TRANSPORT
ISSN 1648-4142 / eISSN 1648-3480
2018 Volume 33 Issue 3: 589–597
https://doi.org/10.3846/transport.2018.2608

EVALUATION OF UAV AUTONOMOUS FLIGHT ACCURACY WHEN CLASSICAL NAVIGATION ALGORITHM IS USED

Ramūnas KIKUTIS[1*], Jonas STANKŪNAS[2], Darius RUDINSKAS[3]

[1]*Dept of Aviation Technologies, Vilnius Gediminas Technical University, Lithuania*
[2,3]*Dept of Aeronautical Engineering, Vilnius Gediminas Technical University, Lithuania*

Received 22 November 2016; revised 20 December 2017; accepted 26 March 2018

Abstract. This article examines and shows mathematical results of classical algorithm, which is used for small Unmanned Aerial Vehicle (UAV) navigation. The research is done with mathematical UAV model, which eliminates aerodynamics while the chosen flight path is followed by using vector field method. Lots of attention is dedicated to show possible flight path error values with representation of modelled flight path trajectories and deviations from the flight mission path. All of the modelled flight missions are done in two-dimensional space and all of the collected data with flight path error values are evaluated statistically. The most possible theoretical flight path error values are found and the general flight path error tendencies are predicted.

Keywords: navigation, algorithm, flight path error, small unmanned aerial vehicle, statistical evaluation, dynamic model.

Introduction

Small Unmanned Aerial Vehicles (UAVs) are becoming more widely used for various practical tasks and applications like carriage of small parcel, states board monitoring, electrical line inspection etc. All of these practical applications and newly created UAV flight rules require small UAVs to fly as accurately as possible. However, small UAVs still cannot prove good reliability and flight path accuracy. On the other hand, current research shows that one of the possible solutions to this problem is to look for the new and more advanced autonomous navigation algorithms. One of the more advanced algorithms is Dubins flight paths. Initially the Dubins model was developed by Dubins in 1957 to be used in a two-dimensional plane. The definition states that a vehicle, which moves in a 2D coordinate system at constant forward speed with a finite turn-rate constraints, the minimum distance path between two configurations is termed a Dubins path (Owen *et al.* 2015). Later it has been adapted to three-dimensional coordinate systems for its use in UAV applications.

In order to evaluate Dubins paths accuracy, first of all we need to investigate the accuracy of the classical navigation methods. For this reason, we chose to use mathematical UAV model, which eliminates aerodynamics. It means that the results from the model are very general and could be later used for any type of UAV airplane. Besides that, chosen model has no external factors influence like wind or turbulence, but these, if needed, could be included when using appropriate equations (Kothari *et al.* 2014; Brezoescu *et al.* 2013). The collected data later could be used to account for any side drift including side drift due to wind as in paper by Brezoescu *et al.* (2011). All in all, by using this model we can be sure that theoretical data acquired from the model is not influenced by any of other factors (Dadkhah, Mettler 2012) and flight path error we get is only due to navigation algorithm itself.

In next section, mathematical model of the UAV is presented.

1. Mathematical model of the UAV

The mathematical model of UAV we choose for this research is based on 5 differential equations, which are according to Nelson *et al.* (2006) and Beard *et al.* (2014). This model imitates the UAV flight when the autopilot makes control actions. For this reason, data collected from this model is further used to evaluate only autonomous UAV navigation. These equations are given below:

$$\dot{p}_n = V_a \cdot \cos\psi + w_n; \tag{1}$$

$$\dot{p}_e = V_a \cdot \sin\psi + w_e; \tag{2}$$

*Corresponding author. E-mail: *ramunas.kikutis@vgtu.lt*

$$\ddot{X} = b_{\dot{X}} \cdot \left(\dot{X}^c - \dot{X}\right) + b_X \cdot \left(X^c - X\right); \tag{3}$$

$$\ddot{h} = b_{\dot{h}} \cdot \left(\dot{h}^c - \dot{h}\right) + b_h \cdot \left(h^c - h\right); \tag{4}$$

$$\dot{V}_a = b_{V_a} \cdot \left(V_a^c - V_a\right), \tag{5}$$

where: $\dot{p}_n$ – change of the north coordinate speed; $\dot{p}_e$ – change of the east coordinate speed; ψ – flight heading; V_a – aircraft airspeed; w_n – north component of the wind vector; w_e – east component of the wind vector; X – aircraft flight course; X^c – autopilot flight course; $\dot{X}$ – change of an aircraft flight course speed; $\dot{X}^c$ – change of an autopilot flight course speed; $\ddot{X}$ – change of an aircraft flight course acceleration; b_X – proportional coefficient for tuning course tracking; $b_{\dot{X}}$ – derivative coefficient for tuning course tracking; h – aircraft flight height; $\dot{h}$ – change of an aircraft flight height speed; h^c – autopilot flight height; $\dot{h}^c$ – change of an autopilot flight height speed; $\ddot{h}$ – change of an aircraft flight height acceleration; b_h – proportional coefficient for tuning height tracking; $b_{\dot{h}}$ – derivative coefficient for tuning height tracking; V_a^c – autopilot airspeed; $\dot{V}_a$ – change of an aircraft flight airspeed; b_{V_a} – proportional coefficient for tuning airspeed tracking.

From Equations (1–5) we see that model includes UAV mathematical position in three dimensional Cartesian coordinate system. We have position in the 2D plane p_n, p_e and altitude h. Furthermore, model includes control of the course angle X and flight airspeed V_a. All of the calculations can be performed in calm meteorological conditions, but the wind w_n, w_e can be simulated as well. Inputs to this model must be the autopilot flight height h^c, the autopilot airspeed V^c, and the autopilot flight course X^c. Here the flight heading of UAV – ψ is calculated by the Equation (6) where γ_a represents air-mass-referenced flight path angle:

$$\psi = X - \sin^{-1}\left(\frac{1}{V_a \cdot \cos\gamma_a} \cdot \begin{pmatrix} w_n \\ w_e \end{pmatrix}^T \cdot \begin{pmatrix} -\sin X \\ \cos X \end{pmatrix}\right), \tag{6}$$

where: γ_a – air-mass-referenced flight path angle.

All of the required inputs to the model are taken from the modelled vector field (Wang *et al.* 2016), which for the straight line path follow is described in Equation (7):

$$X_d(e_p) = -X^\infty \cdot \frac{2}{\pi} \cdot \tan^{-1}(k_p \cdot e_p), \tag{7}$$

where: e_p – deviation from the flight path; k_p – proportional system tuning coefficient; X^∞ – flight course which should be kept at infinite distance from the flight path; $X_d(e_p)$ – flight course function of e_p variable.

In Equation (7) we have flight path error value e_p, which controls course angle. If UAV has drifted from the predefined flight path, corrections are made according e_p. If the flight path error is big, flight course must be altered by 90 degrees. All in all, due to flight path error, course can vary in range as defined in Equation (8):

$$-\frac{\pi}{2} < X^\infty \cdot \frac{2}{\pi} \cdot \tan^{-1}(k_p \cdot e_p) < \frac{\pi}{2}. \tag{8}$$

For this research, classical UAV navigation algorithm was chosen. It states that UAV will make the transition from one straight-line path to another straight-line path when it enters the sphere, which is near the waypoint and has some constant radius. This transition and course change will appear according Equation (9):

$$\left|\left\{p(t)\right\} - \left\{w_i\right\}\right| \le r, \tag{9}$$

where: $\left\{p(t)\right\}$ – aircraft position vector function of time; $\left\{w_i\right\}$ – waypoint Cartesian coordinates vector; r – waypoint sphere radius.

UAV current position is calculated and stored in vector $\left\{p(t)\right\}$ and if the difference between current UAV coordinates and the coordinates of the waypoint $\left\{w_i\right\}$ is less or equal to the size of the radius of the sphere, then it is time to change the flight course and switch to another flight path leg. The visual example of this algorithm is shown in Figure 1.

In Figure 1 we can see that UAV starts its flight mission at waypoint $\left\{w_i\right\}$ and continues along the path until it reaches waypoint $\left\{w_{i+3}\right\}$. After that waypoint UAV makes final turn and returns to initial departure point. During the flight, UAV is controlled with the autopilot commands according the data received from the imaginable vector field. This method is not complicated, but on the other hand, it has some disadvantages:

- UAV will start waypoint transition when it reaches intersection of predefined sphere and current flight leg. UAV is not flying until it is exactly above waypoint;
- If the radius of the sphere is too small and flight accuracy is not so good, the UAV may overshoot and will never reach the waypoint. Mission will fail;
- If the radius of the sphere is too big, the accuracy of the flight will degrade. The flight path error tolerance for the UAV will be bigger.

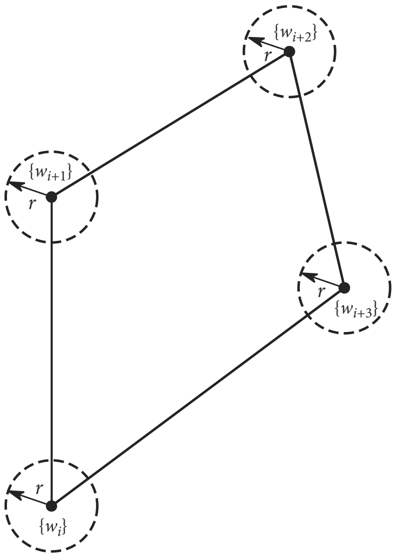

Figure 1. Visualization of the classical UAV autonomous navigation algorithm with four waypoints

All of the pluses and minuses of this algorithm in value terms remain unclear. Due to this, in the next section algorithm evaluation results are presented.

2. Results of the flight path accuracy

To find out the tendencies of the UAV flight accuracy when it has to follow some generated mission trajectory (when using classical course change algorithm), we choose flight path, which has three waypoints (Li *et al.* 2015). These three waypoints are connected with three straight-line segments and has equal length of 500 m. It means all mission is 1500 m long. Along the mission path, UAV must change course two times by 120°.

Some researchers look into this problem from the flight time constrains (Avellar *et al.* 2015; Yeol, Hwang 2016). However, we make an assumption that the accuracy of the flight will be the function of the flight speed and the radius of the sphere:

$$e_p = f\left(V_a, r\right). \tag{10}$$

First of all, the speed was chosen as constant value 10 m/s and sphere radius of different values between 5 and 30 m were taken. The results of this flight can be seen in Figure 2a and closer view is presented in Figure 2b. From these figures, we can clearly see that if we increase the radius of the sphere near the waypoint from 5 m up to 30 m, the flight path overshoot is decreasing. It means that the smallest radius gives the biggest flight path error outside of the defined trajectory. On the other hand, in this case, the error inside of the trajectory is smallest. It still remains unclear, which radius for this flight mission would be more optimal.

Tendencies of the flight path error changes with respect to time are presented in Figure 3 and the closer look is provided in Figure 4. In these time line figures positive values represent flight path error, which is inside of the defined trajectory and the negative values give flight path error, which is outside of the defined trajectory. Another important aspect is that due to the flight path error near the first turn, we get slightly different lengths of the track, which UAV has flown. This is clearly visible near the second turn, as is happens approximately at time interval 100 s. It is visible that the biggest time delay occurs when the sphere radius is 5 m.

Further, the same experiment was done with the constant radius of the sphere and variable flight speed. The radius of the sphere was chosen 10 m and the speed was variable in the range from 5 to 30 m/s with 5 m/s increments. The same flight mission was used which has triangular shape with equal flight leg segments of 500 m. The results can be seen in Figures 5 and 6. Results show that with the increasing flight speed, the flight path overshoot near the turn increases too. One exception in this case is the speed of 30 m/s. After the first turn, UAV has no time to return to the flight mission path. When UAV is close to the second turn it flies near the sphere and never enters it. It means that the algorithm fails and UAV will never accomplish this mission. This example proves that the classical UAV navigation algorithm is not fail safe. Nevertheless, from Figure 6 we can see that UAV will accomplish the mission faster with bigger airspeeds. It makes no difference that for bigger airspeeds we have much longer distances.

From these figures, it is clear (as it is stated in the article by Zhong, Yan (2015)) that there is a danger for the UAV to fly into surrounding obstacles, which could be even outside of the trajectory. However, it is difficult to extract some meaningful or accurate mathematical values from these figures. Due to that in the next section, some statistical analysis is presented.

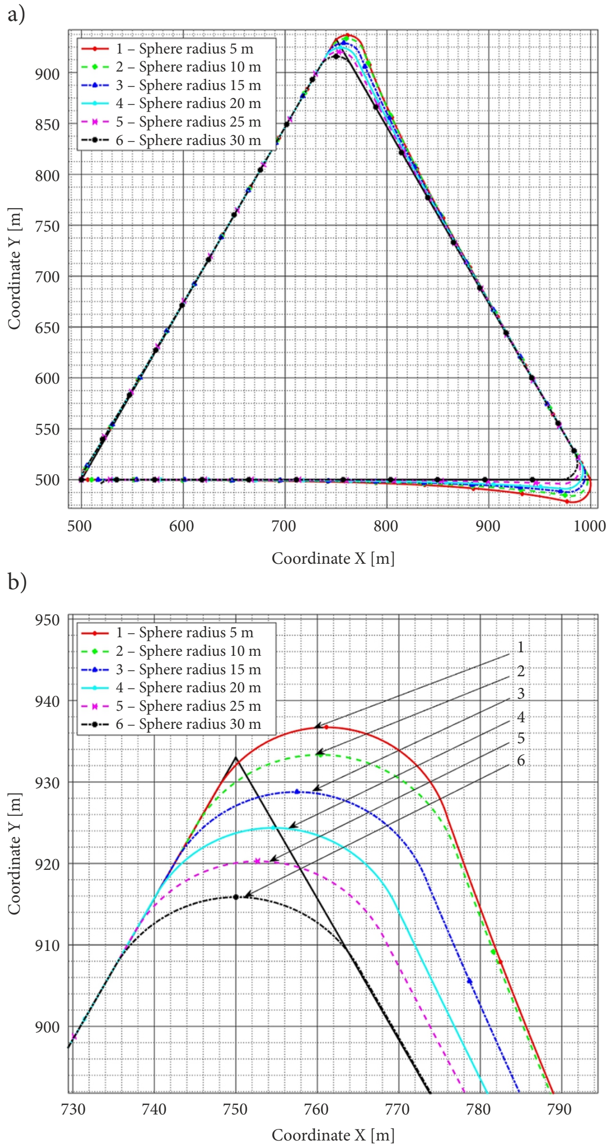

Figure 2. Flight path error when different sphere radius is used: a – full flight mission; b – closer look to the flight path error near the first waypoint

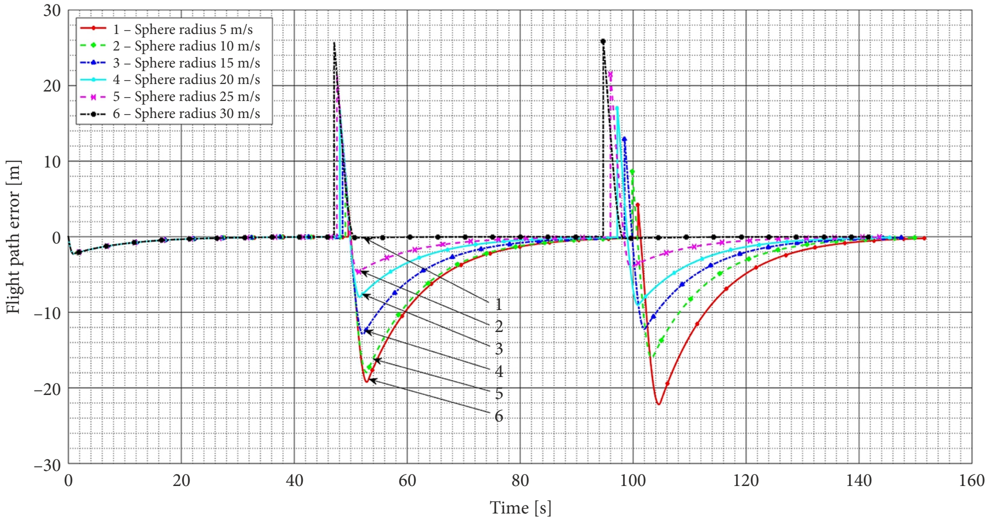

Figure 3. Flight path error values with respect to time when different sphere radius is used

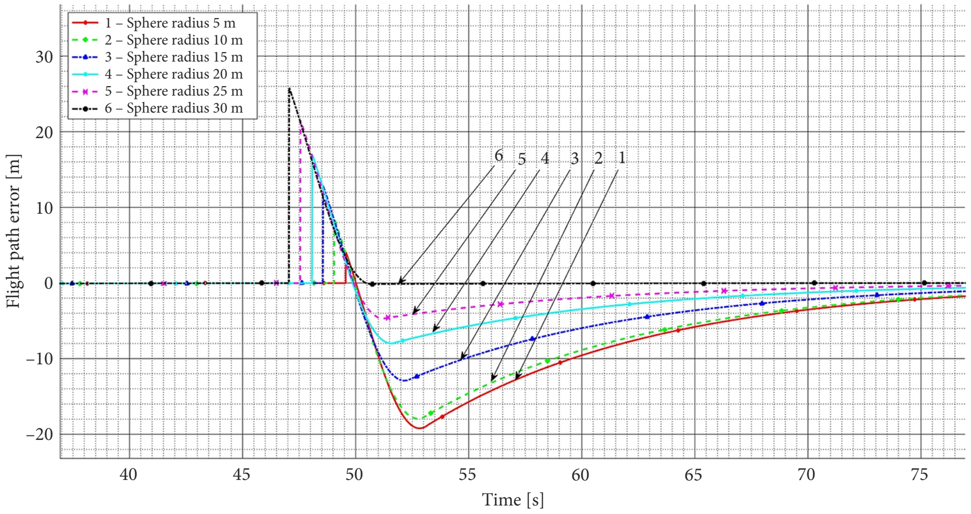

Figure 4. Closer look to the flight path error with respect to time near the first waypoint

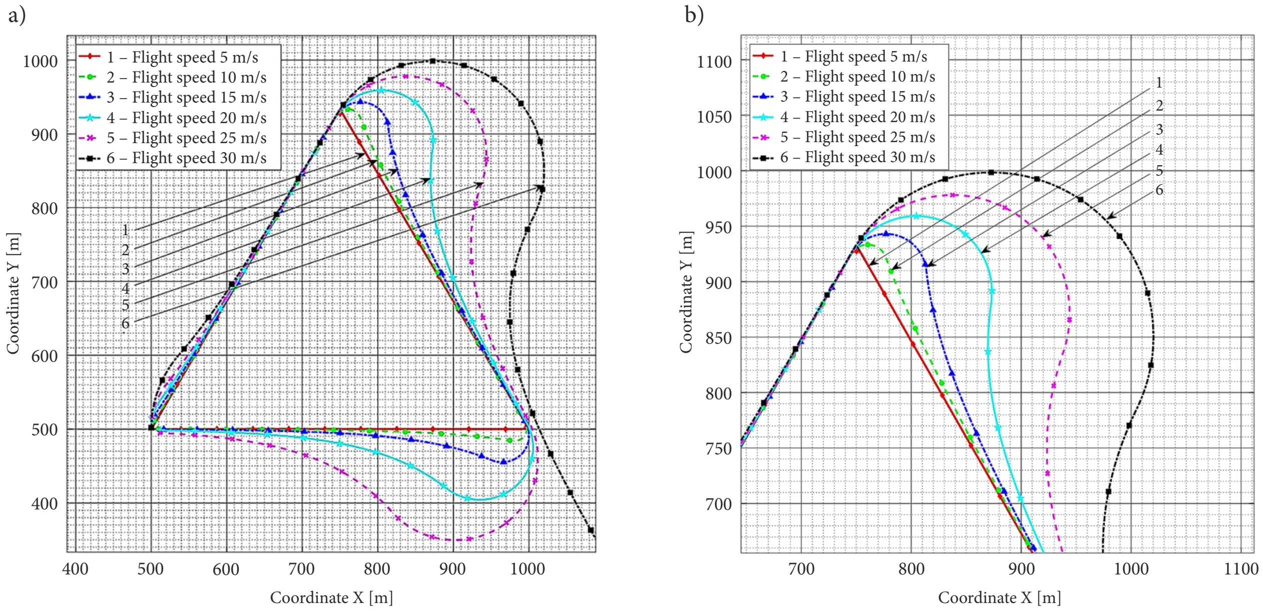

Figure 5. Flight path error when different airspeed is used: a – full flight mission;
b – closer look to the flight path error near the first waypoint

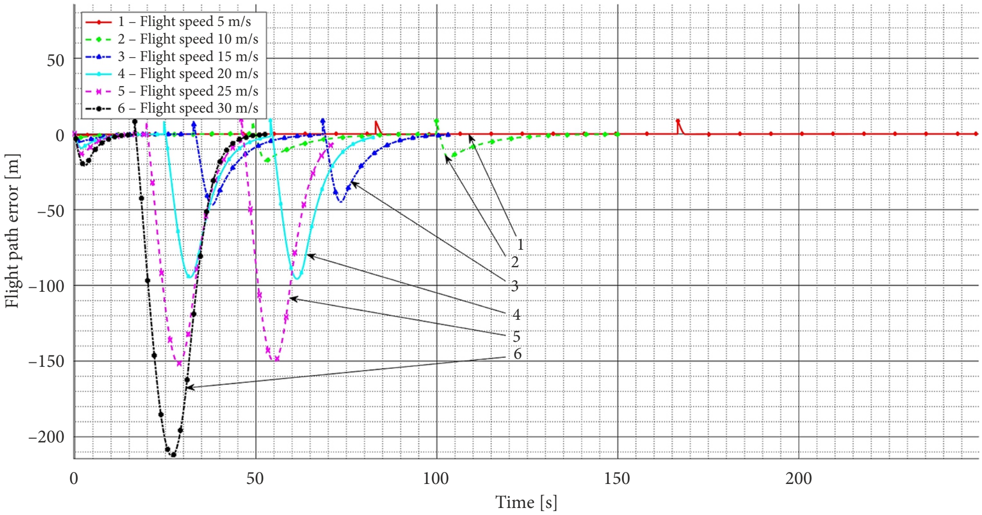

Figure 6. Flight path error values with respect to time when different airspeed is used

3. Statistical results of the flight accuracy

We made an assumption, that the data of the flight path accuracy could have some statistical relationships. In this section, statistical results are presented.

Firstly, statistical data of both flight modes was extracted from the collected log data. The most important statistical values are presented in Table 1. In Table 1 flight mode when the speed was constant and the sphere radius had different values is presented. In the Table 2 we have the same data for the flight mode when the sphere radius was constant and the flight speed was a variable.

In Figure 7 we see a visualization how does the average flight path error change in respect to different sphere radiuses. It is clear that we would get approximate average error of 0 m with the radius of 27.5 m. In addition, it looks that the average error decreases when we increase the sphere radius and this tendency last until the optimal

Table 1. Statistical values of the flight path error for different sphere radius

	r = 5 m	r = 10 m	r = 15 m	r = 20 m	r = 25 m	r = 30 m
$e_{p_{\max}}$, [m]	4.19	8.61	12.88	17.04	21.51	25.89
$e_{p_{\min}}$, [m]	−22.18	−17.93	−12.90	−9.04	−4.69	−2.27
$\bar{e}=\frac{\sum e_i}{n}$, [m]	−3.42	−2.78	−1.99	−1.25	−0.47	−0.32
$\sigma=\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(e_i-\bar{e}\right)^2}$, [m]	5.14	4.23	3.26	2.65	2.52	2.96
$\left[\bar{e}-\sigma;\bar{e}+\sigma\right]$, [m]	[−8.57; 1.72]	[−7.02; 1.45]	[−5.25; 1.26]	[−3.90; 1.39]	[−3.00; 2.05]	[−2.64; 3.28]

Table 2. Statistical values of the flight path error for different airspeeds

	V = 5 m/s	V = 10 m/s	V = 15 m/s	V = 20 m/s	V = 25 m/s	V = 30 m/s
$e_{p_{\max}}$, [m]	8.60	8.62	8.73	9.11	9.77	8.34
$e_{p_{\min}}$, [m]	−0.82	−17.94	−46.85	−95.74	−151.52	−212.71
$\bar{e}=\frac{\sum e_i}{n}$, [m]	−0.03	−2.78	−9.30	−25.85	−52.21	−56.48
$\sigma=\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(e_i-\bar{e}\right)^2}$, [m]	0.58	4.24	12.89	30.96	53.16	72.59
$\left[\bar{e}-\sigma;\bar{e}+\sigma\right]$, [m]	[−0.61; 0.55]	[−7.02; 1.46]	[−22.19; 3.59]	[−56.81; 5.10]	[−105.37; 0.95]	[−129.07; 16.11]

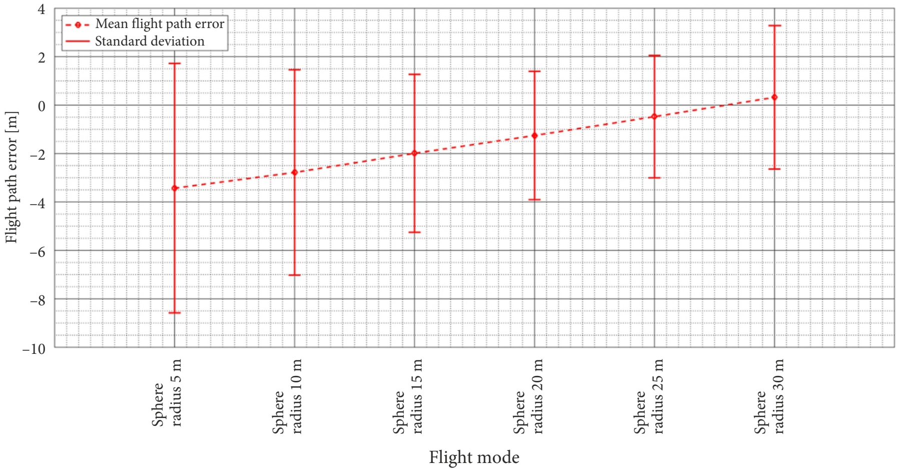

Figure 7. Average flight path error values and standard deviation ranges for different radius of the sphere

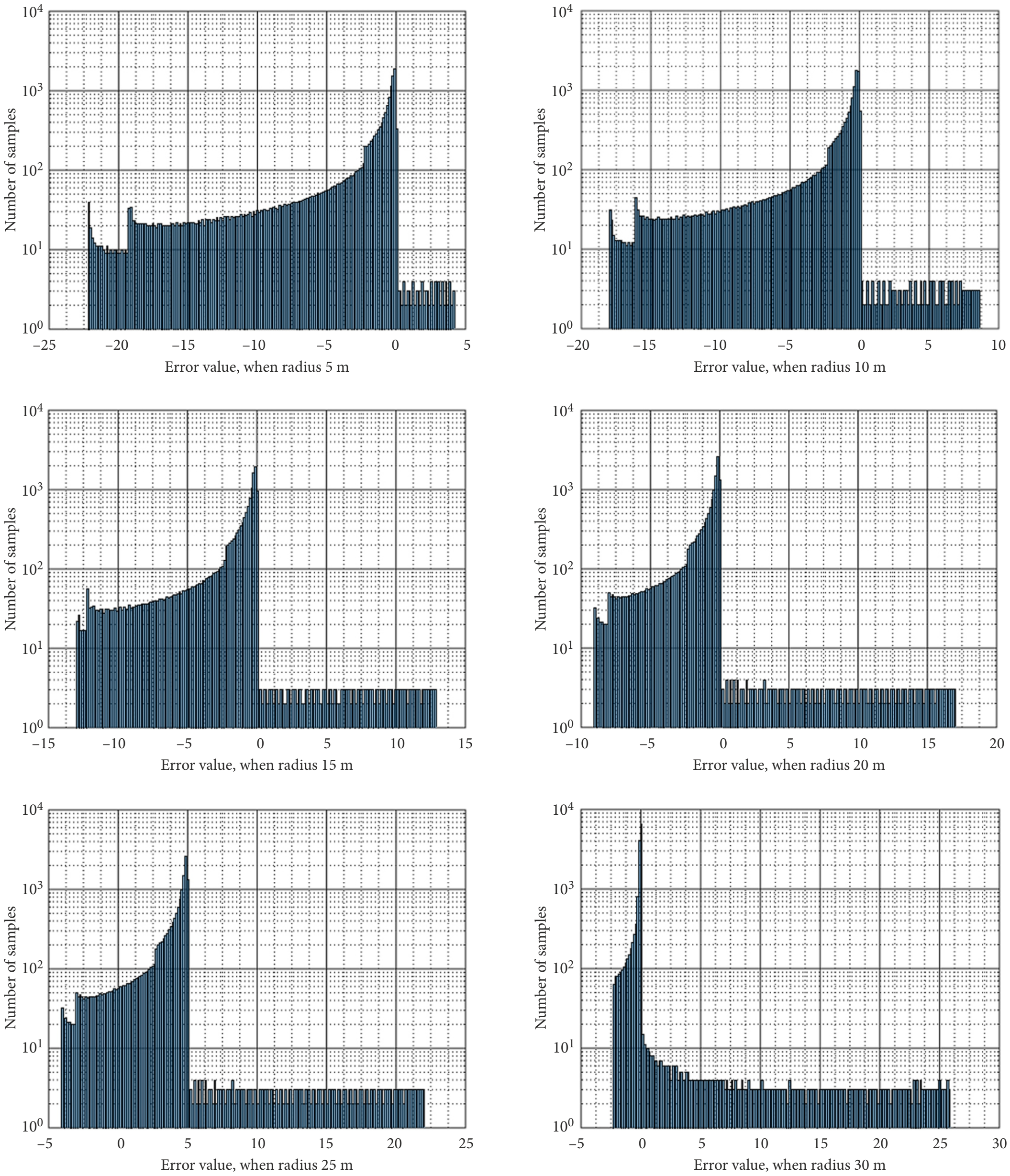

Figure 8. Flight path error distributions when different sphere radius is used

sphere radius value. Later flight path error starts increasing, and the average flight path error changes from negative values to positive values, which represent flight path error outside the trajectory. All tendency looks like linear function.

From the same Figure 7, we could estimate that if the sphere radius is being increased the standard deviation range of the flight path error decrease. From the Figure 7, we can identify, that the minimum standard deviation range will appear slightly before when the value of the average flight path error of 0 m is reached.

From Figure 8, we could find that the shape of the flight path error distribution is like the one-sided Gaussian. However, there are some relatively small number of samples on opposite side of this shape and it has more like an even distribution. Furthermore, we can see that with the increasing value of the sphere radius we get more compact flight path error distribution shape, which is a good result.

In Figures 9 and 10 we have the visualization of the average flight path error change in respect to different flight speed when the constant sphere radius of 10 m was used. In this case, it becomes clear that the average flight path error increases with increasing flight speed. This tendency has nonlinear properties and looks like quadratic function. Besides that, we could estimate that the standard deviation range values of the flight path error increase and also has some quadratic function tendency. These results prove that some increase in an airspeed gives even bigger increase in a flight path error.

Shapes of sample distribution for this case is provided in Figure 10. With small speed values tendency of the distribution shape is approximately the same as in the results provided in Figure 8. However, for higher speeds we can identify the tendency to get some additional spikes near the biggest error values. This tendency means that the biggest flight path error values have some average tendency to appear and it is not the smallest one. As well, for bigger flight speed we get wider flight path error distribution ranges, which is not good if we want to keep our flight as accurate as possible.

4. Single turn flight path error values

After the previous experiment, it is still not clear how accurate flight is near the turn when the UAV needs to make the transition from one flight leg to another. On the other hand, only one turn angle has been checked. That turn angle was of 120°.

In this section, respectively additional results are provided. As we might expect different overshoot or undershoot values for different course change angles, we got the relationship of the flight path error dependency for various sphere radius sizes and course change angles. This relationship is provided in Figure 11.

In Figure 11, we see the tendency that with the increasing turn or course change angle value, firstly UAV flight path error increases until reaching the pike point. Later it starts to decrease. It is important to mention, that initial flight path error is the undershoot and this error is measured from the closest point on the flight mission trajectory line. If some turn or course change angle is reached, tendency changes from the undershoot into the overshoot and further increases with the increasing turn angle size. Besides that, every flight setup will have the optimal sphere radius value. This optimal values depends on the course change angle. As is presented in Figure 11, 5 m sphere is optimal for course change angle of 30°, 10 m – for 60°, 15 m – for 88°, etc.

All of the flight accuracy results could be later applied when making algorithm corrections in order to improve flight reliability.

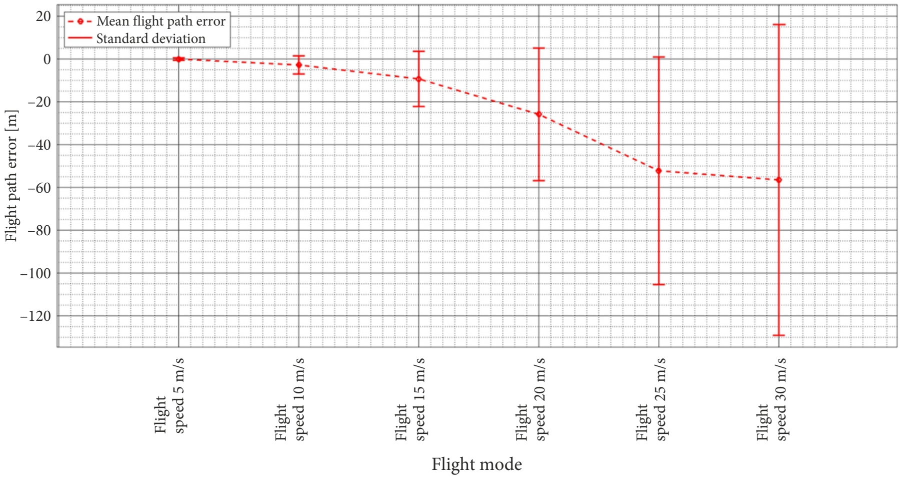

Figure 9. Average flight path error values and standard deviation ranges for different airspeeds

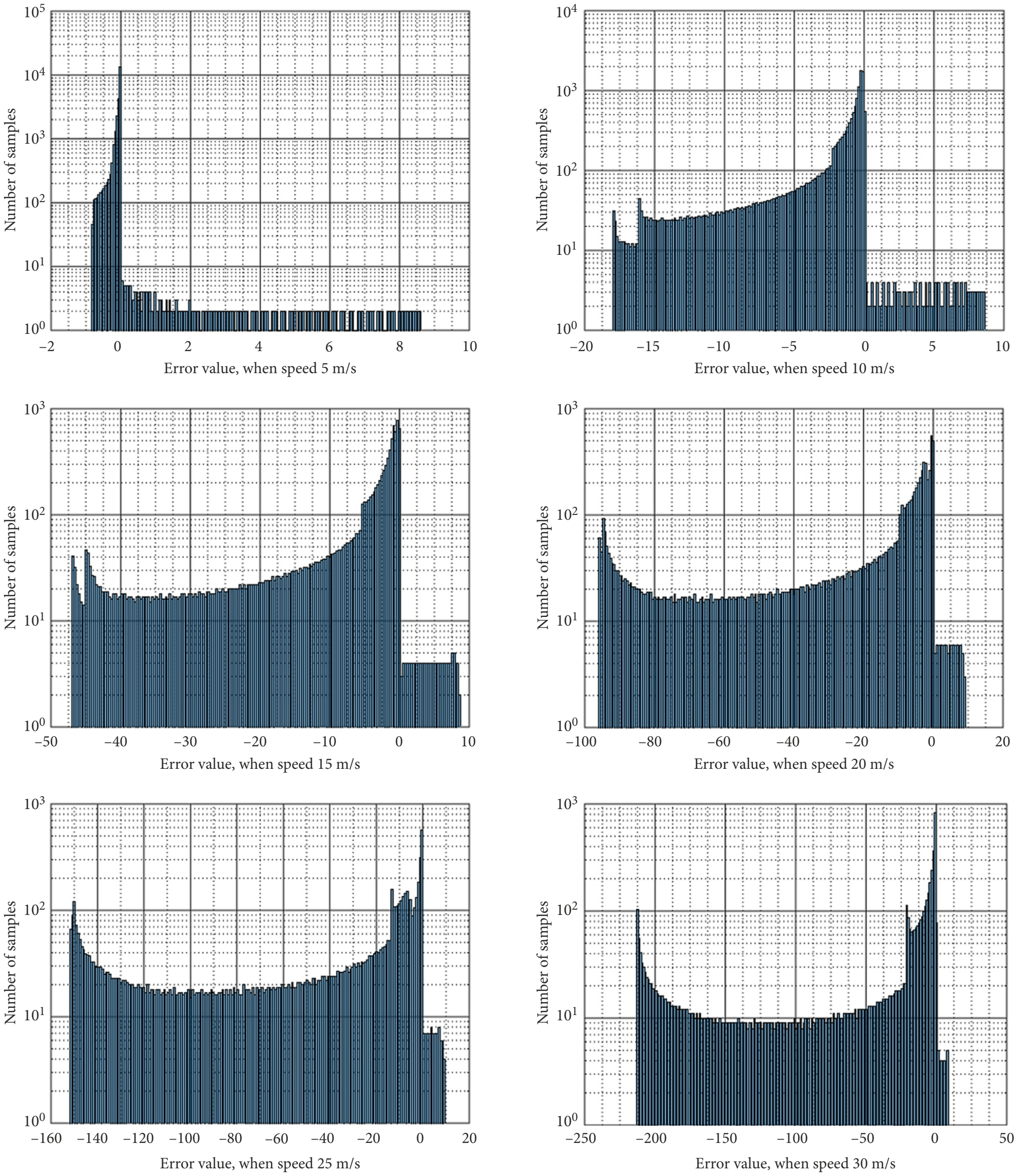

Figure 10. Flight path error distributions when different airspeed is used

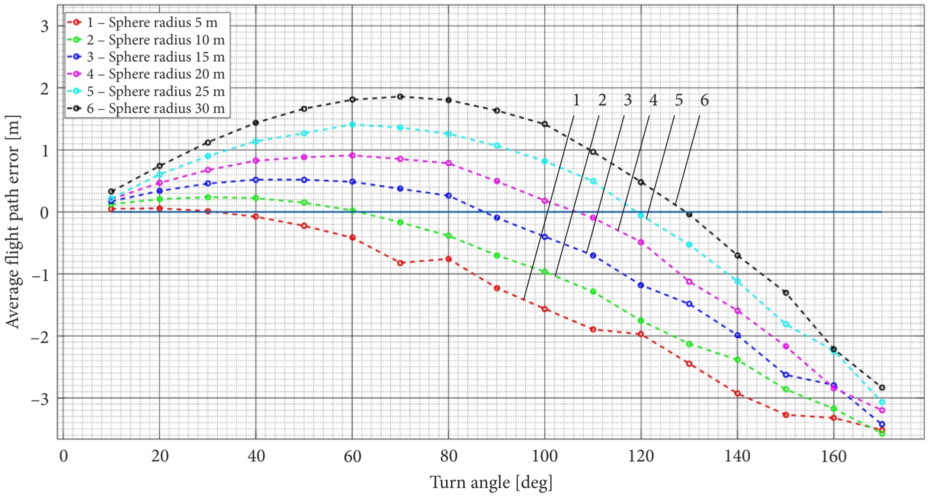

Figure 11. Flight path error for various course change angles when different sphere radiuses of classical waypoint switching algorithm is used

Conclusions

After these experiments we can make the conclusions that the flight path error would be the smallest if we would choose slowest allowed and possible to use flight speed and when the classical navigation algorithm is set-up with the optimal sphere radius (not too big or not too small). However, we have found that for every flight speed we need to look for the different sphere radius optimal value, which is a minimum mathematical extreme. All of these results are universal for all small UAVs, but only if we say that there is no aerodynamical influence.

If we would choose the optimal sphere radius for the chosen flight speed, the flight path will become like "filleted flight path", which is used for Dubins paths. It proves that the Dubins paths are much more accurate and has much smaller average error, which mainly occur inside of the predefined mission trajectory. On the other hand, classical algorithm has much bigger average error, which appear more outside of the predefined mission trajectory (this situation is also more dangerous regarding the outside obstacles). Besides that, classical algorithm is not failsafe as we saw with the example of 30 m/s flight speed when the 10 m sphere radius was used. In this case, UAV has not finished the mission by overshooting second waypoint.

All in all, the research must be extended and the Dubins paths accuracy should be evaluated.

References

Avellar, G. S. C.; Pereira, G. A. S.; Pimenta, L. C. A.; Iscold, P. 2015. Multi-UAV routing for area coverage and remote sensing with minimum time, *Sensors* 15(11): 27783–27803. https://doi.org/10.3390/s151127783

Beard, R. W.; Ferrin, J.; Humpherys, J. 2014. Fixed Wing UAV path following in wind with input constraints, *IEEE Transactions on Control Systems Technology* 22(6): 2103–2117. https://doi.org/10.1109/TCST.2014.2303787

Brezoescu, A.; Castillo, P.; Lozano, R. 2011. Straight-line path following in windy conditions, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 38(1/C22): 283–288. https://doi.org/10.5194/isprsarchives-XXXVIII-1-C22-283-2011

Brezoescu, A.; Espinoza, T.; Castillo, P.; Lozano, R. 2013. Adaptive trajectory following for a fixed-wing UAV in presence of crosswind, *Journal of Intelligent & Robotic Systems* 69(1–4): 257–271. https://doi.org/10.1007/s10846-012-9756-8

Dadkhah, N.; Mettler, B. 2012. Survey of motion planning literature in the presence of uncertainty: considerations for UAV guidance, *Journal of Intelligent & Robotic Systems* 65(1–4): 233–246. https://doi.org/10.1007/s10846-011-9642-9

Kothari, M.; Postlethwaite, I.; Gu, D.-W. 2014. UAV path following in windy urban environments, *Journal of Intelligent & Robotic Systems* 74(3–4): 1013–1028. https://doi.org/10.1007/s10846-013-9873-z

Li, W.; Chen, W.; Wang, C.; Liu, M. Ge. Y.; Song, Q. 2015. A 3D path planning approach for quadrotor UAV navigation, in *2015 IEEE International Conference on Information and Automation*, 8–10 August 2015, Lijiang, China, 2481–2486. https://doi.org/10.1109/ICInfA.2015.7279703

Nelson, D. R.; Barber, D. B.; McLain, T. W.; Beard, R. W. 2006. Vector field path following for small unmanned air vehicles, in *2006 American Control Conference*, 14–16 June 2006, Minneapolis, MN, USA, 5788–5794. https://doi.org/10.1109/ACC.2006.1657648

Owen, M.; Beard, R. W.; Mclain, T. W. 2015. Implementing Dubins airplane paths on fixed-wing UAVs, in K. Valavanis, G. Vachtsevanos (Eds.). *Handbook of Unmanned Aerial Vehicles*, 1677–1701. https://doi.org/10.1007/978-90-481-9707-1_120

Wang, T.; Le Maître, O. P.; Hoteit, I.; Knio, O. M. 2016. Path planning in uncertain flow fields using ensemble method, *Ocean Dynamics* 66(10): 1231–1251. https://doi.org/10.1007/s10236-016-0979-2

Yeol, J. W.; Hwang, Y.-W. 2016. Parametrization of nonlinear trajectory for time optimal 2D path planning for unmanned aerial vehicles, in *2016 2nd International Conference on Control, Automation and Robotics (ICCAR)*, 28–30 April 2016, Hong Kong, China, 335–339. https://doi.org/10.1109/ICCAR.2016.7486751

Zhong, W.; Yan, L. 2014. A target visiting path planning algorithm for the fixed-wing UAV in obstacle environment, in *2014 IEEE Chinese Guidance, Navigation and Control Conference (CGNCC)*, 8–10 August 2014, Yantai, China, 2774–2778. https://doi.org/10.1109/CGNCC.2014.7007603