A Civilian Aircraft Landing Challenge

on-line available from the aerospace benchmark section of the SMAC toolbox

http://w3.onera.fr/smac/

Jean-Marc Biannic* & Josep Boada-Bauxell
ONERA **AIRBUS**

October, 2016

Abstract

From the early developments of the A320 program, thanks to Fly-By-Wire systems, Automatic Control techniques have significantly contributed to improve flight performance and safety of civilian aircraft. Today, most of the flight segments can be managed quite efficiently by the autopilot. However, the final approach and landing phases still remain critical in poor visibility and strong wind conditions. Based on a realistic nonlinear model of a civil transport aircraft in full configuration, the objective of the proposed challenge is to design an autopilot system to enable a correct landing despite parametric variations and **maximized cross wind** conditions.

Acknowledgements

The authors are grateful to Clément Roos, Andrés Marcos (and his team at University of Bristol: Pedro Simplicio, Andrea Lannelli and Diego Navarro-Tapia), Julian Theis, Daniel Ossmann and Harald Pfifer for their useful and constructive feedbacks on this challenge thanks to which bugs were corrected and quality thus improved.

1 Introduction

For all civil transport aircraft, final approach and flare segments still remain critical phases of the flight during which many variables have to be controlled simultaneously and high safety standards must be met. Fortunately, with the help of CAT III instrument landing systems (ILS) now commonly available in a rapidly growing list of airports, automatic landing control laws have recently contributed to secure these two phases notably in degraded weather conditions (fog, crosswinds). However, despite numerous methodological works [3, 1, 4, 5, 2] over the past two decades, the design, tuning and validation process of final approach and flare control systems remains a challenging and time-consuming task. As is observed in [4], where a complete design framework together with a dedicated software is proposed, the tuning phase requires rather tricky multi-objective optimization.

Automatic landing control has to prove robustness to a wide range of system and environmental phenomena dispersion like aircraft weight or wind disturbances. Indeed certification process requires the aircraft parameters to be within a range of a so-called mean and risk dispersion. Therefore the validation campaign, performed in a Monte Carlo framework, exposes the control law not only to scenarios where all the parameters are scattered following their own statistical distribution (mean dispersion) but also to operational situations where one parameter is fixed to its extreme value while dispersing the rest (risk dispersion). Consequently control design task becomes an iterative process

*corresponding author: biannic@onera.fr

where first the mean dispersion requirements are met and then all risk dispersion requirements are tested sequentially. The challenge appears when a risk is not satisfied, then two solutions are available:

- either adjust the control laws to improve the robustness versus a critical parameter which may result in robustness degradation versus others;
- limit the maximum dispersion that one seeks to cover which will then reduce the operational domain of the aircraft.

For example, control designer shall invest a great effort on satisfying a large domain of weight whereas he can accept a reduction of the crosswind capabilities. However, in order to improve aircraft operational reliability and reduce crew workload in difficult situations like crosswind landings the next generation of control architectures shall be able to cope with larger crosswinds domains. Nowadays A320 is certified to land with 20kts of crosswind while A380 is able to cope with 30kts and future aircrafts should target even a larger domain, eventually as large as the crosswind proven in manual flight (35kts for an A320).

This challenge goal is to inspire new control architectures which are able to cope with the largest possible crosswind while validating the certification requirements presented in Section 3. Proposed solutions shall minimize the use of actuators at high frequency ($bandpass < 2Hz$), avoid pitch and bank-bank oscillations that could induce the pilot to disconnect the autopilot (a damped second order response is required) and respect loads constraints (vertical load factor $\leq 2g$).

The quality of a design will be evaluated through its ability to fulfill the requirements detailed in Section 3. However, **methodological aspects** (simplicity, genericity,...) will also be considered.

2 Aircraft model

The aircraft model is representative of a large transport aircraft in full configuration from 1000 ft above runway until touch-down. It is fully implemented in open-access nested simulink blocks. The equations of the model are detailed next.

2.1 General equations

Following a standard approach, the aircraft model is obtained from the dynamics and kinematics equations. Let us denote $\Omega = [p\ q\ r]'$ and $V = [u\ v\ w]'$ the rotations and translations speeds vectors both expressed in a **body axis** frame with respect to the center of gravity of the aircraft. One obtains:

$$\dot{\mathbf{\Omega}} = I^{-1}(\mathbf{M} - \mathbf{\Omega} \times I\mathbf{\Omega}) \tag{1}$$

and:

$$\dot{\mathbf{V}} = \frac{\mathbf{F}}{m} - \mathbf{\Omega} \times \mathbf{V} \tag{2}$$

where m denotes the mass of the aircraft and I is the inertia matrix which is structured as follows:

$$I = \begin{pmatrix} I_{xx} & 0 & I_{xz} \\ 0 & I_{yy} & 0 \\ I_{xz} & 0 & I_{zz} \end{pmatrix} \tag{3}$$

The two vectors $\mathbf{M}$ and $\mathbf{F}$ stand for the moments and forces vectors to be detailed next. But let us conclude with the differential equations first. To this purpose, we need to introduce $\mathbf{\Phi} = [\phi\ \theta\ \psi]'$ and $\mathbf{X} = [x\ y\ z]'$. The first vector contains the Euler's angles. The second contains the position of the center of gravity G of the aircraft. They are both expressed in a **earth-linked** vertical frame. The x-axis is oriented forward, the y-axis is oriented rightward, the z axis is oriented downward.

The kinematics and navigation equations respectively yield:

$$\begin{aligned} \dot{\mathbf{\Phi}} &= \begin{pmatrix} 1 & \sin\phi\,\tan\theta & \cos\phi\,\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\,/\cos\theta & \cos\phi\,/\cos\theta \end{pmatrix} \mathbf{\Omega} \\ &= T(\mathbf{\Phi}).\mathbf{\Omega} \end{aligned} \tag{4}$$

and:

$$\begin{aligned} \dot{\mathbf{X}} &= \begin{pmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{pmatrix} \mathbf{V} \\ &= R_{b\to v}(\mathbf{\Phi}).\mathbf{V} \end{aligned} \tag{5}$$

The 12^{th} order aircraft model may thus be summarized as follows:

$$\begin{cases} \dot{\mathbf{V}} &= \frac{1}{m}\mathbf{F} - \mathbf{\Omega}\times\mathbf{V} \\ \dot{\mathbf{\Omega}} &= I^{-1}(\mathbf{M} - \mathbf{\Omega}\times I\mathbf{\Omega}) \\ \dot{\mathbf{\Phi}} &= T(\mathbf{\Phi}).\mathbf{\Omega} \\ \dot{\mathbf{X}} &= R_{b\to v}(\mathbf{\Phi}).\mathbf{V} \end{cases} \tag{6}$$

2.2 Forces & Moments

The forces applied to the aircraft can be decomposed into three terms (engines thrust, gravity and aerodynamic forces):

$$\mathbf{F} = \mathbf{F}_{eng} + \mathbf{F}_g + \mathbf{F}_a \tag{7}$$

To comply with equation (1), these must be expressed in the body-axis frame. It is assumed that the thrust is aligned with the longitudinal axis, so that:

$$\mathbf{F}_{eng} = \begin{pmatrix} F_x \\ 0 \\ 0 \end{pmatrix} \tag{8}$$

The gravity forces, projected in the body-axis frame are given by:

$$\mathbf{F}_g = R_{v\to b}(\mathbf{\Phi}) \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} = R_{b\to v}(\mathbf{\Phi})' \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} = mg \begin{pmatrix} -\sin\theta \\ \cos\theta\sin\phi \\ \cos\theta\cos\phi \end{pmatrix} \tag{9}$$

The aerodynamic forces are initially expressed in a stability-axis frame which is derived from the body-axis:

$$\mathbf{F}_a = q_d S \underbrace{\begin{pmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{pmatrix}}_{R_{s\to b}(\alpha)} \begin{pmatrix} C_X \\ C_Y \\ C_Z \end{pmatrix} \tag{10}$$

where α, q_d and S denote the angle-of-attack, the dynamic pressure and the reference surface respectively. The drag ($C_X = -C_D$), lateral (C_Y) and lift ($C_Z = -C_L$) coefficients are detailed next.

The moment about the center of gravity G of the aircraft result from the engines and aerodynamic actions:

$$\mathbf{M} = \mathbf{M}_{eng} + \mathbf{M}_a \tag{11}$$

Let us assume that both engines deliver the same thrust, so that the thrust location can be reduced to a single point E whose coordinate along the y-axis is zero. The moment resulting from the thrust is:

$$\mathbf{M}_{eng} = \mathbf{GE}\times\mathbf{F}_{eng} = \begin{pmatrix} 0 \\ z_{eng}.F_x \\ 0 \end{pmatrix} \tag{12}$$

Note that the engines are located below the center of gravity, so that $z_{eng} > 0$ and one observes a pitching moment. The aerodynamic moment exhibit two terms:

$$\mathbf{M}_a = q_d S L \begin{pmatrix} C_l \\ C_m \\ C_n \end{pmatrix} + \mathbf{GA} \times \mathbf{F}_a \tag{13}$$

The first one contains the main contribution and is directly proportional to the moment coefficients C_l, C_m and C_n about roll, pitch and yaw axis. The aerodynamic mean chord is denoted L in this term. The second term is linked to the fact that the aerodynamic forces F_a detailed above are applied to a point A which possibly differs from the center of gravity. This notably depends on the center of gravity location.

2.3 Wind & Atmosphere effects

As is clear from previous subsections, at least through the dynamic pressure:

$$q_d = \frac{1}{2}\rho {V_a}^2 \tag{14}$$

aerodynamic moments and forces strongly depend on the airspeed $V_a = \|\mathbf{V_a}\|$, but also on the air density ρ. The airspeed is clearly affected by wind. The latter denoted $\mathbf{W}$ is generally expressed in a earth-linked vertical axis. The airspeed vector $\mathbf{V_a}$ in body-axis coordinates is thus obtained as:

$$\mathbf{V_a} = \mathbf{V} - R_{v\to b}(\mathbf{\Phi})\mathbf{W} = \mathbf{V} - R_{b\to v}(\mathbf{\Phi})'\mathbf{W} \tag{15}$$

Aerodynamic coefficients also depend on the aerodynamic angle-of-attack α and sideslip angle β which are respectively obtained as:

$$\alpha = \arctan\left(\frac{V_{az}}{V_{ax}}\right), \quad \beta = \arcsin\left(\frac{V_{ay}}{V_a}\right) \tag{16}$$

Let us now detail the expression of ρ which depends on altitude and temperature. Since we focus here on the landing phase, from approximately $1000\,ft$ above runway until touchdown, the altitude and temperature variations are negligible. Then, the air density ρ can be viewed as a fixed parameter depending on the runway altitude and temperature. Let us denote H_{rwy} (expressed in m) the runway altitude and T_0 (expressed in ^oK) the temperature at sea-level. Since the maximum value for H_{rwy} is about $3000m$, the temperature at runway level is given by:

$$T_{RWY} = T_0 - 6.5\,10^{-3} H_{RWY} \tag{17}$$

and

$$\rho = \frac{353}{T_{RWY}}\left(\frac{T_{RWY}}{T_0}\right)^{5.25} \tag{18}$$

At low altitudes and speeds, the calibrated airspeed (denoted V_c) describing the dynamic pressure acting on aircraft surfaces regardless of density is quite close to the equivalent airspeed and is then obtained, from the true airspeed V_a as follows:

$$V_c = \sqrt{\frac{\rho}{\rho_0}}.V_a \tag{19}$$

where $\rho_0 = 1.2257$ stands for the air density at sea-level for a nominal temperature $T_0 = 288^oK$. Finally, the Mach number which impacts the engine efficiency is defined as the ratio between airspeed and sound-speed V_s at runway level:

$$Mach = \frac{V_a}{V_s} \tag{20}$$

with

$$V_s = 20\sqrt{T_{RWY}} \tag{21}$$

2.4 Aerodynamic coefficients

Let us define δ_a, δ_e and δ_r, the ailerons, elevators and rudders deflections. The lift, lateral and drag coefficients are respectively given by:

$$C_L = C_{L_0} + C_{L_\alpha}\alpha + \frac{L}{V_a}C_{L_q}q + C_{L_{\delta_e}}\delta_e + \underbrace{C_{L_H}e^{-\lambda_L H_{LG}}}_{\text{ground effect}} \tag{22}$$

$$C_Y = C_{Y_\beta}\beta + C_{Y_r}\delta_r \tag{23}$$

$$C_D = C_{D_0} + C_{D_\alpha}\alpha + C_{D_{\alpha_2}}\alpha^2 \tag{24}$$

Note that the last term in the lift coefficient describes the ground effect. The lift increases as the aircraft gets closer to the ground. The variable H_{LG} in this term denotes the height of the main landing gear above runway.

Similarly, the moment coefficients about x, y and z axes are given by:

$$C_l = C_{l_\beta}\beta + \frac{L}{V_a}\left(C_{l_p}p + (C_{l_{r_0}} + C_{l_{r_\alpha}}\alpha)r\right) + C_{l_{\delta_a}}\delta_a + C_{l_{\delta_r}}\delta_r \tag{25}$$

$$C_m = C_{m_0} + C_{m_\alpha}\alpha + \frac{L}{V_a}C_{m_q}q + C_{m_{\delta_e}}\delta_e + \underbrace{(C_{m_{H_0}} + C_{m_{H_\alpha}}\alpha)e^{-\lambda_m H_{LG}}}_{\text{ground effect}} \tag{26}$$

$$C_n = (C_{n_{\beta_0}} + C_{n_{\beta_\alpha}})\beta + \frac{L}{V_a}\left(C_{n_r}r + (C_{n_{p_0}} + C_{n_{p_\alpha}}\alpha)p\right) + C_{n_{\delta_a}}\delta_a + C_{n_{\delta_r}}\delta_r \tag{27}$$

Here again, the last term on the pitching moment describes the ground effect.

2.5 Engines & Actuators

During the final approach and landing phases, the aircraft is controlled by the means of:

- twin engines both providing the same thrust in the fuselage direction (x-axis),
- a pair of ailerons whose asymetric deflections generate moments about the x-axis (roll),
- an elevator whose deflections control the y-axis (pitch),
- a rudder whose deflections control the z-axis (yaw).

The dynamics of the engines and actuators are approximated by magnitude and rate limited first-order filters whose characteristics are summarized in Table 1.

Parameter	**Time-constant**	**Lower-bound**	**Upper-bound**	**Rate-Limit**
Engines (EPR)	$2\,s$	0.95	1.6	0.1
Ailerons (δ_A)	$0.06\,s$	$-55\,deg$	$55\,deg$	$60\,deg/s$
Elevators (δ_E)	$0.07\,s$	$-25\,deg$	$25\,deg$	$20\,deg/s$
Rudder (δ_R)	$0.2\,s$	$-30\,deg$	$30\,deg$	$30\,deg/s$

Table 1: Engines & Actuators characteristics

Remark 1 *The effective thrust F_{ENG} at a given altitude is approximated by an affine function of the Exhaust Pressure Ratio (EPR):*

$$F_{ENG} = A_{ENG}(T/T_0).EPR + B_{ENG}(T/T_0) \tag{28}$$

whose coefficients depend on the temperature ratio T/T_0.

Remark 2 *Along the longitudinal axis, the aircraft is trimmed by the horizontal stabilizers whose dynamics are much slower than those of the elevators. It has not been represented here. It is assumed that the stabilizers are not used during the final approach.*

2.6 Inputs & Outputs

As is visible on Figure 1, the AirCraft System model (ACS.slx) exhibits a total of 9 inputs (4 control inputs, 3 wind inputs, 2 ILS noises inputs) and 40 outputs (16 states outputs, 5 non-measured outputs (available for visualization only), 19 measured outputs (available for feedback).

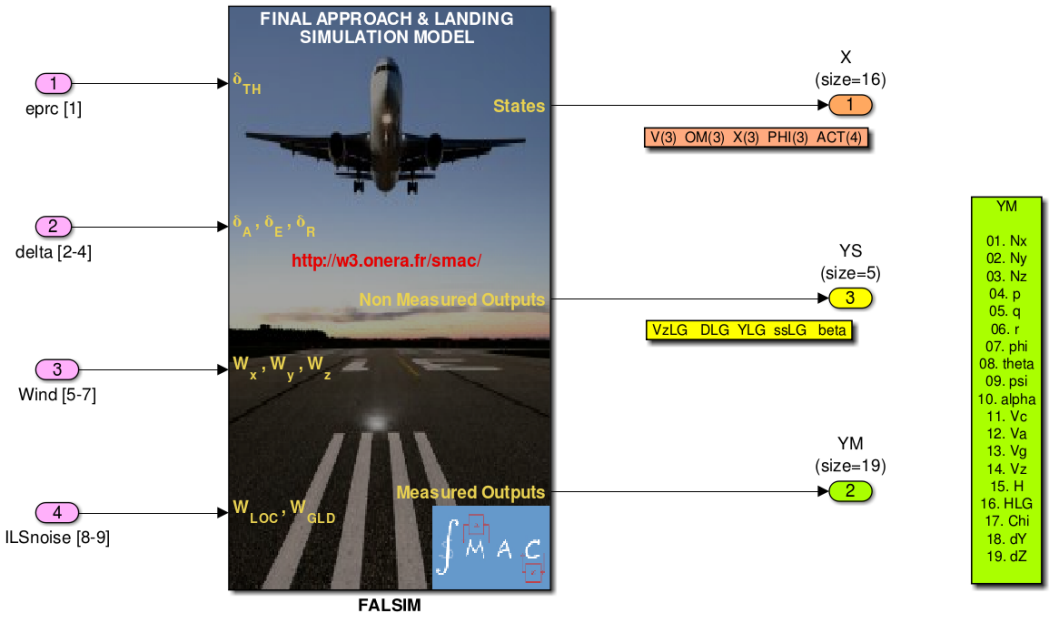

Figure 1: Open-Loop Aircraft Model

These inputs and outputs are summarized in the following tables.

2.6.1 Control inputs & perturbations

#	sym	description	unit
1	eprc	commanded exhaust pressure ratio	dimensionless
2	delta(1)	commanded deflection on ailerons (δ_A)	rad
3	delta(2)	commanded deflection on elevators (δ_E)	rad
4	delta(3)	commanded deflection on rudder (δ_R)	rad
5	Winds(1)	horizontal x-wind input (W_x)	m/s
6	Winds(2)	horizontal y-wind input (W_y)	m/s
7	Winds(3)	vertical wind input (W_z)	m/s
8	ILSnoise(1)	Localizer Noise Input (w_{loc})	μA
9	ILSnoise(2)	Glide Noise Input (w_{gld})	μA

Table 2: Control inputs & perturbations

2.6.2 Measured outputs

#	sym	description	unit	#	sym	description	unit
1	Nx	longitudinal load factor	m/s^2	11	V_c	calibrated airspeed	m/s
2	Ny	lateral load factor	m/s^2	12	V_a	true airspeed	m/s
3	Nz	vertical load factor	m/s^2	13	V_g	ground speed	m/s
4	p	roll rate	rad/s	14	V_z	inertial vertical airspeed	m/s
5	q	pitch rate	rad/s	15	H	altitude	m
6	r	yaw rate	rad/s	16	H_{LG}	landing gear height	m
7	ϕ	bank angle	rad	17	χ	route angle	rad
8	θ	pitch angle	rad	18	Δ_Y	loc deviation	m
9	ψ	heading angle	rad	19	Δ_Z	glide deviation	m
10	α	angle-of-attack	rad -	-	-	-	-

Table 3: Measured outputs

2.6.3 Non measured outputs

#	sym	description	unit
1	Vz_{LG}	landing gear vertical speed / runway	m/s
2	D_{LG}	distance to threshold (neg. before, pos. after)	m
3	Y_{LG}	deviation from runway axis	m
4	ss_{LG}	landing gear sideslip angle	rad
5	β	aerodynamic sideslip angle	rad

Table 4: Non measured outputs

2.6.4 Additional comments on inputs and outputs

- the ILS noises are calibrated such that

$$w_{loc} = 1\mu A \;\Rightarrow\; \delta(\Delta_Y) = 0.7\,m \text{ at threshold}$$

$$w_{gld} = 625\mu A \;\Rightarrow\; \delta(\gamma_{GLD}) = 1\,deg$$

- V_z and $V_{z_{LG}}$ coincide during approach but a discontinuity may appear in $V_{z_{LG}}$ after threshold when a non-zero runway slope is considered,
- The baro-altitude H, during approach is given by $H = H_{LG} + H_{RWY}$. After threshold a difference appears when a non-zero runway slope is considered.

3 Control design objectives

Based on the above 3-axes nonlinear model, the main objective is to design an autoland control system to perform correct approach and landing tasks despite wind, turbulences, ground effects and variations of mass and center-of-gravity location.

Ideally, the aircraft should hit the ground $400\,m$ after threshold, with a vertical speed around $2.5\,ft/s$ ($0.77m/s$). The lateral deviation at touchdown should be kept as small as possible, as well as the roll and main landing gear sideslip angles.

More precisely, the quality of the autoland control system is evaluated through a Monte-Carlo statistical analysis from a set of 2000 landings. These are performed with randomly distributed turbulences profiles, mass values, center-of-gravity locations, runway heights, runway slopes, temperatures, glide slopes and localizer displacements. The distribution characteristics are described in Table 5.

Remark 3 *Note that in this challenge, the crosswind (WY33) conditions are to be maximized, possibly beyond the standard values of Table 5. Control law validation shall be performed in two steps:*

- **Step1:** *First perform a Monte-Carlo analysis considering Table 5 distribution modifying the min and max value of the crosswind (WY33) dispersion to the sought level (i.e.* $\pm 30kts$*).*
- **Step2:** *Then fix the crosswind (WY33) to its maximum and perform a Monte-Carlo analysis dispersing the rest of the parameters following Table 5 distribution.*

Parameter	**distribution**	**mean**	σ	**min**	**max**
Long. Wind (WX33)	normal	$7.5\,kts$	$7.5\,kts$	$10\,T(+10)$	$30\,H(-30)$
Lat. Wind (WY33)	normal	$0\,kts$	$7\,kts$	$20L(-20)$	$20R(+20)$
Mass	uniform	NA	NA	$120\,t$	$180\,t$
CG	uniform	NA	NA	15%	41%
RWY alt	specific	NA	NA	$-1000\,ft$	$9200\,ft$
ISA	uniform	NA	NA	-69^oC	$+40^oC$
RWY slope	normal	0%	0.4%	-2%	2%
GLD slope	normal	-3^o	0.075^o	-2.85^o	-3.15^o
LOC displacement	normal	$0\,\mu A$	$2.5\,\mu A$	$-5\,\mu A$	$5\,\mu A$

Table 5: Distributions for parameters affecting the landing performances

For each landing, the following six parameters:

- **HTP60** : height of the main landing gear above runway, $60\,m$ after threshold
- **XTP** : touch-down point distance from the runway threshold
- **VZTP** : vertical speed at touch-down
- **YTP** : touch-down lateral deviation from the runway axis
- **PHI** : roll angle at touch-down
- **SSTP** : touch-down lateral deviation from the runway axis

are evaluated, plotted in bar-diagrams and interpolated by gaussian functions. From such interpolations, probability levels that the above variables exceed critical values can be computed.
The certification requirements are presented in the last two columns of Table 6.

- ***Average Risks Level*** is the highest probability allowed to a given evaluated risk in Table 6 when dispersing all the parameters following Table 5 distribution (i.e. **step 1** described in Remark 3).
- ***Limit Risks Level*** is the highest probability allowed to a given evaluated risk in Table 6 when a parameter in Table 5 is fixed at its maximum level (i.e. crosswind $WY33 = 30kts$) while dispersing the rest of the parameters following Table 5 distribution (i.e. **step 2** described in Remark 3).

Designers are advised to follow an iterative process:

1. design & and perform an initial tuning of the control law to cope with the average risks requirements,

Evaluated risk	Probability	Avr. Risks Level	Lim. Risks Level
Short landing	$\mathcal{P}(HTP60 < 0)$	10^{-6}	10^{-5}
Long landing	$\mathcal{P}(XTP > 915\,m)$	10^{-6}	10^{-5}
Hard landing (*avr.*)	$\mathcal{P}(VZTP > 10\,ft/s)$	10^{-6}	not applicable
Hard landing (*lim.*)	$\mathcal{P}(VZTP > 12\,ft/s)$	not applicable	10^{-5}
Decentered landing	$\mathcal{P}(YTP > 15\,m)$	10^{-6}	10^{-5}
Steep bank angle	$\mathcal{P}(PHI > 12^o)$	10^{-8}	10^{-7}
Steep sideslip angle	$\mathcal{P}(SSTP > 14^o)$	10^{-6}	10^{-5}

Table 6: Risks evaluation

2. verify and adjust the control law if needed, to satisfy the limit risks requirements.

In this challenge, the objective is to design an autoland control system that meets both average and limit risks requirements with **maximized crosswind** conditions.

The design presenting the highest crosswind will be declared as challenge best design. In case of two design present the same level of maximum crosswind, the design presenting higher margins with respect to Table 6 limits will be considered as the best.

For a **quick start** with the control design problem, a **baseline controller** is proposed and implemented in a closed-loop SIMULINK model (ALS.slx) as shown on Figure 2.

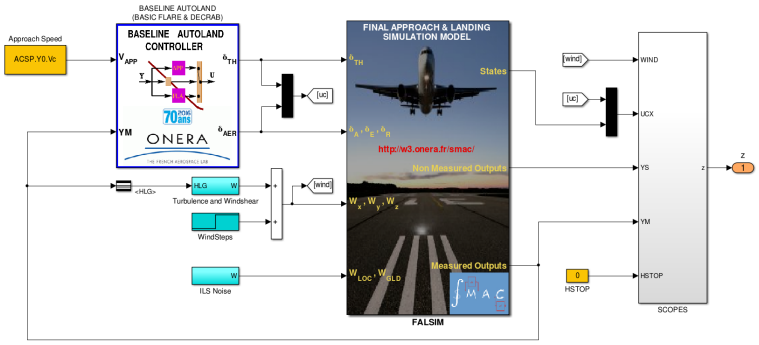

Figure 2: Closed-Loop Model with a Baseline Autoland Control System

This autoland system enables to perform correct landings on a reduced parametric domain with low to medium wind conditions. The corresponding Monte-Carlo analysis results (obtained with the routine MCplots) are displayed on Figure 3.

4 Description of the software package

The software package *CALC.zip* related to the proposed challenge can be downloaded from the SMAC project website **w3.onera.fr/smac** in the benchmarks section. This package contains 6 MATLAB functions, 3 SIMULINK files and 2 MATLAB script files. They are briefly described next. Further details can be found in the help of the main routines. A quick start guide is also provided.

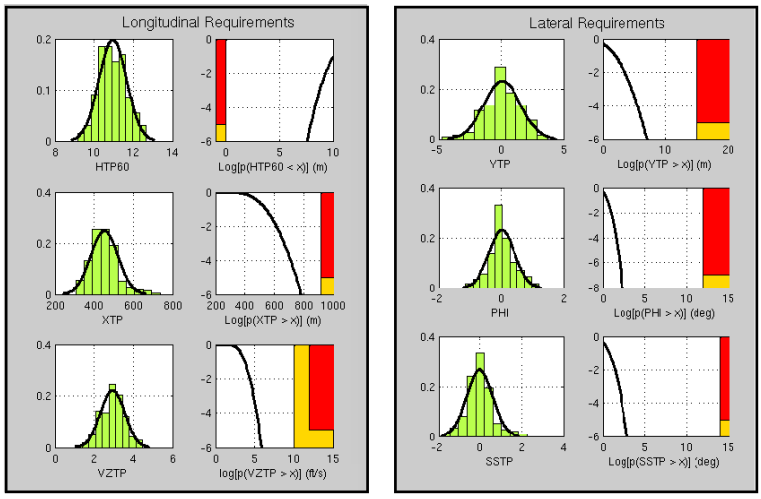

Figure 3: Results of Monte-Carlo analysis on a reduced operating domain

4.1 MATLAB functions & script files

4.1.1 initACSP

This routine initializes to empty matrices the main 13 fields of the global **ACSP** (AiCraft & Systems Parameters) variable. The fields are as follows:

1. **ACSP.U0:** Intitial inputs
2. **ACSP.X0:** Intitial states
3. **ACSP.Y0:** Intitial outputs
4. **ACSP.MIG:** Mass, Inertia & Geometry parameters
5. **ACSP.COEF:** Aerodynamic Coefficients
6. **ACSP.ATM:** Atmospheric data
7. **ACSP.ENG:** Engines characteristics
8. **ACSP.ACT:** Actuators parameters
9. **ACSP.ILS:** ILS parameters
10. **ACSP.RWY:** Runway characteristics (height, slope)
11. **ACSP.TURBW:** Turbulences & wind parameters
12. **ACSP.NOISE:** ILS noise & bias
13. **ACSP.ALS:** AutoLand System data

4.1.2 initALS

This routine initializes the AutoLand System with the proposed baseline controller. The data are stored in 3 subfields of ACSP:

- **ACSP.ALS.KLON:** Longitudinal feedback gain during approach
- **ACSP.ALS.KLAT:** Lateral feedback gain during approach
- **ACSP.ALS.HLON:** Longitudinal feedforward gain during approach
- **ACSP.ALS.HLAT:** Lateral feedforward gain during approach
- **ACSP.ALS.kpGLD:** Longitudinal guidance proportional gain
- **ACSP.ALS.kpLOC:** Lateral guidance proportional gain

- **ACSP.ALS.kdLOC:** Lateral guidance derivative gain
- **ACSP.ALS.HFL:** Flare Altitude ($15\,m$)
- **ACSP.ALS.HDC:** Decrab Altitude ($9\,m$)

Of course, these fields can be freely modified, renamed and additional fields can be defined according to the needs.

4.1.3 ACStrim

This routine is central in the package. It is closely related to the *ACS.slx* file. For given flight conditions stored in the input argument *flightpar* it performs a longitudinal trimming of the aircraft and initializes the first 10 fields of ACSP. Optionnaly (according to the number of outputs arguments), the routine will also deliver a 16^{th}-order linear model around the trimmed conditions. All required details can be found by typing *help ACStrim*.
Remark: *Note that the trimming algorithm has been implemented in a way to provide a reliable solution with a minimized computational time, which is of high interest for Monte-Carlo evaluations. Therefore, the trimming is restricted to the longitudinal dynamics. It is thus assumed that no crosswind is initially applied. Evaluations with crosswind conditions are then performed with a lateral wind that starts from zero and progressively increases. Realistic profiles are generated by a specific Turbulence block implemented in the* SIMULINK *file ALS.slx.*

4.1.4 ALplots

After a time-domain simulation has been performed, either using ALS1.slx, ALS2.slx or any other user-defined closed-loop Simulink model with a compatible output port z (with 36 components), this function plots the results by visualizing the main variables on three figures including subplots. The first one is dedicated to the main outputs to be controlled (landing trajectory, velocity, vertical speed,...). The second figure shows the commanded and realized control inputs while the last one shows the wind inputs.

4.1.5 MCpar

The routine MCpar generates a structured variable with 9 fields of randomly distributed parameters according to Table 5. It should be used to perform Monte-Carlo simulations.

4.1.6 MCplots

Once a set of N landings has been performed and the 6 parameters to be evaluated have been stored in a $6 \times N$ matrix, this routine plots the results in bar-diagrams and performs gaussian interpolations. The probabilities discussed in Table 6 are also evaluated and visualized through cumulative distributions functions.

4.1.7 MCsim

Based on a given SIMULINK file, whose name is to specified as an input argument, this routine runs the Monte-Carlo simulations & analysis *w.r.t* average or limit risks requirements. The above two routines *MCpar* and *MCplots* are used by *MCsim*.

4.1.8 ALeval

This script is dedicated to the evaluation process of the designed autoland system. Nominal approach & landing simulations without wind nor noises are performed first. Next, deterministic wind inputs are introduced. Finally, a turbulence model is considered and Monte-Carlo simulations are performed.

4.2 SIMULINK files

4.2.1 ACS model

This model is an open SIMULINK implementation of the aircraft model , its engines, actuators and ILS system described in Section 2. It only contains standard blocks mainly involving gains, integrators, signal additions and multiplications operators. As is already emphasized in Subsection 2.5, the model is to be run with the first-order fixed-step solver ode1. The sampling time must be fixed to $\tau = 0.05\,s$.
Important Remark: The model ACS is called by the trimming & linearization routine *ACStrim*. Therefore, it should not be modified.

4.2.2 ALS model

This model implements a closed-loop interconnection of the above ACS model with a baseline Autoland control system. The latter has been optimized for specific flight conditions and is not supposed to cover the whole operating domain. It can be used to perform landings subject to different flight conditions, deterministic wind inputs, windshear, turbulences and ILS noises.

4.3 A quick-start guide

For a good and fast understanding of the models and the various routines described above, the best way is to start with the *ALsim* script file and run it line-by-line. The file is well documented and quite easily understandable. Next, to get familiar with the statistical anlaysis and requirements, the second script file *MCsim* should be considered.

5 The evaluation process

Teams are invited to submit their work to the dedicated open-tack session of the forthcoming IFAC world congress in Toulouse. Contributions can be:

- methodology-oriented papers with illustrations on the benchmark
- application-oriented papers with extensive tests (see above) in order to demonstrate that the proposed controller has fulfilled the required objectives.

During the congress, simulations and final tests will be performed to evaluate and compare your autoland system with others! You are then strongly invited to come to the conference with:

- the latest version of your controller in a Matlab/Simulink (2012b or higher) format with the same input/output structure as the baseline solution provided with the benchmark,
- a data file (in mat format) which contains the structured variable ACSP with the ALS field related to your controller. Note that all subfields of ACSP.ALS may be customized according to your own needs provided this is compatible of course with your Simulink model.

Good luck and enjoy with the model !

References

[1] J-M. Biannic and P. Apkarian. A new approach to fixed-order H_∞ synthesis : Application to autoland design. In *Proceedings of the AIAA GNC*, Montreal, Canada, August 2001.

[2] J-M. Biannic and C. Roos. Flare control law design via multi-channel H_∞ synthesis: Illustration on a nonlinear aircraft benchmark. In *Proceedings of the American Control Conference (ACC)*, Chicago, IL. USA., July 2015.

[3] I. Kaminer and P. Khargonekar. Design of the Flare Control Law for Longitudinal Autopilot using H_∞ synthesis. In *Proceedings of the* 29^{th} *IEEE Conference on Decision and Control*, Honolulu, Hawaii, December 1990.

[4] G. Looye and H.D. Joos. Design of Autoland Controller Functions with Multiobjective Optimization. *Journal of Guidance, Control and Dynamics*, 29(2):475–484, March-April 2006.

[5] H. Sadat-Hoseini, A. Fazelzadeh, A. Rasti, and P. Marzocca. Final Approach and Flare Control of a Flexible Aircraft in Crosswind Landings. *Journal of Guidance, Control and Dynamics*, 36(4):946–957, August 2013.

A Numerical data

A.1 Mass and Geometry

- The average mass of the considered aircraft is: $m_0 = 150000\,kg$. According to fuel quantity and payload, it may vary between $m_{min} = 120000\,kg$ and $m_{max} = 180000\,kg$.
- The coefficients of the inertia matrix are given by:

$$\begin{cases} I_{xx} = 10^7 + 45\,(m - m_0) \\ I_{yy} = 1.6\,10^7 + 33\,(m - m_0) \\ I_{zz} = 2.4\,10^7 + 100\,(m - m_0) \\ I_{xz} = -10^6 \end{cases}$$

- $S = S_{ref} = 360\,m^2$, $L = L_{ref} = 7.5\,m$, $z_{eng} = 2\,m$

A.2 Aerodynamic forces coefficients

- $C_{L_0} = 0.90$, $C_{L_\alpha} = 5.5$, $C_{L_q} = 3.3$
- $C_{L_{\delta_e}} = 0.32$, $C_{L_H} = 0.20$, $\lambda_L = 0.12$
- $C_{D_0} = 0.065$, $C_{D_\alpha} = 0.4$, $C_{D_{\alpha_2}} = 1.55$
- $C_{Y_\beta} = -0.7$, $C_{Y_{\delta_r}} = 0.25$
- $C_{l_\beta} = -3$, $C_{l_p} = -15$, $C_{l_{r_0}} = 5$, $C_{l_{r_a}} = 35$
- $C_{l_{\delta_a}} = -0.7$, $C_{l_{\delta_r}} = 0.2$
- $C_{m_0} = -0.3$, $C_{m_\alpha} = -1.5$, $C_{m_q} = -12$, $C_{m_{\delta_e}} = -1.2$
- $C_{m_{h_0}} = -0.09$, $C_{m_{h_\alpha}} = -0.9$, $\lambda_m = 0.15$
- $C_{n_{\beta_0}} = 0.85$, $C_{n_{\beta_\alpha}} = -1.95$, $C_{n_{p_0}} = -3$, $C_{n_{p_\alpha}} = -35$
- $C_{n_r} = -7$, $C_{n_{\delta_a}} = -0.04$, $C_{n_{\delta_r}} = -1.25$