Thirteenth USA/Europe Air Traffic Management Research and Development Seminar (ATM2019)

Iterative Learning Control for Precise Aircraft Trajectory Tracking in Continuous Climb Operations

Almudena Buelta, Alberto Olivares and Ernesto Staffetti
Universidad Rey Juan Carlos
Fuenlabrada, Madrid, Spain
almudenajose.buelta@urjc.es, alberto.olivares@urjc.es, ernesto.staffetti@urjc.es

***Abstract*—In this paper, an iterative learning control method is used to improve precision in aircraft trajectory tracking in which, given a departure procedure, the dynamical model of an aircraft and a trajectory to be followed, the problem consists in defining an iterative learning control scheme which is able to improve the precision of the aircraft in following the trajectory taking into account the deviations suffered by previous flights. It is assumed that all the flights are operated with the same aircraft model and that they successively follow the same trajectory with short time-based separation and therefore are subject to similar recurrent disturbances. In the iterative learning control scheme used in this paper, the control action consists in generating at each iteration a new reference trajectory for the aircraft which compensates for recurrent disturbances. Thus, it can be applied to systems with underlying controllers for trajectory tracking, such as aircraft. In this case, the feedback trajectory tracking control is intended to reduce non-repetitive disturbances while the iterative learning control is intended to reject repetitive disturbances. The iterative learning control problem is solved in two steps: disturbance estimation and aircraft reference trajectory update. Both steps rely on a nominal model of the aircraft in which input and state constraints are explicitly taken into account. Continuous climb operations, defined within a standard instrumental departure, are considered in the simulations. The result show the effectiveness of the method which is able to reduce the trajectory tracking error due to recurrent disturbances in a few iterations, thus improving their predictability. Higher predictability of aircraft trajectories would simplify both management and control of air traffic, would improve the capacity of the air traffic management system and would allow a better exploitation of the infrastructures. Greater predictability of aircraft trajectories would also allow airlines to define and follow trajectories with a smaller number of alterations. This would result in a reduction of costs and emissions.**

***Index Terms*—Aircraft trajectory tracking, iterative learning control, continuous climb operations.**

I. Introduction

The future operational concept of the Single European Sky ATM Research[1] (SESAR) and the other initiatives to increase the efficiency and sustainability of air transportation, is based on four-dimensional (4D) trajectories which result from the integration of time into three-dimensional (3D) aircraft trajectories herein referred to as paths. This means that any delay must be regarded as a deviation from the planned trajectory as much as a spatial deviation from the planned path. In trajectory-based operations, aircraft will be able to follow optimized trajectories based on the preferences of the airlines, with the obligation to precisely fulfill the arrival times at the waypoints that define them. Achieving precision in tracking planned trajectories is difficult due to the multiple random factors that affect the flight of an aircraft, such as the weather conditions. Deviations from the planned trajectories can neither be predicted accurately nor compensated by usual aircraft trajectory tracking controllers, which in general use a feedback control scheme and are limited by the causality of this control method that compensates for disturbances only after they occur [1]. These limitations are addressed in this paper by a learning control approach which is able to use information from previous flights that followed the same path and infer the correct control action to improve precision in trajectory tracking of subsequent flights. More specifically, the Iterative Learning Control (ILC) paradigm has been considered in this paper to improve the predictability of aircraft 4D trajectories, that is, the compliance of actual trajectories with the planned ones. In this paradigm, the control action consists in generating a new reference trajectory for the aircraft which compensates for recurrent disturbances. Improving predictability of 4D trajectories is of special importance in terminal maneuvering areas (TMAs), which are the controlled airspaces that surround one or more aerodromes. Due to the numerous arrival and departure routes they contain, they generally have a high density and are therefore more sensitive to 4D trajectories' spatial or temporal deviations. Furthermore, busy TMAs represent the ideal environment to implement ILC, since many aircraft follow the same routes with short temporal separation and therefore are likely to be subject to similar recurrent disturbances.

ILC is based on the idea that the performance of a control system that executes the same task multiple times can be improved by learning from previous executions. In particular, precision in the execution of a task can be improved

This work has been partially supported by the grant number TRA2017-91203-EXP of the Spanish Government.

[1] https://www.sesarju.eu/

by incorporating error information into the control law for subsequent iterations. In doing so, accurate performance can be achieved with low transient tracking error despite recurrent disturbances. ILC paradigm emerged from industrial robotics applications to improve trajectory tracking precision of robot manipulators in repetitive tasks [2]. It has been widely applied to improve precision in trajectory tracking over the last three decades and recently has been applied to precise trajectory tracking of aerial robots [3]. In [4], the main results in ILC analysis and design are surveyed. Stability, performance, learning transient behavior, and robustness in ILC are discussed and the most popular design techniques are described. They include PD-type, plant inversion methods, $\mathcal{H}_\infty$ methods and quadratically optimal design. Extensions to time-varying, continuous-time, multivariable, and nonlinear systems are outlined. Several textbooks on ILC are available, such as [5], which treats both ILC for linear and nonlinear systems, and [6], which focuses on real time applications.

In this paper, the possibility of applying the ILC paradigm to improve precision in aircraft trajectory tracking has been investigated. However, in this context, additional difficulties arise, since consecutive flights are carried out by different aircraft, so different dynamical systems perform each iteration, whereas basic ILC scheme requires the system dynamics to be repetition-invariant. Furthermore, consecutive flights along the same path follow in general different trajectories, so knowledge acquired by some aircraft following some trajectories must be transferred to other aircraft to follow other trajectories. Due to their complexity, these problems, called knowledge transfer among dynamical systems and among trajectories, are not addressed in this paper and will be subject of future research.

To implement the ILC algorithm to improve precision in trajectory tracking, information about intended and actually flown trajectories by other aircraft in the relevant airspace is needed, which is assumed to be available. Although it is currently not so, this information will be handled in the future Air Traffic Management (ATM) system through the System Wide Information Management (SWIM). This infrastructure will be available by 2030, when the International Civil Aviation Organization (ICAO) expects the concept of SWIM to be worldwide deployed. The purpose of SWIM, which is part of the ICAO Global Air Navigation Plan [7], is to harmonize the exchange of aeronautical, weather and flight information for all airspace users and stakeholders.

Thus, the problem considered in this paper can be stated as follows. Given a departure procedure, the dynamical model of an aircraft and a trajectory to be followed compliant with the procedure, define an ILC scheme which is able to improve the precision of the aircraft in following the trajectory taking into account the deviations suffered by other flights operated with the same aircraft model, assuming that all the flights successively follow the same trajectory with short time-based separation and therefore are subject to similar recurrent disturbances.

In particular, the ILC paradigm has been studied for Continuous Climb Operations (CCOs) in which the execution of a vertical flight profile optimized to the performance of aircraft is allowed, leading to significant reduction of fuel consumption and environmental benefits in terms of noise and emissions reduction. This case is of special interest because for an effective CCO implementation, accurate trajectory tracking control schemes must be employed to avoid potential conflicts between the arriving and departing aircraft together with appropriate airspace design and effective Air Traffic Control (ATC) procedures. In general, CCOs are part of a Standard Instrument Departure (SID) procedure [8].

Due to the obvious difficulty in implementing an ILC algorithm for accurate trajectory tracking using real commercial aircraft, the experiments whose results are reported in this paper have been carried out in a simulated environment. A realistic scenario has been built in the MATLAB/Simulink platform using commercial aircraft data from the Base of Aircraft Data (BADA), which is an Aircraft Performance Model (APM) developed and maintained by EUROCONTROL[2] with the collaboration of aircraft manufacturers and airlines, specifically designed for simulation and prediction of aircraft trajectories for research and operations in ATM. The BADA APM has two components: model specifications, which provide the theoretical models used to calculate aircraft performance parameters, and data sets, which give the aircraft-specific coefficients. There are two families of the BADA APM, based on the same modeling approach and with the same components: BADA Family 3 and BADA Family 4. The latest release of the first one has been used in this paper [9].

The trajectories to be followed have been generated using optimal control techniques. This approach permits the actual dynamical model of the aircraft to be taken into account, which ensures the feasibility of the planned trajectories. The resulting optimal control problems have been solved using DIDO, a MATLAB application based on the pseudospectral method developed by Elissar Global[3]. This application, besides the optimal control and the corresponding optimal trajectory, returns the Hamiltonian, costates, path covectors, and endpoint covectors. This information, together with classical tools such as Pontryagin's Maximum Principle, is essential to verify the optimality of the numerical results.

In this paper the optimization-based ILC method presented in [3] has been applied to aircraft trajectory tracking. In this method, the ILC problem is solved in two steps: disturbance estimation and aircraft control input update. Both steps rely on a nominal model of the aircraft in which input and state constraints are explicitly taken into account. An optimal estimator is used to guess the recurrent disturbance affecting the flight of an aircraft along a trajectory, and optimization techniques are employed to compute the control input for the following aircraft which optimally compensates for the recurrent disturbances in tracking the same trajectory. The ILC method has been implemented in Simulink, a graphical

[2]https://www-test.eurocontrol.int/services/bada
[3]http://www.elissarglobal.com/industry/products/

programming environment for modeling and simulating dynamical systems which, like MATLAB, has been developed by MathWorks[4].

An interesting aspect of this ILC method is that it can be made non intrusive with respect to the trajectory tracking control system of the aircraft by calculating, at each iteration, a reference trajectory for the following aircraft rather than a control input. Thus, it can be applied to systems with underlying feedback controllers for trajectory tracking, such as aircraft. The feedback trajectory tracking control is intended to reduce non-repetitive disturbances while the ILC is intended to reject repetitive disturbances.

In [3], an ILC approach has been applied to precise quadcopter trajectory tracking. It has been shown that trajectory tracking of aerial robots can be achieved by pure feed-forward adaptation of the input signal and that the accuracy of the trajectory tracking is limited by the level of non-repetitive disturbances. This work can be viewed as an extension of the results presented in [10], where a least-squares based learning rule was proposed to perform aggressive maneuvers with quadcopters which consist in steering these systems quickly from one state to another. Other ILC paradigms have been applied to trajectory tracking for Unmanned Aerial Vehicles UAVs. See for example [11] and references therein.

To the best knowledge of the authors, ILC has not been applied yet to precise aircraft trajectory tracking. Hence, its introduction would be an innovative solution to increase predictability of trajectories in the future trajectory-based operations paradigm. Higher predictability of trajectories would simplify both management and control of air traffic, would improve the capacity of the air traffic management system and would allow a better exploitation of the infrastructures. Greater predictability of trajectories would also allow airlines to define and follow trajectories with a smaller number of alterations. This would result in a reduction in costs and emissions.

The general ILC scheme is presented in Section II. A model of the aircraft dynamics is derived in Section III and the method for generating feasible aircraft reference trajectories is presented in Section IV. The experimental setup is described in Section V and the results of the application of the ILC algorithm to aircraft trajectory tracking in Section VI. Finally, Section VII contains the conclusions.

II. Iterative Learning

In this section, following [3], the ILC method implemented in this paper for precise aircraft trajectory tracking will be introduced. The starting point of the learning algorithm is a time-varying nonlinear model of a real dynamic system:

$$\begin{aligned} \dot{\mathrm{x}}(t) &= \mathrm{f}(\mathrm{x}(t), \mathrm{u}(t), t) \\ \mathrm{y}(t) &= \mathrm{g}(\mathrm{x}(t), \mathrm{u}(t), t), \end{aligned} \tag{1}$$

where $\mathrm{u}(t) \in \mathbb{R}^{n_u}$ is the control input, $\mathrm{x}(t) \in \mathbb{R}^{n_x}$ is the state, and $\mathrm{y}(t) \in \mathbb{R}^{n_y}$ is the output, and f and g are assumed to be continuously differentiable in x and u. Constraints on the input $\mathrm{u}(t)$ and the state $\mathrm{x}(t)$, and their time derivatives are represented by

$$\mathrm{Z}\ \mathrm{q}(t) \preceq \mathrm{q}_{max}, \tag{2}$$

where

$$\mathrm{q}(t) = \left[\mathrm{x}(t), \mathrm{u}(t), \dot{\mathrm{x}}(t), \dot{\mathrm{u}}(t), \ldots, \frac{d^m}{dt^m}\mathrm{x}(t), \frac{d^m}{dt^m}\mathrm{u}(t)\right] \tag{3}$$

and $\mathrm{q}_{max} \in \mathbb{R}^{n_q}$. The inequality is defined component-wise and n_q is the total number of constraints, Z is a constant matrix of appropriate dimensions.

The goal of the presented learning algorithm is to precisely track a feasible predefined output trajectory $y^*(t)$ over a finite time interval $t \in \mathcal{T} = [t_0, t_f]$, with $t_f < \infty$. The desired output trajectory $y^*(t)$ is here the result of solving an optimal control problem based on the system dynamics (1).

The system's behavior (1) can be represented as a linear time-varying system

$$\begin{aligned} \dot{\tilde{x}}(t) &= A(t)\tilde{x}(t) + B(t)\tilde{u}(t) \\ \dot{\tilde{y}}(t) &= C(t)\tilde{x}(t) + D(t)\tilde{u}(t), \quad t \in \mathcal{T}, \end{aligned} \tag{4}$$

where the time-dependent matrices $A(t), B(t), C(t), D(t)$ are the corresponding Jacobian matrices of the nonlinear functions f and g with respect to x and u. The triple $(\tilde{u}(t), \tilde{x}(t), \tilde{y}(t))$ represent small deviations from the desired trajectory and its corresponding state and input $(x^*(t), u^*(t), y^*(t))$,

$$\begin{aligned} \tilde{u}(t) &= \mathrm{u}(t) - u^*(t), \\ \tilde{x}(t) &= \mathrm{x}(t) - x^*(t), \\ \tilde{y}(t) &= \mathrm{y}(t) - y^*(t). \end{aligned} \tag{5}$$

In a real system, measurements are only available at fixed time intervals, therefore a discrete-time representation is needed, which results in the following linear, time-varying difference equations,

$$\begin{aligned} \tilde{x}(k+1) &= A_D(k)\tilde{x}(k) + B_D(k)\tilde{u}(k) \\ \tilde{y}(k) &= C_D(k)\tilde{x}(k) + D_D(k)\tilde{u}(k), \end{aligned} \tag{6}$$

where $k \in \mathcal{K} = \{0, 1, ..., N-1\}, N < \infty$ represents the discrete-time index. The desired trajectory is represented by a N-sample sequence

$$(u^*(k), x^*(k+1), y^*(k+1)), \quad k \in \mathcal{K}, \tag{7}$$

with given initial state $x^*(0)$. The input and state constraints (2) are similarly transformed

$$Z\tilde{q}(k) \preceq q_{max}(k), \tag{8}$$

where $\tilde{q}(k)$ is the deviation of $q(k)$ from the corresponding nominal values $q^*(k)$ defined analogously to (5), and discretized.

[4]https://es.mathworks.com/

A. Lifted system representation

It is useful to replace the model described above by a lifted vector representation, mapping the finite input time series $\tilde{u}(k), k \in \mathcal{K}$ into the corresponding output time series $\tilde{y}(k)$, $k \in \mathcal{K}$ for each trial [12]. The deviations with respect to the desired trajectory (7) are then

$$\begin{aligned} u &= [\tilde{u}(0), \tilde{u}(1), \ldots, \tilde{u}(N-1)]^T && \in \mathbb{R}^{Nn_u} \\ x &= [\tilde{x}(1), \tilde{x}(2), \ldots, \tilde{x}(N)]^T && \in \mathbb{R}^{Nn_x} \\ y &= [\tilde{y}(1), \tilde{y}(2), \ldots, \tilde{y}(N)]^T && \in \mathbb{R}^{Nn_y}. \end{aligned} \tag{9}$$

Using this notation, the linear system (6) can be described as

$$\begin{aligned} x &= Fu + d^0 \\ y &= Gx + Hu, \end{aligned} \tag{10}$$

The lifted matrix $F \in \mathbb{R}^{Nn_x \times Nn_u}$ is composed of the matrices $F_{(l,m)} \in \mathbb{R}^{n_x \times n_u}$, $1 \leq l, m \leq N$, that is

$$F = \begin{bmatrix} F_{(1,1)} & \ldots & F_{(1,N)} \\ \vdots & \ddots & \vdots \\ F_{(N,1)} & \ldots & F_{(N,N)} \end{bmatrix}, \tag{11}$$

where

$$F(l,m) = \begin{cases} A_D(l-1)\ldots A_D(m)B_D(m-1) & \text{if} \quad m < l \\ B_D(m-1) & \text{if} \quad m = l \\ 0 & \text{if} \quad m > l. \end{cases}$$

Matrices G and H are block-diagonal and analogously defined by

$$G(l,m) = \begin{cases} C_D(l) & \text{if} \quad l = m \\ 0 & \text{otherwise} \end{cases}$$

and

$$H(l,m) = \begin{cases} D_D(l) & \text{if} \quad l = m \\ 0 & \text{otherwise,} \end{cases}$$

respectively, where, $G_{(l,m)} \in \mathbb{R}^{n_y \times n_x}$ and $H_{(l,m)} \in \mathbb{R}^{n_y \times n_u}$. Vector d^0 contains the free response of the system (6) to the initial deviation $\tilde{x}(0) = \tilde{x}_0 \in \mathbb{R}^{n_x}$,

$$d^0 = \left[(A_D(0)\tilde{x}_0)^T, (A_D(1)A_D(0)\tilde{x}_0)^T, \ldots, \left(\prod_{i=0}^{N-1} A_D(i)\tilde{x}_0 \right)^T \right]^T.$$

B. Disturbance estimation

In order to take into account the repetitive nature of the problem setting, the system (10) is written as

$$\begin{aligned} x_j &= Fu_j + d_j \\ y_j &= Gx_j + Hu_j, \end{aligned} \tag{12}$$

where the subscript j indicates the jth execution of the desired task and d_j represents the repetitive disturbance along the reference trajectory, which shows only slight random changes, ω_j, between iterations. Taking into account process and measurement noise, captured in the random variable μ_j, the evolution of the learning over consecutive trials can be represented as a Kalman filter model [13]

$$\begin{aligned} d_j &= d_{j-1} + \omega_{j-1} \\ y_j &= Gd_j + (GF + H)u_j + \mu_j, \end{aligned} \tag{13}$$

where both stochastic zero-mean Gaussian white noise variables, $\omega_j \sim \mathcal{N}(0, \Omega_j)$ and $\mu_j \sim \mathcal{N}(0, M_j)$, are trial-uncorrelated and assumed to be independent. Matrices Ω_j and M_j represent the noise covariances and can be characterized as diagonal matrices.

The Kalman filter estimates the current error d_j taking into account the output signals $y_0, y_1, \ldots, y_j$ from previous trials. Given the initial values of the error estimate, $\widehat{d_0}$ and the error covariance matrix, $P_0 = E[(d_0 - \widehat{d_0})(d_0 - \widehat{d_0})^T]$, the disturbance estimate is calculated as

$$\widehat{d_j} = \widehat{d}_{j-1} + K_j \left(y_j - G\widehat{d}_{j-1} - (GF + H)u_j \right), \tag{14}$$

where K_j is the optimal Kalman gain

$$K_j = (P_{j-1} + \Omega_{j-1}) G^T \left(G (P_{j-1} + \Omega_{j-1}) G^T + M_j \right)^{-1}.$$

C. Input update

The learning update consists in deriving a model-based update rule that computes a new control input $u_{j+1} \in \mathbb{R}^{Nn_u}$ in response to the estimated disturbance $\widehat{d_j}$, that is, minimizing the deviation from the nominal trajectory in the next trial. Since this deviation x_{j+1} is unknown, the expected value of x_{j+1} given all past measurements is considered,

$$E[x_{j+1}|y_1, y_2, \ldots, y_j] = Fu_{j+1} + \widehat{d_j}. \tag{15}$$

The update rule can be expressed by the following optimization problem:

$$\begin{aligned} &\min_{u_{j+1}} && \| Fu_{j+1} + \widehat{d_j} \|_\ell + \alpha \| Du_{j+1} \|_\ell \\ &\text{subject to} && Lu_{j+1} \leq q_{max}, \end{aligned} \tag{16}$$

where constraints (8) are explicitly taken into account, and $\alpha \geq 0$ and the matrix D have the objective of penalizing the input or approximations of its derivatives in order to enforce the smoothness of the optimal problem solution. The vector norm ℓ, with $\ell \in \{1, 2, \infty\}$, of the minimization problem (16) affects the result and convergence of the learning algorithm and should be chosen in accordance with the performance objectives.

The update law defined in (16) can be formulated as a standard convex optimization problem of the form

$$\begin{aligned} &\min_z \left(\frac{1}{2} z^T V z + v^T z \right) \\ &\text{subject to} \quad Wz \leq w \quad \text{and} \quad \eta_1 \leq z \leq \eta_2, \end{aligned} \tag{17}$$

where $z \in \mathbb{R}^{n_z}$ represents the vector of decision variables. Vectors v, w and matrices V, W have appropriate dimensions.

A scaling of the original signals $\mathrm{u}(t), \mathrm{x}(t), \mathrm{y}(t)$ in (1) is essential to guarantee reasonable results in the optimization problem. The scaling, exemplarily shown on the system's state $\mathrm{x}(t)$, reads as

$$x^s = S_x x, \quad S_x \in \mathbb{R}^{Nn_x \times Nn_x} \tag{18}$$

with x^s representing the scaled version of a lifted state vector x and S_x being the corresponding scaling matrix, usually represented by a diagonal matrix. Additionally, a state weighting matrix may be useful to give greater importance to some of the state variables over the rest.

As mentioned in the introduction, one of the advantages of the iterative learning algorithm is its non intrusiveness with respect to the aircraft's existing trajectory tracking controller, since a new reference trajectory can be provided to the following aircraft rather than a control input. Once the updated input u_{j+1} is calculated in (16), the new reference trajectory is obtained by introducing this input into the lifted model, dismissing the disturbances except for the initial deviation error

$$\begin{aligned} \mathrm{x}_N &= x_N + x_d, \quad \text{with} \quad x_N = F u_{j+1} + d^0_{j+1} \\ \mathrm{y}_N &= y_N + y_d, \quad \text{with} \quad y_N = G x_N + H u_{j+1}, \end{aligned} \tag{19}$$

where $\mathrm{x}_N \in \mathbb{R}^{Nn_x}$ is the new reference state variable lifted vector, $\mathrm{y}_N \in \mathbb{R}^{Nn_y}$ is the new reference output lifted vector, and x_d and y_d are the lifted vectors of the desired state and output, respectively

$$\begin{aligned} x_d &= [x^*(1), x^*(2), \ldots, x^*(N)]^T \quad \in \mathbb{R}^{Nn_x} \\ y_d &= [y^*(1), y^*(2), \ldots, y^*(N)]^T \quad \in \mathbb{R}^{Nn_y}. \end{aligned} \tag{20}$$

III. Aircraft dynamics

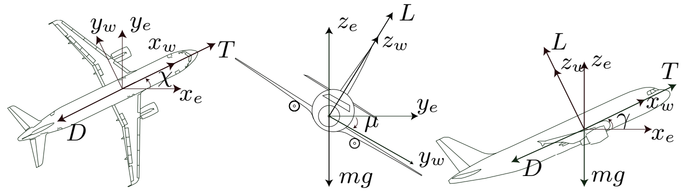
In this section, the dynamic model of the aircraft used in the simulated environment will be described.

A. Equations of Motion

A common three-degree-of-freedom dynamic model has been used which describes the point variable-mass motion of the aircraft over a non-rotating flat Earth model [14]. In particular, a symmetric flight has been considered. Thus, it has been assumed that there is no sideslip and all forces lie in the plane of symmetry of the aircraft. The following equations of motion of the aircraft have been considered:

$$\begin{aligned} \dot{V}(t) &= \frac{T(t) - D(h_e(t), V(t), C_L(t)) - m(t)\cdot g \cdot \sin\gamma(t)}{m(t)} \\ \dot{\chi}(t) &= \frac{L(h_e(t), V(t), C_L(t)) \cdot \sin\mu(t)}{m(t)\cdot V(t)\cdot\cos\gamma(t)} \\ \dot{\gamma}(t) &= \frac{L(h_e(t), V(t), C_L(t)) \cdot \cos\mu(t) - m(t)\cdot g\cdot\cos\gamma(t)}{m(t)\cdot V(t)} \\ \dot{x}_e(t) &= V(t)\cdot\cos\gamma(t)\cdot\cos\chi(t) \\ \dot{y}_e(t) &= V(t)\cdot\cos\gamma(t)\cdot\sin\chi(t) \\ \dot{h}_e(t) &= V(t)\cdot\sin\gamma(t) \\ \dot{m}(t) &= -T(t)\cdot\eta(V(t)). \end{aligned} \tag{21}$$

The three dynamic equations in (21) are expressed in an aircraft-attached reference frame, the wind axes (x_w, y_w, z_w),



(a) Top view (b) Front view (c) Lateral view

Fig. 1. Aircraft state and forces.

and the three kinematic equations are expressed in a ground based reference frame, the Earth reference frame (x_e, y_e, z_e), as shown in Fig. 1.

The states of the system (21) are the true airspeed, V, the heading angle, χ, the flight path angle, γ, the position, x_e, y_e, h_e, and the aircraft mass, m. Thus, $\mathrm{x}(t) = (V(t), \chi(t), \gamma(t), \lambda_e(t), \theta_e(t), h_e(t), m(t))$. The control inputs are the bank angle, μ, the engine thrust, T, and the lift coefficient, C_L. Thus, $\mathrm{u}(t) = (T(t), \mu(t), C_L(t))$.

Parameter η is the speed-dependent fuel efficiency coefficient. Lift, $L = C_L S\hat{q}$, and drag, $D = C_D S\hat{q}$, are the components of the aerodynamic force. Parameter S is the reference wing surface area and $\hat{q} = \frac{1}{2}\rho V^2$ is the dynamic pressure. A parabolic drag polar $C_D = C_{D0} + KC_L^2$, and an International Standard Atmosphere (ISA) model are assumed. The lift coefficient C_L is a known function of the angle of attack α and the Mach number.

B. Flight Envelope Constraints

Flight envelope constraints are derived from the geometry of the aircraft, structural limitations, engine power, and aerodynamic characteristics. The performance limitations model and the parameters have been obtained from BADA.

$$\begin{aligned} &0 \le h_e(t) \le \min[h_{M0}, h_u(t)], && \gamma_{min} \le \gamma(t) \le \gamma_{max}, \\ &M(t) \le M_{M0}, && m_{min} \le m(t) \le m_{max}, \\ &\dot{V}(t) \le \bar{a}_l, && C_v V_s(t) \le V(t) \le V_{Mo}, \\ &\dot{\gamma}(t)V(t) \le \bar{a}_n, && 0 \le C_L(t) \le C_{L_{max}}, \\ &T_{min}(t) \le T(t) \le T_{max}(t), && \mu(t) \le \bar{\mu}. \end{aligned} \tag{22}$$

In (22), h_{M0} is the maximum operational altitude and $h_u(t)$ is the maximum operative altitude at a given mass (it increases as fuel is burned). $M(t)$ is the Mach number and M_{M_0} is the maximum operating Mach number. C_v is the minimum speed coefficient, $V_s(t)$ is the stall speed, V_{M_0} is the maximum operating Calibrated Airspeed (CAS) and $\bar{a}_n$ and $\bar{a}_l$ are, respectively, the maximum normal and longitudinal accelerations for civilian aircraft. Finally, $T_{min}(t)$ and $T_{max}(t)$ correspond to the minimum and maximum available thrust, respectively, and $\bar{\mu}$ corresponds to the maximum bank angle due to structural limitations.

C. Longitudinal dynamics

In this work, ILC is used for precise trajectory tracking of the vertical profile of a CCO. Therefore, the motion of the

aircraft is limited to a vertical plane, i.e., with constant course and thus constant heading angle χ. Without loss of generality, we assume that the heading angle is zero, that is $\chi = 0$. We also suppose that the aircraft performs a leveled wing flight, thus the bank angle is zero, that is $\mu = 0$.
The state variables are then

$$\mathrm{x}(t) = (V(t), \chi(t), \gamma(t), \lambda_e(t), \theta_e(t), h_e(t), m(t))$$

and the control variables,

$$\mathrm{u}(t) = (T(t), C_L(t)).$$

The equations of motion are reduced to:

$$\begin{aligned}
\dot{V}(t) &= \frac{T(t) - D(h_e(t), V(t), C_L(t)) - m(t) \cdot g \cdot \sin\gamma(t)}{m(t)} \\
\dot{\gamma}(t) &= \frac{L(h_e(t), V(t), C_L(t)) - m(t) \cdot g \cdot \cos\gamma(t)}{m(t) \cdot V(t)} \\
\dot{x}_e(t) &= V(t) \cdot \cos\gamma(t) \\
\dot{h}_e(t) &= V(t) \cdot \sin\gamma(t) \\
\dot{m}(t) &= -T(t) \cdot \eta(V(t)).
\end{aligned} \tag{23}$$

The flight envelope constraints remain the same as in (22), except the one referring to the bank angle, since it is assumed to be zero.

IV. Trajectory Planning

In this section, the method to generate a reference trajectory to be followed will be described.

As said before, in this paper, the ILC paradigm is applied to follow a CCO trajectory in the the vertical plane generated using an optimal control technique in which fuel consumption has been minimized. In [8], ICAO defines CCO as "an aircraft operating technique enabled by airspace design, procedure design and facilitation by ATC, allowing the execution of a flight profile optimized to the performance of the aircraft". They also enumerate some advantages of CCO, such as more fuel efficient operations, reduction in both flight crew and controller workload through the design of procedures requiring less ATC intervention, reduction in the number of required radio transmissions, cost savings and environmental benefits through reduced fuel burn, potentially aircraft noise mitigation through thrust and height optimization, and potential authorization of operations where noise limitations would otherwise result in operations being curtailed or restricted. On the other hand, precise trajectory tracking is essential to avoid potential conflict between traffic flows and ensure that safety and capacity are not compromised.

A feasible state trajectory with its corresponding nominal input is the starting point of the iterative learning algorithm. The trajectory is generated via the MATLAB application DIDO using a nominal model of an Airbus A320 aircraft. The input to DIDO is the problem formulation in a structured format, including system dynamics, constraints and cost function. Besides the optimal control and the resulting trajectory, DIDO automatically outputs the Hamiltonian, costates, path

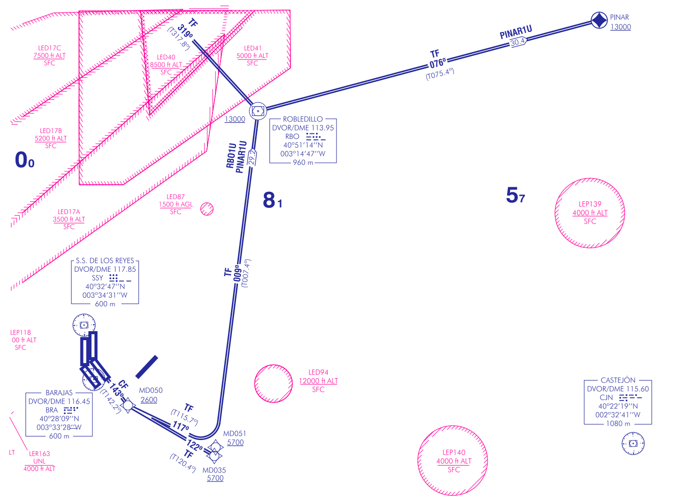


Fig. 2. SID PINAR1U. Source: Spanish AIP service. Not for operational use.

covectors, and endpoint covectors for the verification of the solution. A CCO associated with a SID procedure has been considered. Therefore, airspace constraints that represent the passage through the waypoints that define the SID are introduced in the trajectory generation problem together with the flight envelope constraints. The boundary conditions for the state variables have been selected according to a realistic flight from Adolfo Suárez Madrid-Barajas (LEMD) airport. The trajectory generation starts up to 1000 m after takeoff and ends at 10000 m, when the aircraft reaches cruise level. Initial velocity and path angle have been selected according to standard values.

The selected SID procedure is Madrid Barajas PINAR1U shown in Fig. 2. The definition of the SID can be found in the Spanish Aeronautical Information Publication (AIP) service[5], managed by ENAIRE, where the SID is textually described as follows: "To MD050 on heading 143°M at 2600 ft or above, turn left. To MD051 at 5700 ft or above, turn left. To RBO at 13000 ft or above, turn right. To PINAR at 13000 ft or above".

TABLE I
Coordinates of waypoints and navaids of the SID PINAR1U. Source: Spanish AIP service.

Point	Type	Latitude	Longitude
MD050	RNAV waypoint	40°25′54.0220″N	003°29′37.3611″W
MD051	RNAV waypoint	40°22′15.4740″N	003°19′44.9769″W
RBO	DVOR/DME navaid	40°51′14″N	003°14′47″W
PINAR	RNAV waypoint	40°58′49.0620″N	002°35′56.9980″W

Three of the points that define the SID are Area Navigation (RNAV) waypoints whereas one of them is a DVOR/DME Navigational Aid (NAVAID). The geographical coordinates of these points are listed in Table I.

[5]https://ais.enaire.es/AIP/

V. Experimental Setup

In this section, the simulated environment in which the experiments have been carried out will be described. It is composed by:

- a realistic flight simulator,
- an estimator of the disturbances acting on the aircraft, and
- an ILC controller.

In order to test the proposed learning scheme, a 3-DOF longitudinal model of an Airbus A320 aircraft has been implemented in Simulink, a widely used software in aircraft simulation [15]. The input variables are the thrust and the lift coefficient, $\mathrm{u} = (T, C_L)$, and the vector of state variables $\mathrm{x} = (V, \gamma, x_e, h_e, m)$, where V is the true airspeed, γ is the flight path angle, x_e and h_e are the horizontal position and the height, and m is the aircraft mass. We assume that all states can be measured. Noise is introduced in the model in order to simulate weather perturbations, model uncertainties as well as measurement noise. Therefore, a Kalman filter has been employed to estimate the disturbances acting on the aircraft as described in Section II-B.

The ILC algorithm, which has been implemented in MATLAB, can be summarized as follows.

- Initialization. First, the desired trajectory, $y^* = x^*$, and the corresponding input, u^*, are loaded from a file provided by the optimal control software DIDO. As said in Section IV, the desired trajectory to be followed has been calculated using a nominal model of the aircraft without taking into account perturbations. This model does not coincide with the aircraft model used in the flight simulator, which is more realistic, containing perturbations. Then, settings and parameter values of the learning algorithm are introduced in the main program, and the lifted-domain representation and the Kalman gains are computed. To test the robustness of the ILC controller to modeling errors, the aircraft parameters entered here are slightly different from those used in the flight simulator and in trajectory planning. The optimization problem is now set up.
- j-th iteration. Using the most recent feed-forward input generated by the ILC algorithm, the corresponding trajectory of the aircraft is generated by the flight simulator and the tracking error between desired and actual trajectories is estimated.
- $j+1$-th iteration. Based on the tracking error estimated after the j-th iteration, the simulated aircraft is set to the initial position and the update step of the learning algorithm is executed, which computes a new reference trajectory to be tracked in the $j+1$-th iteration.

VI. Results

In this section, the results of the application of the ILC scheme to precise aircraft trajectory tracking in CCOs will be reported. As said before, although the ILC algorithm is able to tackle 4D trajectories, in this section, for the sake of clarity, only the corresponding 3D paths will be reported.

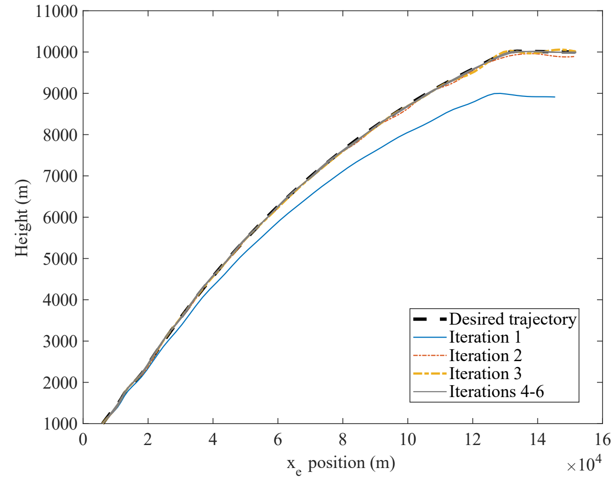


Fig. 3. Evolution of the trajectory $x_e - h_e$ over iterations.

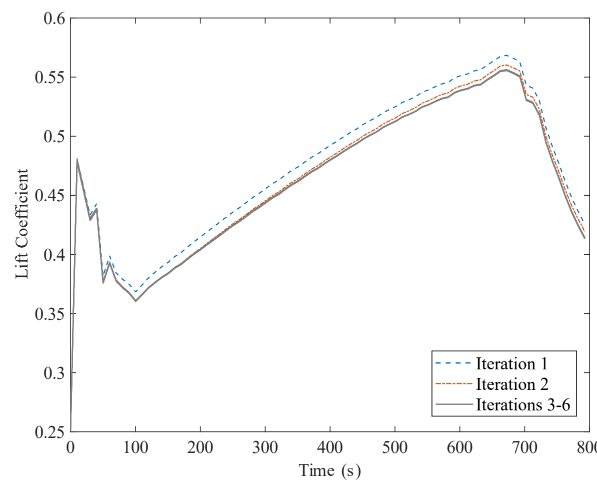


Fig. 4. Evolution of the lift coefficient over iterations.

The desired path associated with the desired trajectory of the aircraft to perform a CCO, obtained in the trajectory planning phase described in Section IV, is shown in dashed black line in Fig. 3. In the first iteration, the corresponding input is applied to the first aircraft and the resulting trajectory is generated by the flight simulator described in Section V.

Due to modeling and disturbance errors considered in the flight simulator, the resulting path falls far below the desired one, initiating the cruise phase more than 1000 m below the planned cruise height. The ILC scheme rapidly learns from the first executions achieving a very precise tracking of the designed CCO trajectory after only three iterations. This means that using the ILC scheme, the third aircraft will be able to compensate for the recurring disturbances and follow the desired trajectory with high precision.

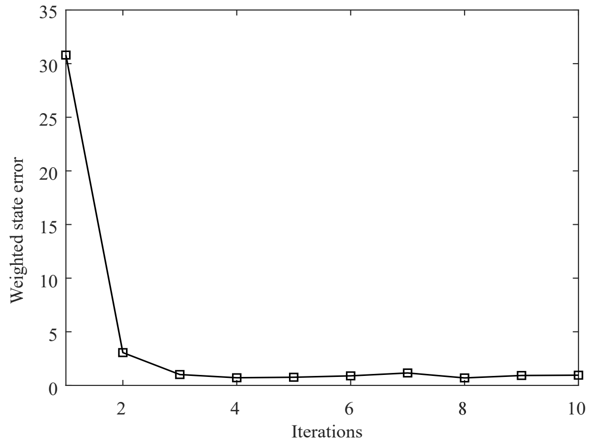


Fig. 5. Evolution of the weighted state error over iterations.

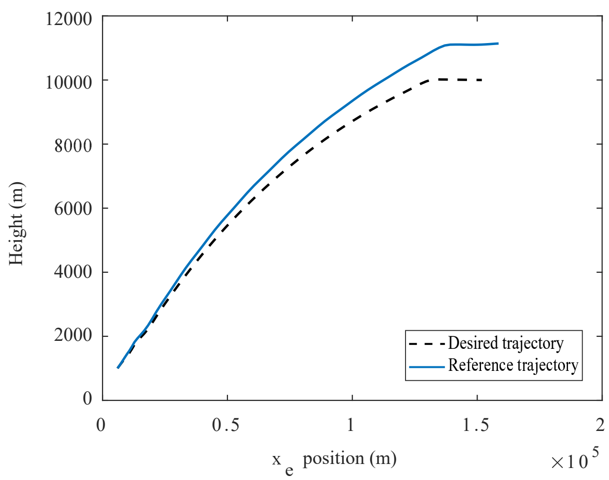


Fig. 6. Reference path associated with the reference trajectory to be provided to the aircraft in the seventh iteration of the ILC algorithm to precisely track the desired trajectory.

As shown in Fig. 4, the lift coefficient input converges after few iterations, and similarly does the thrust. Despite the convergence of the input, both the trajectory and the input vary over iterations due to non-repetitive disturbances, which would be compensated by the feedback trajectory tracking controller of the aircraft.

The state weighted error in Fig. 5 allows the learning performance to be evaluated over iterations. It is calculated as:

$$e_{w,j} = \| S\mathrm{y}_j \|_2, \tag{24}$$

where S is the weighted scaling matrix of the state variables and y_j is the measured output vector. As expected, since the ILC scheme is intended to compensate for repetitive disturbances, the weighted state error converges to the system noise level, and not to zero, due to non-repetitive disturbances.

In Fig. 6 is depicted the reference path associated with the reference trajectory to be provided to the aircraft in the seventh iteration of the ILC algorithm to precisely track the desired trajectory. It is an example of the non-intrusiveness of this control paradigm with respect to the underlying aircraft feedback controller for trajectory tracking by calculating, at each iteration, a reference trajectory for the following aircraft rather than a control input.

VII. Conclusions

In this paper an optimization-based iterative learning approach has been applied to precise aircraft trajectory tracking. In this method, optimality is pursued in both the estimation of the recurring disturbance and in the calculation of the input update, which optimally compensates for the disturbance. The approach has been successfully applied to trajectory tracking of the simulation of commercial aircraft in continuous climb operations. The flight of an Airbus A320 aircraft in the vertical plane has been modeled assuming constant course and leveled wing flight and International Standard Atmosphere conditions. Weather perturbations, model uncertainties and measurement noise have been introduced in the model. These noise signals vary from iteration to iteration and are assumed to be trial-uncorrelated sequences of zero-mean Gaussian white noise. It has been shown that precision in aircraft trajectory tracking can be improved by pure feed-forward adaptation of the control input. Extending this approach to flights not restricted to the vertical plane and applying it to other flight procedures are subjects of current research.

References

[1] A. Tewari, *Advanced Control of Aircraft, Spacecraft and Rockets*. Wiley, 2011.

[2] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *Journal of Robotic Systems*, vol. 1, no. 2, pp. 123–140, 1984.

[3] A. P. Schoellig, F. L. Mueller, and R. D'Andrea, "Optimization-based iterative learning for precise quadrocopter trajectory tracking," *Autonomous Robots*, vol. 33, no. 1-2, pp. 103–127, 2012.

[4] D. A. Bristow, M. Tharayil, and A. Alleyne, "A survey of iterative learning control," *IEEE Control Systems Magazine*, vol. 26, pp. 2039–2114, 2006.

[5] J.-X. Xu and Y. Tan, *Linear and Nonlinear Iterative Learning Control*. Springer, 2003.

[6] J.-X. Xu, S. K. Panda, and T. H. Lee, *Real-time Iterative Learning Control: Design and Applications*. Springer, 2009.

[7] ICAO, "Global air navigation plan," Tech. Rep. 9750-AN/963, International Civil Aviation Organization, 2016.

[8] ICAO, "Continuous Climb Operations (CCO) Manual," Tech. Rep. Doc. 9993 AN/495, International Civil Aviation Organization, 2012.

[9] V. Mouillet, "User Manual for the Base of Aircraft Data (BADA) Revision 3.14," tech. rep., EUROCONTROL Experimental Centre, Brétigny, France, 2017.

[10] O. Purwin and R. D'Andrea, "Performing and extending aggressive maneuvers using iterative learning control," *Robotics and Autonomous Systems*, vol. 59, pp. 1–11, 2011.

[11] X. Liang, M. Zheng, and F. Zhang, "A scalable model-based learning algorithm with application to UAVs," *IEEE Control Systems Letters*, vol. 2, no. 4, pp. 839 – 844, 2018.

[12] B. Bamieh, J. B. Pearson, B. A. Francis, and A. Tannenbaum, "A lifting technique for linear periodic systems with applications to sampled-data control," *Systems & Control Letters*, vol. 17, no. 2, pp. 79 – 88, 1991.

[13] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York, USA: John Wiley & Sons, Inc., 2001.

[14] D. G. Hull, *Fundamentals of Airplane Flight Mechanics*. Berlin Heidelberg: Springer-Verlag, 2007.

[15] L. Moin, A. Baig, and V. Uddin, "State space model of an aircraft using Simulink," *International Journal of System Modeling and Simulation*, vol. 2, no. 4, pp. 1–6, 2017.

Almudena Buelta is a Teaching Assistant of Statistics and a PhD student at the Universidad Rey Juan Carlos in Madrid, Spain. She received her MSc degree in Aeronautical Engineering from the Universidad Politécnica de Madrid. She worked for Airbus Defence and Space, developing training and operational documentation and providing operational support for in-service aircraft. Her research focuses on iterative learning control applied to commercial aircraft trajectory tracking.

Alberto Olivares is a Professor of Statistics and Vector Calculus at the Universidad Rey Juan Carlos in Madrid, Spain. He received his MSc degree in Mathematics and his BSc degree in Statistics from the Universidad de Salamanca, Spain, and his PhD degree in Mathematical Engineering from the Universidad Rey Juan Carlos. He worked with the Athens University of Economics and Business. His research interests include statistical learning, stochastic hybrid optimal control and model predictive control with applications to biomedicine, robotics, aeronautics and astronautics.

Ernesto Staffetti is a Professor of Statistics and Control Systems at the Universidad Rey Juan Carlos in Madrid, Spain. He received his MSc degree in Automation Engineering from the Università degli Studi di Roma "La Sapienza," and his PhD degree in Advanced Automation Engineering from the Universitat Politècnica de Catalunya. He worked with the Universitat Politècnica de Catalunya, the Katholieke Universiteit Leuven, the Spanish Consejo Superior de Investigaciones Científicas, and with the University of North Carolina at Charlotte. His research interests include stochastic hybrid optimal control, iterative learning control and model predictive control with applications to robotics, aeronautics and astronautics.