India in 2010. He received his B.E. in Electronics and Instrumentation Engineering from Institute of Engineering and Technology (I.E.T), DAVV, Indore, India in 2008.*

Aakarsh Rao *received a B.S in Electronics and Communication Engineering from M.S. Ramaiah Institute of Technology, Bangalore, India in 2009 and an M.S in Computer Engineering from University of Arizona where he is currently pursuing his PhD.*

Chris Gniady *is currently Associate Professor in the Department of Computer Science at the University of Arizona. He received his Ph.D. in Electrical and Computer Engineering from Purdue University in 2005. He also received his B.S. degree in Electrical and Computer Engineering from Purdue University in 1997. His research interests include operating systems, multiprocessor architecture, and architectural support for operating systems.*

Wolfgang Fink *is currently an Associate Professor and the inaugural Edward & Maria Keonjian Endowed Chair of Microelectronics with joint appointments in the Departments of Electrical and Computer Engineering, Biomedical Engineering, Systems and Industrial Engineering, Aerospace and Mechanical Engineering, and Ophthalmology and Vision Science at the University of Arizona in Tucson. He was a Visiting Associate in Physics at the California Institute of Technology (2001–2016), and held concurrent appointments as Visiting Research Associate Professor of Ophthalmology and Neurological Surgery at the University of Southern California (2005–2014). Dr. Fink is the founder and director of the Visual and Autonomous Exploration Systems Research Laboratory at Caltech (http://autonomy.caltech.edu) and at the University of Arizona (http://autonomy.arizona.edu). He was a Senior Researcher at NASA's Jet Propulsion Laboratory from 2000 till 2009. He obtained a B.S. and M.S. degree in Physics and Physical Chemistry from the University of Göttingen, Germany, and a Ph.D. in Theoretical Physics from the*

University of Tübingen, Germany in 1997. Dr. Fink's research comprises human/brain-machine interfaces, "smart" service systems, autonomous systems, biomedical engineering for healthcare, and computer-optimized design. Dr. Fink is an AIMBE Fellow, a da Vinci Fellow, and a Senior Member IEEE. He has over 222 publications (including journal, book, and conference contributions) as well as 15 patents awarded to date.

***Dr. Jerzy W. Rozenblit** is University Distinguished Professor, Raymond J. Oglethorpe Endowed Chair in the Electrical and Computer Engineering (ECE) Department, and Professor of Surgery in the College of Medicine at The University of Arizona. From 2003 to 2011 he served as the ECE Department Head. During his tenure at the University of Arizona, he established the Model-Based Design Laboratory with major projects in design and analysis of complex, computer-based systems, hardware/software codesign, and simulation modeling. The projects have been funded by the National Science Foundation, US Army, Siemens, Infineon Technologies, Rockwell, McDonnell Douglas, NASA, Raytheon, and Semicon-ductor Research Corporation. He had served as a research scientist and visiting professor at Siemens AG and Infineon AG Central Research and Development Laboratories in Munich, where he was instrumental in the development of design frameworks for complex, computer-based systems. Currently, jointly with the Arizona Surgical Technology and Education Center, he is developing computer guided training methods and systems for minimally invasive surgery. Co-author of several edited monographs and over two hundred publications, Jerzy holds the PhD and MSc degrees in Computer Science from Wayne State University, Michigan, and an MSc degree from the Wroclaw University of Technology. He presently serves as Director of the "Life-Critical Computing Systems" Initiative, a research enterprise intended to improve the reliability and safety of technology in healthcare and life-critical applications.*