8 (2): 96-105 (2024)

Journal of Aviation

https://dergipark.org.tr/en/pub/jav

e-ISSN 2587-1676

Redundancy in Automatic Flight Control System Design for A General Purpose Helicopter

Mutullah Eşer[1*], Berk Akın Yıldız[2], Umut Türe[3]

[1*]Ankara University, Department of Electrical and Electronics Engineering, Ankara, Türkiye. (eserm@ankara.edu.tr)
[2] Turkish Aerospace Industries, Department of Autopilot Systems, Ankara, Türkiye. (berkakin.yildiz@tai.com.tr)
[3] Turkish Aerospace Industries, Department of Autopilot Systems, Ankara, Türkiye. (umut.ture@tai.com.tr)

Article Info

Received: 15 April 2024
Revised: 05 June 2024
Accepted: 10 June 2024
Published Online: 25 June 2024

Keywords:
Automatic flight control system
Redundancy management
Sensor fault detection
Functional redundancy
Direct redundancy

Corresponding Author: *Berk Akın Yıldız*

RESEARCH ARTICLE

https://doi.org/10.30518/jav.1468684

Abstract

Redundancy is a popular concept among autopilot design engineers due to the importance of safety in aviation. The main goal of redundancy in autopilot design is to eliminate or minimize the loss of a system in case of single point failures. In this study, the designed autopilot flight control system with redundancy for a general purpose helicopter is presented, along with the simulator and flight test results of the design. The designed system combines the elements of direct and functional redundancy to achieve a full redundant system. The simulator tests were done in the flight control simulator in Turkish Aerospace Industries (TAI) in Ankara, and the flight tests were done with the general purpose helicopter over Ankara Mürted airspace. The results of these tests verify the proposed design, ensuring the safety of the flight under different failure scenarios.

1. Introduction

In aviation, safety is vital, achieved through redundancy to boost reliability. Redundancy can be achieved in many ways and in the scope of this study, the redundancy in AFCS (Automatic Flight Control System), which is one of the fundamental systems of an aircraft, is presented. The main purpose of the AFCS is to reduce the workload of the pilot, to increase the fuel efficiency during operations, and increase safety of flight, and today, almost every modern aircraft is equipped with an AFCS (Padfield, 2018). Although certain designs can be made to prepare for failures, some cannot be foreseen, nor can they be prevented, regardless of the quality of the equipment used in the autopilot system. Similarly, it is impossible to take into account every scenario that can happen before, during or after the flight. Therefore, having a replica of a system in the aircraft can prevent single points of failure from resulting in major or catastrophic failures. Where major failure causes significant reduction in the functional capabilities of the aircraft, and catastrophic failure resulting in the total loss of the aircraft.

A system having n number of sub-systems that can execute the tasks of the main system is called an n-redundancy system (Schafer et al., 2018). In order to predict the malfunctions and the faults that can occur, it is important to understand the behavior of all the subsystems with the help of Fault Tree Analysis (FTA). A malfunction does not only affect the system it has occurred in, contrarily, it tends to spread. For this reason, the relation between the component and the rest of the system is as important as the relation between the duplicate components when it comes to redundancy. In case of a failure, the autopilot algorithm should not jeopardize the safety of the system and should operate as robust as possible. Thus, the main goal of the redundancy in the autopilot system is to minimize the damage done and prevent the loss of a system in case of single point failure.

In general, redundancy in an aircraft can fundamentally be achieved in two ways. First, by designing a system having multiple elements all working simultaneously, where each element can operate the system by itself if necessary, and second, by having a standby equipment that can take over the tasks of the faulty one. This can also be done for different levels of equipment, where either the system itself or a component of it can be backed up (Downer, 2009).

In this study, the work done on the hardware, also known as direct, and software, also known as functional, redundancy of AFCS in a utility helicopter is presented. The developed redundancy system has been tested and verified with different malfunction scenarios in HIL (Hardware in the Loop) laboratory tests and in flight tests. The first benefit of the proposed design is maintaining the functionality of the system in case of a malfunction by being an aggregate of several subsystems, and increasing safety by disengaging the autopilot upon detecting that the system can no longer function properly due to a critical fault. The second benefit is being a system where both direct and functional redundancy is utilized.

2. Automatic Flight Control System

Aircrafts have 6 degrees of freedom and each require a form of control during flight. This control can be done by a computer, just like it can be done by a human. Automatic Flight Control System (AFCS) is a system of hydraulic, mechanical and electrical components that can control the aircraft under certain circumstances (Padfield, 2018). AFCS has three main components: sensors, computers and actuators, which can be seen in Figure 1. Sensors are used to measure the relevant parameters and transmit the necessary information to the computer. Computers, which may contain electronic, mechanical or other forms of processing, convert the information coming from the sensors into the signals which are fed to the system output devices (Newman, 1994). The actuator converts the computer signals into a form which will result in the necessary helicopter control movement.

In our system, the AFCS is a conventional four axis limited authority flight control system with conventionally configured cyclic, collective and tail rotor pedal flight controls provided for pilot interface. AFCS architecture is based on two Flight Control Computers (FCC) and their peripherals which provide Fail/Safe operation. The system gives full control authority to the pilot, regardless whether the autopilot is engaged or disengaged. This in turn means that in case the autopilot is engaged and the pilot controls (cyclic, collective and pedals) are controlled by the autopilot, the pilot can still override these controls and suspend the autopilot commands by moving the controls manually.

In aircrafts, such AFCSs employ feedback control to increase the stability of the aircraft, reduce pilot workload, reduce the effect of transients such as gust, and increase flight comfort (Newman, 1994).

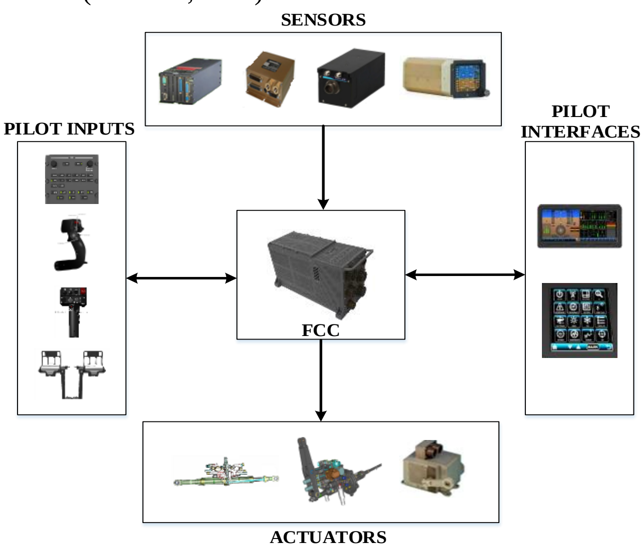

Figure 1. AFCS interface

The AFCS has two main modes, SAS and ATT, and several upper modes such as IAS (Indicated Airspeed Hold) and HDG (Heading Hold). FCC receives mode selections from flight control panel.

The SAS (Stability Augmentation System) improves the handling characteristics of the helicopter by damping the effects of the short-term external aircraft disturbances on pitch, roll and yaw.

The ATT (Attitude Hold) provides long-term attitude retention and stabilization for hands-off flying. The pilot can override the AFCS ATT mode at any time by taking over the controls manually.

The upper modes enable the pilot to focus on other tasks rather than keeping the helicopter at a certain altitude, speed, orientation etc. during flight and can only be engaged from ATT. The details about the designed AFCS architecture is given in section 3.1.

3. Redundancy in AFCS Design

The safety-critical systems in aircrafts have to comply with certain safety and architectural standards, and aircrafts with several of these systems are where redundancy is the most prominent. For this reason, redundancy is a popular concept among AFCS algorithm design and system engineers.

In this section, a brief summary of the works done in literature on this topic is given, and afterwards the steps in redundancy in AFCS design is mentioned.

Ahlström et. al. (2002) performed simulations on different fault handling strategies for distributed flight control systems. JAS 39 Gripen is used as a platform for the simulations. The redundancy for the system is realized by having a separate controller each control surface, for a total of 7 actuator nodes, each receiving data from all the sensor nodes, which are also duplicated. To handle the communication between the nodes, a scheduled communication system is implemented where each node can send message only in their predetermined assigned cycle, which prevents the nodes from sending values outside of their cycle in case of a failure. The three methods discussed are transferring the context of a non-faulty actuator node to the faulty one, letting the fault go through and be handled by the actuator nodes fulfilling replica determinism, and there being no recovery action, letting the disturbed context be corrected by subsequent sensor inputs and approach the non-faulty actuator nodes. The first two methods generated faulty output for only one operation cycle after which they were corrected by the mentioned strategies and give the same results as the non-faulty ones, however they can experience consensus problems. The latter method does not suffer from such problems but also perform worse, always having a small amount of error.

Amato et. al. (2006) proposed nonlinear observer based functional redundancy algorithm for detecting and isolating sensor faults on aircraft. The proposed algorithm decides to isolate the faulty signal if a fault has occurred.

Zhi-hong et. al. (2006) designed a dual-redundancy electrical and mechanical actuator system for improving safety of aircraft. Mechanical redundancy reduces two mechanical outputs into one without using conventional controllers. This way, the controller design is simplified, along with increasing the stability and the reliability of the system in case of a malfunction.

Schafer et. al. (2018) proposed a mixed-integer linear programming model to design the redundancy in an aircraft's door control system. This is a network system and has k-redundancy for the functions.

The main goal in designing an AFCS with redundancy is to keep the systems functioning properly and to not endanger the flight upon the malfunction of a system component. Usually, the design in made by expecting the system to fail in a way that is foreseen or predicted, however, there can be unexpected failures that make the design process harder.

The terms used commonly in designing autopilot systems with redundancy elements are summarized below:

Redundancy: The provision of additional or duplicate systems, equipment, etc. that function in case an operating part or system fails, as in a spacecraft. In other words, the existence of more than one means for accomplishing a given function. Each means of accomplishing the function need not necessarily be identical (Li & Shi, 2020).

Redundancy Management: The portion of the system logic and control (hardware or software) that detects and isolates failures in a fault-tolerant system; and reconfigures the system after the failure is detected and isolated so as to maintain the same or a reduced level of operation.

Active Redundancy: A type of redundancy whereby all redundant items are operating simultaneously, rather than being activated when needed.

Standby Redundancy: A type of redundancy whereby the alternative means of performing the function is inoperative until needed, and is activated upon failure of the primary means of performing the function.

Fail Operational: A design feature that enables a system to continue to operate despite the malfunction or failure of one or more components.

Fail Passive: Refers to the quality whereby the failed device or system ceases to create any active output.

Fail-Safe: Refers to the quality whereby the control device or system ceases to function, but the conditions or consequences resulting from the failure are not hazardous and do not preclude continued safe flight. The condition following failure may be completely passive, or it may involve driving to a predetermined non-active condition.

Failure: Describes the state of having failed. In dealing with fault-tolerant flight control systems, a failure occurs when a device within the system fails to function within prescribed limits without regard to the cause of the failure. Thus a failure may be: (a) any loss of function of any element within the control system; (b) loss of supply power to the system; (c) erroneous hardover conditions or loss of control intelligence at the signal input; or (d) any out-of-tolerance condition that exceeds normal operating limits.

Lane: An independent computational unit which includes processor and input-output capability inside a Flight Control Computer.

Redundancy increases the number of elements in both the software and hardware, which can result in unexpected interactions between the components. This result in a system that is more difficult and complex, to design and to understand (Zolghadri, 2000). Therefore, the autopilot should be designed keeping in mind these interactions, and necessary measures should be taken to prevent undesired responses stemming from the complexity. This results in additional designs in algorithm level to handle the redundant systems, such as the communication delay between the two FCCs, or the malfunction of a sensor.

In autopilot systems, there are two components to redundancy: direct and functional. To have an autopilot system with redundancy to work as expected, these two concepts must be designed to be in compliance with each other. If these do not comply with each other, the error risk increases in the system (Amato et al., 2006).

Direct redundancy is the replication of the physical systems such as FCCs, sensors, actuators, wiring, busses, and these components working in parallel with their counterparts. This way, if a component malfunctions, it can be detected and isolated from the system. However, functionality is not lost and the system keeps working as there is a back-up component. Therefore, the advantages of direct redundancy can said to be the isolation of the malfunction, and safety (Li & Shi, 2020). The disadvantage stems from the necessity for additional physical components, which result in bigger physical space requirement, higher cost, heavier overall component mass, additional wiring and increased system complexity.

Functional redundancy are the special algorithms that determine whether the system functions properly or not, and also make the logic decision enable correct operation. These algorithms are responsible of several critical tasks such as deciding on which of the redundant sensor data to use, making sure the FCCs are in the same state, checking whether the actuators have similar displacements as they should be receiving similar commands from the FCCs and so on.

3.1. Direct redundancy design

Direct redundancy in AFCS is physically having the duplicate of critical systems such as FCC, sensor, wiring and actuator. This hardware redundancy requires the presence of one or more sub-systems or equipment in the design. Direct redundancy can be implemented in two ways: hot (active) and cold (standby).

In standby redundancy, the redundant element is activated only in case of a failure in the main unit by means of a switching and the standby unit is often not energized until operation. Active redundancy requires that both main and redundant units to be operational during the whole mission period. In Figure 2, schematic Reliability Block Diagram presentations of two units (A and B) which are non-redundant, active redundant and standby redundant are given respectively.

According to the results from Fault Tree Analysis, the systems that should have direct redundancy are determined. These systems are processing units, both inside the FCCs and between them, sensors, AFCS related electrical power systems, control panels and actuators.

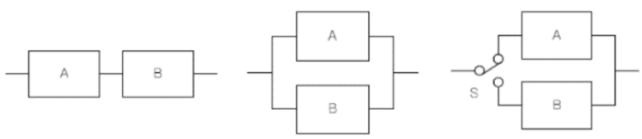

Figure 2. Types of direct redundancy

The designed AFCS system has two channels. Each channel consists of an FCC and a series actuator for each main rotor actuator, MRA, and tail rotor actuator, TRA.

Each FCC has dissimilar processors with different real-time operating systems. In each FCC, there are two independent CPU and IO modules and these CPU-IO pairs are called a lane. Both FCC lanes shall provide the same processing at the same time in order to ensure that both lanes provide the same outputs in absence of hardware failures or common mode failures. The check mechanism to make sure that both lanes generate the same outputs is given as a flowchart in Figure 3.

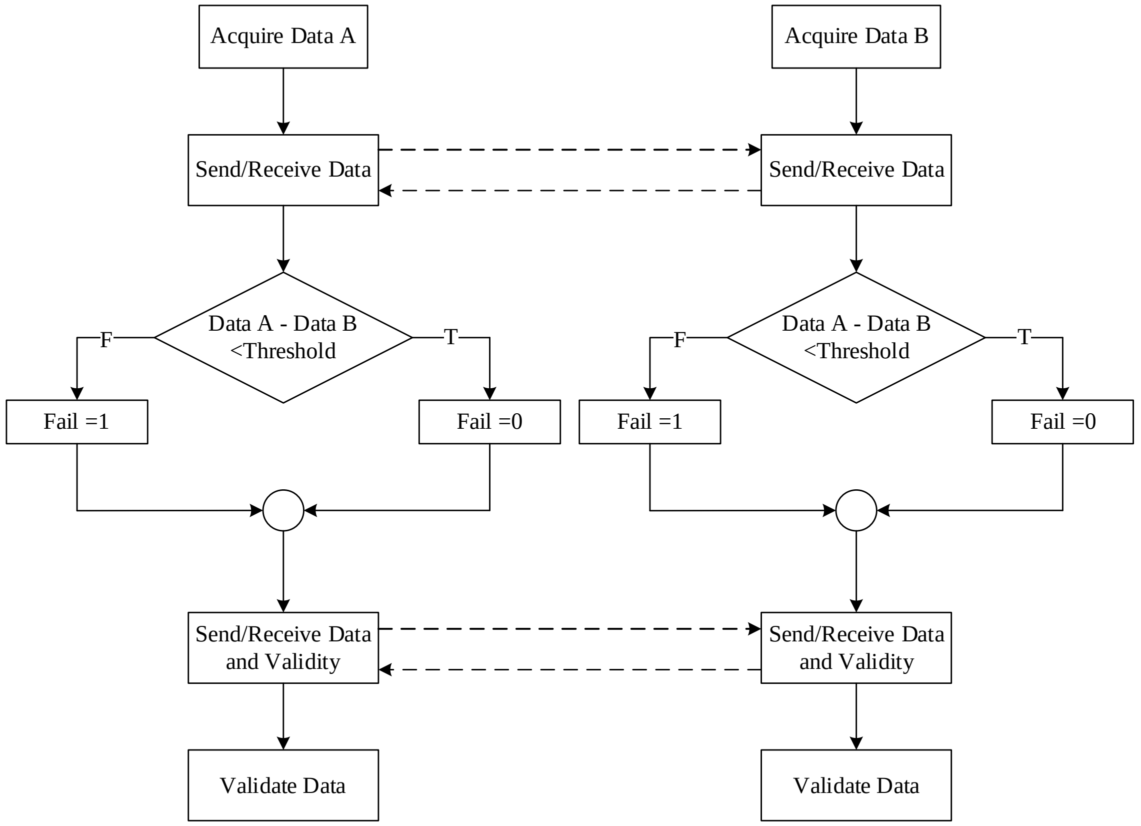

Figure 3. Lane check algorithm

The communication between these lanes is called LDL (Lane Data Link) which shares the input information and the control commands between them. The LDL is implemented from two serial "Point to Point Ethernet Busses" with transformer isolation and galvanic isolation to prevent the cascade of any failures within the one lane coupling into the other lane via the LDL. Data transferred across the link is monitored using "Cyclic Redundancy Checks" and ethernet protocol checks to detect for data bus and data transfer errors. The LDL Interface provides the capability to read the data of the other lane from the LDL RX buffer and to write the data to be sent to the other lane to the LDL TX buffer.

The communication between the FCCs where the mode information, CLAW (Control Law) commands, references and such are shared are maintained by the FDL (FCC Data Link). In case of FDL failure, the system cannot operate with dual FCC redundancy, but the helicopter can still fly with single FCC and autopilot. The FDL and LDL architectures are shown in Figure 4.

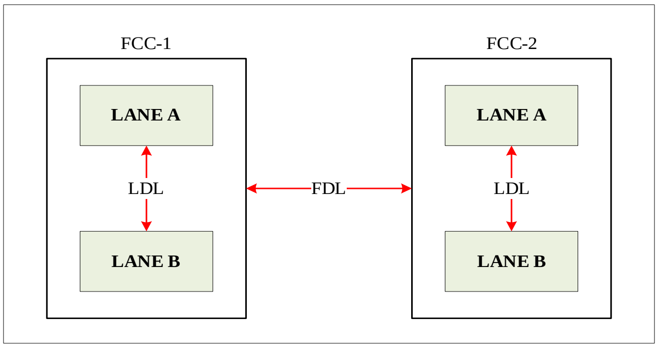

Figure 4. Communication architecture of AFCS

AFCS shall use redundant sensors, EGI (Embedded Global Positioning System Inertial Navigation System), ESIS (Electronic Standby Instrument System), ADU (Air Data Unit), RADALT (Radar Altimeter), for core functions. The backup sources the sensor data can be received from are given in the Table 1.

Table 1. AFCS sensor redundancy

Equipment	Backup sources
EGI1	EGI2, ESIS
EGI2	EGI1, ESIS
ESIS	EGI1, EGI2
ADU1	ADU2, ESIS
ADU2	ADU1, ESIS
RADALT	-

Both of the FCCs are connected to all the sensors to have data availability in case a bus fails. Within Channel-1, the EGI-1 is the primary source for basic data and EGI-2 is the redundant source for integrity validation of basic data, similarly, within Channel-2, the EGI-2 is the primary source for basic data and EGI-1 is the redundant source for integrity validation of basic data. EGI1 and EGI2 sensors are used as primary sensors whereas ESIS is used as an arbiter sensor in the event that the sensor values differ from each other. The validity of the sensor data that is sent to the FCCs and their logical controls are done in Sensor Selection module, which provides functional redundancy. In normal operating conditions, both of the FCCs function actively, receiving data from sensors and sending commands. The power busses that supply the FCCs are also duplicated. Both FCCs have primary and secondary power grids. The details are given in section 3.2.

The series actuators that work in series with the pilot commands and execute the generated AFCS command are also duplicated in each axis. Each Main Rotor Actuator (MRA) and Tail Rotor Actuator (TRA) has two independent SCAS actuators which are integrated on their own chassis to provide the full redundant actuator system. Two FCCs control SCAS actuators with Active/Active mechanism and control Trim actuators with Active/Standby mechanism. In Active-Active mechanism, each FCC is active and drives its own actuators on the MRAs and TRA. In case of one FCC failure, other FCC continues to drive its own actuators. When it comes to trim actuators, there are four trim actuators, one for each axis, and these are controlled by the trim master FCC, decided by the algorithms in each FCC. In Active/Standby mechanism, the FCCs communicate to each other to decide who will drive the Trim Actuator. One FCC becomes the master drives the actuator, while the other waits in Stand-By. In case of any failure in driving (master) FCC, stand-by FCC takes over the control. The way the series and the trim actuators are connected to each other and to the system are given in the Figure 5. Additionally, the hardware redundant AFCS system architectural design is given in Figure 7.

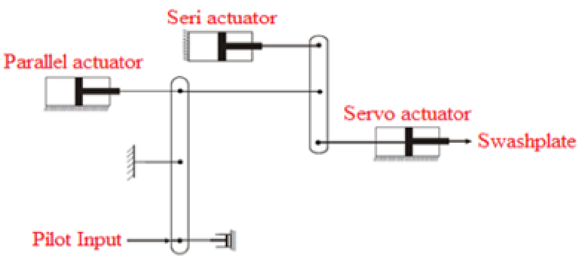

Figure 5. Actuator structure

The abbreviations used in the figure are: AHRS (Attitude and Heading Reference System), ADC (Air Data Computer), Radalt (Radar Altimeter), ESIS (Electronic Standby Instrument System), DCU (Data Concentrator Unit), IMD (Integrated Modular Display), and MRA (Main Rotor Actuator). Since the scope of the study is the design of a system with redundancy, the component details are not given.

3.2. Functional redundancy design

Functional redundancy means the use of logic and mathematical relations to ensure redundancy in the system.

The designed AFCS algorithm of the general purpose helicopter being worked on consists of 3 main parts which are System Govern, Logic and CLAW (Control Law), which is shown in Figure 6. A brief summary of these algorithms is given below, explaining their overall functionality, followed by the logical and mathematical modifications made on them for functional redundancy.

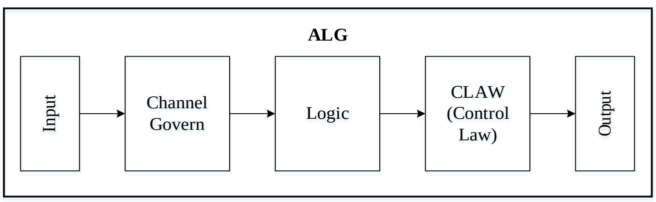

Figure 6. Algorithm layout

System govern: Manages system level algorithms. It allows or prevents the engagement of the autopilot by checking if all the necessary systems, such as the actuators and the busses, are working correctly and as expected. It also helps handle the information coming from the opposite FCC and decides which of the systems should be chosen as master for the upper modes.

Logic: Collects information from the sensors, pilot inputs and flight management system to determine which autopilot mode should be engaged and which reference values should be followed. Logic also handles the transition between the modes that can arise from speed transitions or pilot changing modes. These transitions are crucial for both safety and comfort of flight, they should be smooth and handled correctly. Therefore, the design should be approached with utmost care and the logic should be analyzed in terms of both safety and performance thoroughly. The following points should be considered in Logic design:

- Engagement conditions, determining when should a mode be engaged.
- Disengagement conditions, determining when should a mode be disengaged, or how can the pilot disengage a mode.
- Reference management, determining how the pilot can change the reference values, whether it be an attitude or upper mode reference such as airspeed.
- Limit values, determining the minimum and the maximum references considering the abilities and the limitations of the helicopter.

CLAW: Algorithms that calculate the commands that should be sent to the actuators to achieve the desired helicopter references. There are certain aspects of the control law algorithms that become more complicated with redundancy, increasing in complexity as the number of redundant systems increase. The commands should be handled in such a way that the CLAW algorithms in each FCC is guaranteed to generate similar commands, so that if one fails, there is no significant change in command to affect the helicopter. These transitions should be as smooth as possible as to not discomfort the pilot or to not strain the hydraulics of the actuator.

Basically, system govern determines the state of the systems, deciding whether the FCC is capable of controlling and letting, or preventing, the autopilot to engage. Logic on the other hand determines the state of the autopilot. It decides on which mode should be engaged, which functions should be active and feeds this information to CLAW. CLAW takes this information and, along with the input from sensors, does the necessary calculations to determine the necessary command to maneuver the helicopter or hold the attitude, whichever the pilot desires.

The redundancy starts at system govern module. The algorithm checks if both FCCs are suitable for operation and decides between one channel and two channel modes. This decision is made by ensuring that both channels are powered up and synchronized, the FDL is working correctly, and lastly, channels are operating with the same software and parameters.

In two channel mode, the FCCs communicate with each other through FDL with a transport delay taken into account in the design. The algorithms that use the information from the other FCC execute few cycles after the main logic to handle this delay, which are called second phase logic.

The Logic module has the sensor selection, second phase logic, reference management, command management and FD (Flight Director) mastership algorithms for redundancy and redundancy management.

Starting from the input, to ensure that the correct data is used in the control law calculations, a sensor selection algorithm is used. Designed architecture include dual navigation sensors and these sensors will feed flight displays, AFCS and other subsystems if required. Sensor selection algorithm used to find best usable/healthy source between multiple sensors.

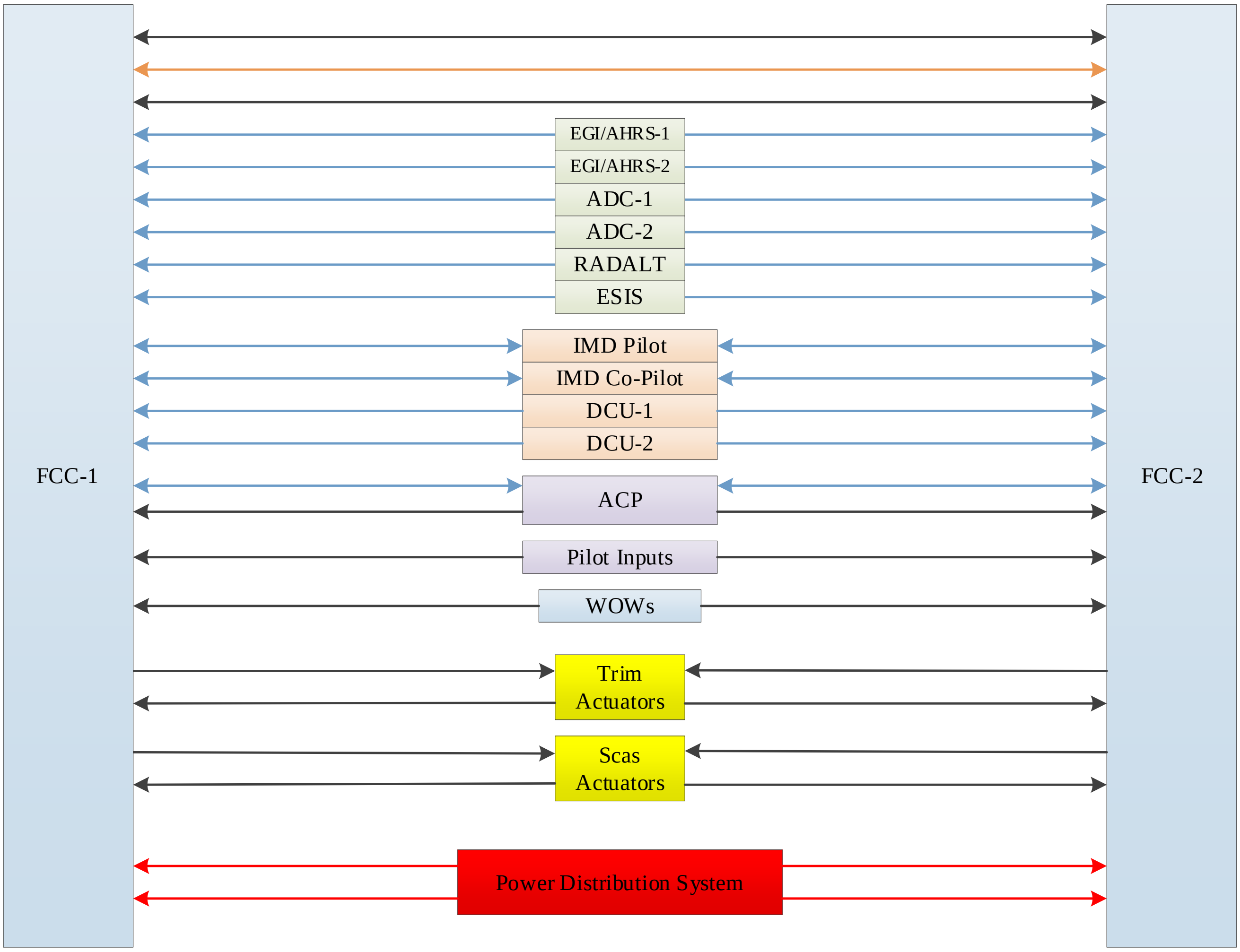

Figure 7. Hardware redundant AFCS system architecture

The Sensor Monitor and Selection unit operates independently from the state of the channel and it continuously gets the computed information to the potential user. The flow chart of the attitude source selection in the AFCS is given in Figure 8.

The AFCS provides the monitor function to determine the usability of attitude parameters fed by the two primary sensors EGI1 and EGI2 to support the selection of the source for the AFCS basic control functions. This algorithm compares the data from the dual redundant AHRS equipment and calculates the deviation of the data between the sensors. Difference is calculated by taking the difference between the sensors.

$$difference = |\mathrm{sensorX} - \mathrm{sensorY}| \quad (1)$$

Where, sensorX/sensorY pair refers to EGI1/EGI2, EGI1 /ESIS, and EGI2 / ESIS.

If the difference is less than a pre-determined threshold value, each FCC uses their assigned sensor's value e.g. FCC1 uses AHRS1 pitch angle value, FCC2 uses AHRS2 pitch angle value. However, if the difference exceeds the threshold, the electronic standby instrument system, ESIS, is used as an arbiter, as without another sensor, it wouldn't be possible to determine which of the sensors give the correct data. The sensor which gives a closer value to the arbiter is deemed as the correct value, and the data is sent to both FCCs. If neither value is within a certain range of ESIS, the sensors enter a fail state instead of sending wrong values.

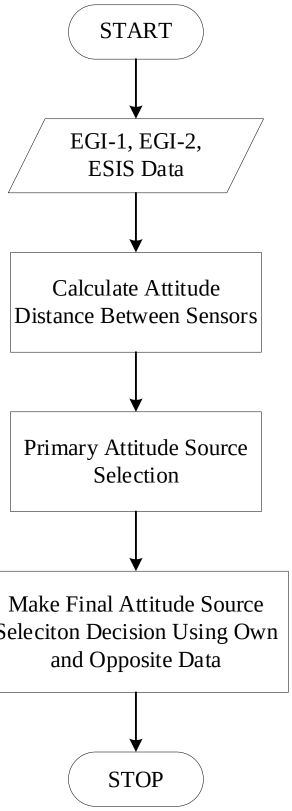

Figure 8. Attitude source selection flowchart

One of the sub-logics in second phase is a logic that compares the own reference values of both FCCs and averages them to obtain final reference (Equation 2). The own reference refers to the reference value computed before the second phase logic in each FCC separately. Afterwards this common reference value is used to calculate the error and multiplied with gain to calculate similar commands (Equation 3-4). However, because the sensor data can differ within a certain range of each other, the integrator commands could deviate between the two FCCs over time, despite the references being the same. To prevent this, the integrator data is sent between the computers and are equalized. Additionally, to ensure that the FCCs share the same states and generate similar commands, upon engaging the autopilot, if the other autopilot is already engaged, the filter states of the engaged autopilot is sent to and used as initial condition by the previously disengaged FCC. Otherwise, the filter outputs wouldn't match and different commands would be observed.

$$final\_reference = (\text{own\_FCC1\_ref} + \text{own\_FCC2\_ref})/2 \tag{2}$$

$$FCC1\_cmd = (\text{final\_reference} - \text{FCC1\_sensor}) \times \text{gain} \tag{3}$$

$$FCC2\_cmd = (\text{final\_reference} - \text{FCC2\_sensor}) \times \text{gain} \tag{4}$$

The status information and the decisions of the FCCs are also sent between them using FDL. By comparing the "health" of the FCCs, the more suitable one is chosen as the master to drive the trim resolver, as there is only one. Even so, the trim commands are compared to each other and averaged before being driven by the master. This way, even if one FCC loses communication with the trim actuators due to software or even hardware fault, the other FCC will take over the mastership and continue normal operation. By looking at a wider range of criteria, the healthier FCC is chosen as flight director, FD, master. Being FD master means that, despite the upper mode commands still being calculated by both FCCs, the final commands sent belong to the master FCC, as it is the healthier, and more reliable one. The FD master FCC still uses the information from the other FCC similar to basic mode operation, but solely uses its own sensor and forces its own FD command in the axis the upper mode is engaged. In case of a failure or degradation, the mastership is taken over by the other FCC and because the calculations are done on both FCCs regardless of the mastership, this transition is smooth and no severe command change is observed.

4. Test Results

The tests results were gathered both from simulator flights and flight tests. The flight tests were done with the Turkish Light Utility Helicopter GÖKBEY over Ankara Mürted airspace and the simulator flights were done in the System Integration Laboratory in Turkish Aerospace Industries, TAI, with Hardware-In-The-Loop tests. The simulator shown in Figure 9 has a physical replica of the helicopter cockpit along with the FCCs, displays, control panel and the trim actuators whereas the series actuators and the sensors are emulated.

Several tests were done to show different aspects and responses of the algorithm in terms of redundancy. In the first test case both FCCs are in Attitude mode. As seen in Figure 10, FCC1 is disengaged in simulator environment and re-engaged while changing only the heading reference and the response of FCC2 is observed. At t = 2287s, the FCC1 autopilot is disengaged manually, which can also happen due to a malfunction in flight, and the own reference value is directly taken as the current sensor value for FCC1. However, the final reference value does not change, as FCC2 now acts as master and forces its reference to FCC1. Upon re-engagement, the own reference of FCC1 is initialized to the existing reference value, which belongs to FCC2. This way, during both transitions, no change in the autopilot commands is observed as both FCCs make their own calculations and the disengagement does not affect the existing command, an example of hot (active) direct redundancy. Similarly, the newly engaged autopilot takes the existing values of the already active autopilot, this way the calculations converge and become similar, as desired and expected, much faster. Therefore, the transition is smooth and happens without any discomfort to the pilot and does not endanger the safety of flight.

Figure 9. TAI flight simulator

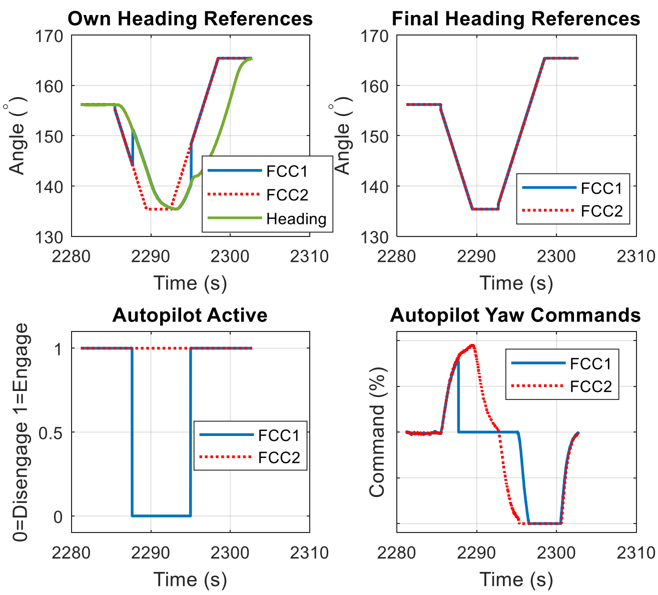

Figure 10. Reference management test

In the second test case, the FCC1 autopilot is disengaged during an agile maneuver where the autopilot generates high commands. The disengagement happens at t = 2373s, at which point the commands of FCC1 are forced to 0, as an extra safety precaution since it cannot move the actuators anyway. The FCC2 commands are not affected by this disengagement, as seen in Figure 11. Even though the FCCs communicate with each other and use information from one another, they can still operate independently an as standalone in case of failure. This way, as seen here, the malfunction of one does not affect the other. The rates are also examined as the rate of change in angular rates, angular acceleration, is felt as a force by the pilots. This change is expected to be minimal as to be smooth and not discomfort the pilot. In Figure 12, it can be seen that there is little no significant change in rates when one of the autopilots disengage.

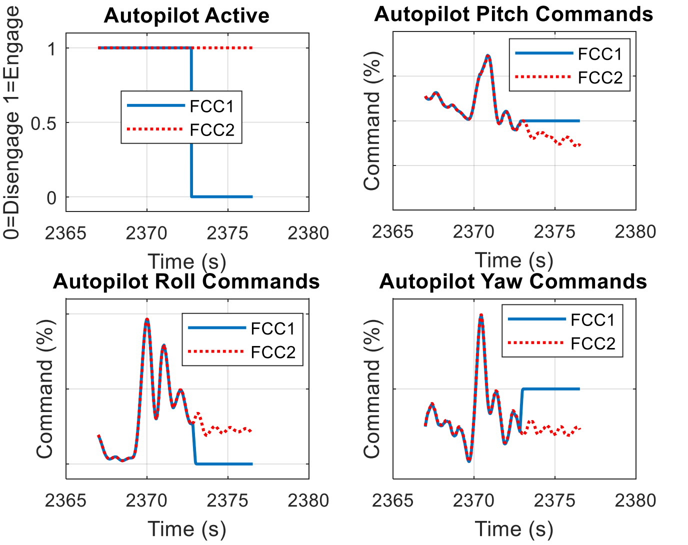

Figure 11. Command change during disengage

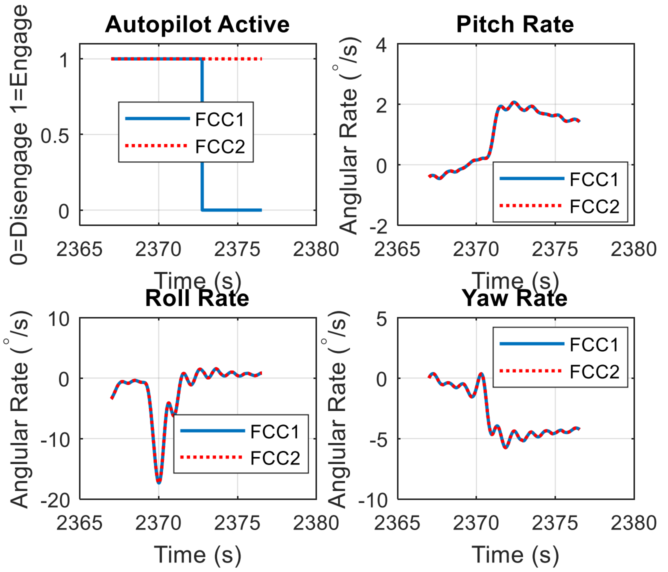

Figure 12. Rates during disengage

In the third test case, FD mastership transition test is made in the simulator. Initially, the IAS upper mode is engaged and FCC1 is FD master. Its commands are being sent to the actuators to control pitch to hold the desired airspeed. FCC2 is also doing its own calculations and sending commands to its own series actuators, however, because it is not the master, the command it sends is overridden by FCC1. This is done to ensure that the healthier FCC sends the upper mode commands, which is where the upper modes differ from SAS and ATT, since upper modes require more parameters from sensors. The mastership transition is made at t = 2577s, where FCC2 becomes the master after only a few operational cycles. At that instance, the FCC1 commands are forced to zero, its reference equated to current airspeed, and FCC2 now sends its own calculated commands to the actuators. Because it was constantly running its calculations, the command change and pitch rate change is almost nonexistent during the transition, which can be seen in Figure 13.

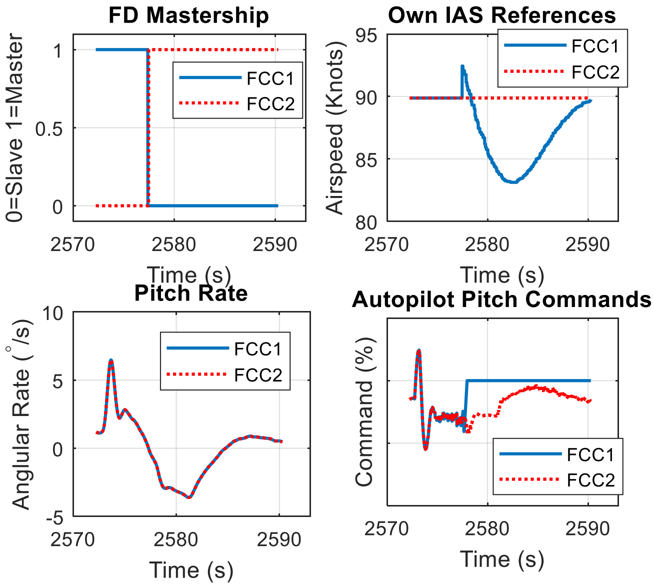

Figure 13. FD mastership transition

In the fourth test case, trim mastership transition is made in the simulator. Initially, FCC1 is the trim master. Despite both FCCs calculating the command to be sent to the trim actuator, FCC1 is the one that controls the actuator and since there is only one, unlike the series actuator where each actuator has its own actuator. At t = 2693s, the communication between the trim resolver and FCC1 is cut, which triggers a series of responses that transfer the trim mastership to FCC2 in a matter of few operational cycles. During this, the rates are examined, which as expected show no change in pitch and yaw, and only a slight change in roll rate at the transition instance, but does not change its overall behavior, as seen in Figure 14. It can be seen that the transition happens smooth with little to no disturbance. Additionally, Figure 15 shows the trim actuators position during fourth test case.

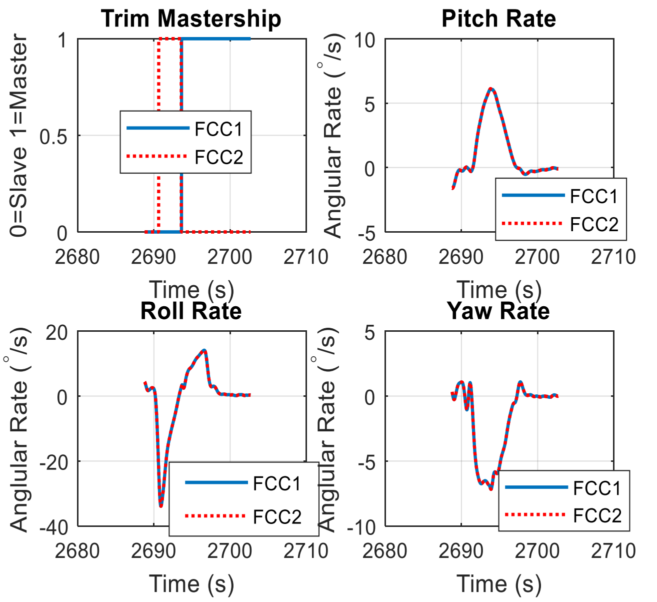

Figure 14. Trim mastership transition

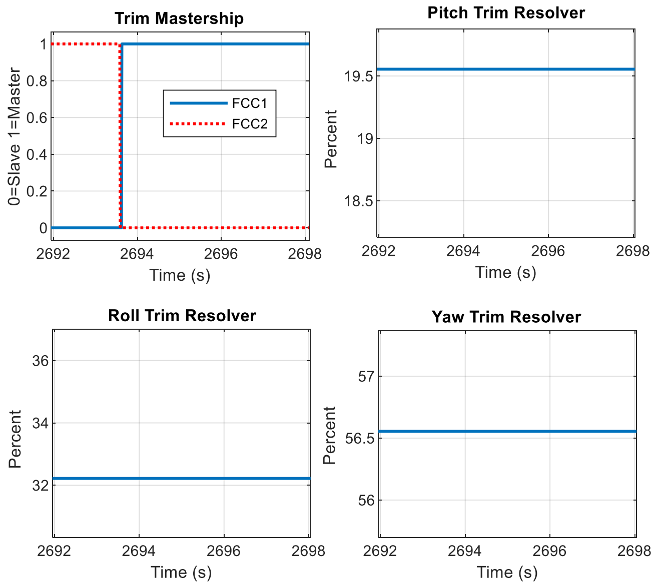

Figure 15. Trim actuator positions

In the fifth test case, the sensor redundancy is tested is simulator. Normally, FCC1 uses attitude data from EGI1, and in this case, the link between the sensor and FCC is cut at t = 2855s, at which point FCC1 starts to use the opposite side sensor, which is EGI2, as seen in Figure 16. During this, no loss of functionality occurs and both FCCs continue their operation as normal. At t = 2862s, the link between FCC1 and EGI2 is cut too, leaving FCC1 with no access to attitude data. The algorithm starts an internal clock and triggers sensor unavailable signal after some time. Once sensor unavail is triggered, FCC1 holds to last valid attitude data as its angles and no longer generates command. Both of these transitions happen without any disturbance as expected since FCC2 continues its normal operation and is not affected by the malfunction in FCC1. At t = 2910s, the links are restored and FCC1 return to its normal operations, receiving attitude data from the sensors. This transition also does not cause any disturbance in the system, to test that the FCC1 correctly receives the data and functions properly a small pilot input is given. From this test, it is seen that the loss of a sensor during flight does not cause a major effect and the flight can safely be continued thanks to redundancy of the sensors.

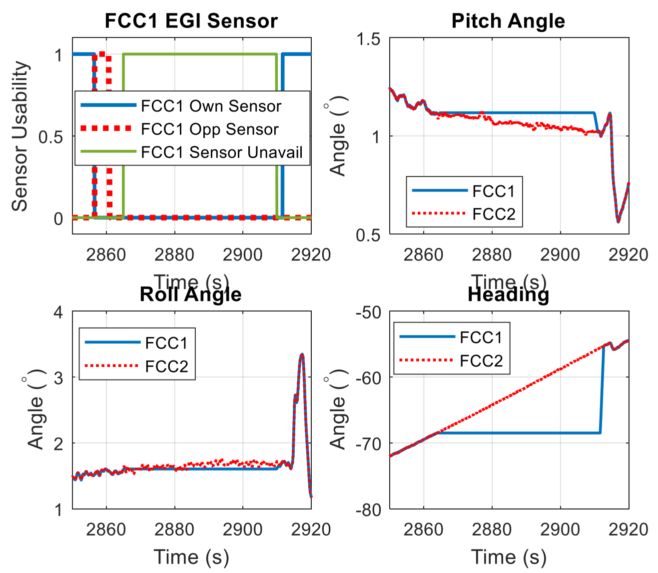

Figure 16. Sensor loss test

In the last test case, the second phase reference management is tested in flight. In Figure 17, it is seen that the EGI1 sensor of FCC1 and the EGI2 sensor of FCC2 give slightly different pitch angle data and therefore the references of these two FCCs are different, which would normally result in the two algorithms trying to hold the helicopter in two different pitch angles. However, these references are sent between the FCCs by the FDL and are used in second phase algorithm, taking the average of these two signals, to create final reference which is the same for both FCCs. This is a method of handling the sensor redundancy in the design, if left unattended, this case would decrease the overall performance of the helicopter and can lead to oscillatory behavior.

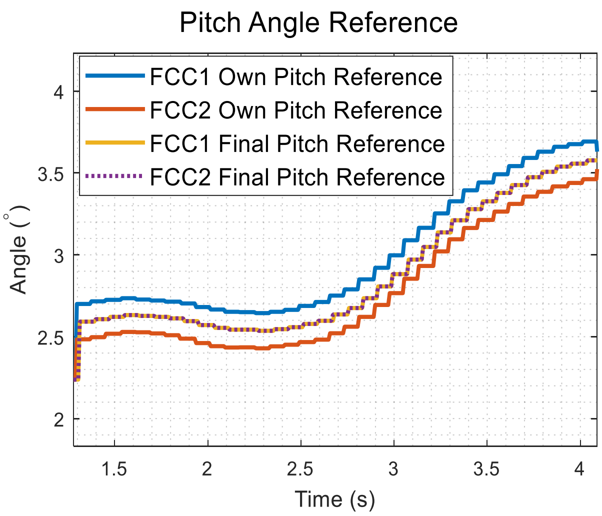

Figure 17. Reference averaging

5. Conclusion

The simulation and flight test results validate the proposed direct and functional redundancy design and meet the expected results. In case of a failure in the systems considered critical for the helicopter, the safety of the flight remains intact and the redundant subsystems can resume normal operation. Some of these cases include sensor failure, autopilot malfunction or mechanical failure such as trim failure, which are all tested and validated in the scope of this study. Additionally, several algorithms that are implemented to handle the dual FCC operation and its drawbacks are mentioned such as the second phase logic and mastership management. Despite the platform used in this study being a limited authority system, such measures are taken to ensure safety, which can further be improved in fly-by-wire flight control systems.

Ethical approval

Not applicable.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

Ahlstrom, K., Torin, J., Fersan, K., & Nobrant, P. (2002). Redundancy Management in Distributed Flight Control Systems: Experience & Simulations. The 21st Digital Avionics Systems Conference. Irvine.

Amato, F., Cosentino, C., Mattei, M., & Paviglianiti, G. (2006). A Direct/Functional Redundancy Scheme for Fault Detection and Isolation on an Aircraft. Aerospace Science and Technology, 10, 338-345.

Chen, J., & Patton, R. (1999). Robust Model-Based Fault Diagnosis for Dynamic Systems. The International Series on Asian Studies in Computer and Information Science.

Downer, J. (2009). When Failure is an Option:Redundancy, reliability and regulation in complex technical systems. The London School of Economics and Political Science, 53.

Eykeren, L., & Chu, P. (2011). Nonlinear Model-Based Fault Detection for a Hydraulic Actuator. AIAA Guidance, navigation, and control conference, (pp. 1-8). Portland.

Gobbo, D., Napolitano, M., Famouri, P., & Innocenti, M. (2001). Experimental Application of Extended Kalman Filtering for Sensor Validation. IEEE Transactions on Control Systems Technology, 9, 376-380.

Li, J., & Li, L. (2017). Reliable Control for Bilateral Teleoperation Systems with Actuator Faults Using Fuzzy Disturbance Observer. IET Control Theory Appl, 11, 446-455.

Li, W., & Shi, G. (2020). Redundancy Management Strategy for Electro-Hydraulic Actuators Based on Intelligent Algorithms. Advances in Mechanical Engineering, 12, 1-16.

Ljung, L., & Söderstrom, T. (1983). Theory and Practice of Recursive Identification. Cambridge.

Navatha, A., Bellad, K., & Hiremath, S. (2016). Dynamic Analysis of Electro Hydrostatic Actuation System. Procedia Technology, 1289-1296.

Newman, S. (1994). The Foundations of Helicopter Flight. London: Edward Arnold.

Padfield, G. (2018). Helicopter Flight Dynamics. Oxford: Blackwell Science.

Patton, R. (1997). Fault-Tolerant Control: The 1997 Situation. IFAC Symposium on Fault Detection, Supervision and Safety. Kingston Upon Hull.

Rehage, D., Carl, U., & Vahl, A. (2005). Redundancy Management of Fault Tolerant Aircraft System Architectures-Reliability Synthesis and Analysis of Degraded System States. Aerospace Science and Technology, 9, 337-347.

Schafer, L., Garcia, S., Mitschke, A., & Srithammavanh, V. (2018). Redundancy system design for an aircraft door management system. Computer and Operations Research, 94(1), 11-22.

Stefano, S., & Everline, C. (2014). The Use of Redundancy to Improve Reliability of Deep Space Missions Using an ASRG Power Source. 12th International Energy Conversion Engineering Conference. Cleveland.

Xiao, B., & Yin, S. (2018). An Intelligent Actuator Fault Reconstruction Scheme for Robotic Manipulators. IEEE Trans Cybern, 48, 639-647.

Zhang, Y., & Deng, A. (2015). Redundancy Rules Reduction in Rule-Based Knowledge Bases. 12th International Conference on Fuzzy Systems and Knowledge Discovery, (pp. 639-643). China.

Zhi-hong, J., Heng, Z., Qing-tang, F., & Ji-hong, Z. (2006). Study of High-Reliable Differential Dual-Redundancy EMA without Load-Averaging Controller for Aircraft. IMACS Multiconference on “Computational Engineering in Systems Applications” (CESA). Beijing.

Zolghadri, A. (2000). A Redundancy-Based Strategy for Safety Management in a Modern Civil Aircraft. Control Engineering Practice, 8, 545-554.

Cite this article: Eser, M., Yildiz, B.A., Ture, U. (2024). Redundancy in Automatic Flight Control System Design For A General Purpose Helicopter. Journal of Aviation, 8(2), 96-105.