

Flight control investigation using a 3D-printed radio-controlled demonstrator

Roger Larsson¹

¹*Dept. of Aeronautical Engineering, Saab Aeronautics, Linköping, Sweden
E-mail: roger.gunnar.larsson@saabgroup.com*

Abstract

In the Swedish national aeronautical research program, the project: Flight Control Subscale Flight Testing, it is investigating how to use 3D-printed flying demonstrators for testing of new and innovative flight control laws. The aim of this project is to show that it is possible to test new technologies, quickly and at low cost for aeronautical engineering purposes. This can lead to the possibility to explore many different ideas early on during the concept phase of an aircraft. The timing is right since Sweden is looking into designing the next generation fighter. But green civil aviation programs could benefit from this type of design initiative. The work being done in Sweden in the field of subscale flight testing have caught the interest from international groups and a NATO project has shown interest in using subscale flight to test different control law solutions. To design the control laws a simulation environment is developed that can used together with the same hardware as is installed in the flying demonstrator. Validation and verification can then be done in this simulation environment to test the control laws before incorporating these into the actual flying demonstrator. This makes it possible to secure a seamless and quick integration of the control laws before flight testing. First flight is in 2025, first without a control law implemented to see that the 3D-printed demonstrator behaves as expected. Later, flight with several other control laws will be done.

Keywords: Subscale Flight testing, Flight Control

1 Introduction

In the early aviation history, the pilot was the main part of the flight control system (FCS) compensating for poor stability characteristics (feedback) while manoeuvring the aircraft (feed forward). This worked for the limited envelope available at that time. As higher speeds and altitudes were achieved, the pilot's task became so advanced that augmentation systems became necessary to safely operate the aircraft. These systems are nowadays not helping the pilot to manoeuvre the aircraft. The pilot can today largely let the aircraft fly itself and only monitor that this is done properly. This has led to both complicated and complex control systems to gain safety and flight performance.

The FCS needs to be deterministic to be able to prove the safety of the system through verification and validation of the flying and handling characteristics. This is today done in simulation, ground and flight testing. The design process for such an FCS is an expensive and time-consuming task. To be able to reduce cost and time spent, much is gained. Also, just as the effort to develop families of aircraft to reduce costs, there are ideas to be able to use the same control system concept in several different aircraft. The goal is to be able to use the same system architecture, with a minimum of changes, so that time and resources for development and

testing can be reduced and thereby decrease costs. This should be done in a way that does not decrease the light performance.

This paper describes the goal and activities done so far within a project in the Swedish national research program (NFFP) called Flight Control SUBscale Flight Testing (FCoSUFT). This is a collaboration between Saab and Linköping University (LiU). At LiU the use of subscale demonstrators, and how these can be used early in the development process of a new aircraft, have been of interest for some time [1]. To bring Swedish industry and academia together to try to, in an efficient way, make use of the opportunities that subscale flight testing brings, a one-year NFFP project, Fast Development of a Flying Technology Demonstrator (Flying TeD) was started in 2018. This was then succeeded in 2019 by a three-year continuation called Flying TeD II, where the goal was to run through the full process from design to flight within a relatively short time. For this, the Generic Future Fighter (GFF), a carbon fibre subscale aircraft, was used. The result was presented at ICAS 2022 in Stockholm [2]. For this only the wing planform was changed to limit the scope to fit within the project. Several planforms were looked at. The chosen design change, which was flown is shown in Figure 1.

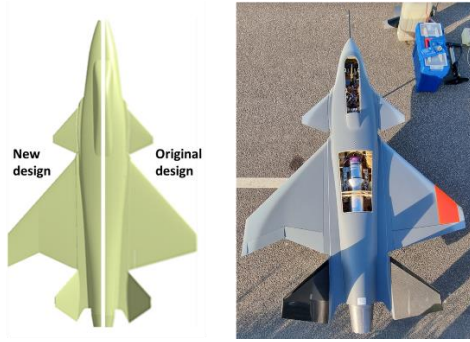


Figure 1: GFF design change within Flying TeD II.

The next step in the development of subscale demonstrators was to try to make use of 3D-printing to reduce the manufacturing time and to use this as a risk mitigating aspect while keeping the rapid prototyping and testing of new technologies. This was done in the NFFP-project ADvanced Subscale Flight Testing (ADSUFT) that was a one-year project. Part of this was run as a master thesis project [3]. For this development, a 3D-printer was purchased. Different materials and ways to construct the structure as well as how to use the 3D-printers ability to manufacture the aircraft was investigated. The project ended with a flight test in 2024. Figure 2 shows the printer and the resulting geometry of the demonstrator. Note that this used the original wing planform.



Figure 2: A 3D-printer and the resulting demonstrator.

With the described results it was clear that rapid prototyping, using 3D-printing was a way forward to be able to design, build and fly cost effective subscale demonstrators. The goal of the FCoSUFT project is to show that the process developed can be used to demonstrate new technologies. The choice of technology in this project is flight control laws and control allocation.

2 The Demonstrator platform

Much time and effort has been put into the GFF platform, and the geometry and aerodynamic characteristics have been studied as shown in [2]. It was natural to continue to use the newer version of the GFF, the one with a cranked wing, as a platform for demonstrating of control allocation since it has more actuators that can be used, see Figure 3. There is a total of eight actuators, two canard surfaces, four elevons at the wing trailing edge and two for the thrust vector control (TVC) nozzle, used for pitch and yaw manoeuvring.

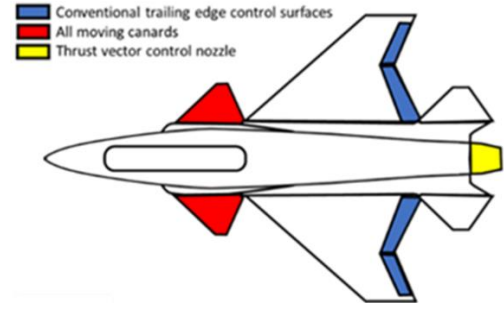


Figure 3: GFF with a cranked wing.

This redundancy in actuators is necessary for the allocation part of the control. Each control actuator needs to be treated separately and not in pairs like traditional elevators and ailerons in more conventional control settings.

The engine to be used is an EDF Ducted Fan JP Hobby 90mm + 12s Motor with a power of 5280 W and an installed thrust of about 50 N. the engine will work with a SEQUIRE SQESC 12200 Brushless Electric Speed Controller 5-12S with a Power Supply of 200A. The aircraft weight will be approximately 70 N. This gives a thrust-to-weight ratio of 0.71, which is considered sufficient for the demonstration purposes.

Figure 4 shows a computer aided design (CAD) drawing of the internal layout of the GFF. A lot of time and effort have been made to reduce the weight so that as high performance as possible will be available.

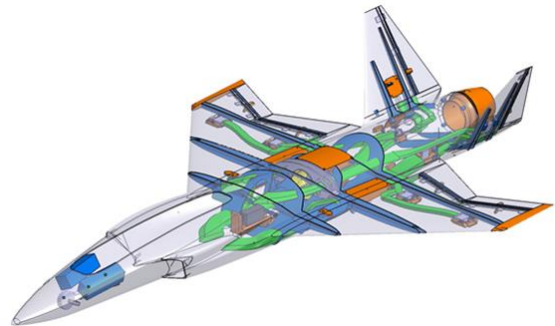


Figure 4: CAD drawing of the internal structure layout.

The manufacturing of the first of three fuselages is ongoing. As an example, a fuselage section and the right main wing, without control surfaces, are shown in Figure 5. One can see where the engine should be fitted and where the control surfaces servos should be placed.



Figure 5: 3D-printed engine section and main wing without control surfaces.

One important part of the demonstrator is the flight control hardware (HW) and software (SW) for the flight control system (FCS). For this the PixHawk, Cube Red HW shown in Figure 6, together with ArduPilot SW is to be used. A good feature with Cube Red is that it is possible to switch between two different implemented control laws. This makes it possible to change to the original control law if things go wrong with the more innovative versions. This also reduces the risk of losing an aircraft.



Figure 6: PixHawk, Cube Red.

The idea is that the different flight control concepts should be developed in the ArduPilot SW that is connected to a simulation environment. When the control laws are verified and validated in the simulation environment, the SD-card in the PixHawk can be moved to the one mounted in the demonstrator aircraft for a seamless implementation before flight test.

3 Flight control concept

Several different flight control laws have been discussed as interesting variants to flight test. Here, some general descriptions of some control law are given. A general, nonlinear dynamic system can be described as

$$\begin{aligned}\dot{x} &= f(x, u) + w \\ y &= h(x) + v\end{aligned}\quad (1)$$

where x is the system state, u is the input, y is the measured output. The system is disturbed by noise w and the measurements by v . A noise free model of the system will be denoted by

$$\begin{aligned}\dot{x} &= f(x, u) \\ y &= h(x)\end{aligned}\quad (2)$$

This will be used to give the description of the different control laws.

3.1 Linear Quadratic Regulator (LQR) with gain scheduling

A common control law design that is used in many of today's aircraft is the LQR concept, which is used together with gain scheduling. A system view of this is shown in Figure 7.

For the LQR design, a linear version of the nonlinear system in given in Eq. (2) around some state is used

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx\end{aligned}\quad (3)$$

where A is the system matrix, B is the control matrix and C is the measurement matrix. For this, a feedback control law on the form

$$u = -K_x x + K_r r \quad (4)$$

is proposed.

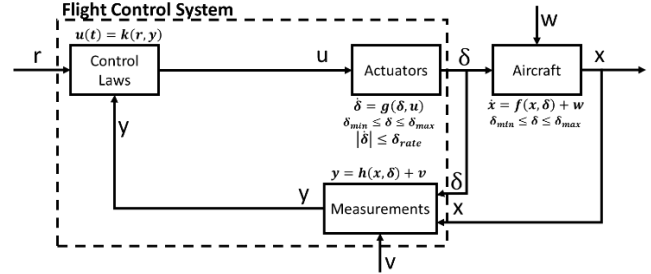


Figure 7: System view of a LQR design.

For the feedback, optimisation is used to find K_x . The goal is to find an input u that minimize the cost function

$$J = \frac{1}{2} \int_0^T (x^T Q x + u^T R u) dt \quad (5)$$

where Q and R are positive and non-negative weight matrices. By choosing Q and R different solutions are found. The solution will be

$$u = -R^{-1} B^T P x = -K_x x \quad (6)$$

where P is the solution to the Algebraic Riccati Equation (ARE)

$$0 = PA + A^T P + Q - PBR^{-1}B^T P \quad (7)$$

The problem can be solved by using the Matlab command

$$K = lqr(A, B, Q, R)$$

The process of getting a control system for the whole flight envelop includes the investigation of several state points since the system is nonlinear. This is where the gain scheduling comes into play. This ties together the point that have been investigated. More of the LQR theory can be found in [4].

The Feed Forward part can be solved using PID control reference tracking of the command. PID theory can also be found in [4].

3.2 Nonlinear Dynamic Inversion (NDI)

NDI is a feedback linearisation technique that uses a model of the system to cancel nonlinearities. This is a way to deal with the gain scheduling problem of ensuring stability characteristics between the points used for the scheduling. For the control law design, it is assumed that the input part of $F(x, u)$ in Eq. (2) can be separated, giving

$$\begin{aligned}\dot{x} &= f(x) + g(x)u \\ y &= h(x)\end{aligned}\quad (8)$$

The idea behind the NDI is to solve for u in Eq. (8), which gives a control law

$$u = g^\dagger(x)(\dot{x} - f(x)) \quad (9)$$

where $\dagger$ denotes the Moore-Penrose inverse $g^\dagger = (g^T g)^{-1} g^T$, which is used if g is not a square matrix. By introducing a linear outer loop control v that describes the desired state motion, the control law becomes

$$\begin{aligned} u &= g^\dagger(x)(v - f(x)) \\ v &= \dot{x} \end{aligned} \quad (10)$$

or if a control law $k(r, y)$ using a feed forward from the pilot reference input, r , and feedback from the measurements, y , for the desired motion

$$\begin{aligned} u &= g^\dagger(x)(v - f(x)) \\ v &= k(r, y) \end{aligned} \quad (11)$$

The top equation in Eq. (11) is here described as the control allocation. For this control law a model of $g(x)$ and $f(x)$ is needed. The accuracy of these will affect the efficiency of the NDI implementation. A description with an aircraft analysis using NDI can be found in [5].

A system diagram of the system is given in Figure 8. Here the actuator dynamics and limitations are also included as well as the system and measurement noise contributions.

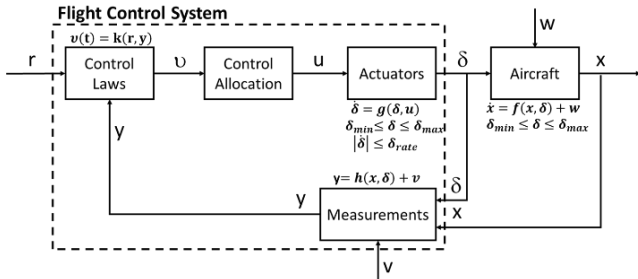


Figure 8: The NDI flight control concept.

3.3 Incremental Nonlinear Dynamic Inversion (INDI)

INDI is closely related to NDI, but it is based on a Taylor expansion of the state equation in Eq. (2) around the current state, x_0 , and input, u_0 , as

$$\dot{x} = \dot{x}_0 + F(x_0, u_0)(x - x_0) + G(x_0, u_0)(u - u_0) \quad (12)$$

where

$$F(x_0, u_0) = \left. \frac{\partial f(x, u)}{\partial x} \right|_{x_0, u_0} \quad G(x_0, u_0) = \left. \frac{\partial f(x, u)}{\partial u} \right|_{x_0, u_0} \quad (13)$$

In INDI it is assumed that the input u changes much faster than the state x . This results in the assumption that $\dot{x} \approx \dot{x}_0$. With this Eq. (12) reduces to

$$\dot{x} = \dot{x}_0 + G(x_0, u_0)(u - u_0) \quad (14)$$

Solving for the input u and introducing the linear outer loop control v similar as for NDI results in the control law

$$u = u_0 + G^\dagger(x_0, u_0)(v - \dot{x}) \quad (15)$$

The similarities between NDI and INDI are obvious when comparing Eq. (11) and Eq. (15). However, INDI is less dependent on the model of the system since $F(x_0, u_0)$ is removed from the control law. This does also reduce the some needed computations.

An example of using INDI for a quad-plane application can be found in [6].

3.4 Dynamic control allocation using constrained quadratic programming

To dynamic control allocate control actions can be made by solving a minimisation problem as stated in Eq. (16), where v is a virtual control command, u is the commanded actuator position, δ is the actual actuator position, x is the system state and y is the measurements. The virtual command v is a desired motion of the aircraft that should be realised by allocating the actuators to different positions. Another way to do this, different from the NDI or INDI, is by using constrained quadratic programming [7] as shown in Eq. (16).

$$\begin{aligned} \min_{u(t)} & \|W_1(u(t) - u_s(t))\|_2^2 + \\ & \|W_2(u(t) - u(t-T))\|_2^2 \\ \text{s. t. } & Bu(t) = v(t) \\ & u_{\min}(t) \leq u(t) \leq u_{\max}(t) \\ & u_{\min}(t) = \max\{\delta_{\min}, u(t-T) - \delta_{\text{rate}}T\} \\ & u_{\max}(t) = \min\{\delta_{\max}, u(t-T) + \delta_{\text{rate}}T\} \end{aligned} \quad (16)$$

Where W_1 and W_2 are weight matrices, B is the control effectiveness matrix and T is the sample time.

The interesting idea here is that constraints for the possible control deflections and deflection rates can be addressed. This means that only commands that lead to feasible control actions will be given. There are some similarities to the NDI and INDI control laws in that the commanded control action is based on the virtual control command. The difference is that the control action will be solved online using an optimisation algorithm. This has to come up with a feasible solution in time to secure safe flight of the aircraft.

3.5 Real-time Certified Model Predictive Control (MPC).

Another possible control law strategi is to use model predictive control. For this a model of the system is needed, much like for many of the previous mentioned control laws. Even in this case an optimisation problem is solved. This looks like

$$\min_{u_k, x_k} \sum_{k=0}^{N-1} (x_k^T Q x_k + u_k^T R u_k) + \Psi(x_N)$$

$$\begin{aligned} \text{s.t. } & x_{k+1} = Ax_k + Bu_k \quad \forall k = 0, \dots, N-1 \\ & x_k \in \mathcal{X} \quad \forall k = 0, \dots, N-1 \\ & u_k \in \mathcal{U} \quad \forall k = 0, \dots, N-1 \\ & u_N \in \mathcal{T} \end{aligned} \quad (17)$$

where $\mathcal{X}$ and $\mathcal{U}$ are convex sets of possible states x and inputs u . Eq. (17) is a linear version of the MPC that show the principle of the optimisation to be solved. The philosophy of MPC is to run the optimisation, trying to follow a reference trajectory, from the current state and input (x_0, u_0) to a time horizon $k=N-1$ to get an optimal sequence $u_k, k=0 \dots N-1$. Then take one timestep, i.e., to (x_1, u_1) and then reset the optimisation problem meaning that the new current state is defined as the new (x_0, u_0) and run the optimisation again, see Figure 9. More on MPC can be found in [8], also for a nonlinear version.

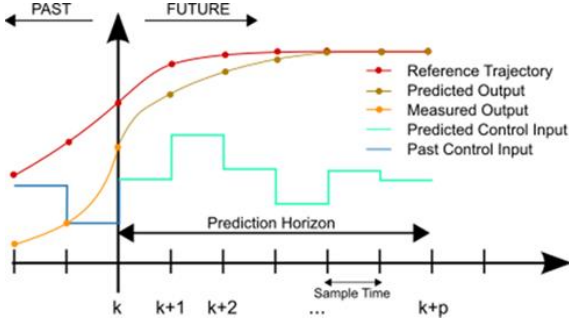


Figure 9: MPC philosophy (By Martin Behrendt, via Wikimedia Commons)

Just as the dynamic control allocation using constrained quadratic programming, a feasible control input needs to be found in time to be able to run a save control law. In [] this problem is addressed.

3.6 Other possible control strategies

The above-mentioned control strategies are interesting versions, but other strategies might be tested. For example, algorithms that use machine learning and neural networks. It will be possible to test all kinds of control laws as long as they have been verified and validated before flight.

4 Simulation environment

For the control law development and design, a flight mechanical simulation environment is built based on the Generic Future Fighter (GFF) configuration. Computational Fluid Dynamic (CFD) calculations for different control surface deflections of the GFF have been performed to describe the aerodynamics that support this simulation environment. The same computer hardware that is installed in the flying demonstrator for control shall be connected to the simulation environment to test control laws before incorporating these into the actual flying demonstrator, which makes it possible to secure a seamless and quick integration

of the control laws before flight testing. A definition description of the GFF variables is given in Figure 10.

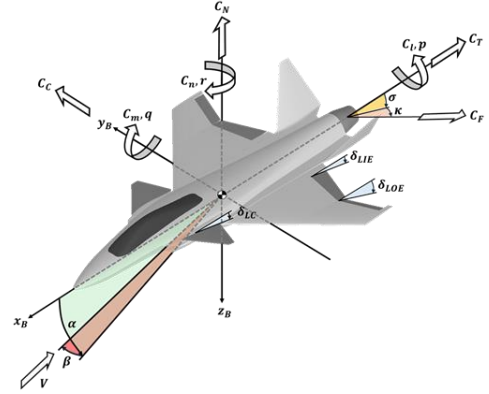


Figure 10: Definitions of aerodynamic and control quantities.

Figure 11 shows results from the CFD calculations for the left side control surfaces. To get the effect of the right-hand side, mirroring technique can be used. It can be seen that the different control surfaces are almost equally efficient. However, the canard characteristics is more nonlinear for large positive deflections.

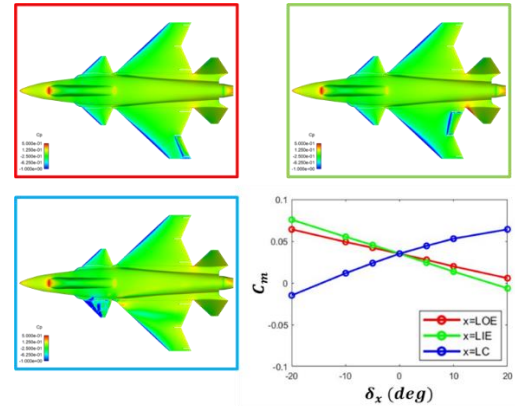


Figure 11: CFD calculations for the left outer elevon, left inner elevon and the left canard.

The pitching moment coefficient for the angle-of-attack and sideslip angle is shown in Figure 12 as an example of how nonlinear the aerodynamic characteristics are for the GFF.

The flight mechanical simulation environment is built up in Matlab in a similar way as described by the system diagram shown in Figure 8 for the NDI flight control concept. As of now only the basic environment, without a control system, is finished and ready for use. The part where the different control laws are to be implemented will be modified to be able to use the simulation environment for the different designs.

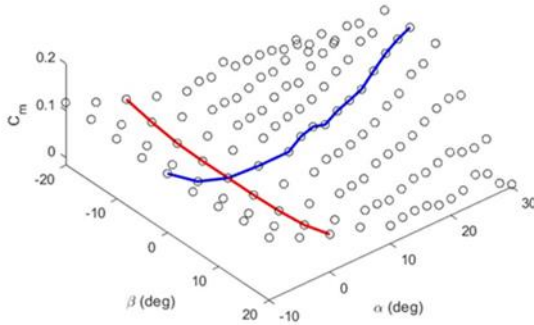


Figure 12 : Pitching moment coefficient for angle-of-attack and side slip angle.

Figure 13 shows an example of the simulation response to a pitch stick input from the simulated radio controller. As can be seen, the possibility to add actuator dynamics into the simulation environment is implemented. For the demonstrator the hardware will have very fast responses, but there will be the possibility to, in the demonstrator software, include different actuator dynamics and even to inject failures of control surfaces to investigate how the implemented control strategies will handle these kinds of situations.

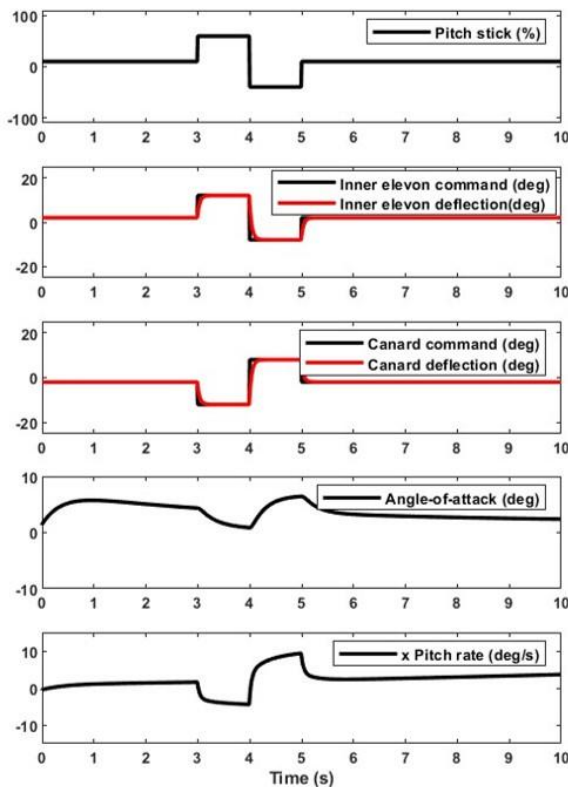


Figure 13: A simulation example using the basic flight mechanic environment.

5 Conclusion

In this paper the ongoing work within the NFFP project Flight Control SUBscale Flight Testing (FCoSUFT) have been presented. The goal of this project is to test different flight control laws in a 3D-printed subscale demonstrator aircraft. This is done to show that using this approach is a way to get interesting results at a low cost and risk early on in new aircraft development projects. Several control law strategies have been proposed, but other may appear during the FCoSUFT project. Right now, the first of three demonstrators is being manufactured and a first version of a simulation environment, to support the control law designs, is in place. The first flight with the GFF demonstrator will be performed during 2025.

References

- [1] D. Lundström & K Amadori, "RAVEN - A Subscale Radio Controlled Business Jet Demonstrator", *In 26th Congress of the International Council of the Aeronautical Sciences*, Anchorage, Alaska, USA. International Council of the Aeronautical Sciences, (ICAS), September 2008.
- [2] R. Larsson et. al., "Rapid prototyping in aircraft design using CFD, wind tunnel and flight testing", *In Proceedings of the 33rd Congress of the International Council of the Aeronautical Sciences*, Stockholm, Sweden. International Council of the Aeronautical Sciences, (ICAS), 2022.
- [3] A. Sen, "Advanced FDM 3D Printing Methods for Lightweight Aeronautical Prototypes", *Master thesis project at LiU, FLUMES*, Sweden, 2024.
- [4] T. Glad & L. Ljung, "Control Theory", CRC press, 2018
- [5] B Pavlovic, "Dynamic Performance Analysis of a Fighter Jet with Nonlinear Dynamic Inversion", *Degree project in vehicle engineering*, KTH, 2019
- [6] L. Zhou et. Al., "Incremental nonlinear dynamic inversion based path-following control for a hybrid quad-plane unmanned areal vehicle", *Int J Robust Nonlinear Control*, 2023.
- [7] O. Härkegård, "Dynamic control allocation using constrained quadratic programming", *Journal of Guidance, Control, and Dynamics*, vol. 27, nr. 6, pp 1028-1034, 2004.
- [8] D. Simon, "Model Predictive Control in Flight Control Design: Stability and Reference Tracking", *Licentiate Thesis*, Linköpings Universitet, Sweden, 2014.
- [9] D. Arnström, "Real-time Certified MPC Reliable Active-Set QP Solvers", *PhD Dissertation*, Linköpings Universitet, Sweden, 2023.