Non-Cooperative Model Predictive Control for Capturing a Remotely Piloted Target Drone

Patrícia Rodrigues[1] and Bruno Guerreiro[1,2]

[1] NOVA School of Science and Technology, UNINOVA-CTS, LASI, NOVA University Lisbon, Caparica, Portugal
[2] Institute for Systems and Robotics (ISR/LARSYS), Instituto Superior Técnico, University of Lisbon, Lisbon, Portugal
E-mails: pag.rodrigues@campus.fct.unl.pt, bj.guerreiro@fct.unl.pt

Abstract. This paper addresses the use of model predictive control (MPC) for pursuit-evasion games, where a shuttle drone aims at capturing a remotely piloted target drone. Several models for shuttle and target vehicles are developed, considering different degrees of complexity and reliance in inner-loop controllers. Based on these models MPC strategies are defined for each vehicle, for joint tracking operation, as well as for differential pursuit-evasion games between shuttle and target vehicles. Lastly the human-in-the-loop is added to the model and respective MPC algorithms of the fixed-wing target. Simulation results are provided, showing several scenarios where each vehicle has the advantage in terms of physical capabilities, or the disadvantage of being remotely-piloted.

Keywords: Model Predictive Control, Differential games, Pursuit-Evasion Games, Unmanned Aerial Vehicles, Human-in-the-loop

1 Introduction

The interest in unmanned aerial vehicles (UAVs), or simply drones, has been growing from a research and military standpoint. Drones are being increasingly used for aerial surveillance, 3D mapping, among other things. Doing such tasks autonomously requires reliable navigation from the drones, particularly in unknown areas. One example of the usage of drones is in the event of natural disasters, and in some cases the prevention of one. The authors in [13] show various examples of how drones can be used in this scenario. Many industries use drones in their day-to-day business, for example, the mining industry to do ore control, 3D mapping, blasting management, etc [14]. However, these devices aren't used only in the research and industry fields, as many people have incorporated drones in their day-to-day lives, which raises concerns in safety and privacy. A good example of that are airports, where the sighting of a drone within the airport's area can cause a temporary shutdown [6].

The focus of this paper is to consider the use of a shuttle drone, assumed to be a rotary-wing unmanned aerial vehicle (RUAV), equipped with capture mechanism and able to model the typical human reaction times and predict the most

probable motion of a remotely piloted target drone. The control of UAVs has been constantly evolving, and lately departing from classical approaches, such as the proportional–integral–derivative (PID) controller [15], nonlinear control, adaptive control, or nonlinear model predictive control (NMPC) [3]. Model predictive control allows for the representation of complex and dynamic systems with multiple inputs and outputs while taking into account sets of constraints created by either the system itself or its surroundings [11,8].

Human behavior is complex and, often, unpredictable, yet simplified models of the human sensori-motor reactions are available in the literature. In [1] a human works cooperatively with the controller or takes a more passive role, being observed in driving scenarios. In [2] a compensatory control is studied, where the human controller acts only on the error between the system output and the reference. For this type of control two models are shown, one is called the extended crossover model that approximates linear controller dynamics. The last one is a second order model that describes human dynamics in a wider frequency range while also considering neuromuscular dynamics, called the simplified precision model. When dealing with dynamical systems models, a simple way of incorporating the human-in-the-loop is to introduce a delay which is meant to stand in as the reaction time (RT). Reaction time can be defined as the interval of time between the appearance of the stimulus and the voluntary response given by the subject, which is found to be typically between 250.12 ± 18.50 ms and 229.80 ± 16.73 ms, depending on the activity levels of the individuals [7].

Differential games allow us to generalize an optimal control problem as seen from two conflicting perspectives. This is the case for the underlying pursuit-evasion games (PEGs) addressed in this paper, where the pursuer shuttle drone has the goal of capturing a target drone, whereas the target drone will try to frustrate this goal [16]. Alternatively, [12] focuses on cooperative differential games where the players maximize the sum of their payoffs, and the game develops along a cooperative trajectory. The authors of [17] combined PEGs with map building in a probabilistic approach, whereas in [9] a distributed formation is studied using differential game theory where UAVs are able to communicate with one another by using a directed graph.

The contributions of this paper include the development of simple shuttle and target models, including a basic human-in-the-loop behavior model for the latter, the definition of PEGs differential game based on NMPC where the shuttle and target drones are engaged in capturing of evading each other. Simulation results for various scenarios are provided to validate the proposed approach.

The remainder of the paper is organized as follows. The models of a target fixed-wing drone and the shuttle rotary-wing drone are provided in Section 2, followed by the explanation of the NMPC controller for each vehicle with and without human-in-the-loop (HIL), in Section 3. Section 4 shows the results for the Pursuit-Evasion Game (PEG) with and without HIL as well. Finally, some concluding remarks and future work is provided in Section 5.

2 Vehicles Models

This section starts by introducing the modeling concepts for the RUAV drone and the fixed-wing UAV, defining the coordinate frames used to deduce these models, followed by the modeling of the human being.

Before modeling any of the vehicles it is important to explain and categorize the reference frames applied. In this case the mass center of the earth is considered as the origin of the frame, as the Earth-Centered Earth-Fixed (ECEF) with the z-axis pointing towards the north pole and the x-y plane coinciding with the earth's equatorial plane, where the x-axis is aligned with the international reference meridian. For aerospace vehicles the usual convention used is the North-East-Down (NED), however given the fact that we have different vehicles, the East-North-Up (ENU) was used as the inertial frame (I), represented in Figure 1 with the axes $x_i = \begin{bmatrix}1 & 0 & 0\end{bmatrix}^T$, $y_i = \begin{bmatrix}0 & 1 & 0\end{bmatrix}^T$, and $z_i = \begin{bmatrix}0 & 0 & 1\end{bmatrix}^T$. In this convention the x-axis is pointing east, the y-axis is pointing north, and the z-axis is pointing up. The body frame is attached to the vehicle, originating at the center of mass of the object. The axes x_{B_a} and y_{B_a} are in the plane defined by the center of the vehicle, while z_{B_a} is perpendicular to that plane pointing upwards as seen in 1.

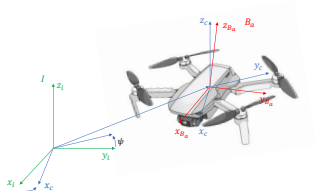


Fig. 1: Inertial and body frame referential for the aerial vehicle.

The following models are continuous, and a discretization of our system is needed considering the tools we have used to simulate the vehicles behaviors. The generalized model equation is given by

$$\dot{x}_i(t) = f_{i_c}(x_i(t), u_i(t)) \tag{1}$$

where f_{i_c} is the corresponding continuous model equations of the vehicle i, where f_{a_c} is for the RUAV and f_{fw_c} is for the fixed-wing UAV. As mentioned before the next step is to discretize the non-linear model which is now

$$x_i(k+1) = f_i(x_i(k), u_i(k)) \tag{2}$$

where f_i is obtained using the Euler forward method and is given by

$$f_i(x_i(k), u_i(k)) = x_i(k) + T_s f_{i_c}(x_i(k), u_i(k)) \tag{3}$$

where T_s is the sampling time of the simulation.

Let p_a and v_a be the linear position and velocity of the shuttle RUAV vehicle in frame I, while its attitude in terms of Euler angles roll (ϕ), pitch (θ), and yaw (ψ) is $\lambda_a \in \mathbb{R}^3$, and the angular velocity of the vehicle described in the shuttle body frame A is $\omega_a \in \mathbb{R}^3$. Using the kinematic and Newton-Euler equations the shuttle drone model can be written as:

$$\begin{aligned} \dot{p}_a &= v_a \\ \dot{v}_a &= m^{-1} R(\lambda_a)(f_e + f_a) - g z_i \\ \dot{\lambda}_a &= Q(\lambda_a)^{-1} \omega_a \\ \dot{\omega}_a &= I^{-1}(-\omega_a \times I\omega_a + \tau_e + \tau_a) \end{aligned} \tag{4}$$

where m is the mass of the drone, g is the gravity's speed, I is the inertia matrix of the vehicle, $R(\lambda_a) \in SO(3)$ denotes the rotation matrix based on the Z-Y-X Euler angles vector, representing the attitude of the body frame relative to the inertial frame, whereas $Q(\lambda_a)$ is the matrix that maps the Euler angle rates into the angular velocities of the vehicle described in the body frame. The aerodynamic force ($f_a \in \mathbb{R}^3$) and torque ($\tau_a \in \mathbb{R}^3$) produced by mainly by the aerodynamic drag are, respectively, $f_a == -a_T R(\lambda_a)^T v_{air}$ with $v_{air} = v_a - v_w$ where v_w is the wind's velocity vector in the inertial frame, and $\tau_a = -a_R \omega_a$, where a_T and a_R are the linear and rotational air friction coefficients, respectively. For the external force ($f_e \in \mathbb{R}^3$) and torque ($\tau_e \in \mathbb{R}^3$) we have that $f_e = \begin{bmatrix} 0 & 0 & f_{ez} \end{bmatrix}^T$ and $\tau_e \in \mathbb{R}^3$. The values for these parameters are further detailed in [10].

The shuttle RUAV model is highly non-linear and has a significant complexity, which creates a need for high computational power. To circumvent this issue, and since the objective is to control the RUAV movements so it follows the other vehicle, the model for the NMPC was simplified. This is possible given that we have a non-linear controller that allows for the combined control of the complete system. This version of the RUAV model becomes

$$\begin{aligned} \dot{p}_a &= v_a \\ \dot{\psi}_a &= \omega_{a_z} \end{aligned} \tag{5}$$

where the states for the RUAV are now $x_a = \begin{bmatrix} p_a & \psi_a \end{bmatrix}^T$ and the inputs are $u_a = \begin{bmatrix} v_a & \omega_{a_z} \end{bmatrix}^T$. For this model the same discretization process was applied except for the simplified version, f_{a_c} is found in Equation (5).

The target fixed-wing vehicle is modelled considering similar state variables as for the RUAV, denoting the 3-D position of the vehicle in the inertial frame as $p_{fw} \in \mathbb{R}^3$ and the 3-D linear velocity vector in the vehicle body frame FW as $v_{fw} \in \mathbb{R}^3$. Nonetheless, the angular motion is simplified to account for motion constraints related to the air flow, defining the angular velocity as $\omega_{fw} \in \mathbb{R}^2 = \begin{bmatrix} \omega_{fw_y} & \omega_{fw_z} \end{bmatrix}^T$, and the angle vector $\lambda_{fw} = \begin{bmatrix} \gamma & \psi \end{bmatrix}^T$, where γ is the flight-path

angle and ψ is the yaw angle. The model is then defined as

$$\begin{aligned} \dot{p}_{fw} &= R_{fw}(\lambda_{fw}) v_{fw} \\ \dot{v}_{fw} &= a_{fw} \\ \dot{\lambda}_{fw} &= \omega_{fw} \end{aligned} \tag{6}$$

where $R_{fw}(\lambda_{fw})$ is the rotation matrix between the wind frame and the inertial frame. The vehicle state vector is then $x_{fw} = \left[p_{fw}^T \; v_{fw}^T \; \lambda_{fw}^T\right]^T$, whereas the input vector combines the linear acceleration vector, $a_{fw} \in \mathbb{R}^3$, and the angular velocities. $\omega_{fw} \in \mathbb{R}^2$, as $u_{fw} = \left[a_{fw}^T \; \omega_{fw}^T\right]^T$.

Considering the human-in-the-loop (HIL) modeling, there are two fundamental models used throughout the literature: the extended crossover model and the simplified precision model. While the extended crossover model accounts for the delay in the reaction time, simplified precision model is a second order model that accounts for the neuro-muscular dynamic. These two models are then connected to form the full compensatory model [2].

In this paper, a simple human reaction model is achieved by introducing a reaction time constant as a first order model, with continuous-time transfer function given by

$$\frac{X_u(s)}{U(s)} = \frac{1}{T_H s + 1} \tag{7}$$

where the value of T_H represents the human reaction time. This is a simplistic version of modeling the human reaction so that the NMPC has a straightforward notion of the disadvantage of HIL, which can still be identified in the future with real data of human piloting. When applied to our system, we define an additional state variable $x_{u_{fw}} = \left[a_{fw}^H \; \omega_{fw}^H\right]$, resulting in the augmented model equations given by

$$\begin{aligned} \dot{p}_{fw} &= R_{fw}(\lambda_{fw}) v_{fw} \\ \dot{v}_{fw} &= a_{fw}^H \\ \dot{\lambda}_{fw} &= \omega_{fw}^H \\ \dot{x}_{u_{fw}} &= -\frac{1}{T_H} x_{u_{fw}} + \frac{1}{T_H} u_{fw} \end{aligned} \tag{8}$$

3 NMPC for drone capture

In this section a NMPC strategy is developed, in a first stage for each vehicle individually, and in a second stage, a PEG differential game based on NMPC is proposed for the two vehicles.

3.1 Single vehicle NMPC

For a general vehicle, taking into account the output obtained at a given time k, the controller predicts the future actions of the system over the prediction

horizon and determines which input to apply to the next instance. The optimal control problem for a given time horizon, N, consists of finding the optimal state control sequence u_k^* for $k \in 0, ..., N-1$ that drives the system along a state trajectory x_k^* for $k \in 0, ..., N$ according to the system dynamics and within input and state constraints, such that the specified performance index J_0 is minimized relative to a reference output signal $\bar{y}_k$, that is

$$\begin{aligned} \min_{u_0,...,u_{N-1}} \quad & J_0 = (y_N - \bar{y}_N)^T P (y_N - \bar{y}_N) + \sum_{K=0}^{N-1} (y_k - \bar{y}_k)^T Q (y_k - \bar{y}_k) + u_k^T R u_k \\ s.t \quad & x_{k+1} = f(x_k, u_k), \forall k = 0, ..., N-1 \\ & y_k = C x_k \\ & x_0 = z_0 \end{aligned}$$

where z_0 is the initial condition, C is the output matrix, and $f(x_k, u_k)$ is the equality constraint. In order to accomplish this, we use the receding horizon control strategy defined in Algorithm 1. The initial condition instantiates the system model, given by $x_0(k) = x(k-1)$, which is used to predict the behavior, with the actual systems state at that sampling instant.

Algorithm 1 Receding horizon control

Require: Current state x

1: Consider the initial condition x_0 is the current state x_{t_k}.
2: Obtain the optimal control sequence U^* solving the non-linear optimal control problem by minimizing the cost function over the prediction horizon.
3: If the problem is unfeasible then terminate the algorithm.
4: Apply the first control action from U^* to the system, $u_{t_k} = u_0^*$
5: Repeat from step 1.

Considering the stacked state, output, and input sequences defined respectively as $X = \begin{bmatrix} x_0^T \cdots x_N^T \end{bmatrix}^T$, $Y = \begin{bmatrix} y_0^T \cdots y_N^T \end{bmatrix}^T = HX$, and $U = \begin{bmatrix} u_0^T \cdots u_{N-1}^T \end{bmatrix}^T$, the optimal problem can be rewritten in batch notation as

$$\begin{aligned} \min_{U} \quad & J = (Y - \bar{Y})^T \bar{Q} (Y - \bar{Y}) + U^T \bar{R} U \\ s.t \quad & F(X, U) = 0 \\ & Y = HX \end{aligned} \tag{9}$$

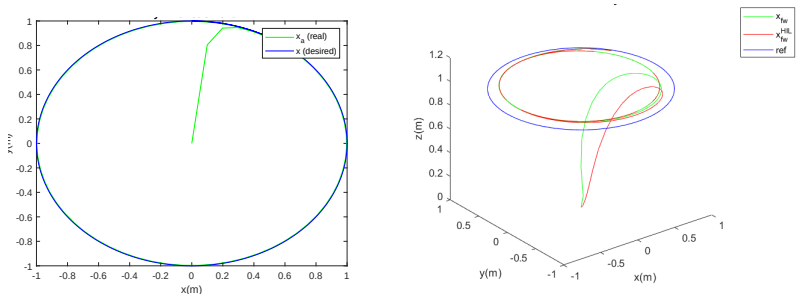
where the function for the model equality constraint is given by

$$F(X, U) = \begin{bmatrix} x_0 - z_0 \\ x_1 - f(x_0, u_0) \\ \vdots \\ x_N - f(x_{N-1}, u_{N-1}) \end{bmatrix}. \tag{10}$$

Stability of these strategies is analyzed through the initial feasibility of the optimal control problem, in other words, the input trajectory exists, and all the constraints are satisfied [4]. One of the big advantages for this type of control is that it is versatile.

Considering the RUAV shuttle and FW target vehicles introduced in the previous section, we define the model equality constraint using the models defined in Equations (6) and (5), respectively. Furthermore, considering a sampling time of 0.1 seconds and the prediction horizon of 15 samples (or 1.5 seconds), Algorithm 1 is applied to each of the individual vehicles.

The results for tracking a trajectory can be seen for each vehicle, respectively, in Figures 2a and 2b, which show that both vehicles are able to effectevely follow a reference trajectory. For the FW target drone, both the regular model and the augmented model considering HIL are considered, from where we can clearly identify the delayed response of the latter case.



(a) RUAV controller (b) FW controller

Fig. 2: Trajectories of RUAV and FW individual NMPC.

3.2 Differential game for drone capture

For the PEGs each player will have an objective and apply control towards that goal. Considering players A and B, the joint dynamical system can be given by

$$x_{k+1} = f(k, x_k, A, B)$$

with initial condition $x_0 = x(t_0)$ and payoff $P(A, B)$ that can be described as

$$P(A, B) = g(N, x_N) + \sum_{k=0}^{N} h(k, x_k, A, B)$$

where A has the goal to maximize the $P(A, B)$ while B wants to minimize it. The difficulty is that in a given instant k both players are aware of their previous

Algorithm 2 Pursuer controller

1: **procedure** SOLVE MAXMIN $J(E, U_p, U_e)$
2: Maximize $J_e(E, U_p^*, U_e)$ to obtain U_e^*, considering the previously calculated U_p^*.
3: Minimize $J_p(E, U_p, U_e^*)$ to obtain U_p^*
4: Repeat the steps 2 and 3 k times, with k being the level of thinking
5: Set final U_p^* as the optimal input trajectory
6: **end procedure**
7: Set $u_p(1)$ as the current pursuer input

control actions as well as their opponents, but don't know what the future control of the adversary is going to be [5]. It is important to describe the optimization problem again since it changes slightly from the one described before. There are now two types of cost for the differential game. For the pursuer the specified cost J_p must be minimized while considering a defined value for the control action of the evader. In the case of the evader, the cost J_e must be maximized while considering a fixed control from the pursuer. Using batch notation, where the state, output, and input variables for the complete horizon are stacked together, the optimal control problem for the pursuer can be described as

$$\begin{aligned} U_p^* = \underset{U_p}{\text{argmin}} \quad & J_p := (Y_p - Y_e)^T \bar{Q}_p (Y_p - Y_e) + U_p^T \bar{R}_p U_p \\ s.t \quad & F_e(X_e, U_e^*) = 0 \ , \ Y_e = H_e X_e \\ & F_p(X_p, U_p) = 0 \ , \ Y_p = H_p X_p \end{aligned} \tag{11}$$

where the function for the equality constraints, F_E, can be defined by the stacking of both vehicles' equality constraints, or $F_E = \begin{bmatrix} F_p & F_e \end{bmatrix}$, and the input sequence for the evader, U_e^*, is that of the previous iteration of the algorithm. In a similar fashion, the evader optimal control problem can be defined as

$$\begin{aligned} U_e^* = \underset{U_e}{\text{argmax}} \quad & J_e = (Y_p - Y_e)^T \bar{Q}_e (Y_p - Y_e) + U_e^T \bar{R}_e U_e \\ s.t \quad & F_e(X_e, U_e) = 0 \ , \ Y_e = H_e X_e \\ & F_p(X_p, U_p^*) = 0 \ , \ Y_p = H_p X_p \end{aligned} \tag{12}$$

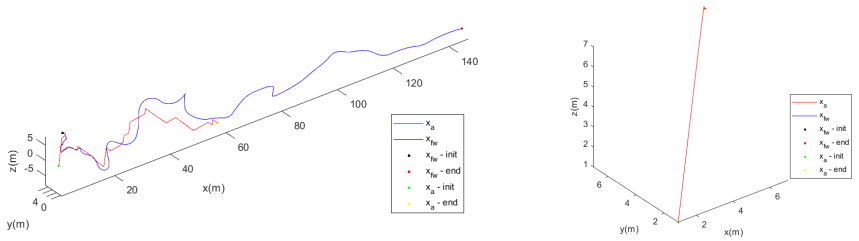
The evader and the pursuer have two separate algorithms. Similarly to the cost, the algorithms are mirrors of each other, where for the evader steps 2 and 3 are switched and the previous optimal control is considered in the opposite vehicles' optimization. To solve these algorithms `fmincon` was once again used, this time each controller runs it twice, once to minimize the pursuer, and once to maximize the evader, and vice-versa.

Using the model and OCP described before we can obtain results for the simulation of the NMPC with a human-in-the-loop. The gains were kept the same in order to compare the control in the two cases. For this the time T_H considered was 0.2 seconds in accordance with the literature mentioned, where we consider a slightly faster human then the ones studied.The 3D plot can be

seen in Figure 2b, We can see that when the HIL is applied the fixed-wing UAV is slower to arrive at the reference, having a similar value than the simulation without HIL, only a few instances later, as was expected.

4 Results and Analysis

For the Shuttle drone vs. Fixed-wing Target with Acceleration Control scenario there will be two tests, one where the fixed-wing has the advantage, and one where the RUAV has the advantage, which will be done by changing the RUAV's velocity bounds. The target vehicles' input constraints will be $\pm \begin{bmatrix} 8 & 8 & 8 & 2\pi & 2\pi \end{bmatrix}^T$ for both scenarios. For a faster fixed-wing the RUAV's constraints will be $\pm \begin{bmatrix} 10 & 10 & 10 & 2\pi \end{bmatrix}^T$ and for a faster RUAV they are $\pm \begin{bmatrix} 16 & 16 & 16 & 2\pi \end{bmatrix}^T$. The output matrix, which until now was the identity matrix, changes so that only the four states are chosen, the three position states and the yaw angle. Figure 3 shows the XYZ plot of the simulations. We can see that the fixed-wing UAV moves less erratically, making less jagged lines, when compared with the behavior of the RUAV.

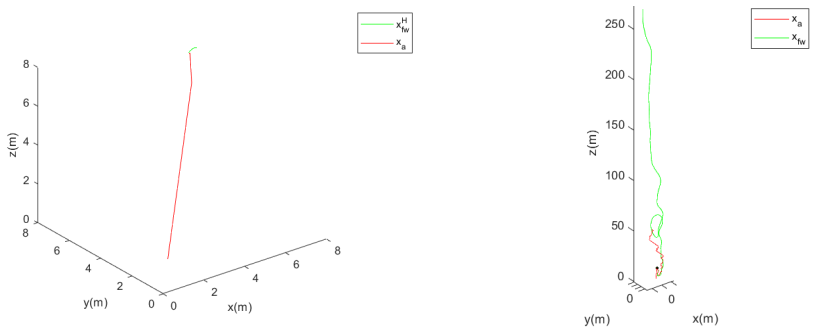


(a) Faster fixed-wing (b) Faster RUAV

Fig. 3: Trajectories of RUAV vs. FW

It is also important to analyze the runtime of each simulation. For the differential game for a faster fixed-wing UAV the total runtime is 2574 seconds (accounting for all iterations), which when compared to the NMPC of the same vehicle is 411 seconds, which is 6 times bigger. As expected, the more complex the system model and the scenario (following a reference vs. differential game) the more computational weight the simulation has and the more time it takes to process.

With the human-in-the-loop, and considering the same matrix values as the simulation without it where the fixed-wing won, we can see that the human controlled fixed-wing gets caught almost immediately, as seen in Figure 4a. To test the algorithm with HIL in the case where the human controlled vehicle



(a) Faster fixed-wing lost. (b) Even faster fixed wing wins.

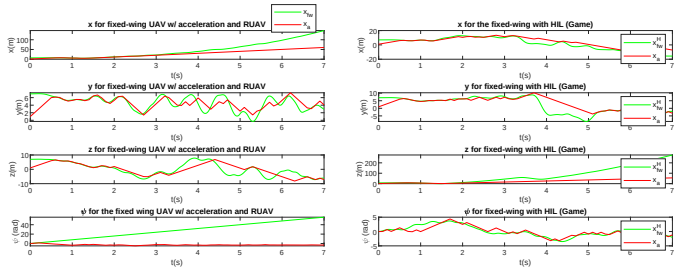
Fig. 4: Trajectories of RUAV vs. FW-HIL

escapes, another simulation scenario is considered where the input of the fixed-wing drone has two times less constraints, which are now $\pm \begin{bmatrix} 4\pi & 4\pi & 15 & 15 & 15 \end{bmatrix}^T$. In Figure 4b we can see that RUAV follows the fixed-wing very closely but that around the halfway point it manages to escape in the z-axis. In Figures 5a and 5b we can see the position states of the differential game with and without HIL, respectively. We can see that the vehicle shows the same behaviors in each simulation, with different results. Without HIL we see the fixed-wing UAV focusing on escaping in the x-axis, having an effect on its yaw rotation. With HIL the target vehicle escapes in the z-axis instead.

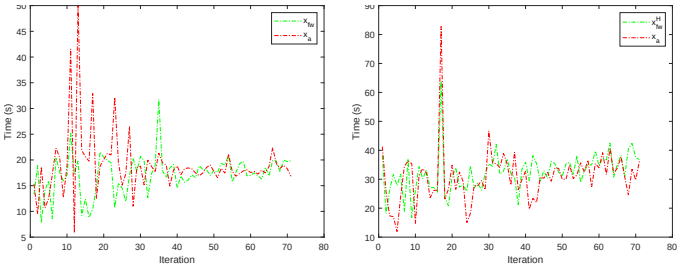
Finally, Figures 5c and 5d show the CPU time for each iteration for each case. We can see that overall the scenario with the human-in-the-loop takes longer to calculate its values at each iteration, since it also has more variables to consider. It is also of note that iteration takes more time than the sampling time because `Simulink` waits for the calculation to be done. This can be improved with more efficient libraries. It is important to note that with different optimization libraries, such as `YALMIP`, `CASADI`, or `ACADOS`, this problem will be able to be optimized and implemented in real-time.

5 Conclusion

This paper presents a differential game strategy based on NMPC to enable a shuttle drone to capture a target drone eventually piloted by a human operator. After defining the models for each vehicle, considering also a basic human delayed reaction time for the target drone, NMPC algorithms for each vehicle are developed. Then, a pursuit-evasion game between the target and shuttle drone is proposed, which is validated through simulation results. These results confirm the expected outcome, in which when the shuttle drone has an advantage in terms of its capabilities, it catches its target easily, whereas when the target vehicle has the advantage, it can escape. Nonetheless, when the human reaction



(a) Position without HIL (b) Position with HIL



(c) CPU time without HIL (d) CPU time with HIL

Fig. 5: Position and CPU time evolution.

time is considered, for the fixed-wing to escape it is necessary to diminish the input constraints roughly 100%.

Based on these conclusions there are some areas that can be further investigated. Firstly, additional work should be developed so that the control is able to work with the complete nonlinear models. In the same vein, the modeling of the human being should be further studied so that both the behavior and the resourcefulness of people are fully represented. Finally, one should consider more efficient and scalable optimization libraries.

Acknowledgements

This work was partially funded by the FCT projects CAPTURE (https://doi.org-/10.54499/PTDC/EEI-AUT/1732/2020), CTS (UIDB/EEA/00066/2020), and LARSYS (UIDB/50009/2020).

References

1. Chipalkatty, R., Daepp, H., Egerstedt, M., Book, W.: Human-in-the-loop: Mpc for shared control of a quadruped rescue robot. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4556–4561. IEEE (2011)

2. Drop, F.M.: Control-Theoretic Models of Feedforward in Manual Control, vol. 46. Logos Verlag Berlin GmbH (2016)
3. Eklund, J., Sprinkle, J., Sastry, S.: Implementing and testing a nonlinear model predictive tracking controller for aerial pursuit/evasion games on a fixed wing aircraft. In: Proceedings of the 2005, American Control Conference, 2005., pp. 1509–1514 vol. 3 (2005). DOI 10.1109/ACC.2005.1470179
4. Findeisen, R., Imsland, L., Allgower, F., Foss, B.A.: State and output feedback nonlinear model predictive control: An overview. European Journal of Control **9**(2), 190–206 (2003). DOI 10.3166/ejc.9.190-206
5. Friedman, A.: Differential games. Courier Corporation (2013)
6. HEANEY, S.: Drone sighting causes flights to be suspended at dublin airport. Irish Examiner (2023). URL https://www.irishexaminer.com/news/arid-41056655.html
7. Jain, A., Bansal, R., Kumar, A., Singh, K.: A comparative study of visual and auditory reaction times on the basis of gender and physical activity levels of medical first year students. International journal of applied and basic medical research **5**(2), 124–127 (2015). DOI 10.4103/2229-516X.157168
8. Limon, D., Alamo, T., Salas, F., Camacho, E.: Input to state stability of min–max mpc controllers for nonlinear systems with bounded uncertainties. Automatica **42**(5), 797–803 (2006). DOI 10.1016/j.automatica.2006.01.001
9. Lin, W.: Distributed uav formation control using differential game approach. Aerospace Science and Technology **35**, 54–62 (2014). DOI 10.1016/j.ast.2014.02.004
10. Magnusson, T.: Attitude control of a hexarotor. Master's thesis, Linköpings University (2014)
11. Mayne, D.Q.: Model predictive control: Recent developments and future promise. Automatica **50**(12), 2967–2986 (2014). DOI 10.1016/j.automatica.2014.10.128
12. Perelman, A., Shima, T., Rusnak, I.: Cooperative differential games strategies for active aircraft protection from a homing missile. Journal of Guidance, Control, and Dynamics **34**(3), 761–773 (2011)
13. Restas, A., et al.: Drone applications for supporting disaster management. World Journal of Engineering and Technology **3**(03), 316 (2015)
14. Shahmoradi, J., Talebi, E., Roghanchi, P., Hassanalian, M.: A comprehensive review of applications of drone technology in the mining industry. Drones **4**(3) (2020). DOI 10.3390/drones4030034
15. Szafranski, G., Czyba, R.: Different approaches of pid control uav type quadrotor. In: International Micro Air Vehicle conference and competitions (IMAV) (2011)
16. Tzannetos, G., Marantos, P., Kyriakopoulos, K.J.: A competitive differential game between an unmanned aerial and a ground vehicle using model predictive control. In: 2016 24th Mediterranean Conference on Control and Automation (MED), pp. 1053–1058 (2016). DOI 10.1109/MED.2016.7535979
17. Vidal, R., Shakernia, O., Kim, H., Shim, D., Sastry, S.: Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation. IEEE Transactions on Robotics and Automation **18**(5), 662–669 (2002). DOI 10.1109/TRA.2002.804040