A Model Predictive Control Approach for Longitudinal Control of Helicopter

Frédéric Larocque (5481023)

***Abstract*—The longitudinal model of a helicopter was derived and linearized around a hover flight condition. A regulation, trajectory tracking and offset-free output reference model predictive controllers (MPC) were designed with state and input constraints and then proved to be stable in their attractor set. Multiple simulations were completed to evaluate the performance of each controller, compare it to an LQR controller and analyze the impact of various MPC parameters on its performance and stability.**

I. Introduction

Rotorcraft are highly complex dynamic systems. Due to interaction between the constant rotation of the rotor, interaction between aerodynamics, elastic forces (aeroelasticity) and the rigid body dynamics of the fuselage, helicopters are highly non-linear and present strong coupling between axis[1]. Helicopters are inherently unstable in the hover, as the rotor flaps back to counteract the fuselage motion [2]. Skilled pilots can learn to master the helicopter and make it stable. Stability augmentation systems (SAS) are now present in almost all commercial helicopters, mostly based on feedback control using classical control methods and gain scheduling [2].

A. Related Work

To try to reduce coupling between axis present in classical single-input single-output (SISO) feedback methods, multiple-input multiple-output (MIMO) methods have been implemented for helicopters. For example, Kim et al. implemented a H_∞ MIMO method that provides better robustness against uncertainty and disturbance than a classical SISO method [3]. MPC has started to be used in the aeronautical industry to take advantage in its capacity to provide maneuver and envelope protection [4]. MPC has been used with success to provide trajectory tracking for constrained 6 DoF helicopter models[5], [6]

As the main aim of the course SC42125 is to evaluate the feasibility, stability and performance of an MPC controller on a non-linear system, our paper will model and implement a MPC controller on the 3 degrees of freedom longitudinal dynamics of a helicopter. Although model fidelity for such a simplified model is low, it is deemed ideal to get a grasp on fundamental dynamics and trends of helicopter dynamics [1] and of the MPC controller.

Frédéric Larocque is an aerospace engineering master student at TU Delft, The Netherlands. E-mail address: f.larocque@student.tudelft.nl.

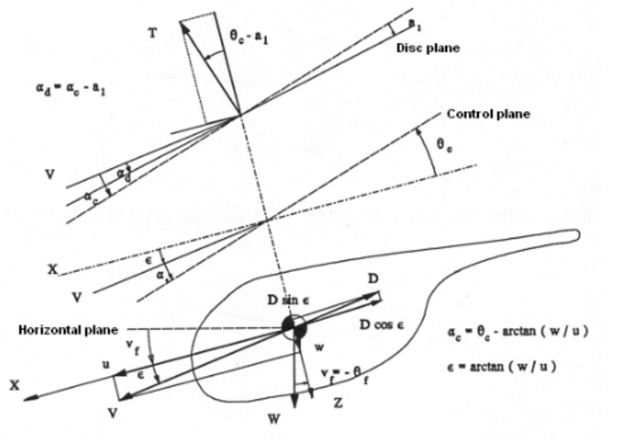

Fig. 1. Axis Used for Longitudinal Model [7]

B. Non-Linear Helicopter Dynamics

The non-linear model of the longitudinal axis of the helicopter is built based on work by [1], [7], with helicopter physical parameters derived from the H145 [8]. Figure 1 shows the main forces acting on the helicopter in flight, thrust (T) and drag (D) in addition to the reference frame used for the model. The airframe equations of motion are first derived in Equation 1 to 4.

$$\dot{u} = -g\sin\theta_f - \frac{D}{m}\frac{u}{V} + \frac{T}{m}\sin\left(\theta_c - a_1\right) - qw \tag{1}$$

$$\dot{w} = g\cos\theta_f - \frac{D}{m}\frac{w}{V} - \frac{T}{m}\cos\left(\theta_c - a_1\right) + qu \tag{2}$$

$$\dot{q} = -\frac{T}{I_y}h\cdot\sin\left(\theta_c - a_1\right) \tag{3}$$

$$\dot{\theta} = q \tag{4}$$

In addition to the airframe equation of motion, the system should be augmented with the flapping dynamics of the rotor. However, as the flapping response is very rapid compared to the body dynamics, using the quasi-static assumption, it is assumed that the blade system responds instantaneously to control inputs and flight conditions [9]. Thereafter, the inflow (λ_i) dynamics (or induced flow by the rotor system) are modeled following Equation 5 to 8. For detailed information regarding notation and parameters values, the reader is referred to [9], [1] and [7].

$$a_1 = \frac{\frac{8}{3}\mu\theta_0 - 2\mu\left(\lambda_c + \lambda_i\right) - \frac{16}{\gamma}\frac{q}{\Omega}}{1 - \frac{1}{2}\mu^2} \tag{5}$$

$$C_{T_{BEM}} = \frac{1}{4} c_{l\alpha}\sigma \left[\frac{2}{3}\theta_0\left(1 + \frac{3}{2}\mu^2\right) - \left(\lambda_c + \lambda_i\right)\right] \tag{6}$$

$$C_{T_{Glau}} = \lambda_i \sqrt{\left(\frac{V}{\Omega R}\cos\left(\alpha_c - a_1\right)\right)^2 + \left(\frac{V}{\Omega R}\sin\left(\alpha_c - a_1\right) + \lambda_i\right)^2} \tag{7}$$

$$\tau\frac{dv_i}{dt} = C_{T,\,\text{elem}} - C_{T,\,\text{Glauert}} \tag{8}$$

Brought together, the airframe and inflow dynamics create a non-linear time invariant system composed of 7 states and two inputs (enumerated in Equation 10 and 11), defined by:

$$\dot{x} = f(x) + g(u) \tag{9}$$

where:

$$x = \begin{bmatrix} x & z & u & w & q & \theta & \lambda_i \end{bmatrix}^T \tag{10}$$

$$u = \begin{bmatrix} \theta_0 & \theta_c \end{bmatrix}^T \tag{11}$$

More precisely, the states are:

- x : x position (m)
- z : z position (m)
- u : x velocity (m/s)
- w : z velocity (m/s)
- q : pitch rate (rad/s)
- θ : pitch angle (rad/s)
- λ_i : inflow (non-dimensionnal)

And the inputs are:

- θ_0 : collective position (as a percentage of maximum possible angle), generally responsible for increasing thrust
- θ_c : cyclic position (as a percentage of maximum possible angle), generally responsible for tilting the thrust force forward to move

C. Linearized Dynamics

Trim for the following system can be achieved around the hover position. All states except inflow are set to zero. Initial inflow can be calculated iteratively following [1], depending on the total airspeed. With the calculated trim position, the system can be numerically linearized around it to obtain a linear time invariant state space system:

$$\dot{x} = A\,x + B\,u \tag{12}$$

$$y = C\,x \tag{13}$$

For the state feedback case, the matrix C is set to the identity matrix, while for output feedback, it is set to obtain the x and y position, in addition to the pitch rate q. The whole system is discretized using a first order hold with a sample time of 0.1s. This sample time was chosen to reduce computationnal load while still being able to stabilize the system, as the helicopter dynamics are much slower (step response of around 2-3 sec). The system obtained was verified to be observable and controllable, but not stable, as some poles are outside of the unit circle. Comparison between the non-linear and linear system in hover is completed in subsection IV-A and deemed satisfactory to continue the controller design around this linearized condition.

II. Model predictive control design

In this section, multiple MPC controllers are designed for this paper.

1) Regulation MPC
2) Trajectory Tracking MPC
3) Offset-free Output Tracking MPC

For all controllers, constraints are set on the system. As the system is linearized and the inputs are described as a percentage of the maximum input, the inputs are easily constrained as follows to obtain $\mathbb{U}$:

$$|u(k)| \leq \begin{bmatrix} 0.25 \\ 1 \end{bmatrix} = u_{lim} \tag{14}$$

Collective input is more constrained than cyclic to make sure that we don't exceed the maximum collective input, that is already set to around 65% at equilibrium point.

States are constrained to limit high airspeed, pitch angle and rate that are unrealistic for a helicopter transporting passengers and most importantly, could lead to the linearized system to diverge from the non-linear system. Position and inflow are not constrained and therefore their constraint is set to a high enough value to not interfere with calculations. States constraints are set as follows:

$$|x(k)| \leq \begin{bmatrix} 10^3\,m \\ 10^3\,m \\ 5\,m/s \\ 5\,m/s \\ 5\,deg/s \\ 15\,deg \\ 10^3 \end{bmatrix} = x_{lim} \tag{15}$$

As the MPC controller optimizes inputs, these state constraints that define $\mathbb{X}$ need to be transformed back to inputs using the prediction equation:

$$\boldsymbol{x_{n+1}} = T\,x_0 + S\,\boldsymbol{u_N} \tag{16}$$

Ultimately, the state constraints can be enumerated as a function of the sequence of input as follows:

$$S\,\boldsymbol{u_N} \leq -T\,x_0 + x_{lim} \tag{17}$$

$$S\,\boldsymbol{u_N} \geq -T\,x_0 - x_{lim} \tag{18}$$

For all MPC controllers, a receding control horizon $N = 25$ is used as it provides a big enough attractor set $\mathbb{X}_n$ while not having a too high computational load. The effect of N will be studied in subsubsection IV-B2.

State weight Q and input weight R matrices were set (refer to Equation 19 and 20) to put a greater importance on states and to leave inflow weighted low as it follows the other states. In subsubsection IV-B1, these values are modified to see their impact on the performance of the MPC controller.

$$Q = 10^1 diag(\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 10^{-4} \end{bmatrix}) \tag{19}$$

$$R = diag(\begin{bmatrix} 1 & 1 \end{bmatrix}) \tag{20}$$

A. Regulation MPC

The regulation MPC is designed following section 2.0 of [10] using full state feedback. The main cost function is set as $V_N(x_0, \boldsymbol{u_N}) = \sum_{k=0}^{N-1} \ell(x(k), u(k)) + V_{\text{f}}(x(N))$, with $\ell(x,u) = x(k)^T Q x(k) + u(k)^T R u(k)$ and $V_{\text{f}}(x) = \beta x(N)^T P x(N)$, where P is the discrete-time algebraic Riccati equation solution, solvable on Matlab using *dare()* and β is a gain used to make it possible to remove the terminal set constraints, as explained in the next paragraph. With this selection, when in $\mathbb{X}_f$, the MPC controller with effectively perform as a LQR controller.

No constraint has been set on the terminal set $\mathbb{X}_f$, following section 2.6 from [10]. Indeed, adding this terminal constraint puts additional hard constraints on the problem that can be avoided by instead making sure the initial state lies in a subset of $\mathbb{X}_n$ and using a sufficiently high β value. This was tested empirically using simulation and a β value of 1 was deemed satisfactory to provide stability to the system.

In the end, the optimal control problem to solve for the regulation MPC is as follows:

$$\mathbb{P}_n(x_0,t) : \begin{cases} \min_{\boldsymbol{u_N}} & V_N(x_0, \boldsymbol{u_N}) \\ \text{s.t.} & S\,\boldsymbol{u_N} \le -T\,x_0 + x_{lim} \\ & S\,\boldsymbol{u_N} \ge -T\,x_0 - x_{lim} \\ & |\boldsymbol{u_N}| \le u_{lim} \end{cases} \tag{21}$$

B. Tracking MPC

For the tracking MPC, the controller must follow an output reference in position y_{ref} using state feedback. As the controller no longer works in regulation around 0 but around a translated $\mathbb{X}_f$, we must solve the optimal target selection (OPS) problem each time the reference changes:

$$(x_{ref}, u_{ref})(y_{ref}) : \begin{cases} \text{argmin}_{x_r,u_r} & J(x_r,u_r) \\ \text{s.t.} & \begin{bmatrix} I-A & -B \\ C & 0 \end{bmatrix} \begin{bmatrix} x_r \\ u_r \end{bmatrix} = \begin{bmatrix} 0 \\ y_{ref} \end{bmatrix} \\ & (x_r,u_r) \in \mathbb{Z} \\ & y_{ref} \in \mathbb{Y} \end{cases} \tag{22}$$

where $J(x_r,u_r) = x_r^T Q x_r + u_r^T R u_r$, with Q and R as identity matrices.

This brings a new stage and terminal cost calculation, where $\ell(x,u) = (x(k) - x_{ref})^T Q (x(k) - x_{ref}) + (u(k) - u_{ref})^T R (u(k) - u_{ref})$ and $V_{\text{f}}(x) = \beta (x(N) - x_{ref})^T P (x(N) - x_{ref})$. It is also important to make sure the translated terminal set $\mathbb{X}_f$ is still inside $\mathbb{X}$. If so, stability can be guaranteed. We can assume a terminal constraint set small enough that its translation still keeps it in $\mathbb{X}$[4]. The optimal problem to solve follows the same structure as Equation 21, but with a new stage and terminal cost calculation.

Additionally, the following tracking problem was extended to provide y_{ref} as a vector of length N_{ref}, corresponding to how steps in advance the MPC controller knew of the future trajectory to follow, the standard MPC having a $N_{ref} = 1$. The same OPS and OPC as in Equation 21 and 21 is used, but with y_{ref} as a vector. The change of N_{ref} is evaluated in subsubsection IV-C1.

C. Output Offset-free MPC

The output offset-free MPC follows the same logic as the last section. However, only output feedback can be used, to simulate a case where the helicopter has limited sensors. It can measure its position x_{pos}, y_{pos} and its pitch rate q. In addition, the system is subject to wind, acting as constant unknown disturbances d on the x_{pos}, y_{pos} states. The original state space system must therefore be augmented with the disturbances d:

$$\begin{bmatrix} x^+ \\ d^+ \end{bmatrix} = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix} \begin{bmatrix} x \\ d \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u \tag{23}$$

$$y = \begin{bmatrix} C & C_d \end{bmatrix} \begin{bmatrix} x \\ d \end{bmatrix} \tag{24}$$

The original and augmented system are verified to be observable, in addition to verifying that:

$$\text{rank} \begin{bmatrix} I-A & -B_d \\ C & C_d \end{bmatrix} = n + n_d \tag{25}$$

where n and n_d are respectively the number of states in the original system and the number of disturbances.

In order to estimate the states of the system (including the disturbances), a Luenberger observer is constructed, by pole placement, with arbitrary poles around 0.5 in the z-plane that gives gains L that provide stability to the $A_{aug} - L\,C_{aug}$ system. A Kalman observer was also implemented and tested but with lower performance. The Kalman observer most probably needed precise tuning for it covariance matrices.

The OPS must be calculated at each MPC iteration as the best current estimation of the disturbance evolves at each iteration. The OPS, is then:

$$(x_{ref}, u_{ref})(\hat{d}, y_{ref}) : \begin{cases} \text{argmin}_{x_r,u_r} & J(x_r,u_r) \\ \text{s.t.} & \begin{bmatrix} I-A & -B \\ C & 0 \end{bmatrix} \begin{bmatrix} x_r \\ u_r \end{bmatrix} = \begin{bmatrix} B_d \hat{d} \\ y_{ref} - C_d \hat{d} \end{bmatrix} \\ & (x_r,u_r) \in \mathbb{Z} \\ & C\,x_r + \hat{d} \in \mathbb{Y} \end{cases} \tag{26}$$

To guarantee asymptomatic stability, it is assumed that the estimation of the disturbance will over time converge to a constant disturbance value. This assumption is verified a-posteriori in Figure 13.

III. Asymptotic stability

Asymptotic stability of the closed loop system is proved using the assumptions and theorems from section 2.5.4 of [10], as our system is initially unstable, with control and state constraints.

A. Assumption 2.2: Continuity of system and cost

Assumption 2.2 is verified as our linear system $f(x,u)$, the stage cost ℓ and terminal cost $V_{\text{f}}(x)$ are all continuous and are zero at the origin.

B. Assumption 2.3: Properties of constraints sets

The admissible inputs $\mathbb{U}$ and admissible states $\mathbb{X}$ are compact sets that contain the origin. Assumption 2.3 is therefore verified.

C. Assumption 2.14: Basic stability assumption

As per [10], for all $\boldsymbol{x} \in \mathbb{X}_f$, there must exist a sequence of inputs $\boldsymbol{u}$ that lead to a Lyapunov decrease $V_\mathrm{f}(f(x,u) - V_\mathrm{f}(x) \leq -\ell(x,u)$ and keep $\boldsymbol{x}$ in $\mathbb{X}_f$. As we choose the terminal cost as $V_\mathrm{f}(x) = \beta x(N)^T P x(N)$, where P is the solution to the discrete Riccati equation, which also gives out the optimal gain K, it can be shown (refer to [10]) that the system follows a Lyapunov decrease:

$$V_f\left(A_K x\right) + (1/2) x' Q_K x - V_f(x) \leq 0 \tag{27}$$

where $A_K = A + BK$ and $Q_K = Q + K^T R K$

$a)$ can also be showed a-posteori, as shown in Figure 2, where the stage cost is always less or equal than the difference of the terminal costs.

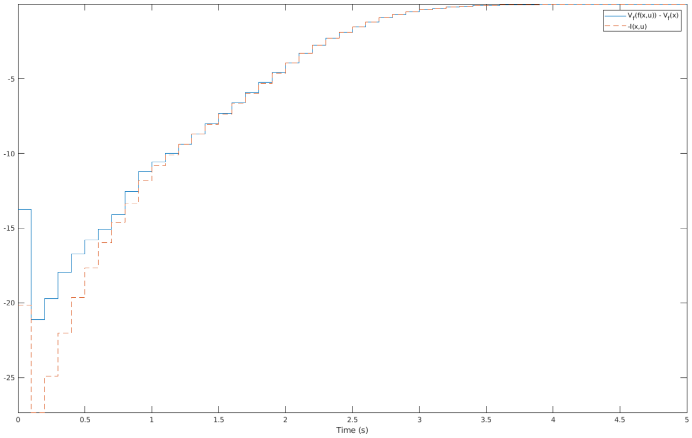

Fig. 2. Evolution of stage cost and terminal cost when starting outside $\mathbb{X}_N$

$b)$ holds as matrices Q,R,P are all positive definite as they are diagonal matrices with positive coefficients, making stage cost and terminal cost positive definite.

D. Exponential Stability in $\mathbb{X}_N$

To make sure that the controller always satisfies the state and input constraints, the terminal constraints set $\mathbb{X}_f$ is chosen to be the maximal invariant constraint admissible set for our system, as per [10]. This means that if the initial state is in $\mathbb{X}_f$ the controller will follow constraints and keep the state in $\mathbb{X}_f$.

As all assumptions are satisfied and $\mathbb{X}_f$ contains the origin, by theorems 2.19 and 2.21 of [10], the system is exponentially stable in $\mathbb{X}_N$, the set of states that can be brought to $\mathbb{X}_f$ in N steps.

Stability of the non-linear system was not evaluated nor pursued due to its inherent complexity, although simulations have shown empirically that the regulation MPC is stable close to $\mathbb{X}_f$. More analysis needs to be conducted on that matter.

E. Evaluating $\mathbb{X}_f$

To evaluate $\mathbb{X}_f$, the algorithm by Gilbert and Tan [11] was used, with a slight modification to calculate state admissible set instead of output admissible set (set $C = [I; K]$). In short, each constraint (state and input constraint) is optimized until it gives a positive value. If it does not, the system is elevated to higher power and optimized again. The reader is referred to [11] for more details. At the end of the process, a set of constraints define the terminal set $\mathbb{X}_f$:

$$\mathbb{X}_f = \{x \in \mathbb{R}^7 | H\,x \leq h\} \tag{28}$$

Using this inequality, although we cannot visualize it in 7 dimensions, we can set certain states to zero and visualize the terminal set. The terminal set was plotted in Figure 3, for position and velocity in the x axis. Refer to the Matlab code supporting this paper[12] for more $\mathbb{X}_f$ figures.

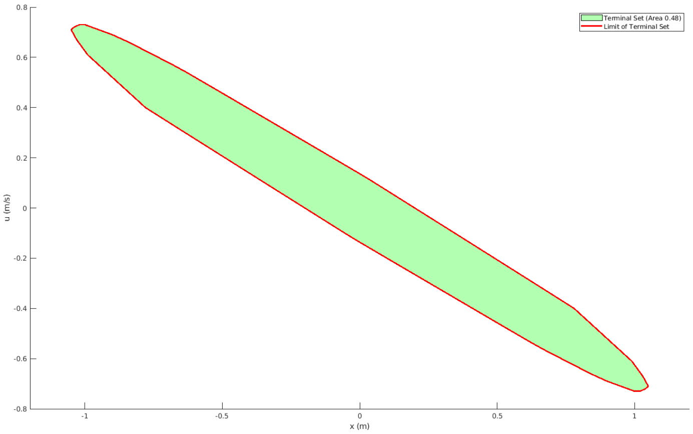

Fig. 3. Terminal set $\mathbb{X}_f$ when setting all states except x and u to zero

F. Estimating $\mathbb{X}_N$

$\mathbb{X}_N$ is defined as the set of states from which the MPC controller can bring the states towards $\mathbb{X}_f$ in N steps while following states and input constraints. It is possible to estimate $\mathbb{X}_N$ using algorithms from [10], however, for this problem, as the set would be in $\mathbb{R}^7$ it was deemed too time consuming to calculate the whole domain. Instead, $\mathbb{X}_N$ was estimated when all states except the position were set to 0. An algorithm was built to explore the 2D position domain and verify for each point if the MPC controller could converge to $\mathbb{X}_f$ in N steps. If it did, the point was added to $\mathbb{X}_N$. Figure 4 shows the attraction and terminal set when all states except x position and velocity are set to 0.

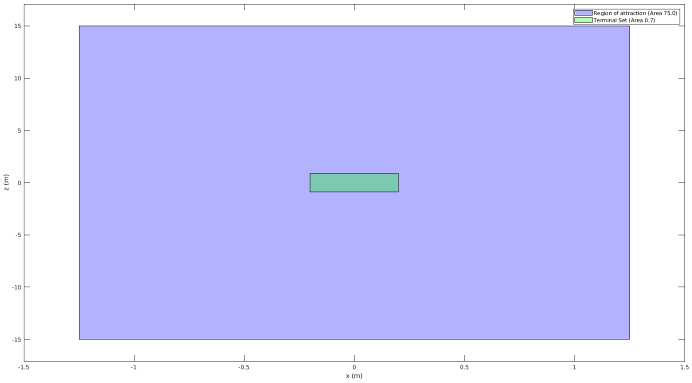

Fig. 4. Attraction set $\mathbb{X}_N$ when setting all states except x and z to zero

IV. Numerical Simulations

Multiple simulations were completed testing the various MPC controllers designed and to analyze the effect of multiple parameters.

A. *Linear vs Non-Linear System*

The non-linear and linearized system was compared in Figure 5 to verify graphically that the linearized system did provide a good estimation of the system in hover. It is observed that the linear system follows the linear system, even when the pitch angle and rate are high ($\geq$ 30 deg). As our MPC controller limits those states to much smaller values, the linear model should present even higher fidelity. However, it is worth mentioning that the altitude coupling with pitch is non-present in the linear model.

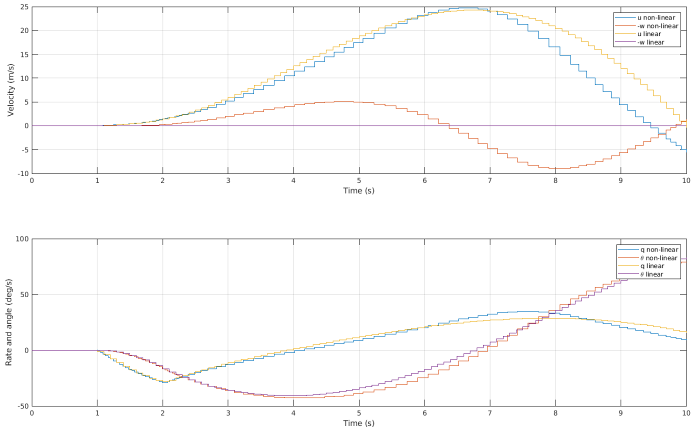

Fig. 5. Comparison of linear and non-linear system using a step input of 0.25 on the cyclic at $t = 1$ and release at $t = 2$

B. *Regulation MPC*

The regulation MPC is the first and most basic controller implemented. It is a good candidate to test the most basic parameters of the controller. First, it can be observed in Figure 10, that for an initial condition outside $\mathbb{X}_f$, but in $\mathbb{X}_N$, the regulation MPC controller provides asymptotic stability and converges to the origin. This confirms empirically the theoretical proof of subsection III-D.

1) Modifying Q and R weights: The whole Q matrix is progressively modified from low to higher values and is shown on Figure 6. As Q increases, the time response of the states is decreased and the input effort is increased. This is expected, as Q puts more weight in the optimization on states being close to 0, at the expense of inputs. The opposite behavior can be observed on Figure 7, where R is varied. Interestingly, the response between $R = 10^2$ and $R = 10^4$ is almost the same, although the actuators have not attained their constraints. This is actually due to the pitch having attained a state constraint. As the state constraint is attained, it actually limits the speed at which the helicopter can come back to the origin.

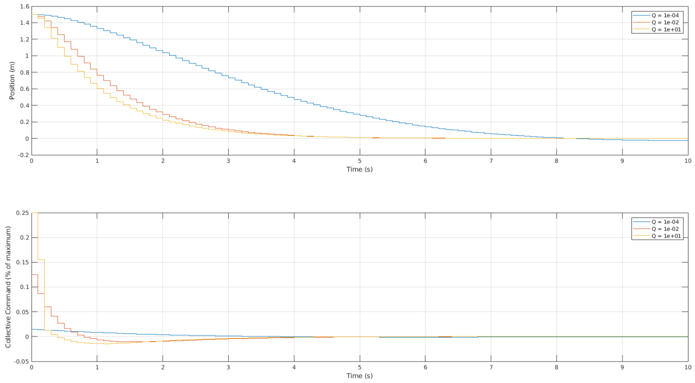

Fig. 6. Varying state weight Q through $\begin{bmatrix}10^{-4} & 10^{-2} & 10^{1}\end{bmatrix}$ from initial condition of 1.5 m in x

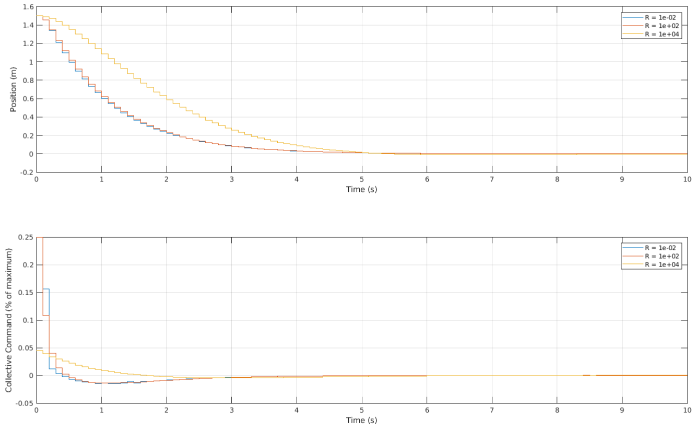

Fig. 7. Varying input weight R through $\begin{bmatrix}10^{-2} & 10^{2} & 10^{4}\end{bmatrix}$ from initial condition of 1.5 m in x

2) Modifying Control Horizon N: By modifying the control horizon N, the attraction set changes size and therefore, for a set initial condition, stability might no longer be guaranteed. This is explored in Figure 8, where N is successively decreased from its nominal value of $N = 25$. As the control horizon decreases, the settling time increases and oscillations increases. At $N = 5$, the system is unstable. The difference between $N = 15$ and $N = 20$ is marginal for an initial condition of $x = 2$, although $N = 20$ increases the overall attraction set and was therefore kept.

3) Comparison to LQR: As the terminal cost was set using the solution of the discrete Riccati equation, the LQR and MPC controller should perform the same when in $\mathbb{X}_f$. This is verified in Figure 9. When outside $\mathbb{X}_f$ but still inside $\mathbb{X}_N$ (shown in Figure 10), the LQR controller becomes aggressive and does not respect the input constraints. Its overall input effort is increased, while the MPC stays inside constraints and with a input effort.

C. *Trajectory Tracking MPC*

The trajectory tracking MPC is tested in two ways: by modifying the advance reference knowledge it possesses and by making it track a complex accelerating trajectory to stress test the controller.

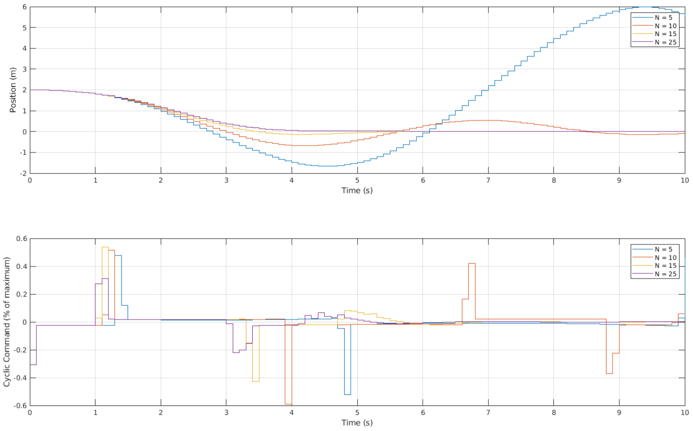

Fig. 8. Varying control horizon N through $\begin{bmatrix}5 & 10 & 15 & 25\end{bmatrix}$ from initial condition of 2 m in x

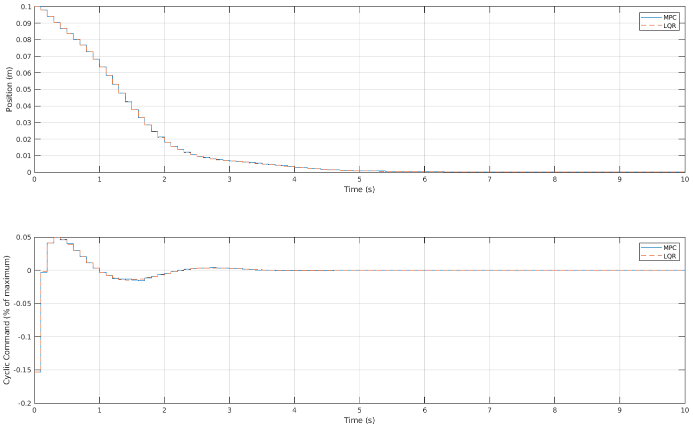

Fig. 9. Comparing LQR and MPC controller when initial condition is in $\mathbb{X}_f$

1) *Modifying Reference Horizon N_{ref}:* By increasing the reference horizon N_{ref}, the MPC now performs the optimization by taking into account the future reference and should therefore try to change its states even before the reference changes. This is shown on Figure 11. As N_{ref} increases, the controller issues command before the reference change, leading to an overall decreased input effort. The downside to this advance knowledge is a decrease in the response time, as the state response becomes shallower. This issue has been identified in related work and solutions based on a two-step design process (optimize the feed forward component of MPC after the classic MPC control loop) have been proposed [13].

2) *Circular Trajectory Tracking:* To stress test the controller, the MPC controller is asked to follow a sinusoidal trajectory increasing in speed as time increases in Figure 12, with a reference horizon $N_{ref} = 25$. At first, the MPC controller can follow the reference while confirming to state and input rules. As speed of the trajectory increases, the phase lag of the controller increases for both x and z while magnitude decreases for z. At approximately 120s, the controller hits a state constraint on the maximum pitch rate (it can be observed as the saturation on one of the pitch rate triangular curve). Effectively, the translated $\mathbb{X}_f$ due to the reference is no longer in the set of admissible states $\mathbb{X}$, which leads to instability, as mentionned in subsection II-B. The controller is no longer able to track the reference and becomes unstable, the onset of which can be observed on the cyclic and collective command around 120s.

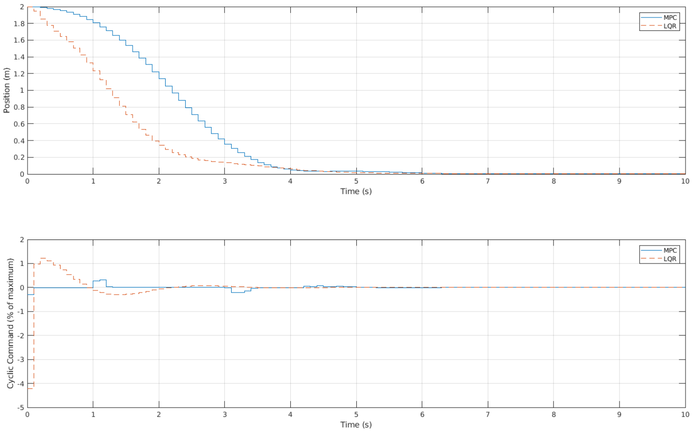

Fig. 10. Comparing LQR and MPC controller when initial condition is outside $\mathbb{X}_f$, but in $\mathbb{X}_N$

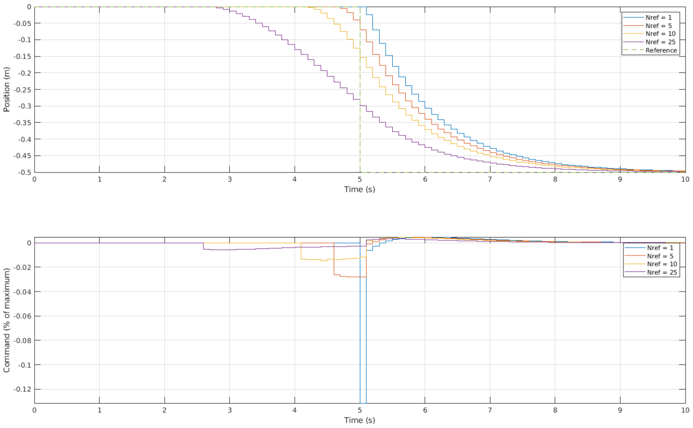

Fig. 11. Varying reference horizon N_{ref} through $\begin{bmatrix}1 & 5 & 10 & 25\end{bmatrix}$ from initial condition of 1.5 m in x

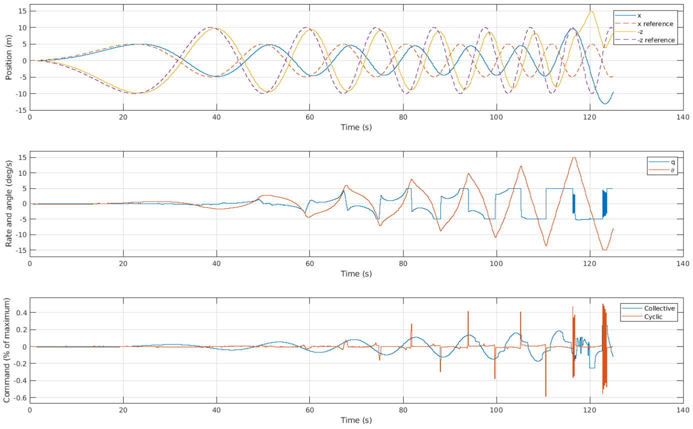

Fig. 12. MPC following circular trajectory ($x_{ref} = 5sin((t/100)\,2\,\pi\,t/20)$ and $y_{ref} = 10sin((t/100)\,2\,\pi\,t/20)$) accelerating in time, with reference horizon $N_{ref} = 25$

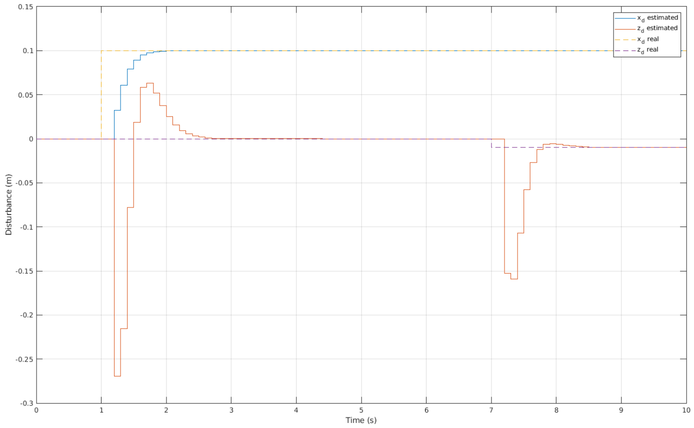

Fig. 13. Disturbance recognition for offset-free MPC, with step x_d disturbance of 0.1 at time $t = 1$ and step z_d disturbance of 0.01 at time $t = 7$

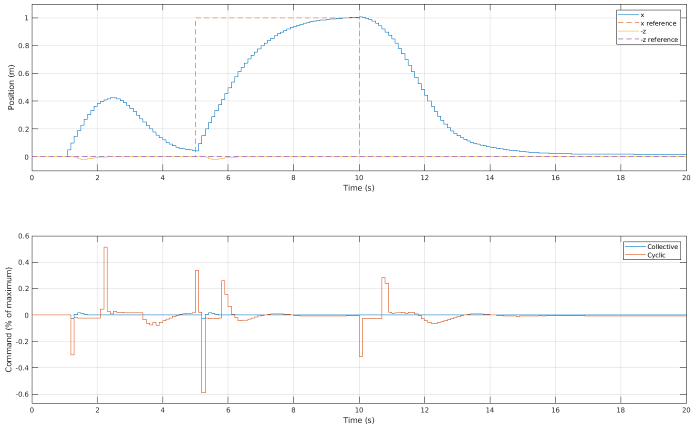

Fig. 14. Offset-free MPC following x_{ref} step trajectory from time $t = 5$ to $t = 10$, with x_d disturbance acting at time $t = 1$ to $t = 5$

D. *Offset-Free Output Tracking MPC*

Offset-Free output tracking MPC estimates the states and disturbances using a Luenberger observer. In Figure 13, the system is faced with a frontal gust (x_d) at time $t = 1$, which the system detects in around 1 sec (settling time), which is significantly less than the average settling time of 5 sec of the system. This proves that the MPC controller can therefore recognize a disturbance rapidly and correct for its action on the system. This proves a-posteriori the assumption required to prove the stability of the offset-free output tracking MPC, as described in subsection II-C. It is worth mentioning that the upward gust (z_d) estimator responds with high amplitude to any disturbance, which might lead to control issues if the upward disturbance is of high amplitude. This problem was identified but not explored further.

Figure 14 shows the whole system subject to a frontal $x_d = 0.1$ (equivalent to 1m/s wind) disturbance from time $t = 1$ to $t = 5$ and to a step reference change at time $t = 5$ and back to the origin at $t = 10$. The MPC controller is able to correct for the disturbance and then follow the reference, with a settling time of approx. 5 sec, the same settling time as the regulation MPC, showing that the offset-free MPC was correctly implemented and shows equivalent performance.

V. Conclusion

The non-linear model of the longitudinal dynamics of a helicopter was derived and then linearized around the hover condition. A regulation, trajectory tracking and offset-free MPC controllers were designed specifically for this flight condition with input and state constraints to keep the helicopter near its linearization point. Multiple MPC controllers were designed and proved to be stable when the states start in their attractor set.

The regulation controller made it possible to analyze the effect of various parameters of the MPC controller on its state and input performance and stability. Changing Q and R modified the aggressivity of the MPC controller to come back to the origin while R mostly affected the stability of the system, by changing the attractor set. The trajectory MPC was a chance to test advance knowledge of the reference and how it decreased overall input effort at the expanse of a shallower and slower settling time. Finally, the offset-free MPC was designed to properly recognize when the system is subject to a frontal and/or vertical wind gust, while following a reference, all using output feedback.

Overall, this paper was a good way to get familiarized with the basic concepts of MPC on a non-linear and coupled system while identifying potential improvements that could be applied to the MPC controllers designed. As future improvement, these MPC algorithms could be applied to a full 6DoF helicopter model with higher order rotor dynamics.

References

[1] M. Innocenti, "Helicopter flight dynamics: The theory and application of flying qualities and simulation modeling," *Journal of Guidance, Control, and Dynamics*, vol. 22, no. 2, pp. 383–384, 1999. [Online]. Available: https://doi.org/10.2514/2.4396

[2] R. W. Prouty and H. Curtiss Jr, "Helicopter control systems: A history," *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 1, pp. 12–18, 2003.

[3] B. Kim, Y. Chang, and M. H. Lee, "System identification and 6-dof hovering controller design of unmanned model helicopter," *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, vol. 49, no. 4, pp. 1048–1057, 2006.

[4] D. Simon, "Model predictive control in flight control design," *Licentiate Thesis, Linköping University, Dept. of Electrical Engineering*, vol. 17, 2014.

[5] T. Oktay and C. Sultan, "Constrained predictive control of helicopters," *Aircraft Engineering and Aerospace Technology*, vol. 85, no. 1, pp. 32–47, Jan 2013. [Online]. Available: https://doi.org/10.1108/00022661311294021

[6] H. Zhou, H. Pei, and Y. Zhao, "Trajectory tracking control of a small unmanned helicopter using mpc and backstepping," in *Proceedings of the 2011 American Control Conference*, June 2011, pp. 1593–1597.

[7] P. Marilena, M. B., H. Th. van, and M. J.A., "Helicopter performance, stability and control," *TU Delft*, 2002.

[8] Airbus, "Eurocopter 145 technical data," *Airbus*, 2006.

[9] R. W. Prouty, *Helicopter performance, stability, and control*, 1995.

[10] J. Rawlings and D. Mayne, *Model Predictive Control: Theory and Design*. Nob Hill Publishing, 2008.

[11] E. Gilbert and K. Tan, "Linear systems with state and control constraints: the theory and application of maximal output admissible sets," *IEEE Transactions on Automatic Control*, vol. 36, no. 9, pp. 1008–1020, Sep. 1991.

[12] F. Larocque, "A model predictive control approach for longitudinal control of helicopter system," https://github.com/lampadaire45/helicopter_MPC, April 2022.

[13] J. Rossiter and G. Valencia-Palomo, "Feed forward design in mpc," *2009 European Control Conference, ECC 2009*, 01 2009.