

Autonomous Navigation and Landing Tasks for Fixed Wing Small Unmanned Aerial Vehicles

Sefer Kurnaz, Omer Çetin

Turkish Air Force Academy, ASTIN Yesilyurt, Istanbul, 34807 Turkey
s.kurnaz@hho.edu.tr, o.cetin@hho.edu.tr

Abstract: Autonomous control of UAVs has become a popular research topic in recent years. This paper is concerned with the flight of UAVs (Unmanned Aerial Vehicles) and proposes fuzzy logic based autonomous flight and landing system controllers. Three fuzzy logic modules are developed under the main navigation control system and three more for the autonomous landing control system to control of the altitude, the speed, and the position against the runway, through which the global position (latitude-longitude) of the air vehicle is controlled. A SID (Standard Instrument Departure) and TACAN (Tactical Air Navigation) approach is used and the performance of the fuzzy-based controllers is evaluated with time based diagrams under MATLAB's standard configuration and the Aerosim Aeronautical Simulation Block Set which provides a complete set of tools for rapid development of detailed 6 degree-of-freedom nonlinear generic manned/unmanned aerial vehicle models. The Aerosonde UAV model is used in the simulations in order to demonstrate the performance and the potential of the controllers. Additionally, some visual tools are deployed in order to get visual outputs that aid the designer in the evaluation of the controllers. Despite the simple design procedure, the simulated test flights indicate the capability of the approach in achieving the desired performance.

Keywords: UAV, Fuzzy Logic Controller, Autonomous Flight Control, Autonomous Landing System

1 Introduction

In the literature, it can be seen that the research interests in control and navigation of UAVs has increased tremendously in recent years. This may be due to the fact that UAVs increasingly find their way into military and law enforcement applications (e.g., reconnaissance, remote delivery of urgent equipment/material, resource assessment, environmental monitoring, battlefield monitoring, ordnance delivery, etc.). This trend will continue in the future, as UAVs are poised to replace the human-in-the-loop during dangerous missions. Civilian applications of UAVs are also envisioned such as crop dusting, geological surveying, search and rescue operations, etc.

One of the important endeavors in UAV related research is the completion of a mission completely autonomously, i.e. to fly without human support from take off to land on. For unmanned aerial vehicle systems to achieve full autonomy, smarter airplanes need to be developed. Full autonomy means performing takeoffs, autonomous waypoint navigation and, especially landings while the craft is hardest to control under computer control autonomously. The ground station control operator plans the mission and the target destination for reconnaissance and surveillance. UAV then takes off, reaches the target destination, completes the surveillance mission, and turns back to the base and lands on autonomously. Navigation is the topic which is most studied about. In literature, many different approaches can be seen related to the autonomous control of UAVs; some of the techniques proposed include fuzzy control [1], adaptive control [2], [3], neural networks [4], [5], genetic algorithms [6]. Capabilities of autopilot systems are important to successfully complete the mission of an UAV. A number of different autonomous capabilities may be required to be exhibited during a flight, like autonomous take off, navigation and autonomous landing.

In Section 2 of the paper the design of the navigation system with fuzzy controllers used for the autonomous control of the UAV is described and the autonomous landing model is defined by using the landing parameters given. Section 3 starts with the definition of the basic flight pattern for a UAV and then explain a sample mission plan which includes SID (Standard Instrument Departure) and TACAN (Tactical Air Navigation) procedures. A number of simulation studies are carried out with the fuzzy logic based autonomous navigation and landing system and some typical results are presented in Section 4. Finally the concluding remarks and some plans about future work are given in Section 5.

2 Fuzzy Logic-based System Design

As shown in Fig. 1, there basically are two separate computers used in an autonomous UAV. One of them is the flight computer and the other is the mission computer. UAV flight computer basically sets the control surfaces to the desired positions by managing the servo controllers in the defined flight envelope supplied by the UAV navigation computer as a command, reads sensors and communicates with the mission computer and also checks the other systems in the UAV (engine systems, cooling systems, etc.). The navigation computer and the landing system are a part of the mission computer. The mission computer carries out many other duties beside navigation, like payload control, communication with GCS, etc., the navigation computer is used for flying over a pattern which is designed before the flight or while flying in LOS (Line Of Sight). When GCS (Ground Control Station) gets control of UAV, the navigation computer goes into a passive state. During autonomous flight, the navigation computer gets the position values from

sensors (GPS receiver, altimeter, INS (Internal navigation System), etc.) and then matches these values (the current position) with the desired position values (the waypoint values). The navigation computer then determines the required maneuvers of the UAV to reach the goal position and sends these data to the flight computer to apply them to control surfaces.

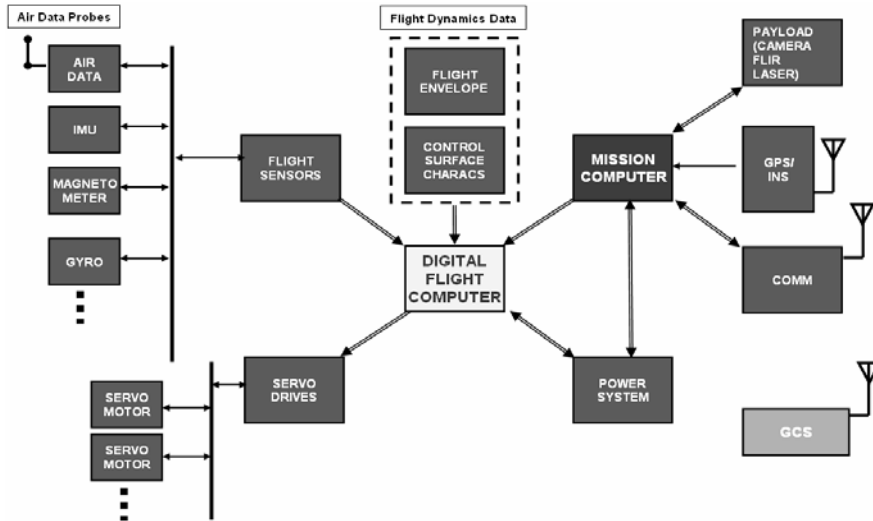


Figure 1

UAVs electronics architecture

Landing is one of the most critical parts of a flight, because, like in traditional aircrafts, UAVs aim to land at minimum air speed, consequently the stability conditions are severe and the maneuvering abilities are limited. A total of 65 Predators have crashed to date and thirty-six of the crashes were attributed to human error, and half of those occurred during landing [7]. The difficulties in taking off and landing arise mostly from instinctual factors, because pilots use their feelings in these periods of flight, such as feeling the ground rush and having peripheral vision. For a kite pilot, landing is a process which aims to see the wings inside of runway, but when a pilot is not in the cockpit, such feelings do not exist any longer. Because of these reasons, the manual control of an UAV from the ground is not a good alternative in the case of an emergency, especially during takeoff and landing.

The architecture used by the authors in [1] for their work on a Fuzzy Logic Based Navigation Control System (FLBNCS) forms the basis of the architecture for the Fuzzy Logic Based Autonomous Landing System (FLBANS). The mission computer has two main parts in this work, FLBNCS and FLBALS, as shown in Fig. 2. After getting the sensor values from the sensor interface, both FLBNCS and FLBANS calculate the desired attitude of the UAV attitudes which must be achieved by the flight computer. Then flight computer selects the correct

commands between the navigation c and the landing system commands. If UAV is in the final approach pattern it uses the landing systems commands, else it uses the navigation computer commands as inputs. The flight computer then calculates the control surfaces and the throttle positions by using its direct sensor inputs and the command inputs to reach the desired attitudes. The flow of this process can be seen in Fig. 2.

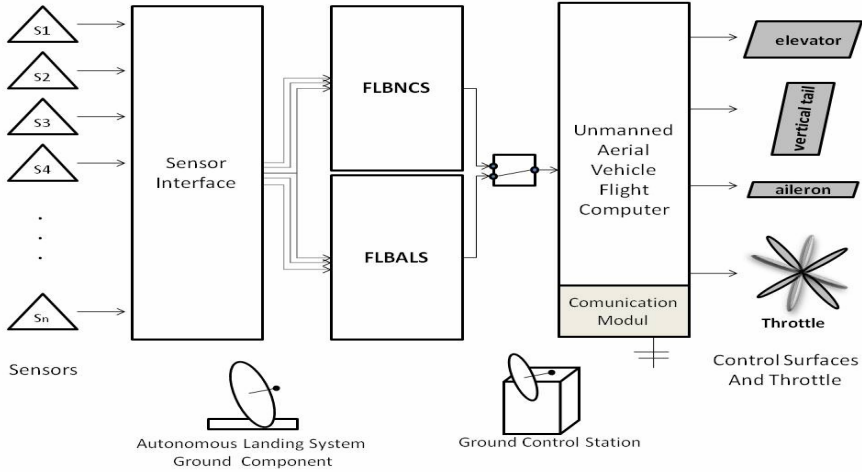


Figure 2
Autonomous System Design

The operation of the navigation computer proposed in this paper is fuzzy logic based. Basically, a fuzzy logic system consists of three main parts: the fuzzifier, the fuzzy inference engine and the defuzzifier. The fuzzifier maps a crisp input into some fuzzy sets. The fuzzy inference engine uses fuzzy IF-THEN rules from a rule base to reason for the fuzzy output. The output in fuzzy terms is converted back to a crisp value by the defuzzifier.

In this paper, Mamdani-type fuzzy rules are used to synthesize the fuzzy logic controllers, which adopt the following fuzzy IF-THEN rules:

$$R^{(i)} : \text{If } (x_1 \text{ is } X_1^i) \text{ AND } \dots \text{ AND } (x_n \text{ is } X_n^i) \text{ THEN } y_1 \text{ is } Y_1^i, \dots, y_k \text{ is } Y_k^i \quad (1)$$

where R^l is the l^{th} rule $x = (x_1, \dots, x_n)^T \in U$ and $y = (y_1, \dots, y_n)^T \in V$ are the input and output state linguistic variables of the controller respectively, $U, V \subset \mathfrak{R}^n$ are the universe of discourse of the input and output variables respectively, $(x_1, \dots, x_k)^T \subset U$ and $(y_1, \dots, y_k)^T \subset V$ are the labels in linguistic terms of input and output fuzzy sets, and n and k are the numbers of input and output states respectively.

We consider a multi-input and single-output (MISO) fuzzy logic controller ($k = 1$), which has singleton fuzzifier. Using triangular membership function, algebraic product for logical AND operation, product-sum inference and Centroid defuzzification method, the output of the fuzzy controller has the following form:

$$y_j = \frac{\sum_{l=1}^M \left(\prod_{i=1}^N \mu_{x_i^l}(x_i) \right) y_l}{\sum_{l=1}^M \prod_{i=1}^N \mu_{x_i^l}(x_i)} \quad (2)$$

Where N and M represent the number of input variables and total number of rules respectively. $\mu_{x_i^l}$ denote the membership function of the l^{th} input fuzzy set for the i^{th} input variable.

If the fuzzy controller types in literature are reviewed, it can be seen that there are two main classes of fuzzy controllers: one is position-type fuzzy controller which generates control input (u) from error (e) and error rate (Δe), and the other is velocity-type fuzzy logic controller. The former is called PD Fuzzy Logic Controller and the latter is called PI Fuzzy Logic Controller according to the characteristics of information that they process and system has two inputs, the error $e(t)$ and change of error $\Delta e(t)$, which are defined by

$$e(t) = y_{ref} - y \quad (3)$$

$$\Delta e(t) = e(t) - e(t-1) \quad (4)$$

Where y_{ref} and y denote the applied set point input and plant output, respectively. The output of the Fuzzy Logic Controller is the incremental change in the control signal $\Delta u(t)$. PD type fuzzy logic controller can be seen in Fig. 3. Then, the control signal is obtained by

$$\Delta u(t) = u(t-1) + \Delta u(t) \quad (5)$$

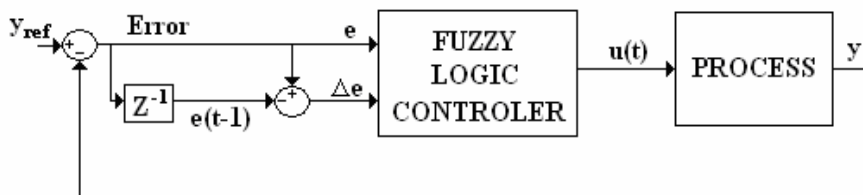


Figure 3

PD Type Fuzzy Logic Controller

As the second system in the architecture, FLBALS system uses the position inputs to calculate the exact location against the runway. It determines the error and calculates the corrective maneuvers by using three additional fuzzy logic subsystem blocks (in addition to the three blocks used for navigation). First fuzzy block is the lateral fuzzy logic controller which resolves the lateral errors in FLBALS. The second block is the vertical fuzzy logic controller which resolves the altitude errors and the last one is the speed fuzzy logic controller which tries to achieve the desired speed for the current conditions in FLBALS.

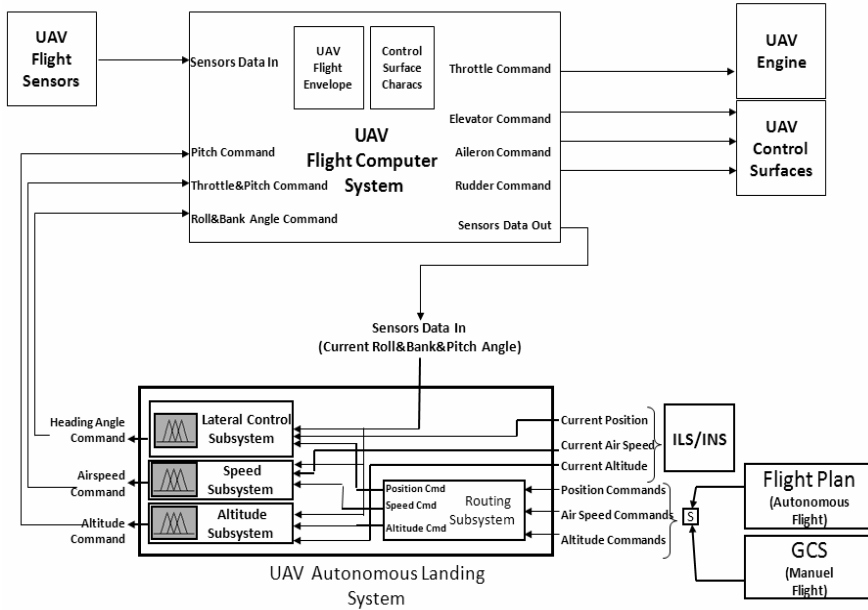


Figure 4

Fuzzy Logic Based Autonomous Landing System Design

The inputs to these fuzzy logic blocks are provided by different systems like ILS/INS and GPS [8, 9], laser based systems [10] or by vision based algorithms [11]. Other inputs of these blocks are the landing pattern flight plan or the manual commands issued by the Ground Control Station (GCS). The inputs of the fuzzy blocks can be seen in Fig. 4 and the surface diagrams of the blocks can be seen in Fig. 5.

3 Autonomous Flight Model

The test pattern used in this study includes a box pattern of Yalova Airport (LTBP) 18-26 Runway to show that UAV can fly autonomously a pattern which is designed for aircrafts if it has enough performance parameters. In classical SID and TACAN maps, a waypoint is defined with the radial angle and the distance between the VOR (VHF Omni-directional Radio Range) station and the waypoint. After transformation of the waypoints as GPS coordinates, UAV can apply SID departure and TACAN approach as a mission plan without a VOR receiver. In Fig. 6, the top and the side views of the test flight pattern is shown. This is a kind of box pattern. There are some important points which must be defined as GPS coordinates, like the initial approach point (IAP), the last turn point (LTP), the last approach point (LAP), the minimum altitude point (MIN) and the downwind turn point (DWTP). The UAV must reach the minimum altitude before the MIN point after takeoff. Then the UAV continues to the MIN point and starts to turn to reach DWTP. The particular set of these points that is used in the simulation studies is shown in Table 2. Each point of the pattern is represented by three values, the latitude and the longitude as the GPS position and the altitude as the vertical position.

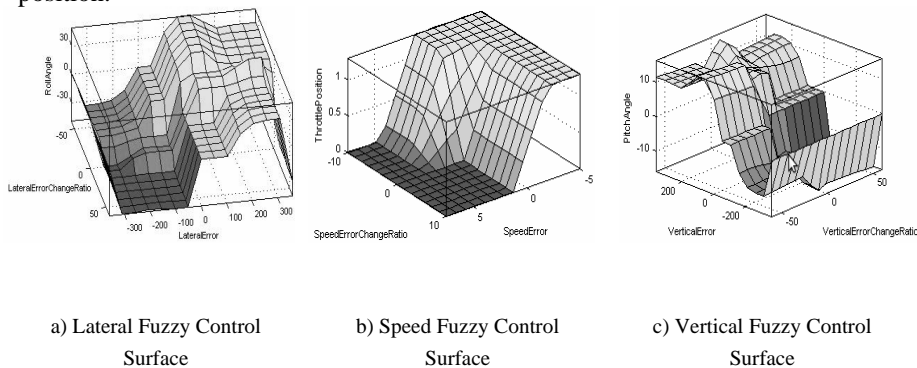


Figure 5
Control Surfaces

In this study, the UAV is considered to take off manually. Autonomous navigation starts when the UAV reaches the MIN point. It then reaches the way points in order and finishes when the UAV reaches the REP. The definition of each point includes speed, altitude and position (longitude and longitude coordinates) values.

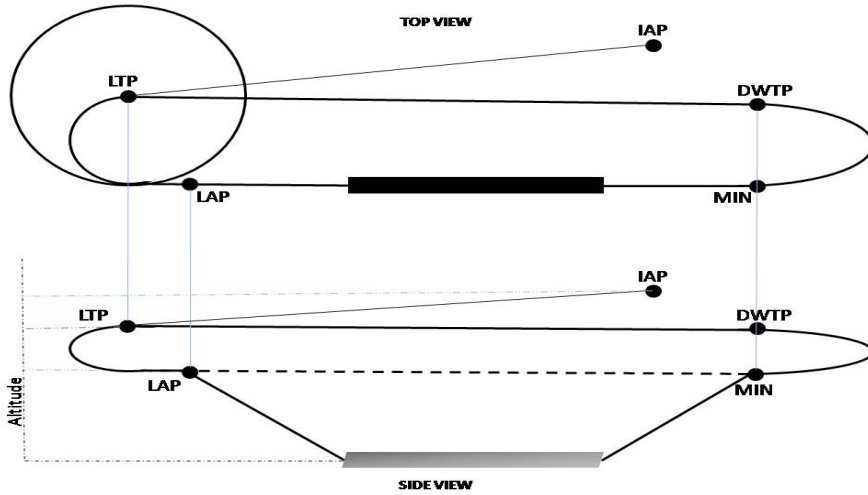


Figure 6
The test pattern of the UAV autonomous flight

To land, the aircraft must reach to the IAP and then it aims to reach the LTP and the LAP in order. After reaching the LAP, if airfield is not suitable for landing, it goes into a holding pattern. When the airfield becomes ready to land, the UAV completes the turn until the LAP is reached and goes into the final approach stage.

Table 1
Definitions of test pattern waypoints.

Point Name	Coordinate (GPS)	Altitude (feet)
Runway Starting Point (RSP)	N40 41 39.20 E29 22 34.81	6
Minimum Altitude Point (MIN)	N40 49 56.67 E29 22 34.81	1500
Down Wind Turn Point (DWTP)	N40 49 56.67 E28 18 43.00	1500
Initial Approach Point (IAP)	N40 41 50.00 E29 17 50.00	1700
Last Turn Point (LTP)	N40 32 26.55 E29 18 43.00	1500
Last Approach Point (LAP)	N40 32 26.55 E29 22 34.81	1200
Runway End Point (REP)	N40 40 56.54 E29 22 34.81	6

The values in Table 1 are contour a huge left side box pattern for Yalova (LTBP) Airfield 18-36 runway. First turn point in pattern is MIN point and it is 9 nm away from runway. Downwind leg is 3 nm away from runway and parallel to the 18-36 runways. Also LAP is 9 nm away from runway too. All the altitude values are mean sea level (MSL) in table.

To accomplish a successful landing, there are three main attributes which must be under control. First of them is the lateral position of the UAV with reference to the runway. As has already been stated, the goal is to touchdown on the lateral middle point of the runway like in Fig. 7. The second attribute is the vertical position, which is the AGL (above ground level) altitude of the UAV. It is a dynamic value since it changes according to the distance to the runway, but the usual glide path angle is 3 degree in aviation literature as in Fig. 7. The glide path angle is 3 degrees in nearly all the airfields in the world if there is no obstacle in this 30 path. The last main attribute is the speed. The speed value is a static value and it depends on the aircraft characteristics. The main aim is keep the desired speed value during the period of the final approach.

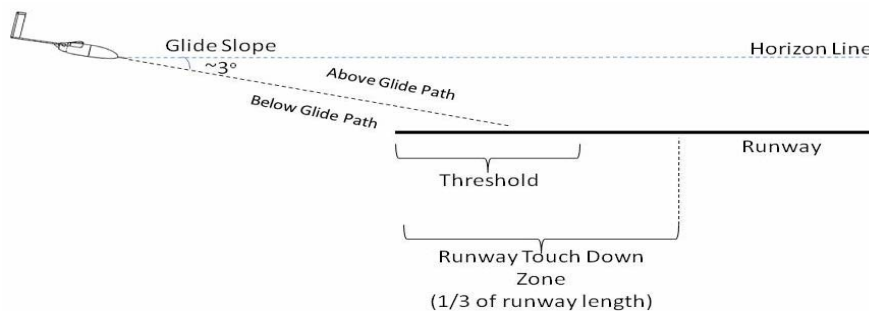


Figure 7

Desired Downward Velocity

In order to obtain the lateral position of the UAV with reference to the runway, different techniques can be used, like image processing [11] or radio based position calculators [8] and ILS (instrument landing systems) [9]. To measure the altitude and the speed of the UAV, laser altimeters and pito systems can be used respectively [10]. In this work, it is assumed that accurate measurements of these three parameters are available.

4 Simulation and Simulation Results

The performance of the proposed system is evaluated by simulating a number of test flights, using the standard configuration of MATLAB and the Aerosim Aeronautical Simulation Block Set [12], which provides a complete set of tools for rapid development of detailed 6 degree-of-freedom nonlinear generic manned/unmanned aerial vehicle models. As a test air vehicle, a model which is called Aerosonde UAV [13], shown in Fig. 8 together with its characteristics is utilized. The great flexibility of the Aerosonde, combined with a sophisticated command and control system, enables deployment and command from virtually any location.



Weight	27-30 lb,
Wing Span	10 ft
Engine	24 cc, 1.2 kw
Flight	Fully Autonomous / Base Command
Speed	18 – 32 m/s
Range	>1800 miles
Altitude Range	Up to 20,000 ft
Payload	Maximum 5 lb with full fuel

Figure 8

The Aerosonde and its specifications

In order to get visual outputs that aid the designer in the evaluation of the controllers, a number of aircraft instruments which are developed by using Delphi programming Active X components are deployed as shown in Fig. 9. Additionally, Flightgear open source flight simulator [14] is used to visualize the flight, like shown in Fig. 10. The details of these visual aids can be found in [15]. In order to be able to visualize the position of the UAV in GPS coordinate system, diagrams like the one shown in Fig. 15 are also plotted.



Figure 9

UAV Aircraft Instruments to get visual outputs of UAV parameters and mission planning



Figure 10

Visualization of landing by the use of FlightGear

The autonomous landing system test pattern can successfully be achieved by using the fuzzy logic based navigation computer system. In this work, the final approach period of landing pattern is handled by FLBALS. Final approach begins with the LAP and finishes at the touchdown point of the runway. The coordinates of these points and the elevations are given in Table 1 and navigation computer Simulink architecture can be seen in Fig. 11.

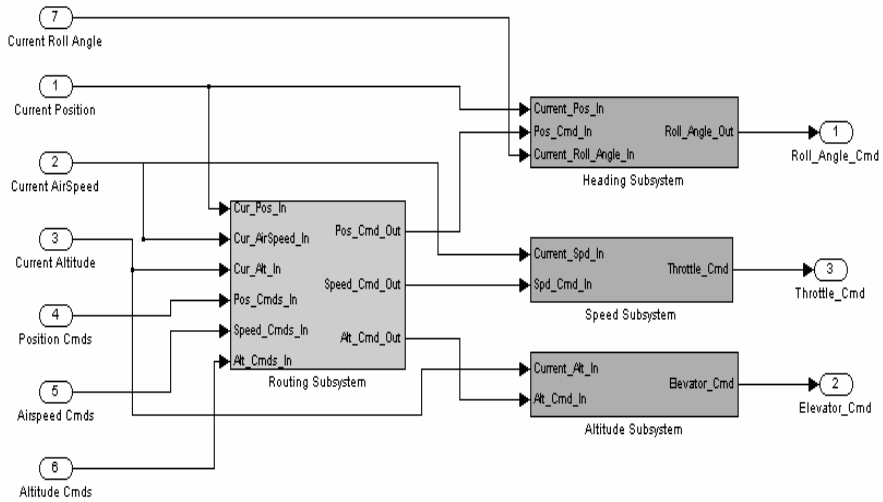


Figure 11
Navigation Computer in Simulink

5 Discussions on the Simulation Results

The UAV must reach exact altitude values during the flying pattern as shown in Fig. 6. There are some levels which depend on the distance from the runway. The dashed line shows the altitude command and the other one shows the current altitude at that simulation time. As we can see in Fig. 12, fuzzy logic-based autonomous landing system gets the desired altitude values in desired time. Also it manages not to sway too much from the 30 glide path angle throughout the pattern.

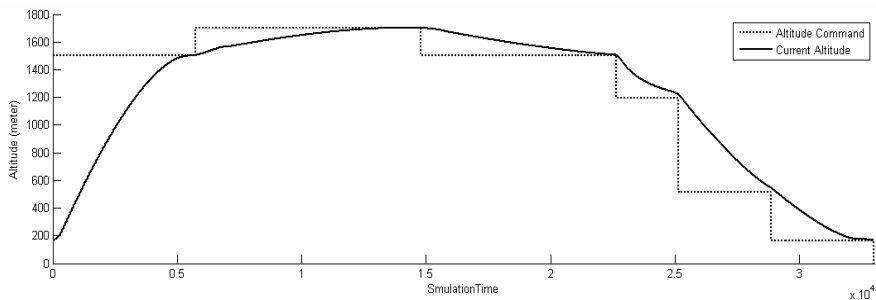


Figure 12
Current and Command Altitude – Simulation Time Diagram

In Fig. 13, UAV's current and command pitch degrees can be seen. Pitch commands produced by FLBNCS and FLBALS during simulation. UAV flight computer manage to hold desired pitch values during test flight. Pitch commands represents the desired values and dashed line shows the instant pitch value of UAV.

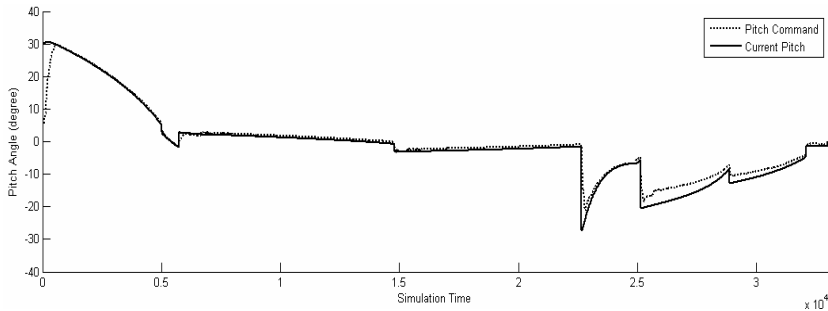


Figure 13

Current and Command Pitch Degree – Simulation Time Diagram

The last approach air speed of Aerosonde UAV is approximately 60 knot. The fuzzy logic-based autonomous landing system therefore tries to hold 60 knots during approach as shown in Fig. 14. The dashed line shows the desired air speed value and continues one indicates the current air speed of UAV in that simulation time.

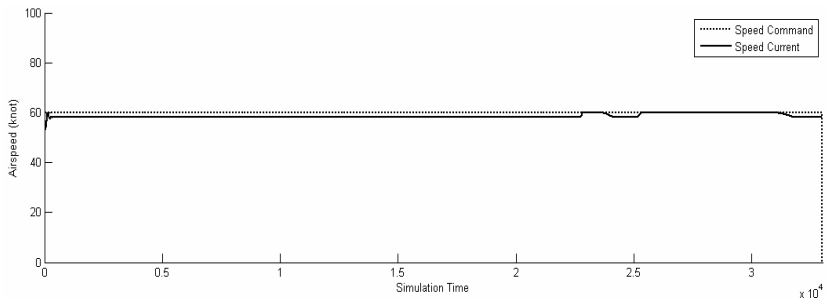


Figure 14

Current and Command Air Speed – Simulation Time Diagram

The vertical control and the air speed UAV are parameters that are related to each other. When the UAV pitches up, its speed decreases in parallel. The opposite of this is true also, when the UAV pitches down its speed increase. However, in this work there is no control relation between the air speed and the vertical control. The control of the air speed is provided by just using the throttle. The Aerosonde UAV is a kind of small fixed wing UAV. So this technique works to get airspeed of UAV under control. But major UAVs air speed must be controlled by using pitch angle and throttle together. So the architecture of control must be definitely different one.

In Fig. 15, the instant position of the UAV is depicted in the two dimensional space during the test pattern. The two dimensions of space are the GPS coordinate frames to show the exact position of the UAV. The other dimension is the altitude of UAV in meters, shown in Fig. 15. By looking at these diagrams, it can be stated that fuzzy logic based autonomous landing system manages to hold UAV in the correct position during the test pattern.

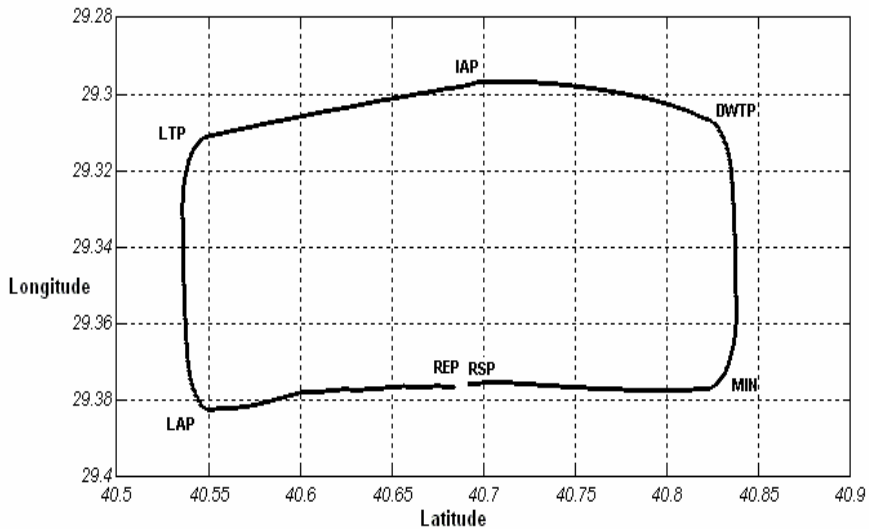


Figure 15
UAV Position in 2D GPS Diagram

As shown in Fig. 16, the UAV reaches the waypoints which have been defined in test pattern waypoints table (Table 1). After manual take off, the fuzzy logic based navigation computer system (FLBNCS) takes the control of the UAV until it reaches the IAP. After reaching the IAP, the UAV starts to be controlled by the fuzzy logic based autonomous landing system (FLBALS). The figure indicates that both of the fuzzy logic based systems successfully manage to navigate UAV in the test pattern.

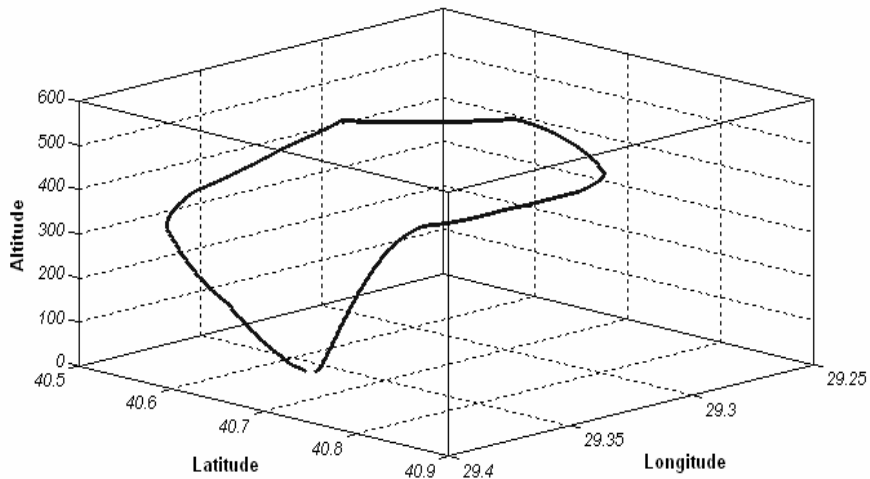


Figure 16
Current Position of UAV in Test Pattern Diagram

Conclusions

The mission computer design described in this paper provides autonomy to the UAV in all phases of a typical mission except take off. The main purpose of the navigation computer is to enable the UAVs to accomplish its mission autonomously, without any (or with minimal) input from the operator. In such a way the airplane is endowed with improved dynamic stability by the regulation of the flight parameters within limited ranges and with, at the same time, a successful tracking of the UAV mission plan.

The main contribution of this paper has been to demonstrate that the navigation and the landing tasks of small, fixed wing aerial vehicles can autonomously be achieved by fuzzy logic based systems. The simulation studies have shown adequate overall performance of the controllers. The main objective of this work is to keep the UAV in a pattern and also in a frame which is critical to hold the correct position during the final approach. This frame will be smaller when the UAV gets closer to the runway. The controllers must therefore show high performance against disruptive effects like wind. In our future work we will demonstrate the performance of fuzzy logic based autonomous landing system under disruptive effects.

Although there are many control law architectures, the classic PID control approach augmented with online gain scheduling provides the ideal mix of robustness and performance for typical aircraft dynamics. The stability and control loops can be tuned to provide the desired performance and robustness specifications by adjusting a set of autopilot parameters or gains. But this is done through linear analysis – the nonlinear aircraft model is linearized for a

representative set of flight conditions that cover the operating envelope of the aircraft. The linear dynamics of the closed-loop system (aircraft + autopilot) are analyzed in terms of stability and control responses (overshoot, settling time). By using fuzzy controllers, this difficult design process is avoided; nevertheless stable control and fast reaction time over conventional autonomous UAVs can be achieved as shown in this paper. The capability to do a dynamic planning of the desirable flight pattern is also important and this is done in this paper by using the current position of the moving UAV and the stationary target positions.

This paper also demonstrates that an UAV can apply a SID and TACAN approach without human control if it has enough performance like an aircraft. A series of SID for an airport can be planned just before the mission and the SID allowed by ATC (Air traffic controller) and meteorology can be applied autonomously. UAVs can to apply a TACAN approach too if they are properly instrumented with VOR and TACAN receivers. Without a VOR receiver, a SID can be applied by using GPS. Based on these attributes, fuzzy logic has been identified as a useful tool for developing controllers for UAVs so that they will be able to perform autonomous navigation.

In this paper, a fuzzy logic based autonomous flight controller for UAVs is proposed. The simulation studies presented verify that the UAV can follow the pre-defined trajectories despite the simplicity of the controllers. In future studies, the goals are to develop algorithms for a better tuning of membership functions which utilize the well known ANFIS (Adaptive Neuro Fuzzy Inference Systems) approach or possibly type-2 fuzzy sets. Autonomous take off will also be tried.

References

- [1] S. Kurnaz, O. Çetin, O. Kaynak, "Fuzzy Logic Based Approach to Design Flight Control and Navigation Tasks for Autonomous UAVs", *Journal of Intelligent and Robotic Systems*, v. 54, 2009, pp. 229-244
- [2] C. J. Schumacher, Rajeeva Kumar: Adaptive Control of UAVs in Close-coupled Formation Flight. *Proceedings of the American Control Conference*, Volume 2, 2000, pp. 849-853
- [3] B. Andrievsky, A. Fradkov: Combined Adaptive Autopilot for an UAV Flight Control. *Proceedings of the 2002 International Conference on Control Applications*, Volume 1, 2002, pp. 290-291
- [4] W. R. Dufrene, Jr.: Application of Artificial Intelligence Techniques in Uninhabited Aerial Vehicle Flight. *The 22nd Digital Avionics Systems Conference*, Vol. 2, 2003, pp. 8.C.3 - 8.1-6
- [5] Y. Li, N. Sundararajan, P. Sratchandran: Neuro-Controller Design for Nonlinear Fighter Aircraft Maneuver using Fully Tuned RBF Networks. *Automatica*, Vol. 37, 2001, pp. 1293-1301

- [6] J. A. Marin, R. Radtke, D. Innis, D. R. Barr, A. C. Schultz: Using a Genetic Algorithm to Develop Rules to Guide Unmanned Aerial Vehicles. Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, Vol. 1, 1999, pp. 1055-1060
- [7] J. Hodges, "Computer Co-pilots", C4ISR Journal, September 2009, pp. 30-32
- [8] K. L. Koester, W. Vaillancourt, "TALONS 95 GHz Radar Sensor for Autonomous Landing Guidance" Aerospace and Electronic Systems Magazine, IEEE Volume 7, Issue 7, July 1992, pp. 40-44
- [9] E. F. Roy, J. W. Davison, "Autonomous Landing Guidance Systems" Aerospace and Electronic Systems Magazine, IEEE Volume 1, Issue 5, May 1986, pp. 10-15
- [10] J. Geske, M. MacDougal, R. Stahl, J. Wagener, D. R. Snyder, "Miniature Laser Rangefinders and Laser Altimeters" Avionics, Fiber-Optics and Photonics Technology Conference, IEEE, Sept. 30-Oct. 2, 2008, pp. 53-54
- [11] Y. Fan, S. Haiqing, W. Hong, "A Vision-based Algorithm for Landing Unmanned Aerial Vehicles" Computer Science and Software Engineering, 2008 International Conference on Volume 1, Dec. 12-14, 2008, pp. 993-996
- [12] Unmanned Dynamics, Aerosim Aeronautical Simulation Block Set Version 1.2 User's Guide: <http://www.u-dynamics.com/aerosim/default.htm> [Last reached 12/27/2009]
- [13] FlightGear Open-Source Flight Simulator www.flightgear.org [Last reached 12/27/2009]
- [14] Global Robotic Observation System, Definition of Aerosonde UAV Specifications <http://www.aerosonde.com> [Last reached 12/27/2009]
- [15] O. Cetin, M. Ç. Kaplan, A. Aydın; "Fuzzy Logic-based Autonomous UAV Ground Control Station Simulator Design" Proceedings of 3rd National Defense Applications Modeling and Simulation Conference proceedings, USMOS METU, 2009