



**HAL**  
open science

## **Model-based Robust Transient Control of Reusable Liquid-Propellant Rocket Engines**

Sergio Pérez-Roca, Julien Marzat, Hélène Piet-Lahanier, Nicolas Langlois, Marco Galeotta, François Farago, Serge Le Gonidec

► **To cite this version:**

Sergio Pérez-Roca, Julien Marzat, Hélène Piet-Lahanier, Nicolas Langlois, Marco Galeotta, et al.. Model-based Robust Transient Control of Reusable Liquid-Propellant Rocket Engines. *IEEE Transactions on Aerospace and Electronic Systems*, 2021, 57 (1), pp.129-144. <10.1109/TAES.2020.3010668>. <hal-03143086>

**HAL Id: hal-03143086**

**<https://hal.science/hal-03143086v1>**

Submitted on 16 Feb 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

# Model-based Robust Transient Control of Reusable Liquid-Propellant Rocket Engines

Sergio Pérez-Roca, Julien Marzat, H el ene Piet-Lahanier, Nicolas Langlois, Marco Galeotta, Fran ois Farago and Serge Le Gonidec

**Abstract**—Reusable liquid-propellant rocket engines (LPREs) imply more demanding robustness requirements than expendable ones due to their extended capabilities. Therefore, the goal of this work was to develop a control loop adapted to all the operating phases of LPRE, including transients, and robust to internal parametric variations. Firstly, thermo-fluid-dynamic simulators representative of the gas-generator-cycle engines were built. These simulators were subsequently translated into nonlinear state-space models. Based on these models, the continuous sub-phase of the start-up transient is controlled to track pre-computed reference trajectories. Beyond the start-up, throttling scenarios are managed with end-state-tracking algorithm. Model Predictive Control has been applied in a linearised manner with robustness considerations to both scenarios, in which a set of hard state and control constraints must be respected. Tracking of pressure (thrust) and mixture-ratio operating points within the design envelope is achieved in simulation while respecting constraints. Robustness to variations in the predominant parameters, to external state perturbations and to the possible impact of an observer on the loop, is demonstrated.

**Index Terms**—Liquid-propellant rocket engines, control-oriented nonlinear modelling, transients, Model Predictive Control (MPC), robustness, parameter-varying, trajectory planning

## I. INTRODUCTION

THE current context of launcher vehicles design is strongly related to the reusability feature. From the automatic control perspective, this potential need for reusable liquid-propellant rocket engines (LPREs) is translated into stricter robustness requirements, mainly forced by their multi-restart and thrust-modulation capabilities. These demanding requirements arise from the possible endogenous perturbations due to components faults or evolving parameters and from exogenous perturbations related to the more complex mission profiles envisaged by new launchers. In LPREs, the control references generally correspond to the two variables defining their operating envelope: combustion-chamber pressure (related to thrust)

S. P erez-Roca is on leave from *DTIS, ONERA, Universit  Paris-Saclay*, 6 chemin de la Vauve aux Granges, 91123 Palaiseau, France and *CNES - Direction des Lanceurs*, 52 Rue Jacques Hillairet, 75612 Paris, France (e-mail: sergio.perez\_roca@onera.fr).

J. Marzat and H. Piet-Lahanier are with *DTIS, ONERA, Universit  Paris-Saclay*, 6 chemin de la Vauve aux Granges, 91123 Palaiseau, France (e-mails: julien.marzat@onera.fr, helene.piet-lahanier@onera.fr)

N. Langlois is with *Normandie Universit , UNIROUEN, ESIGELEC, IRSEEM*, Rouen, France (e-mail: nicolas.langlois@esigelec.fr)

M. Galeotta and F. Farago are with *CNES - Direction des Lanceurs*, 52 Rue Jacques Hillairet, 75612 Paris, France (e-mails: marco.galeotta@cnes.fr, francois.farago@cnes.fr)

S. Le Gonidec is with *ArianeGroup SAS*, For t de Vernon, 27208 Vernon, France (e-mail: serge.le-gonidec@ariane.group)

Manuscript received 21/01/2020; revised 14/05/2020

## NOMENCLATURE

$\alpha$	Valve opening angle [�]
$\alpha_P$	Auxiliary variable in QIH
$\mathbf{X}, \mathbf{U}$	State and control-inputs trajectories
$\mathbf{x}, \mathbf{u}$	States and control inputs
$\mathbf{z}$	Integrator decision variables
$\Delta$	Uncertainty in dynamic matrices
$\dot{m}$	Mass flow [kg/s]
$\gamma$	Minimised scalar in epigraph <i>minimax</i>
$\omega$	Rotational speed [rad/s]
$A$	Area // Valve section [m <sup>2</sup> ]
$A, B$	State-space matrices
$a_i - h_i$	Internal-parameter composites in ODE
$C^*$	Characteristic exhaust speed [m/s]
$I$	Scenarios indices set
$i$	Igniter/starter activation flag // Perturbed scenario
$J$	Cost function
$J_{perf}$	Global performance criterion
$K_I$	MPC integrator gain matrix
$MR$	Mixture ratio [-]
$n, m$	Number of states and of control inputs
$N_p, N_u$	Prediction and control horizons in steps
$P$	Positive-definite LYAPUNOV matrix
$p$	Pressure [bar]
$Q, R, S$	Weight matrices in cost
$T$	Temperature [K]
$t$	Time [s]
$w_t$	Exogenous input

and mixture ratio (oxidiser to fuel ratio), which are usually controlled by means of adjustable valves. One of the design capabilities of the future European *PROMETHEUS* engine is to throttle down to 30% of thrust [1]. Tracking and robustness have to be maintained at those low throttle levels, where physical phenomena are more difficult to anticipate.

In the literature, this problematic has been partially faced, generally attaining narrow and non-continuous throttling envelopes [2]. Reference [3] remarks that thrust and mixture-ratio control in real flights can be achieved in open loop (OL) if a high accuracy is not required, or if off-line optimisation strategies are pertinent [4], [5]. However, steady-state closed-loop (CL) control is performed in the majority of publications. Most of authors have selected conventional PID-based techniques for controlling around a given nominal point making use of linearised models [6]–[11]. The only reference making use of closed-loop nonlinear techniques [12], considered damage modelling and control in their loop, as other NASA reusable-engine publications have done [4], [13], [14]. Other more complex approaches present in the literature, incorporating some hybrid [14], [15] or robust techniques facing uncertainties and some faulty scenarios [14], [16], [17],



combustion pressure models has been applied. Concretely, the binary variables corresponding to igniter and starter activations manage combustion-related terms in differential equations (as in (8)-(9)).

After its construction, the *Vulcain 1* simulator was validated qualitatively in terms of transient behaviour and throttling and quantitatively in terms of the nominal steady state. The entire start-up transient, which is the main operation that the simulator has to capture, is satisfactorily predicted in terms of pressures, mass flows and rotational speeds, as validated by experts.

The developed engine simulator was then translated into nonlinear state-space models by joining components equations symbolically. That is to say, a global expression of systems dynamic behaviour as a function of its state variables, internal parameters and control inputs was targeted. This process was performed via the symbolic mathematical environment *Maple*, taking into account the internal definition of each LPRE component and of the global system (causal interconnection of equations, following the schematics in Fig. 1).

At this stage, the model is referred to as *complex NLSS* (nonlinear state-space) or  $f_c$ :

$$\dot{\mathbf{x}} = f_c(\mathbf{x}, \mathbf{u}_c, \mathbf{u}_d, \mathbf{w}_t), \quad (1)$$

where  $\mathbf{x}$  is the state vector,  $\mathbf{u}_c$  is the vector of continuous inputs,  $\mathbf{u}_d$  is the vector of discrete inputs and  $\mathbf{w}_t$  is an exogenous time-varying input corresponding to the injected starter mass flow. Along the symbolic component-connection process, some simplifications with respect to simulators equations had to be performed so as to obtain globally defined expressions. That is to say, equations containing internal state-conditional or delayed-feedback statements distancing the model from the mainly continuous state-space formalism were modified. Even so, this first state-space model presented an excessive complexity for control design, since many ODEs contained a high number of coupled nonlinearities in states and control inputs. Thus, it was further reduced until attaining the here-called *simplified NLSS* such that  $\dot{\mathbf{x}} = f_s(\mathbf{x}, \mathbf{u}_c, \mathbf{u}_d, \mathbf{w}_t)$ , more tractable for the derivation of control laws. Physical assumptions and mathematical rewriting were carried out in order to reduce state dependencies and the amount of terms in equations (more details in [21]). The same amount of states and inputs has been maintained since all of them are relevant to systems dynamics, having similar time scales and coupling effects.

The number of states is  $n = 12$  and the number of control inputs is  $m = 5$ . Here, the state vector  $\mathbf{x}$  comprises turbopumps speeds, the several pressures in the system (combustion chamber, GG, and pre-turbine cavities), and mass flows streaming through valves and pre-turbine pipes, as:

$$\mathbf{x}_V = [\omega_H \quad \omega_O \quad p_{CC} \quad p_{GG} \quad p_{LTH} \quad p_{VGC} \quad \dot{m}_{LTH} \quad \dot{m}_{VCH} \quad \dot{m}_{VCO} \quad \dot{m}_{VGH} \quad \dot{m}_{VGO} \quad \dot{m}_{VGC}]^T. \quad (2)$$

The states with higher tracking importance are incorporated into a reduced state vector  $\mathbf{x}_z$  (of length  $n_z$ ):

$$\mathbf{x}_z = [p_{CC} \quad \dot{m}_{VCH} \quad \dot{m}_{VCO} \quad \dot{m}_{VGH} \quad \dot{m}_{VGO}]^T. \quad (3)$$

The control vectors contain inputs of continuous and discrete nature, which renders the model hybrid from the control perspective. The sections of control valves are the continuous ones:

$$\mathbf{u}_c = [A_{VCH} \quad A_{VCO} \quad A_{VGH} \quad A_{VGO} \quad A_{VGC}]^T, \quad (4)$$

whereas igniter and starter activations consist in the discrete ones:

$$\mathbf{u}_d = [i_{CC} \quad i_{GG}]^T. \quad (5)$$

Besides, all equations, states and control variables are rendered non-dimensional with respect to the nominal equilibrium values. An overview of the *Vulcain 1* simplified state-space system  $\dot{\mathbf{x}} = f_s(\mathbf{x}, \mathbf{u}_c, \mathbf{u}_d, \mathbf{w}_t)$  is presented in (6)-(17), where  $a_i, b_i, \dots, h_i \in \mathbb{R}^+$  are internal-parameters composites that serve to express the ODE in a more compact manner. Since they are only dependent on parameters and on the exogenous input influence ( $\mathbf{w}_t$ ), they are either constant or time-dependent (indicated by subscript  $t$ ). Temperatures  $T_{CC}$  and  $T_{GG}$  are considered as varying parameters depending on  $MR$  (polynomial regressions).

$$\dot{x}_1 = \left( -\frac{a_{1,t}}{\sqrt{T_{GG}}} x_5 x_1 + b_{1,t} x_5 - c_1 (x_8 + x_{10}) x_1 - d_1 (x_8^2 + x_8 x_{10}) \right) u_{d_2}, \quad (6)$$

$$\dot{x}_2 = \left( a_2 x_2^2 - \frac{b_{2,t}}{\sqrt{T_{GG}}} x_6 x_2 + c_{2,t} x_6 - d_2 (x_9 + x_{11}) x_2 - e_2 (x_9^2 + x_9 x_{11}) \right) u_{d_2} \quad (7)$$

$$\dot{x}_3 = (a_3 - b_3 T_{CC}) (x_8 + x_9) u_{d_1} + (c_3 + d_3 T_{CC}) (x_8 + x_9) - e_3 \sqrt{T_{CC}} x_3, \quad (8)$$

$$\dot{x}_4 = (a_4 - b_{4,t} T_{GG}) (x_{10} + x_{11}) u_{d_2} + (c_{4,t} + d_{4,t} T_{GG}) (x_{10} + x_{11}) - e_{4,t} T_{GG} (x_7 + x_{12}) + (f_{4,t} + g_{4,t} T_{GG}) w_t, \quad (9)$$

$$\dot{x}_5 = a_{5,t} T_{GG} x_7 - b_{5,t} \sqrt{T_{GG}} x_5, \quad (10)$$

$$\dot{x}_6 = a_{6,t} T_{GG} x_{12} - b_{6,t} \sqrt{T_{GG}} x_6, \quad (11)$$

$$\dot{x}_7 = a_7 (x_4 - x_5) - \frac{b_{7,t} T_{GG} x_7^2}{x_4}, \quad (12)$$

$$\dot{x}_8 = \frac{(a_8 x_1^2 - b_8 x_8^2 - c_8 x_{10} x_8 - d_8 x_3 + e_8) u_{c_1}^2 - f_8 x_8^2}{(g_8 u_{c_1} + h_8) u_{c_1}}, \quad (13)$$

$$\dot{x}_9 = (a_9 x_2^2 - b_9 x_9 x_2 - c_9 x_9^2 - d_9 x_3 + e_9) u_{c_2} - \frac{f_9 x_9^2}{u_{c_2}}, \quad (14)$$

$$\dot{x}_{10} = (a_{10} x_1^2 - b_{10} x_8^2 - c_{10} x_{10} x_8 - d_{10} x_{10}^2 - e_{10} x_4 + f_{10}) u_{c_3} - \frac{g_{10} x_{10}^2}{u_{c_3}}, \quad (15)$$

$$\dot{x}_{11} = (a_{11} x_2^2 - b_{11} x_9 x_2 - c_{11} x_9^2 - d_{11} x_{11}^2 - e_{11} x_4 + f_{11}) u_{c_4} - \frac{g_{11} x_{11}^2}{u_{c_4}}, \quad (16)$$

$$\dot{x}_{12} = \left( x_4 - x_6 - \frac{a_{12,t} T_{GG} x_{12}^2}{x_4 u_{c_5}^2} \right) u_{c_5}. \quad (17)$$

It is clear that this system representation, even if it has been simplified, presents numerous nonlinearities and is non-affine

with respect to control inputs. These GG-LPRE state-space system representations for control purposes with time-varying coefficients and discrete inputs consist in one of the main contributions of this work, as reviewed in [18]. The most usual modelling approaches had concerned linear or linearised models coming from identification or simplified fluid mechanics. The closest control-oriented nonlinear modelling approach is the staged-combustion LPRE model in [25], used in the articles [4], [12], [26], [27]. That approach also considered causal component interconnections but concerned time-invariant coefficients and only continuous inputs. The *PROMETHEUS* engine has also been modelled via the approach in this article in [23].

### III. ANALYSIS OF PROPOSED MODELS FOR CONTROL PURPOSES

The analysis of the previous models is necessary in order to select and build the most appropriate control strategy. Several aspects can be analysed, ranging from the effect of simplifications, stability, observability, controllability and sensitivity to parameter variations. A concise, less detailed analysis was provided in [23]. An example of control-oriented analysis of thermodynamic systems is [28]. The first aspect to introduce is the fact that the derived nonlinear state-space models do not present an analytical solution to their equilibrium point  $\dot{\mathbf{x}} = \mathbf{0}$  even in their simplified form. Knowledge on the desired equilibrium point in some states is used to numerically reconstruct a full-state equilibrium vector in Section IV, as a part of the controller.

*a) Effect of simplifications:* the different simplifications carried out, such as the consideration of some constant thermodynamic properties in  $f_s$ , obviously increase the modelling error slightly. It is specially present in mass flows, which can present a mismatch in start-up simulations up to 15% at each step of simplification (simulator,  $f_c$  and  $f_s$ ). Errors in the rest of states remain below 12% at each step. The steady-state value of  $p_{CC}$  is predicted within 1% accuracy and slope changes and time scales are well synchronised. Linearised models are accurate along a short interval around a selected equilibrium point ( $\pm 0.1s$  in start-up).

*b) Stability:* stability is an inherent property of these GG-LPRE models, which describe a system that self-compensates disturbance in the flow cycle due to its high coupling [29]. The possible high-frequency destabilising phenomena in LPREs, such as combustion instabilities or frequency couplings, are not modelled here since their control is out of scope with the available actuators (bandwidth under 10Hz). Hence, when analysing linearised models about transient trajectories, all eigenvalues present negative real parts.

*c) Observability:* the state is assumed to be completely measurable in the real engine. This is a realistic assumption for  $\omega$  and  $p$ . However, mass flows are normally not measured in LPREs, but estimated through pressure, temperature and volumetric flow measurements [30]. During the discrete part of the transient, observability scarcities can appear due to the low volumetric flows characteristic of that phase [9]. During the fully continuous phases, the linearised models derived in this

work locally fulfil the observability criteria for time-invariant systems.

*d) Controllability:* controllability can be verified after linearising  $f_s$  about the whole continuous start-up trajectory. In other words, all states can be controlled via valves during the continuous sub-phase. However, as commented for observability, controllability is not ensured during the discrete sub-phase, where there are valves that have to remain closed in order to execute a safe sequence. The analysis shows that the system is controllable with the GG valves alone, the last ones to open in the sequence.

*e) Parameters sensitivity:* in the case of the SSME (Space Shuttle Main Engine), studies on parameter estimation via genetic algorithms have been carried out [31]. Engine parameters in this work have been provided for the nominal case. In order to analyse the effect of their variations on the transient behaviour of the engine, which is relevant to this control problem, a series of sensitivity analyses have been performed. The list of considered varying engine parameters is summarised in Table I. There may be different sources of deviation. Some parameters might vary during operation or between different engine runs. Others are estimated for modelling and hence are not 100% certain. Inter-engine material discrepancy can also be a source of variation. Indeed, tanks pressures and combustion efficiencies might oscillate during flight or between engine executions. Fluidic resistances, inertias and combustion efficiencies cannot be easily determined; they have to be estimated from tests or simulations. Orifices in the cycle may present varying effective hydraulic sections during operation and/or inter-engine discrepancies. The Sobol global sensitivity analysis method [32] has been applied. It is of interest here to study the variations in terms of a global performance criterion  $J_{perf}$ , defined as the weighted addition of several simulated indicators:

$$J_{perf} = |err_{pCC}| + |err_{MRCC}| + |err_{MRGG}| + |err_{MRPI}| + 0.001 \cdot |A_{pCC}| + 0.01 \cdot |os_{pCC}|, \quad (18)$$

where  $err$  are static steady-state errors in the main operating quantities (contained in  $\mathbf{x}_z$ ),  $os$  is overshoot (in %) and  $A_{pCC}$  is the surface between the perturbed  $p_{CC}$  start-up transient curve and the nominal desired one (in %). Thus, the lower the criterion the better the performance. The sampling of parameters variations according to Table I has been carried out via the Latin Hypercube Sampling (LHS), recommended for calculating the Sobol global indices. In this fashion, a set of 1000 parameter-variation combinations, with their corresponding output  $J_{perf}$  (computed with the simulator model), is provided to a Kriging-based Sobol algorithm based on [33]. This algorithm creates a Kriging model [34] based on the provided input-output data, which serves as an interpolator and hence avoids the execution of further costly simulations. After running 1 million Monte Carlo evaluations of that less expensive model, the most influential parameters, related to the higher Sobol global indices, can be identified. These are  $AthCC$ ,  $AthT$  and  $p_{PI}$ , whose indices present an order of magnitude of 0.1 (normalised with respect to one), whereas

Parameter	Definition	Considered variation range [%]	Variation source
$p_{PI}$	Tanks pressures, coupled with inlet densities and temperatures	$\pm 10$	Oscillation during/between operations
$Res_{LC}$	LC lines fluidic resistances	$\pm 10$	Modelling uncertainty
$Res_I$	Injectors fluidic resistances	$\pm 10$	Modelling uncertainty
$Res_{CR}$	Cooling-circuit fluidic resistance	$\pm 10$	Modelling uncertainty
$Ine_{CR}$	Cooling-circuit fluidic inertia	$\pm 10$	Modelling uncertainty
$\eta_{CC}$	CC combustion efficiency	$\pm 5$	Modelling uncertainty, oscillation during/between operations
$\eta_{GG}$	GG combustion efficiency	$\pm 5$	Modelling uncertainty, oscillation during/between operations
$Res_{LTH}$	LTH line fluidic resistance	$\pm 10$	Modelling uncertainty
$Ine_{LTH}$	LTH line fluidic inertia	$\pm 10$	Modelling uncertainty
$V_{cav}$	Pre-turbine cavities volume	$\pm 10$	Modelling uncertainty
$Ath_{CC}$	CC throat section	$\pm 1$	Inter-engine discrepancy, alteration during/between operations.
$Ath_T$	Turbine inlet sections	$\pm 1$	Inter-engine discrepancy, alteration during/between operations.
$Ath_{LE}$	Turbine outlet sections	$\pm 1$	Inter-engine discrepancy, alteration during/between operations.

TABLE I: List of *Vulcain 1* engine parameters considered in sensitivity analysis

$Res_I$ ,  $Res_{CR}$  and  $\eta_{GG}$  indices are about 0.01. The effects of variations in the rest of parameters can be neglected.

f) *Worst-case scenario*: building on the previous sensitivity analysis, it is of interest to find extreme parameter-varying cases that would degrade the performance criterion (18) to a higher extent. This information is used later in robust control design (Section IV). From the practical experience of LPRE system behaviour in simulations and in flight, it is known that  $p_{PI}$  can more probably vary in reality, especially in the new reusability scenarios. Thus, a higher priority is put on them in the following. Furthermore, a worst-case scenario of variations of the previous shortlist of influential parameters has also been computed. A Kriging-based black-box maximisation of the performance criterion (18) based on the EGO algorithm [35], as a function of the possible combinations of parameters variations, defined in Table I, has been carried out. The outcome of this search, which practically attains the bounds of intervals, corresponds to a degradation of 106% in  $J_{perf}$  with respect to the nominal OL.

#### IV. CONTROL APPROACHES

The control goals on this system mainly concern reaching a desired end-state or following a predefined trajectory while complying with a set of hard constraints on  $\mathbf{x}$  and  $\mathbf{u}$ . This second objective is somewhat more important than tracking in order to avoid excessive temperatures (related to elevated mixture ratios) or rotational speeds during engine's operation, which could lead to catastrophic failures or to prevent the reuse of certain parts. The robustness of the solution against internal parametric variations is also required. All these goals, required for reusability, are to be attained at acceptable computational times.

The main control diagram considered in this article is depicted in Fig. 2. The controller receives full state and control references from an off-line preprocessor block and computes the corresponding control on-line, which is sent to valves (actuators block). To the right there is the simulation of the rocket engine, performed via the integration of the simulator

presented in Section II. Its initial conditions, when simulating the continuous part of the start-up transient, are taken as the outcome of the nominal discrete sub-phase ( $t = 1.5s$  in *Vulcain 1*). The inputs of the state-space model used for control are valve sections  $\mathbf{u}_c$  (4), as explained in Section II, in the following simply referred to as  $\mathbf{u}$ . However, the actuators mechanical model (internal hydraulic actuators of valves) requires an input in terms of opening angles  $\alpha$ . Thus, a conversion block, characterised by static and monotone non-linear functions, is needed. That internal actuator subsystem is considered as a second-order system for simplification, neglecting phenomena such as hysteresis and solid friction. Even though the estimation has not been developed in this article, the influence of the observer is simulated in the robustness analysis from Section V.

##### A. Preprocessing: reference generator

The preprocessing block, first presented in [22], serves as a novel model-based off-line reference generator for multivariable LPRE controllers. Indeed, the set of steady-state reference commands derived from launcher needs and considered in this work, a total of four, concern thrust or  $p_{CC}$  (interrelated) and mixture ratios in chambers (CC and GG) and in global terms. Since  $\mathbf{x}$  presents twelve states (presented in Section II for the *Vulcain 1* case (2)), the desired references are not sufficient to provide a complete target equilibrium point to an engine controller based on these models. A way of restoring full state and control reference vectors is necessary. Indeed, without these final ( $\mathbf{x}_r$ ,  $\mathbf{u}_r$ ), the posterior controllers would not attain the tracking goal with high precision, since they are used for linearisation and to build reference trajectories.

The four reference inputs are  $p_{CC,r}$ ,  $MR_{PI,r}$ ,  $MR_{CC,r}$  and  $MR_{GG,r}$ . The last three, in contrast to  $p_{CC}$ , do not directly correspond to states in the model. They establish relations between  $\tilde{m}_i$ . In the following, the presence of a tilde ( $\tilde{\cdot}$ ) on top of a quantity means that it is dimensional and its absence means the contrary. In the first place, thanks to the selected

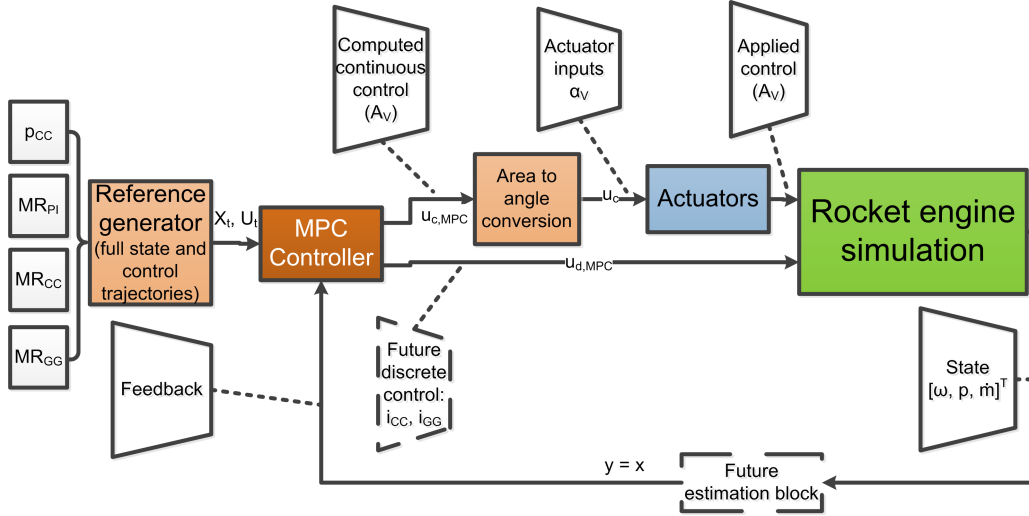


Fig. 2: General control-loop diagram

pressure  $p_{CC,r}$  and  $MR_{CC,r}$ , the choked-flow static equation

$$\dot{m}_{CC} = \frac{p_{CC} A_{th}}{C^*} \quad (19)$$

can provide the injected mass flows into the CC  $\dot{m}_{VCH,r}$  and  $\dot{m}_{VCO,r}$ , seeing that the characteristic velocity  $C^*$  depends on  $MR$ .  $A_{th}$  is throat area. Then, the rest of states at equilibrium, concerning rotational speeds, pressures in GG and pre-turbine cavities and the remaining mass flows, as well as the control inputs  $u_r$  making that state possible, are to be computed. Discrete control inputs  $u_d$  are considered active since a running steady-state is sought. The first two valve sections in  $u_c$ , corresponding to CC valves, are chosen to be opened at their nominal position (100%). This choice stems from the observation that the throttability of the engine is directly related to the GG valves, which can govern the whole state. The tuning of CC valves is not convenient in this kind of engines, even if off-nominal operating points are targeted. All the remaining equilibrium variables are determined by solving the following overdetermined system of nonlinear equations:

$$\begin{cases} \dot{\mathbf{x}} = f_c(\mathbf{x}_r, \mathbf{u}_r) = \mathbf{0} \quad (\dot{p}_{CC} = 0) \\ \frac{\dot{m}_{VCO} + \dot{m}_{VGO}}{\dot{m}_{VCH} + \dot{m}_{VGH}} = MR_{PI,r} \\ \frac{\dot{m}_{VGO}}{\dot{m}_{VGH}} = MR_{GG,r} \\ \dot{m}_{VGH} + \dot{m}_{VGO} = \dot{m}_{LTH} + \dot{m}_{VGC}. \end{cases} \quad (20)$$

The first set of equations forces the ODEs of the complex NLSS in *Vulcain 1* to be at equilibrium, the second and the third ones determine the indicated  $MR$  and the last one enforces the equilibrium of  $\sum \dot{m}_i$  in the GG. The resolution is performed numerically via nonlinear least squares due to the unavailability of an analytic solution of the system, of either  $f_c$  or  $f_s$ . The complex model  $f_c$  has been chosen to increase the accuracy with respect to the simulator and hence the real engine. This computation lasts around 2-3s by means of the *MATLAB*® *lsqnonlin* function. In the case of off-nominal points, the resulting  $u_r$  serves as a reference OL control in the event that closed-loop would not be available in the engine. In

flight, it is usually the case that ad-hoc parameters are tuned to achieve an alternative thrust level. Thus, this method can be useful for performing a 120%-thrust start-up for instance, even though robustness is not guaranteed at all. Performance indicators of these OL controllers in different scenarios are included in the comparison Tables II and III.

a) *Reference-trajectories generation*: once these end targets  $(x_r, u_r)$  have been computed, a reference start-up trajectory  $(X_t, U_t)$  from some given initial  $x_k$  and  $u_0$  can be built, as introduced in [24]. Throughout the article, the index  $k$  denotes the current time step and  $X$  and  $U$  are stacked vectors with future  $x$  and  $u$  at each time step  $j$  along a horizon  $N$  ( $N_p$  for states and  $N_u$  for control inputs):

$$\begin{aligned} X &= [x_k, \dots, x_{k+j}, \dots, x_{k+N_p}]^T \\ U &= [u_k, \dots, u_{k+j}, \dots, u_{k+N_u}]^T \end{aligned} \quad (21)$$

Thus,  $x_k$  represents the initial state, which in this work is assumed to be measurable.

In order to compute start-up trajectories, which is a new approach with respect to the LPRE-control literature, an optimisation and model predictive-based scheme is used. It can be regarded as an OL finite-horizon MPC scheme in which the prediction horizon is set to cover the duration of the start-up build-up transient, considered between 1.5s, the end of the discrete sequential phase, and 2.5s, the desired reference-crossing time in the studied engine *Vulcain 1*. This is a common approach in the literature of trajectory planning [36]. Other trajectory-planning methods have been proposed, especially in the field of robotic and UAV (unmanned aerial vehicle) motion, but they are also mostly based on optimisation [37], [38]. Here, it is of interest to take advantage of the nonlinear models developed.

This algorithm is based on the minimisation of the discretised

quadratic cost function  $J_{OL}$ , defined as:

$$J_{OL}(\mathbf{X}_t, \mathbf{U}_t) = \left( \sum_{j=0}^{N_{p,OL}} \Delta \mathbf{x}_{t,k+j}^T Q_{OL} \Delta \mathbf{x}_{t,k+j} + \sum_{j=0}^{N_{u,OL}} \Delta \mathbf{u}_{t,k+j}^T R_{OL} \Delta \mathbf{u}_{t,k+j} \right) \Delta t, \quad (22)$$

where  $\Delta \mathbf{x}_{t,k} = \mathbf{x}_{t,k} - \mathbf{x}_r$  and  $\Delta \mathbf{u}_{t,k} = \mathbf{u}_{t,k} - \mathbf{u}_r$  are the variables to cancel.  $Q_{OL} = I_n$  and  $R_{OL}$  are the weight matrices associated to states and control respectively. Diagonal terms in  $R_{OL}$  are set to  $10^{10}$  so as to minimise control action.  $N_{p,OL}$  and  $N_{u,OL}$  are states and control prediction horizons, which in this case are taken equal to the horizon (1s) over a discretisation time of  $\Delta t = 10$ ms. This selection of  $\Delta t$  is constrained by control limitations.

Concerning the dynamics considered to predict the behaviour at each time step  $k + j$ , as shown in Section II, the system is highly nonlinear. Neglecting nonlinear dynamics at points far from the equilibrium can lead to non-negligible prediction errors. However, the main repercussion of imposing nonlinear constraints in optimisation problems is generally the loss of convexity of the optimised function and hence the increase in resolution complexity. The compromise chosen here, related to the specific behaviour of the system, is the inclusion of nonlinear dynamic constraints until the system approaches its reference values to within 90% (step  $j = N_{p,OL90\%}$ ). This coincides approximately with the first half of the transient, where modelling errors of linearisation would be relevant if linear dynamics were used. Concretely, the aforementioned simplified NLSS  $f_s$  (6)-(17) is discretised via an Euler implicit scheme:

$$\mathbf{x}_{t,k+j+1} = \mathbf{x}_{t,k+j} + f_s(\mathbf{x}_{t,k+j+1}, \mathbf{u}_{t,k+j+1}, \mathbf{w}_{t,k+j+1}) \Delta t, \quad j \in [0, N_{p,OL90\%} - 1]. \quad (23)$$

This scheme has been selected since it is the most numerically stable among the first-order integration methods, required for lowering the complexity of the optimisation by reducing the interdependencies between decision variables. Once the pressure  $p_{CC}$  ( $x_3$  or  $x_{z,1}$ ) attains its reference value, the  $\mathbf{x}_z$  states (3) (the most relevant for tracking) are forced to be equal to the end-reference values and only linear dynamics is imposed, in order to simplify the optimisation. Linear dynamics ( $A_c, B_c$ ) stems from the end-state linearisation of  $f_s$  about  $(\mathbf{x}_r, \mathbf{u}_r)$ , which is then discretised via zero-order hold (exact discretisation) at  $\Delta t$  ( $A_d, B_d$ ):

$$\Delta \mathbf{x}_{t,k+j+1} = A_d(\mathbf{x}_r, \mathbf{u}_r) \Delta \mathbf{x}_{t,k+j} + B_d(\mathbf{x}_r, \mathbf{u}_r) \Delta \mathbf{u}_{t,k+j}, \quad j \in [N_{p,OL90\%}, N_{p,OL} - 1]. \quad (24)$$

Having defined the different dynamics, the optimisation algorithm which is executed once for the whole horizon (OL

trajectory planning) under constraints is the following:

$$\min_{\mathbf{X}_t, \mathbf{U}_t} J_{OL}(\mathbf{X}_t, \mathbf{U}_t) \quad (25)$$

$$\text{s.t. } \mathbf{X}_t \in X_{OL}, \quad \mathbf{U}_t \in U_{OL} \quad (26)$$

$$A_{ineq,OL}[\mathbf{X}_t \quad \mathbf{U}_t]^T \leq \mathbf{b}_{ineq,OL} \quad (27)$$

$$\begin{aligned} \mathbf{x}_{t,k+j+1} &\leq \mathbf{x}_{t,k+j} \\ &+ f_s(\mathbf{x}_{t,k+j+1}, \mathbf{u}_{t,k+j+1}, \mathbf{w}_{t,k+j+1}) \Delta t + \varepsilon, \\ &\quad \forall j \in [0, N_{p,OL90\%} - 1] \end{aligned} \quad (28)$$

$$\begin{aligned} \mathbf{x}_{t,k+j+1} &\geq \mathbf{x}_{t,k+j} \\ &+ f_s(\mathbf{x}_{t,k+j+1}, \mathbf{u}_{t,k+j+1}, \mathbf{w}_{t,k+j+1}) \Delta t - \varepsilon, \\ &\quad \forall j \in [0, N_{p,OL90\%} - 1] \end{aligned} \quad (29)$$

$$\begin{aligned} A_{eq,OL}[\Delta \mathbf{X}_t \quad \Delta \mathbf{U}_t]^T &= \mathbf{b}_{eq,OL} \\ &\text{(including } \mathbf{x}_{t,N_{p,OL}} = \mathbf{x}_r). \end{aligned} \quad (30)$$

$X_{OL}$  and  $U_{OL}$  from (26) are the allowable sets for states and control (compact subsets of  $\mathbb{R}^{n(N_{p,OL}+1)}$  and  $\mathbb{R}^{mN_{u,OL}}$  respectively). Regarding the rest of constraints, (25) contains linear inequality constraints (27) (defined by  $A_{ineq,OL}$  and  $\mathbf{b}_{ineq,OL}$ ), for satisfying *MR* and actuators sectional-velocity bounds, as well as for defining a monotonically increasing pressure build-up. Nonlinear dynamic constraints (28) and (29) are not defined as strict equality constraints, but are treated as inequalities with a small margin  $\varepsilon = 10^{-2} \times \mathbf{1}_{n \times 1}$  (non-dimensional) so as to simplify the computation of a feasible solution. Linear dynamics (24), initial conditions, and end-state reaching are considered in the equality constraints (30) (defined by  $A_{eq,OL}$  and  $\mathbf{b}_{eq,OL}$ ). The interior-point optimisation software *IPOPT* [39] is used to solve (25)-(30). Due to the inclusion of nonlinear constraints in this quadratic-cost optimisation problem, the solution might not be global. Even so, reference values are attained while respecting constraints. Computation times in *MATLAB*® are about 1 minute in a standard computer. Graphical results concerning control-inputs trajectories can be found in [24].

## B. MPC controllers design

After the analysis of the state-space models from Section III, and having reviewed alternative robust control methods (such as  $H_\infty$  [17], [40], Linear Parameter-Varying (LPV) systems [41] or Lyapunov-based design [42]) the main selected control method in this article has been MPC. This approach, not used in LPRE to date, predicts the future system behaviour along a horizon, and optimises control inputs according to a cost function usually related to a reference trajectory or to an end state. It has already been used in other aerospace applications, such as in fighter aircraft manoeuvring [43] and in rocket landing guidance [44]. Indeed, the complexity of the models in this article, being highly nonlinear and non-affine in control; the relatively high number of states to be controlled by few inputs and the need for a strict verification of hard constraints during transients motivated the selection of this approach. Moreover, it can be extended to include robustness considerations. The versatility of the controller with

respect to different engines and cycles was also a design goal. The MPC approach described in this section is easy to adapt to distinct engines, provided that a similar modelling approach (as in Section II) is carried out. The availability of symbolic nonlinear and linearised models describing the whole operating span of LPREs is a valuable asset which can be exploited with this model-based control approach.

The start point for the on-line MPC schemes designed in this article is the generic continuous-time nominal MPC formulation [19]:

$$\begin{aligned} \min_{\mathbf{u}} \quad & J(\mathbf{x}(t), \mathbf{u}(t)) \\ \text{s.t.} \quad & \mathbf{x} \in X \quad \forall t \in [t, t + T_p] \\ & \mathbf{u} \in U \quad \forall t \in [t, t + T_u] \\ & \dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)) \quad \forall t \in [t, t + T_p], \end{aligned} \quad (31)$$

where  $J$  is an integral quadratic cost function over time,  $X$  and  $U$  are allowable sets for states and inputs, and  $T_p$  and  $T_u$  are the continuous prediction and control horizons respectively.

In this article, different controllers are proposed according to the scenario that the engine is facing. As explained before, one of the goals of this work is to control the continuous sub-phases of start-up transients in CL. Apart from the start-up, operating-point changes or throttling scenarios (fully continuous) are also targeted. The tracking goal is different whether throttling or start-up is to be controlled. Preliminary versions of these controllers, with differences in terms of robustness considerations and optimal tuning, were presented in [22]–[24].

1) *Throttling via end-state-tracking MPC (E.MPC)*: An operating-point change is performed here by tracking a new desired end equilibrium  $(\mathbf{x}_r, \mathbf{u}_r)$ , without specifying a reference trajectory and respecting constraints on  $\mathbf{x}$  and  $\mathbf{u}$ . There is a special focus on having a small tracking error in  $p_{CC}$  and in the three different  $MR$ , at the CC, GG and pump inlets, all represented in  $\mathbf{x}_z$ .

a) *Dynamics*: the dynamic model used in this state-feedback MPC controller is considered as a linearisation of the continuous-time nonlinear  $f_s$  (6)–(17) about the previously computed  $(\mathbf{x}_r, \mathbf{u}_r)$  from (20). The MPC problem is initially considered in continuous time (31). However, in order to solve it numerically, it is discretised via zero-order hold at the same  $\Delta t$  stated before, 10ms:

$$\Delta \mathbf{x}_{k+1} = A_d(\mathbf{x}_r, \mathbf{u}_r) \Delta \mathbf{x}_k + B_d(\mathbf{x}_r, \mathbf{u}_r) \Delta \mathbf{u}_k. \quad (32)$$

Nonlinear models have also been tested for determining dynamic relationships. However, the on-line resolution of the optimisation problem defined later in this section was not practical, requiring longer computational times ( $\sim \times 10$ ) and usually leading to unfeasible and sub-optimal solutions. Therefore, linear dynamics has been implemented. Since the exogenous input  $\mathbf{w}_t$  is zero at final equilibrium points, it has been omitted in (32). Besides, internal-actuators characteristics are directly considered in constraints.

b) *Quasi-infinite horizon (QIH)*: the approach here is partially based on the QIH approach by [45], because it presents proofs for CL stability and feasibility of MPC by incorporating the notion of a terminal region  $\Omega$ . In [45], a

terminal-state nonlinear inequality constraint and a penalty term are defined to enforce those features as Lyapunov arguments. The related  $P \in \mathbb{R}^{n \times n}$  matrix of the Lyapunov function  $V(\mathbf{x}) = \Delta \mathbf{x}^T P \Delta \mathbf{x}$  can be computed via a continuous-time Lyapunov equation. The terminal-region constraint is expressed as  $V(\mathbf{x}_{k+N_p}) \leq \alpha_P$ , where  $\alpha_P$  is iteratively calculated as indicated in the aforementioned reference.

c) *Integral action*: an integral action is also included to enforce a more precise tracking on  $\mathbf{x}_z$ , based on [46]. Indeed, the CL performance of model-based control is highly dependent on modelling errors and disturbances. One way to mitigate it is augmenting the decision variables with states representing tracking-error integration, as in [46]. The additional integral decision variables are denoted here by  $\mathbf{z}$  and present a corresponding weight matrix  $S \in \mathbb{R}^{n_z \times n_z}$  in the cost, whose diagonal is  $[1, 0.1, 0.1, 0.1, 0.1]$ . The dynamics of these  $\mathbf{z}$  variables is defined in constraints (41), where the constant gain matrix  $K_{I,end} \in \mathbb{R}^{n_z \times n}$  determines the velocity of their response. If this gain is set too large, unwanted oscillations in the related states can appear.

d) *Cost definition*: thus, the discretised MPC cost  $J_{end}$  is defined in a first step as:

$$\begin{aligned} J_{end}(\mathbf{X}, \mathbf{U}, \mathbf{Z}) = & \left( \sum_{j=0}^{N_p-1} \Delta \mathbf{x}_{k+j}^T Q_{end} \Delta \mathbf{x}_{k+j} \right. \\ & \left. + \sum_{j=0}^{N_u} \Delta \mathbf{u}_{k+j}^T R_{end} \Delta \mathbf{u}_{k+j} + \sum_{j=01}^{N_p-1} \mathbf{z}_{k+j}^T S \mathbf{z}_{k+j} \right) \Delta t \\ & + V(\mathbf{x}_{k+N_p}), \end{aligned} \quad (33)$$

which consists in the traditional quadratic cost on states and controls plus the added integral and terminal costs, with a prediction horizon  $N_p = 10$  steps ( $T_p = 0.1s$ ) and a control horizon  $N_u = 5$  ( $T_u = 0.05s$ ). Implicitly, the last control  $\mathbf{u}_{k+N_u}$  is used for  $j \geq N_u$ . Further extensions of these horizons did not improve the solutions in terms of tracking or constraints satisfaction.  $Q_{end}$  and  $R_{end}$  are positive-definite symmetric weighting matrices  $Q_{end} \in \mathbb{R}^{n \times n}$ ,  $R_{end} \in \mathbb{R}^{m \times m}$ , whose diagonals have been computed off-line via Kriging-based black-box optimisation as in [35]. An analogous procedure to the search of a worst-case scenario from Section III has been employed here. In this optimal weight selection, a Kriging model is built upon the following input-output data. Inputs consist in the diagonal terms of  $Q_{end}$  and  $R_{end}$  and the chosen output is the simulated performance criterion (18). The required simulations to quantify this criterion are executed on the whole CL during the nominal transient. An EGO scheme based on EI is then used on the resulting Kriging model to minimise that criterion by varying the inputs. In order to obtain this optimal tuning on the case studies considered (Section V), about 500 iterations, implying around 60 hours of calculation, are required.

e) *Robustness considerations*: moreover, some robust considerations have been implemented. The minimisation of the previous  $J_{end}$  (33) under constraints is not intrinsically robust. Indeed, robustness against parameters and initial conditions variations, perturbations and modelling error is very

important in this application. Robust MPC approaches generally make use of the *minimax* optimisation, which minimises the worst-case scenario of endogenous or exogenous perturbations. Other approaches are the tube-based MPC [47] or explicit robust MPC [48]. The robust *minimax* approach has been selected due to the elevated complexity of building invariant sets along trajectories (tube-based) or off-line state partitions (explicit MPC) seeing the highly multivariable nature of this system. However, solving generic *minimax* for compact uncertainty sets  $\Delta_c$  is generally not tractable on-line [49], and especially too costly in this application. Hence, a finite set of uncertain scenarios has been selected, within the so-called scenario-based MPC [50]. In addition, the equivalent epigraph formulation based on [49] has been implemented so as to avoid the maximisation. Concretely, this formulation allows to shift the robustness considerations into the list of constraints and hence pose the problem as a minimisation of a scalar  $\gamma \in \mathbb{R}^+$ , which is equivalent to performing scenario-based *minimax*. In this approach, that  $\gamma$  constrains the  $J_{end}$  (33) of the nominal non-robust problem evaluated at several perturbed states propagations  $\mathbf{X}_i$ :

$$\begin{aligned} \mathbf{X}_i &= [\mathbf{x}_{i,k}, \dots, \mathbf{x}_{i,k+j}, \dots, \mathbf{x}_{i,k+N_p}]^T, \quad i \in I, \\ \Delta \mathbf{x}_{i,k+1} &= A_d(\mathbf{x}_r, \mathbf{u}_r, \Delta_{i,k}) \Delta \mathbf{x}_{i,k} + B_d(\mathbf{x}_r, \mathbf{u}_r, \Delta_{i,k}) \Delta \mathbf{u}_k, \\ & \quad i \in I, \quad (34) \end{aligned}$$

where  $\Delta_{i,k}$  are certain selected internal parameter variations belonging to the non-compact  $\Delta_I = \{\Delta_{i,k}, i \in I\}$ .  $I$  is a finite set, which serves to index the considered perturbed cases. The series of  $\Delta_i$ , where  $I = \{1, 2, 3, 4\}$ , are taken constant for all  $k$  and  $j$  and represent the following parameter-variation cases, selected according to the sensitivity and worst-case analysis from Section III:

- $i = 1$ : nominal parameters.
- $i = 2$ : the most probably varying parameters are increased by 10%: interdependent tanks pressures  $p_{PI}$  and temperatures  $T_{PI}$ .
- $i = 3$ : the most probably varying parameters are decreased by 10%: interdependent  $p_{PI}$  and  $T_{PI}$ .
- $i = 4$ : worst-case variations combination defined in Section III. This off-line estimation of an extreme dynamics-perturbing scenario would correspond to the compact uncertainty set  $\Delta_c$  (related to Table I), for which a costly *minimax* would be necessary.

f) *Optimisation problem*: the resulting NLP is convex, which is highly important in order to compute real-time solutions in aerospace systems, as pointed out in [51]. This smooth convex NLP, in which decision variables are extended

to consider all  $\mathbf{X}_i$ , is:

$$\min_{\mathbf{X}_i, \mathbf{U}, \mathbf{Z}_i, \gamma} \quad \gamma \quad (35)$$

$$\text{s.t.} \quad J_{end}(\mathbf{X}_i, \mathbf{U}, \mathbf{Z}_i) \leq \gamma \quad \forall i \in I \quad (36)$$

$$\mathbf{X}_i \in X, \quad \mathbf{U} \in U \quad \forall i \in I \quad (37)$$

$$A_{ineq}[\mathbf{X}_i \quad \mathbf{U}]^T \leq \mathbf{b}_{ineq} \quad \forall i \in I \quad (38)$$

$$A_{i,eq,end}[\mathbf{X}_i \quad \mathbf{U}]^T = \mathbf{b}_{i,eq,end} \quad \forall i \in I \quad (39)$$

$$\Delta \mathbf{x}_{i,k+N_p}^T P_i \Delta \mathbf{x}_{i,k+N_p} \leq \alpha_P \quad \forall i \in I \quad (40)$$

$$\begin{aligned} \mathbf{z}_{i,k+j+1} &= \mathbf{z}_{i,k+j} + \Delta t K_{I,end} \Delta \mathbf{x}_{z,i,k+j} \\ & \quad \forall i \in I, j \in [0, N_p - 1]. \quad (41) \end{aligned}$$

$X$  and  $U$  in (37) are the allowable sets for states and control (compact subsets of  $\mathbb{R}^{n(N_p+1)}$  and  $\mathbb{R}^{mN_u}$  respectively). The set  $U$  for the first control  $\mathbf{u}_{MPC} \equiv \mathbf{u}_k$ , the one which is really applied to the plant, is specially bounded to comply with actuators capacity [52]:

$$\mathbf{u}_{MPC} \in [\max(\underline{U}, \mathbf{u}_0 - \dot{\mathbf{u}}_{max} \Delta t), \min(\overline{U}, \mathbf{u}_0 + \dot{\mathbf{u}}_{max} \Delta t)], \quad (42)$$

where  $\mathbf{u}_0$  is the previous-step control (warm start is performed by feed-backing the entire  $U$ ),  $\dot{\mathbf{u}}_{max}$  is the maximum sectional velocity of valves and  $\underline{U}$  and  $\overline{U}$  are the inferior and superior bounds of  $U$ . Thanks to saving the full  $U_0$  computed in the previous step, from which only the first control is transmitted to the plant, an initial guess for the whole  $\mathbf{X}_i$  can be made by propagating dynamics (34) from  $\mathbf{x}_k \equiv \mathbf{x}_0$ . The initial value of  $\gamma$  is set large enough ( $10^5$ ).

Regarding the rest of constraints, (35) contains equality constraints (39) (defined by  $A_{i,eq,end}$  and  $\mathbf{b}_{i,eq,end}$ ) for determining initial conditions and the different linear dynamics (34) along the whole horizon. Linear inequality constraints (38) (defined by  $A_{ineq}$  and  $\mathbf{b}_{ineq}$ ) are set for complying with  $MR$  and actuators sectional-velocity limits for all  $\mathbf{X}_i$ . Indeed, the vital mixture ratios limits establish linear inequalities between states; and actuators constraints are defined similarly to (42). The difference in control inputs between two time steps must not exceed  $\pm \dot{\mathbf{u}}_{max} \Delta t$ .

Inequalities (40) represent the nonlinear terminal-region constraints [45], where a different  $P_i$  is used according to each perturbed scenario  $i$ . Imposing these constraints helps the algorithm to find the optimum more quickly, enhancing feasibility. The last constraints (41) correspond to the integrator dynamics [46], where the  $K_{I,end}$  matrix has been computed off-line in the same manner as  $Q_{end}$  and  $R_{end}$ . Performance indicators are very sensitive to these gains.

It is important to emphasise the fact that the resulting  $U$  obtained in (35)-(41) has been confronted to all these scenarios and that all propagated perturbed states must comply with all constraints, thereby improving the robustness of the controller. This approach with equality constraints within an uncertain problem is only valid because of the finite choice of  $\Delta_{i,k}$ .

2) *Start-up via trajectory-tracking MPC (TMPC)*: If the start-up transient is to be executed, another approach is proposed, in which the set of planned trajectories is tracked, and not only a final point. Trajectory tracking is more convenient in this case where the system evolves in a highly nonlinear way and where trajectories can be pre-computed without tight

computational-time limits. In a quick throttling request, there might not be enough time to generate adequate trajectories (around 1min. in *MATLAB*), but only an end reference (about 3s).

The main MPC structure hitherto explained is reused in this case, with certain differences so as to track the pre-generated trajectories  $(\mathbf{X}_t, \mathbf{U}_t)$  from (25). The main control goals, including the verification of constraints, remain the same. However, the target of the controller is to drive  $\mathbf{X}$  and  $\mathbf{U}$  to  $\mathbf{X}_t$  and  $\mathbf{U}_t$  respectively. Dynamics in this controller is predicted in a linearised way for tractability reasons. Indeed, the discrete-time matrices  $A_d$  and  $B_d$  are evaluated about each step in the trajectory, improving the prediction representativeness with respect to the use of single end-state matrices in (34):

$$\begin{aligned} \Delta \mathbf{x}_{i,k+1} &= A_d(\mathbf{x}_{t,k}, \mathbf{u}_{t,k}, \mathbf{w}_{t,k}, \Delta_{i,k}) \Delta \mathbf{x}_{i,k} \\ &+ B_d(\mathbf{x}_{t,k}, \mathbf{u}_{t,k}, \mathbf{w}_{t,k}, \Delta_{i,k}) \Delta \mathbf{u}_k, \quad i \in I. \end{aligned} \quad (43)$$

The matrix  $A_d$  is always stable along the trajectory. Therefore, the discretised cost  $J_{traj}$  to be minimised in nominal MPC is defined as:

$$\begin{aligned} J_{traj}(\mathbf{X}, \mathbf{U}, \mathbf{Z}) &= \\ &\left( \sum_{j=0}^{N_p-1} (\mathbf{x}_{k+j} - \mathbf{x}_{t,k+j})^T Q_{traj} (\mathbf{x}_{k+j} - \mathbf{x}_{t,k+j}) \right. \\ &+ \sum_{j=0}^{N_u} (\mathbf{u}_{k+j} - \mathbf{u}_{t,k+j})^T R_{traj} (\mathbf{u}_{k+j} - \mathbf{u}_{t,k+j}) \\ &\quad \left. + \sum_{j=0}^{N_p-1} \mathbf{z}_{k+j}^T S \mathbf{z}_{k+j} \right) \Delta t + \\ &(\mathbf{x}_{k+N_p} - \mathbf{x}_{t,k+N_p})^T P (\mathbf{x}_{k+N_p} - \mathbf{x}_{t,k+N_p}). \end{aligned} \quad (44)$$

The same  $N_p = 10$  and  $N_u = 5$  are used.  $Q_{traj}$  and  $R_{traj}$  are positive-definite symmetric weighting matrices  $Q_{traj} \in \mathbb{R}^{n \times n}$ ,  $R_{traj} \in \mathbb{R}^{m \times m}$ , whose diagonals have also been tuned off-line via Kriging-based black-box optimisation as in [35], leading to values different to E.MPC. This controller also incorporates the QIH notions from [45], with some differences concerning the linearised dynamics. The Lyapunov equation is modified and solved every  $k$  for the continuous-time matrices evaluated at the end of the on-line horizons  $(N_p, N_u)$ , which slide with time, and for every perturbed case  $i \in I$ . The same  $\alpha_P$  can be obtained for all cases. This promotes CL stability and feasibility in all scenarios. Robustness is treated likewise to E.MPC. Thus, the minimisation problem for trajectory tracking

accounting for all  $\mathbf{X}_i$  is:

$$\min_{\mathbf{X}_i, \mathbf{U}, \mathbf{Z}_i, \gamma} \gamma \quad (45)$$

$$\text{s.t. } J_{traj}(\mathbf{X}_i, \mathbf{U}, \mathbf{Z}_i) \leq \gamma \quad \forall i \in I \quad (46)$$

$$\mathbf{X}_i \in X, \quad \mathbf{U} \in U \quad \forall i \in I \quad (47)$$

$$A_{ineq}[\mathbf{X}_i \quad \mathbf{U}]^T \leq \mathbf{b}_{ineq} \quad \forall i \in I \quad (48)$$

$$A_{i,eq,traj}[\mathbf{X}_i \quad \mathbf{U}]^T = \mathbf{b}_{i,eq,traj} \quad \forall i \in I \quad (49)$$

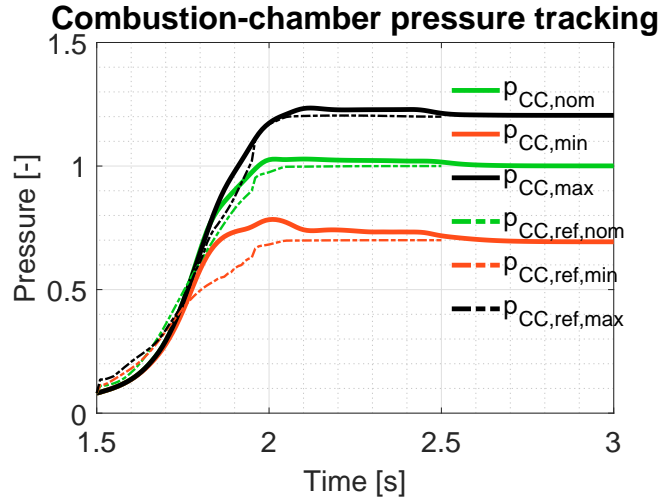
$$\begin{aligned} (\mathbf{x}_{i,k+N_p} - \mathbf{x}_{t,k+N_p})^T P_i (\mathbf{x}_{i,k+N_p} - \mathbf{x}_{t,k+N_p}) \\ \leq \alpha_P \quad \forall i \in I \end{aligned} \quad (50)$$

$$\begin{aligned} \mathbf{z}_{i,k+j+1} &= \mathbf{z}_{i,k+j} \\ &+ \Delta t K_{I,traj} (\mathbf{x}_{z,i,k+j} - \mathbf{x}_{z,t,k+j}) \\ &\quad \forall i \in I, j \in [0, N_p - 1]. \end{aligned} \quad (51)$$

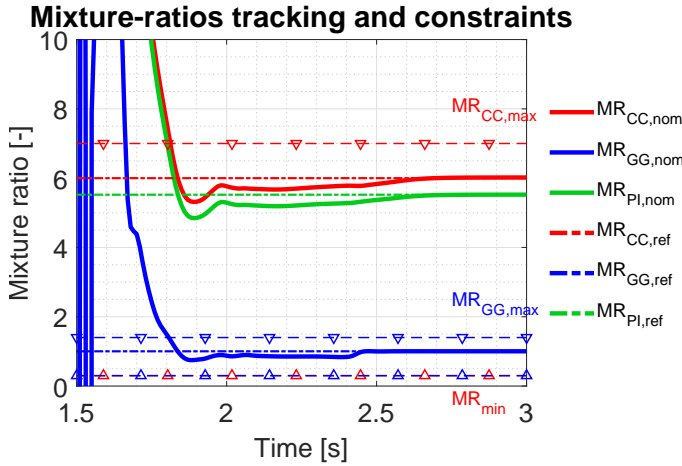
Apart from the already defined bounds  $X$  and  $U$ , in (45) there are also equality constraints (49) (defined by  $A_{i,eq,traj}$  and  $\mathbf{b}_{i,eq,traj}$ ) for initial conditions (measured  $\mathbf{x}_k$ ) and the different linear dynamics along the whole horizon (43). Besides, the same linear inequality constraints (48) as in E.MPC (38) are applied to all  $\mathbf{X}_i$ . Integrator dynamics (51) is also analogous, but different values for  $K_{I,traj}$  have been computed off-line in the same manner as  $Q_{traj}$  and  $R_{traj}$ . In (50), a specific  $P_i$  is used in each perturbed case and MPC step since matrices are varying.

## V. RESULTS AND DISCUSSION

Simulations of the control loop of Fig. 2 are run in *Simulink*® at a maximum integration time step of  $10^{-5}$ s, as required by the developed engine simulator. However, the maximum computation frequency allowed to the controller is 100Hz, i.e., control orders are sent every 10ms. Integration is performed with a fixed step and *ode3* (Bogacki-Shampine scheme), seeing that higher-order and variable-step methods led to identical results despite the associated longer simulation times. The interior-point optimisation software *IPOPT* [39] has been used to solve the smooth convex NLP (35) and (45) within the *MATLAB*® environment. This tool has been selected instead of *MATLAB*® *fmincon* or other solvers within *YALMIP* [53] owing to its efficient interior-point conception, which intrinsically respects barriers on decision variables; and to its coding flexibility, allowing the introduction of user-defined gradients, Jacobians and Hessians in the calculation. This, together with the consideration of sparse matrices, alleviate the computational burden. The resulting computational times in *MATLAB*® are of the order of ten times longer than real time, which does not rule out a future real-machine implementation. The number of decision variables considered in the selected MPC design is  $length(I) \times (N_p + 1) \times n + (N_u + 1) \times m + length(I) \times N_p \times n_z + 1$  (for  $\gamma$ ) = 759. Apart from the individual bound constraints on those variables, the number of equality and inequality constraints combined is 1904. As explained throughout the article, two main continuous-control scenarios are simulated: the continuous sub-phase of the start-up transient as well as the throttling operation, which consists in a change of operating point.

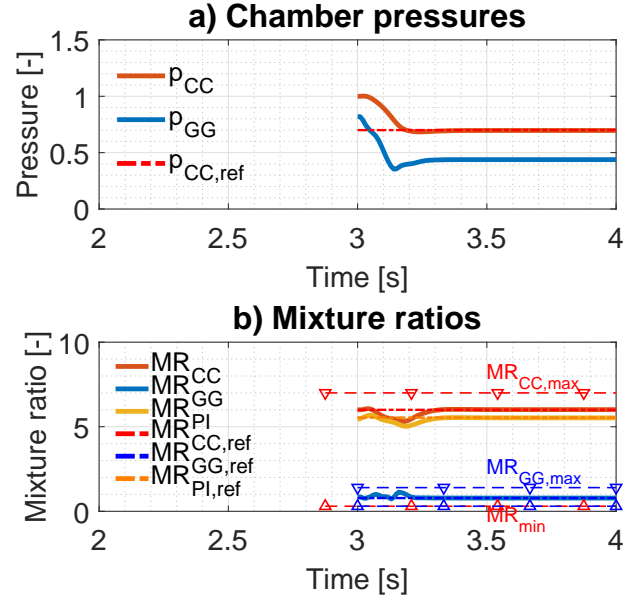


**Fig. 3:** T.MPC start-up tracking results in combustion-chamber pressure  $p_{CC}$  for  $p_{CC,r} = 1$  (nominal),  $p_{CC,r} = 0.7$  (minimum) and  $p_{CC,r} = 1.2$  (maximum)



**Fig. 4:** T.MPC start-up tracking results in mixture ratios  $MR$  for  $p_{CC,r} = 1$  (nominal)

*a) Start-up transient:* the control of the start-up transient is performed via T.MPC, solving (45) during the time frame between  $t_{cont} = 1.5$ s and 3s after the start command. That interval corresponds to the time window in which continuous control is possible in engine start-up transients. Fig. 3 illustrates the results of  $p_{CC}$  tracking for three main operating points in *Vulcain 1* design envelope (70%, 100% and 120%). The reference mixture ratios remain the same  $MR_{CC,r} = 6$ ,  $MR_{GG,r} = [0.8, 1.1]$  and  $MR_{PI,r} = 5.25$ .  $MR$  tracking for the 100% case is shown in Fig. 4. These ratios naturally start in these transients from values very far from the allowable area, due to the low initial mass flows that hinder the definition of quotients. Indeed, chambers are not physically fully ignited during the first instants (even if igniters are active); hence,  $MR$  are not really representative there. Tracking is accomplished with sufficient accuracy in  $p_{CC}$  for all cases (under 0.064% in nominal thrust and under 0.94% in off-nominal) and with little error in  $MR$  (under 0.32% in nominal, under 3.2%



**Fig. 5:** E.MPC throttling tracking results in pressures and mixture ratios for throttle-down to  $p_{CC,r} = 0.7$  from  $p_{CC,r} = 1$  (100% to 70%)

in off-nominal). Simultaneously, constraints are respected up from the time when it is considered feasible and acceptable to respect them in practice ( $t \approx 1.9$ s). The slight overshoot present until around  $t = 2.5$ s, before achieving the final tracking, is generated by the nonlinear exogenous influence of the GG-starter input mass flow. It is more pronounced in the minimum case since the relative influence of the starter is more elevated. All these performance indicators are explicitly listed in Table II. The controller is able to achieve analogous tracking performance after altered initial conditions (independent, uniformly distributed random  $\pm 50\%$ ) coming from the sequential transient.

*b) Throttling transient:* a throttle-down operation from nominal to minimum thrust (100% to 70%) has been selected as a representative case. This is a scenario that reusable LPREs must face during their operation. Fig. 5 depicts the pressure and  $MR$  tracking results via the E.MPC controller (35). The engine is already active at its nominal state at  $t = 3$ s and a step-wise order to track 70% of thrust is sent, with the corresponding pre-calculated  $(x_r, u_r)$ . It can be seen that the operation shift is safely performed (no constraints violated) within 0.34s. Precise performance indicators are summarised in Table II. Other throttling scenarios such as an increase to 120% are also satisfactorily accomplished.

#### A. Comparison with OL and PID/LQR

Table II summarises the comparison between these MPC tracking proposals, conventional PID or LQR controllers (the most adapted to each case) and OL applied to the selected scenarios, in terms of some performance indicators: settling time to 99% of  $p_{CC,r}$ , the start of constraints verification, overshoot in  $p_{CC}$ , static errors in  $p_{CC}$ ,  $MR_{CC}$ ,  $MR_{GG}$  and

$MR_{PI}$ , and the overall-performance criterion  $J_{perf}$  (18). The nominal OL is engine's original command, which is precisely tuned for the standard case, as traditionally done in flight-ready engines. Conversely, the minimum and maximum OL constant commands have been computed by means of the preprocessor from Section IV-A. These OL commands already represent relevant solutions for the off-nominal steady-state multivariable control. The improvement of CL with respect to OL can be observed in the reduction in  $J_{perf}$  in all cases (minimum value is the best case), even though some individual indicators deteriorate during the start-up transient. Indeed, the real gain of this CL MPC control appears for operating points different from the nominal, where multivariable tracking was difficult to achieve with high performance while respecting constraints during the transient. Furthermore, valves openings are not saturated with this predictive approach, even for the maximum thrust case. The enhancement in the down-throttling operation is also noticeable in all indicators. Even though conventional PID and LQR controllers stand out in certain indicators, their overall performance is inferior to MPC. Moreover, these techniques do not guarantee constraints satisfaction. Especially in the maximum thrust case, rotational speeds limits are violated with those controllers, as shown in [22].

### B. Robustness analysis

In Section IV-B, the different perturbed cases have been listed. The system has to be robust to those scenarios, and hence they have been simulated by intentionally altering the involved parameters in the simulator. Furthermore, other sources of perturbation, not explicitly considered in the control algorithm, have been simultaneously introduced for testing robustness. These are namely the effects of external, additive perturbations in the state and the possible impact that a state estimator would represent, not considered in previous works by the authors [22]–[24]. For the former, a band-limited white-noise block has been added to the simulator differential equations before integration, with a power spectral density corresponding to perturbation spans of  $\pm 10\%$  of the nominal state. For the latter, a unitary delay (of 10ms) and an estimation error source (multiplication by a uniform random number) have been inserted in the measurement fed back to the controller, remaining within  $\pm 1\%$  of the true values. The resulting performance indicators obtained in the different perturbed scenarios for attaining a nominal-thrust start-up and a throttle-down to 70% are shown in Table III.

It is observable that results point to considerably greater robustness to parameters variations with respect to OL in both transient phases simulated. Constraints are satisfied when required and overshoot and tracking errors generally diminish with respect to OL control, especially the mixture-ratios errors. The performance criterion  $J_{perf}$  diminishes in all scenarios. The impact of a possible estimator and external perturbations are also well mitigated. Hence it can be concluded that these synthesised MPC controllers fulfil the control goals defined in Section IV.

## VI. CONCLUSIONS

The evolving design of reusable launchers and their associated liquid-propellant rocket engines (LPREs) imposes more demanding robustness requirements on control systems. Transient phases like start-up have traditionally been executed in OL, which does not guarantee robustness to possible parameter variations in the engine. In this article, approaches to control the continuous transient phases of GG-cycle LPREs have been proposed. The main objective is to track combustion-chamber pressure and mixture ratios via valve-opening adjustment while satisfying engine constraints. Engines transient behaviour was first captured in 0-D simulators via thermo-fluid-dynamic and mechanical conservation equations based on the available parameters. Secondly, these simulators were re-expressed in the form of state-space models. The most influential parameters and the worst-case scenario of parametric variations have been identified via Sobol analysis and Kriging-based EGO respectively. Based on these models, strategies relying on Model Predictive Control (MPC) have then been developed for the different transient scenarios that these engines face. An end-state-tracking algorithm for throttling operations, as well as a trajectory-tracking one for the start-up transient, have been synthesised. These linearised MPC controllers with integral action and quasi-infinite horizon are able to track the reference signals with acceptable accuracy within the design envelope and constraints are respected when necessary in simulations. Robustness to the set of more influential parameters in the model is also considered (scenario-based *minimax* in epigraph formulation) and demonstrated. Furthermore, neither the simulated impact of a possible future state observer nor external perturbations deteriorate performance to a relevant extent.

Future work will involve the development of state estimators, since mass flows might not be measurable in real engines. Discrete inputs to the system, involved in the start-up sequence, are also to be managed in order to robustify that sequence, as in [54]. A first proposal to optimise the planning of events timing was published in [23], which still needs to be validated in more extensive simulations. Further verification steps will consist in hardware-in-the-loop simulations and test-bench operation.

### ACKNOWLEDGEMENT

The authors thank Émilien Flayac (ONERA) and Mircea Lazar (TU Eindhoven) for their valuable advice throughout the development of this work.

### REFERENCES

- [1] P. Baiocco and C. Bonnal, "Technology demonstration for reusable launchers," *Acta Astronautica*, vol. 120, pp. 43–58, Mar. 2016.
- [2] S. Le Gonidec, "An overview of connections between scientific automatic topics and their applications in the propulsive systems," in *Journal of Physics: Conference Series: 13th European Workshop on Advanced Control and Diagnosis (ACD 2016)*, ser. 783/011001. Lille, France: IOP Publishing, Jan. 2017.
- [3] D. K. Huzel and D. H. Huang, "Chapter 7: Design of Rocket-Engine Control and Condition-Monitoring Systems," in *Modern Engineering for Design of Liquid-Propellant Rocket Engines*. AIAA Progress in Astronautics and Aeronautics Richard Seebass, 1992.

**TABLE II:** Performance-indicators comparison between MPC, PID/LQR controllers and OL for start-up and throttling transients

Operating point	Nominal start-up			Minimum start-up			Maximum start-up			Throttle-down to 70%		
	OL	LQR	T.MPC	OL	PID	T.MPC	OL	LQR	T.MPC	OL	LQR	E.MPC
Settling time ( $p_{CC,r} \pm 1\%$ ) [s]	2.76	<b>2.54</b>	<b>2.54</b>	-	2.72	<b>2.57</b>	2.7	-	<b>2.51</b>	-	<b>0.31</b>	0.34
Overshoot (% in $p_{CC}$ )	6.29	9.98	<b>2.84</b>	15.18	70.77	<b>12.01</b>	3.43	2.98	<b>2.9</b>	3.19	<b>0.026</b>	2.29
Constraints verified from [s]	<b>1.81</b>	2.25	<b>1.81</b>	1.83	<b>1.78</b>	1.79	<b>1.77</b>	1.98	1.84	<b>0</b>	0.06	<b>0</b>
$p_{CC}$ static error (%)	0.21	<b>0.044</b>	0.064	2.67	<b>0.003</b>	0.94	0.45	1.14	<b>0.43</b>	3.19	<b>0.026</b>	0.498
$MR_{CC}$ static error (%)	<b>0.18</b>	1.2	0.32	<b>2.66</b>	2.7	3.2	3.25	2.38	<b>0.8</b>	2.72	4.63	<b>0.18</b>
$MR_{GG}$ static error (%)	1.41	0.65	<b>0.069</b>	1.35	<b>0.075</b>	0.23	1.28	0.46	<b>0.31</b>	1.37	1.09	<b>0.05</b>
$MR_{PI}$ static error (%)	1.41	0.81	<b>0.022</b>	2.92	3.06	<b>2.62</b>	3.48	2.63	<b>1.04</b>	2.98	3.99	<b>0.29</b>
$J_{perf}$ overall performance	4.16	5.9	<b>1.27</b>	11.45	9.63	<b>8.43</b>	9.54	7.92	<b>3.49</b>	10.29	9.74	<b>1.04</b>

**TABLE III:** Performance-indicators comparison between MPC and OL in perturbed scenarios for start-up and throttling transients

Transient Perturbed cases	Start-up to 100%								Throttle-down to 70%			
	Nominal conditions		Case 2		Case 3		Case 4 (worst case)		Nominal conditions		Case 4 (worst case)	
Indicator	OL	T.MPC	OL	T.MPC	OL	T.MPC	OL	T.MPC	OL	E.MPC	OL	E.MPC
Settling time ( $p_{CC,r} \pm 1\%$ ) [s]	2.76	<b>2.54</b>	2.84	<b>2.52</b>	2.8	<b>2.52</b>	-	<b>2.51</b>	-	<b>0.34</b>	-	-
Overshoot (% in $p_{CC}$ )	6.29	<b>2.84</b>	6.5	<b>3.31</b>	6.18	<b>2.95</b>	4.39	<b>2.94</b>	3.19	<b>2.29</b>	5.86	<b>5.2</b>
Constraints verified from [s]	<b>1.81</b>	<b>1.81</b>	<b>1.72</b>	1.85	<b>1.72</b>	1.85	<b>1.8</b>	1.84	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>
$p_{CC}$ static error (%)	0.21	<b>0.064</b>	0.78	<b>0.54</b>	0.47	<b>0.35</b>	1.34	<b>0.36</b>	3.19	<b>0.498</b>	5.86	<b>1.07</b>
$MR_{CC}$ static error (%)	<b>0.18</b>	0.32	<b>0.45</b>	0.68	<b>0.05</b>	0.33	1.18	<b>0.39</b>	2.72	<b>0.18</b>	4.73	<b>0.53</b>
$MR_{GG}$ static error (%)	1.41	<b>0.069</b>	1.39	<b>0.44</b>	1.42	<b>0.39</b>	1.62	<b>1.3</b>	1.37	<b>0.05</b>	2.68	<b>1.14</b>
$MR_{PI}$ static error (%)	1.41	<b>0.022</b>	1.1	<b>0.38</b>	1.5	<b>0.033</b>	2.86	<b>0.097</b>	2.98	<b>0.29</b>	4.94	<b>1.03</b>
$J_{perf}$ overall performance	4.16	<b>1.27</b>	5.03	<b>3.51</b>	4.65	<b>2.6</b>	8.08	<b>3.45</b>	10.29	<b>1.04</b>	18.29	<b>3.82</b>

- [4] X. Dai and A. Ray, "Damage-Mitigating Control of a Reusable Rocket Engine: Part II-Formulation of an Optimal Policy," *Journal of Dynamic Systems, Measurement, and Control*, vol. 118, no. 3, pp. 409–415, Sep. 1996.
- [5] B. N. Kiforenko and A. M. Kharitonov, "Control of Thrust of Liquid Rocket Engines: Simulation and Optimization," *Journal of Automation and Information Sciences*, vol. 32, no. 8, pp. 47–63, 2000.
- [6] T. Kai, K. Niu, K. Obase, W. Sakai, Y. Fukuda, T. Hashimoto, M. Sato, S. Takada, T. Kimura, Y. Naruo, H. Ogawa, T. Yagishita, and T. Ito, "Engine Control System for the Main Engine of the Reusable Sounding Rocket," in *Proceedings of the International Astronautical Congress, IAC*, ser. IAC-15,C4,3,2,x28758, vol. 10. Jerusalem, Israel: IAF, 2015, pp. 7389–7394.
- [7] M. Klein, D. Hayoun, S. Le Gonidec, and S. Reichstadt, "Method and a circuit for regulating a rocket engine," SNECMA (ArianeGroup), Issy-les-Moulineaux, France, US 2017/0101963 A1, United States Patent Application Publication.
- [8] S. Le Gonidec and O. Faye, "Device for adjusting an operating variable of an engine," SNECMA (ArianeGroup), France, US9037380B2, United States Patent.
- [9] E. Nemeth, R. Anderson, J. Ols, and M. Olsasky, "Reusable rocket engine intelligent control system framework design, phase 2," Rockwell International, Canoga Park, California, Tech. Rep. NASA Contractor Report 187213, Sep. 1991.
- [10] E. W. Otto and R. A. Flage, "Control of combustion-chamber pressure and oxidant-fuel ratio for a regeneratively cooled hydrogen-fluorine rocket engine," NASA Lewis Research Center, Cleveland, USA, Tech. Rep. Technical note D-82, 1959.
- [11] H. Sunakawa, A. Kurosu, K. Okita, W. Sakai, S. Maeda, and A. Ogawara, "Automatic Thrust and Mixture Ratio Control of the LE-X," in *44th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, vol. 2008-4666. Hartford, USA. AIAA, 2008.
- [12] C. F. Lorenzo, A. Ray, and M. S. Holmes, "Nonlinear control of a reusable rocket engine for life extension," *Journal of Propulsion and Power*, vol. 17, no. 5, pp. 998–1004, 2001.
- [13] C. F. Lorenzo, W. C. Merrill, J. L. Musgrave, and A. Ray, "Controls Concepts for Next Generation Reusable Rocket Engines," in *American Control Conference*, vol. FP1-4:10. Seattle, USA. IEEE, 1995, pp. 3942–3950.
- [14] J. L. Musgrave, T. H. Guo, E. Wong, and A. Duyar, "Real-time accommodation of actuator faults on a reusable rocket engine," *IEEE transactions on control systems technology*, vol. 5, no. 1, pp. 100–109, 1996, - NASA.
- [15] F. Zheng, M. Cheng, and W. B. Gao, "Variable structure control of time-delay systems with a simulation study on stabilizing combustion in liquid propellant rocket motors," *Automatica*, vol. 31, no. 7, pp. 1031–1037, Jul. 1995.
- [16] A. Le Fur, C. Voisin, M. H. Guillard, and M. M. Verge, "Study of control laws of an expander-cycle rocket engine - Etude de lois de commande pour moteur a cycle expander," ArianeGroup, ENSAM, LAAM, Vernon, France, ArianeGroup internal report, 1997.
- [17] R. Saudemont and S. Le Gonidec, "Study of a robust control law based on  $H_\infty$  for the Vulcain rocket engine - Etude d'une commande robuste a base de commande  $H_\infty$  pour le moteur Vulcain," ArianeGroup, ESTACA, Vernon, France, ArianeGroup internal report, 2000.
- [18] S. Pérez-Roca, J. Marzat, H. Piet-Lahanier, N. Langlois, F. Farago, M. Galeotta, and S. Le Gonidec, "A survey of automatic control methods for liquid-propellant rocket engines," *Progress in Aerospace Sciences*, vol. 107, pp. 63–84, May 2019.
- [19] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Sokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, Jun. 2000.
- [20] J. Zhang, P. Yuan, and K.-S. Chin, "Model Predictive Control for the Flow Field in an Intermittent Transonic Wind Tunnel," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 1, pp. 324–338, Feb. 2018.
- [21] S. Pérez-Roca, N. Langlois, J. Marzat, H. Piet-Lahanier, M. Galeotta, F. Farago, and S. Le Gonidec, "Derivation and Analysis of a State-Space Model for Transient Control of Liquid-Propellant Rocket Engines," in *2018 9th International Conference on Mechanical and Aerospace Engineering (ICMAE)*, Budapest, Hungary, Jul. 2018, pp. 58–67.
- [22] S. Pérez-Roca, J. Marzat, E. Flayac, H. Piet-Lahanier, N. Langlois, F. Farago, M. Galeotta, and S. Le Gonidec, "An MPC Approach to Transient Control of Liquid-Propellant Rocket Engines," in *IFAC-PapersOnLine*, ser. 21st IFAC Symposium on Automatic Control in Aerospace ACA 2019, vol. 52-12, Cranfield, UK, Aug. 2019, pp. 268–273.

- [23] S. Pérez-Roca, J. Marzat, H. Piet-Lahanier, N. Langlois, F. Farago, M. Galeotta, and S. Le Gonidec, "Robust Transient Control of Reusable Liquid-Propellant Rocket Engines," in *70th International Astronautical Congress (IAC) (to appear)*. Washington D.C., USA. International Astronautical Federation (IAF), Oct. 2019.
- [24] —, "Trajectory planning and tracking via MPC for transient control of liquid-propellant rocket engines," in *15th European Workshop on Advanced Control and Diagnosis, ACD 2019 (to appear)*. Bologna, Italy. Nov. 2019.
- [25] A. Ray and X. Dai, "Damage-Mitigating Control of a Reusable Rocket Engine for High Performance and Extended Life," Lewis Research Center, Tech. Rep. NASA Contractor Report 4640, 1995.
- [26] X. Dai and A. Ray, "Damage-Mitigating Control of a Reusable Rocket Engine: Part I-Life Prediction of the Main Thrust Chamber Wall," *Journal of Dynamic Systems, Measurement, and Control*, vol. 118, no. 3, pp. 401–408, Sep. 1996.
- [27] M.-K. Wu and A. Ray, "Damage-Mitigating Control of Power Systems for Structural Durability and High Performance," *Journal of Engineering for Gas Turbines and Power*, vol. 117, no. 2, pp. 307–313, Apr. 1995.
- [28] J. A. Dillon, W. W. Weaver, R. D. Robinett, III, and D. G. Wilson, "Electro-Mechanical-Thermal Performance and Stability of Aircraft Energy Networks with Pulse Power Loads," *IEEE Transactions on Aerospace and Electronic Systems*, vol. Early access, pp. 1–1, 2019.
- [29] G. P. Sutton and O. Biblarz, *Rocket propulsion elements*, 7th ed., ser. Wiley-Interscience publication. New York, John Wiley and Sons, 2001.
- [30] S. Le Gonidec, "A method of controlling the pressure and a mixture ratio of a rocket engine, and corresponding device," France Patent FR3 039 859 B1, 2017, - Safran Aircraft Engines SAS (ArianeGroup), Institut National de la Propriété Industrielle (France).
- [31] C. C. Peck and A. P. Dhawan, "SSME parameter model input selection using genetic algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 1, pp. 199–212, Jan. 1996.
- [32] I. M. Sobol, "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates," *Mathematics and Computers in Simulation*, vol. 55, no. 1, pp. 271–280, Feb. 2001.
- [33] B. Iooss and P. Lemaître, "A Review on Global Sensitivity Analysis Methods," in *Uncertainty Management in Simulation-Optimization of Complex Systems: Algorithms and Applications*, ser. Operations Research/Computer Science Interfaces Series. Boston, MA, USA. Springer US, 2015, pp. 101–122.
- [34] J. Lefebvre, H. Roussel, E. Walter, D. Lecoq, and W. Tabbara, "Prediction from wrong models: the Kriging approach," *IEEE Antennas and Propagation Magazine*, vol. 38, no. 4, pp. 35–45, Aug. 1996.
- [35] J. Marzat, E. Walter, H. Piet-Lahanier, and F. Dammone, "Automatic tuning via Kriging-based optimization of methods for fault detection and isolation," in *2010 Conference on Control and Fault-Tolerant Systems (SysTol)*, Nice, France, Oct. 2010, pp. 505–510.
- [36] Y. Kuwata, T. Schouwenaars, A. Richards, and J. How, "Robust Constrained Receding Horizon Control for Trajectory Planning," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*. San Francisco, USA. American Institute of Aeronautics and Astronautics, Aug. 2005.
- [37] J. T. Betts, "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 193–207, 1998.
- [38] A. Belkadi, H. Abaunza, L. Ciarletta, P. Castillo, and D. Theilliol, "Design and Implementation of Distributed Path Planning Algorithm for a Fleet of UAVs," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 55, no. 6, pp. 2647–2657, Mar. 2019.
- [39] A. Waechter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, Mar. 2006.
- [40] W. S. Levine and R. T. Reichert, "An introduction to H/sub infinity / control system design," in *29th IEEE Conference on Decision and Control*, Dec. 1990, pp. 2966–2974 vol.6.
- [41] C. Briat, "Robust Control and Observation of LPV Time-Delay Systems," Ph.D. Thesis, Institut National Polytechnique de Grenoble - INPG, Nov. 2008.
- [42] H. K. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ, USA. Prentice Hall, 1996.
- [43] D. Simon, "Fighter Aircraft Maneuver Limiting Using MPC: Theory and Application", Linköping University Electronic Press, vol. 1881, 2017
- [44] J. Wang, N. Cui, and C. Wei, "Optimal rocket landing guidance using convex optimization and model predictive control," *Journal of Guidance, Control, and Dynamics*, American Institute of Aeronautics and Astronautics, vol. 42, no. 5, pp. 1078–1092, 2019
- [45] H. Chen and F. Allgöwer, "A Quasi-Infinite Horizon Nonlinear Model Predictive Control Scheme with Guaranteed Stability," *Automatica*, vol. 34, no. 10, pp. 1205–1217, 1998.
- [46] L. O. Santos, P. A. F. N. A. Afonso, J. A. A. M. Castro, N. M. C. Oliveira, and L. T. Biegler, "On-line implementation of nonlinear MPC: an experimental case study," *Control Engineering Practice*, vol. 9, no. 8, pp. 847–857, Aug. 2001.
- [47] D. Q. Mayne, S. V. Rakovic, R. Findeisen, and F. Allgöwer, "Robust output feedback model predictive control of constrained linear systems," *Automatica*, vol. 42, no. 7, pp. 1217–1222, Jul. 2006.
- [48] D. Muñoz de la Peña, A. Bemporad, and C. Filippi, "Robust explicit MPC based on approximate multi-parametric convex programming," in *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*, vol. 3, Dec. 2004, pp. 2491–2496 Vol.3.
- [49] J. Lofberg, *Minimax Approaches to Robust Model Predictive Control*, ser. Linköping Studies in Science and Technology Dissertations. Uni-Tryck, Linköping University Electronic Press, Apr. 2003, no. No. 812.
- [50] M. Maiworm, T. Baethge, and R. Findeisen, "Scenario-based Model Predictive Control: Recursive Feasibility and Stability," *IFAC-PapersOnLine*, vol. 48, no. 8, pp. 50–56, Jan. 2015.
- [51] Y. Mao, D. Dueri, M. Szmuk, and B. Açikmeşe, "Convexification and real-time optimization for MPC with aerospace applications," *Handbook of Model Predictive Control*, Springer, pp. 335–358, 2019
- [52] Y. Luo, A. Serrani, S. Yurkovich, D. B. Doman, and M. W. Oppenheimer, "Model predictive dynamic control allocation with actuator dynamics," in *Proceedings of the 2004 American Control Conference*, vol. 2, Boston, USA, 2004, pp. 1695–1700.
- [53] J. Lofberg, "YALMIP : a toolbox for modeling and optimization in MATLAB," in *2004 IEEE International Conference on Robotics and Automation (IEEE Cat. No.04CH37508)*. Taipei, Taiwan. IEEE, 2004, pp. 284–289.
- [54] J. Lunze and F. Lamnabhi-Lagarrigue, *Handbook of Hybrid Systems Control: Theory, Tools, Applications*. Cambridge University Press, Oct. 2009.



**Sergio Pérez-Roca** received the B.Sc in aerospace engineering from the Polytechnic University of València (UPV, Spain) in 2014, spending two semesters in the University of Stuttgart (Germany); and the M.Sc in aeronautical and space systems from ISAE-Supaéro Toulouse (France) in 2016. He has recently finished his PhD in automatic control from the University Paris-Saclay.

Since 2017 he is with the French Space Agency CNES and the French Aerospace Lab ONERA in the Paris area. His research focusses on space-propulsion systems design, including thermo-fluid-dynamic modelling and simulation, automatic control and estimation, orbital analysis and systems engineering, with a high appreciation of experimental proof-of-concept developments.



**Julien Marzat** graduated as an engineer from ENSEM (INPL Nancy) in 2008 and completed his PhD Thesis from University Paris-Saclay in 2011. He is currently a Research Scientist at ONERA. His research interests include guidance, control and fault diagnosis for autonomous robots and aerospace systems.



**H el ene Piet-Lahanier** graduated from SupA ero (Toulouse) and obtained her PhD in Physics and her Habilitation Diriger les Recherches (HDR), both from the University Paris XI, Orsay. She is currently Scientific Deputy of the department of Information Processing and Systems in ONERA. Her research interests include modelling under uncertainty, bounded error estimation, cooperative guidance, MPC design and event-triggered estimation. She is a member of the IFAC technical committee on Aerospace.



**Nicolas Langlois** received the PhD and HDR (Habilitation to Supervise Research) degrees in Automatic Control and signal processing from the University of Rouen Normandy, France, in 2001 and in 2008 respectively. In 2000, he joined the graduate school of electrical engineering ESIGELEC, Rouen, France. At present, he is the head in charge of skills acquisition through research of ESIGELEC where he currently teaches control systems and digital signal processing. He is also the Leader of the Automatic Control and Systems research team at the research

laboratory IRSEEM since 2008. His research interests include mainly predictive control and fault tolerant control.



**Marco Galeotta** graduated from Politecnico di Milano in 2003 with a master thesis on the metal agglomeration and the combustion mechanisms of Aluminium Nitrate based solid rocket propellants, he joined the CIRA (Italian Aerospace center) in 2004 as a test engineer for the transonic wind-tunnel facility. After a few years spent in the industry (OHB Italy) as a system engineer for the development of the ground facilities for the upcoming VEGA launcher, he joined Arianespace in 2010 as a production engineer and technical authority for the storable

liquid rocket stages of the European launch vehicle family (*Ariane 5-ES*, *VEGA* and *Soyuz* from Kourou). In 2017 he joined the CNES Technical sub-directorate (engine system department) as the responsible for the liquid rocket engines demonstrators and technological research activities from a system point of view. His main fields of interest are system modelling, engine testing, optimisation, HMS (Health Monitoring System) and control.



**Fran ois Farago** is a GNC engineer at the Launchers Directorate of the French space agency CNES. He graduated from Ecole Polytechnique in 2006 and completed his PhD in celestial mechanics at Observatoire de Paris in 2010. He works on the control of space launchers.



**Serge Le Gonidec** started his career in 1986 at the 'Soci et e Europ eenne de Propulsion' in a support team 'control, measurement and command system' on the engine test benches of the *Ariane* launcher. In the early 90s, the *Ariane 5* (A5) program gave him the opportunity to work on the test-bench control engineering (flow control, tank pressurisation, turbine speed, etc.). These new A5 test benches allowed introducing the digital control and its generalisation for the engine or components tested under a high safety level.

In 1994, he joined the team of the System department. His activities were mainly oriented on the future control and on the system stability analyses for the system developments.

After 1997, he has contributed to the engine design (e.g. *Vinci*) or evolution (e.g. *Vulcain 2*) deploying the linear Multi-variable Predicted Control. In the same time, in relation with ESA or CNES, he worked on R&T activities dedicated to the engine control and the Health Monitoring System.

These activities were a support to the future launchers preparation and paved the way to the reusable launchers, control and monitoring interaction/IVHM/AFTC, etc.

He is currently member of the General Inspection team.