

Evaluation of two strategies NMPC into HIL applied to the operation of an internal combustion engine

José Albeiro Valencia Chica¹  · Adalberto Gabriel Díaz Torres²

Received: 30 November 2016 / Accepted: 3 May 2017 / Published online: 18 May 2017
© Springer-Verlag France 2017

Abstract The hardware in the loop (HIL) platforms constitute an accelerated engineering design medium that subsequently reduces the risk of failure in the implementation process. This fact is enhanced when the designer is faced with extremely complex tasks and when design errors can lead to destructive and costly testing. The design of complex algorithms such as non-linear controllers applied to electronic control units of combustion engines, is a task of higher complexity level, and tests with incorrectly adjusted drivers can result in engine breakdown. This document reports the performance and effectiveness of a two-HIL scheme as an interactive design platform used for the design of electronic control units ECU, for internal combustion engines. The implementation and adjustment of two NMPC strategies aimed at optimizing the operation of the engine from three fronts: energetic, economic and environmental, involving independent and simultaneous objectives, give an account of the usefulness of the interactive design scheme.

Keywords Hardware in the loop · Interactive design scheme · Internal combustion engine · NMPC

1 Preface

The problem of optimizing engine operation involves, minimizing fuel consumption, minimizing air emissions and

maximizing production of torque, this constitutes the three optimization fronts as shown in Fig. 1. The latter aspect is point greater interest in engineering, which can also be understood as minimizing the tracking error to a speed reference, as set out in the formulation of the optimization problem. It is usual in engineering problems that one or more variables of a process to follow a desired reference, and this issue has been solved by implementing systems open-loop control, when it has an accurate model of the process and is free of disturbances, and closed-loop when disturbances and inaccuracy of the model, which is common and actual contemplated. For the case, a fixed reference is set for the lambda-controlled variable λ , and for the angular velocity ω_n , the reference changes at will of the operator according to the desired operating point. The exigency tracking these references depends on the optimal configuration at any given time is established by weighting optimization goals. The solution of dynamic optimization problem, which involves tracking these references, and at the same time provides for the optimization of resources while maintaining the same restrictions on availability, and outputs and states for safety, it is contemplated in the theories of optimal control.

For the case, it is justified to note the resulting complexity in the setting and test of the control algorithms. The three control variables mentioned, limits of optimization, restrictions of state variables, adjustment weights of the optimization objectives, simultaneity of control objectives, and the inherent elements to the precision and speed of execution of the controllers, results in a system with many degrees of freedom, leading in the first place to a large number of tests for adjustments and consequently to an increase in the probability of errors in the tests. For the proposed control algorithms, given their complexity, making such adjustments on the physical process, i.e. on the actual machine, is impractical and also risky. So for the effect of design and implementation of

✉ José Albeiro Valencia Chica
albeirovalencia@misena.edu.co

Adalberto Gabriel Díaz Torres
gdiaz@eafit.edu.co

¹ Servicio Nacional de Aprendizaje SENA, Dg 104 # 69 – 120, Medellín, Colombia

² Universidad EAFIT, Cra. 49 #7sur - 50, Medellín, Colombia

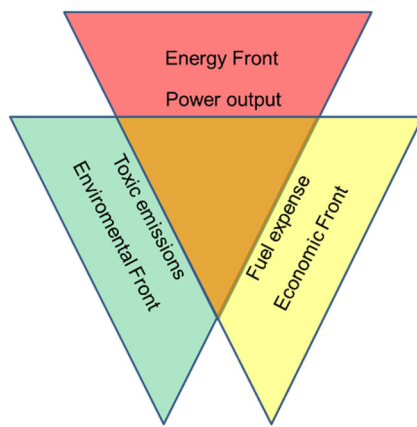


Fig. 1 Optimization possibilities

ECU's using NMPC algorithms has been arranged an interactive design platform based on a HIL scheme with two embedded systems: one of them supports the model simulation of the internal combustion engine spark ignition, and another executes the algorithm of control of optimal operation of the machine, such that, once the algorithm is adjusted correctly, it becomes the ECU designed. This is a differentiating aspect with the usual HIL schemes where the algorithm design resides in a personal computer. The platform includes the development and implementation of the detailed model of the engine containing the modeling of the behavior of the sensors and also involving the real actuators in the control loop, assuring their simulation in real time, the report of these details are found in previous publications [32]. A Human–Machine interface offers the possibility of continuous interaction for each of the design parameters of the ECU and the underlying non-linear optimum controllers: boundaries of control variables, matrices of weighting of optimization objectives, alteration of dynamic conditions of the process, introduction of adverse measurement conditions, introduction of uncertainty in modeling parameters, disturbances in the output and input variables. It also offers the possibility to visualize the actual behavior of all the model state variables and the internal variables of the nonlinear controllers to observe the performance of the optimization algorithms. As a result the platform allows the design of ECUs interacting in real time with the parameters, evaluating the behavior of the machine, analyzing the performance of the controllers, their robustness and optimality in the three work fronts, i.e. by interactively designing controllers for engine operating scenarios by minimizing fuel consumption, or delivering maximum engine power, or reducing atmospheric emissions, or scenarios where optimization of two or three fronts can occur simultaneously. According to the reliability of the response of the platform and hence of the simulation model the ECU designed is a product ready to use with a real engine.

In what follows a short review of the concept of optimal control, formulations practical possibilities within these predictive model based control (MPC), justified by its broad industrial use. Two formulations initially explored in the current research project, focusing on an approach to nonlinear MPC structures, or NMPC then expands. Common aspects are reviewed in the performance of dynamic controllers such as stability and robustness. Finally aspects of the implementation of the controllers, hardware platforms real time and a summary of the results are outlined.

The first tests of the reliability of the implemented algorithms have been obtained in a previously developed, hardware-in-the-loop scheme, involving in the control loop the physical actuators of the combustion engine. The control algorithms have been implemented in control hardware in real time, as well as the dynamic model of the plant in question, increasing the reliability of the analyzes. In addition, the dynamic characteristics of the sensors have been implemented at the FPGA level, to model the responses of the sensors independently. In addition to this scheme, an interface for the design and adjustment of controllers, parameter changes in the model, simulation of limit situations and disturbances are a technological solution for the interactive design of electronic control units for internal combustion engines spark ignition.

2 Materials and methods

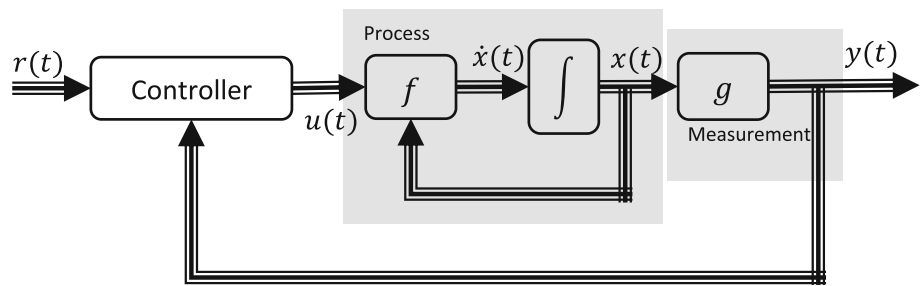
2.1 Optimal control

In the analytical treatment of control systems it is common to use block diagrams represent the flow of iterative information presented between components. Figure 2 shows a control system of a nonlinear plant [1].

The process is here described by a set of equations of state and output of the form:

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

The signal $r(t)$ denotes the reference for the controlled or output $y(t)$ variables, and $u(t)$ is the signal generated by the controller. The aim of the controller is to take the output $y(t)$ to the reference value given $r(t)$, for this deviation is determined and according to gains, adjusted by the designer, the amount and speed of action is calculated to apply in the input signals or excitation system $u(t)$. Considering only the knowledge of the plant and assuming an ideal environment without disturbances or errors, the output feedback is unnecessary, and so be in a system of open loop control. In the real case, the feedback is necessary to add robustness to the control system, especially for cases where controllers use

Fig. 2 Basic Operating diagram of a control system

the process model. Note that if the function $g(\cdot)$ is linear basically what is done is a state feedback process, hence the control is basically a feedback weighted state by a gain controller $u(t) = Kx(t)$. If not all state variables are measurable, a state estimator or observer, that determines the complete status from the current measurements is used. Moreover, regarding the control system of Fig. 2, if $f(\cdot)$ and $g(\cdot)$ are linear functions, i.e. the process behavior it is linear, then the status and output are described by:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t)\end{aligned}\quad (2)$$

The constant matrix of technological coefficients A , is defined by constant process parameters, B and C are matrices with gains of process instrumentation respectively for the actuators and sensors. Then better sketch for linear systems is Fig. 3.

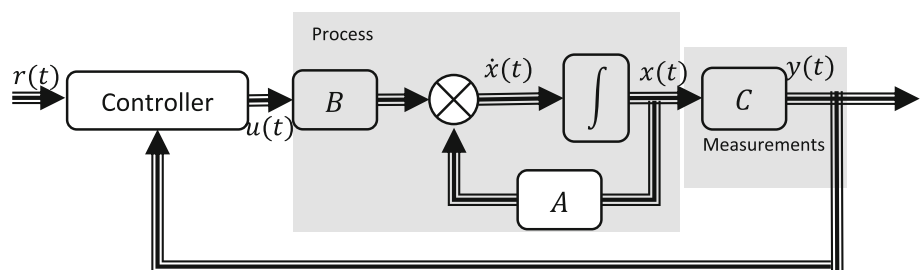
The process of interest is the combustion machine both in its nonlinearly as their linear approximations. His description and model is widely documented [2] and specifically for this report [3], the controller is what concerns at this time. A specific design theory involving objectives of this research is the theory of optimal control. The interest in designing a controller not always closes to the simple fact for tracking a reference, it may be necessary that the reference is reached in a certain time, remove an overshoot in the response of the plant, or merely stabilize a system naturally unstable. For example, in optimal control, performance measurement and therefore the design criteria is supported on an objective function, thus also named as performance index, in addition to the controller output always contemplate maintain bounds for variables state and for inputs, considering that the decision

variables are not static but move in time obeying a representation of the process or model. Hence the problem of control can be formulated as a dynamic optimization problem since it involves time and the elements of a formulation of an optimization problem.

With reference to Figs. 1 and 2, interest is to find a control function or profile optimal $u^*(t)$ Control that directs the process or plant an initial state $x(t_0)$ to a final state $x(t_f)$ verifying limits on the control signals and state variables minimizing an objective function. This requires a mathematical description of the process to be controlled, a specification of the performance index, and establish physical boundary conditions for the actuators and state variables. The first requirement for the case study presented in [3,4], and is a set of equality constraints for the optimization problem, dimensions or boundaries then constitute inequality constraints, although simplify denominating as the feasible solution set.

The performance index takes several forms that change depending on the purpose, for example, if the design criteria is to take the process from an initial to a final state in the minimum time the performance index can be written as $J = \int_{t_0}^{t_f} dt = t_f - t_0$, if you want to minimize expenditure mass transferred to the process, which refers to the input variable, then it can be $J = \int_{t_0}^{t_f} |u(t)| dt$, or similar energy expenditure $J = \int_{t_0}^{t_f} u^2(t) r dt$, where r is the rate of consumption, which for multiple inputs can be written in matrix as $J = \int_{t_0}^{t_f} u^T(t) Ru(t) dt$.

For a tracking system reference is usual minimize the integral of squared error $J = \int_{t_0}^{t_f} [r(t) - x(t)]^T Q [r(t) - x(t)] dt$, and if the position of the state at the end time is relevant, then you can be introduced $J = [r(t_f) - x(t_f)]^T F [r(t_f) - x(t_f)]$, matrices Q and F are weighting matrices of a state

Fig. 3 Control systems for a linear process

variable against another, they must be positive semidefinite, so as to generate indexes convex performance. The generalized manner used as an index of performance for optimal control systems is given by:

$$J = z(t)^T F z(t) + \int_{t_0}^{t_f} \left[z(t)^T Q z(t) + u(t)^T R u(t) \right] dt$$

$$z(t) = r(t) - x(t) \quad (3)$$

The performance index 3 is called quadratic performance index, as has been used for the formulation of the optimization problem of engine operation. As it is, it's called a problem Boltza, if it contains only the terminal cost is a problem called Mayer problem, if only contains the integral term is called problem of Lagrange problem. These forms lead to optimal control schemes more effective [5].

The formulations given in the literature to solve this problem are many, but only some have little practical from the point of view of engineering. To introduce the concept of the solution briefly we review some theories.

From the the calculus of variations [5], similar to the methods based on Newton's method, optimal $u^*(t)$ path (t) or $x^*(t)$ is determined by the conditions of change of the first order and second order. To derive this, suppose the following formulation of an optimal control problem:

$$\min_{x(t), u(t)} J(\dot{x}(t), x(t), u(t), t)$$

$$= \int_{t_0}^{t_f} V(\dot{x}(t), x(t), u(t), t) dt$$

s.a

$$g_i(\dot{x}(t), x(t), u(t), t) = 0 \quad i = 1, 2, 3, \dots, m. \quad (4)$$

With boundary conditions for the state $x(t_0)$ and $x(t_f)$. Forming a Lagrangian, which leads to an augmented cost function:

$$\mathcal{L}(\dot{x}(t), x(t), u(t), t)$$

$$= V(\dot{x}(t), x(t), u(t), t) + \lambda^T g_i(\dot{x}(t), x(t), u(t), t)$$

$$J_a = \int_{t_0}^{t_f} \mathcal{L}(\dot{x}(t), x(t), u(t), t) dt \quad (5)$$

Then the Euler–Lagrange equations derived from variational first of J_a apply.

$$\left(\frac{\partial \mathcal{L}}{\partial x} \right)_* - \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{x}} \right)_* = 0$$

$$\left(\frac{\partial \mathcal{L}}{\partial u} \right)_* - \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{u}} \right)_* = 0$$

$$\left(\frac{\partial \mathcal{L}}{\partial \lambda} \right)_* - \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\lambda}} \right)_* = 0 \quad (6)$$

The subscript * indicates that the application of the above relationships gives rise to a set of linear combinations of the optimal functions $x^*(t)$, $u^*(t)$ and $\lambda^*(t)$, from which is possible to obtain a function, or family of functions for $x^*(t)$ and $u^*(t)$. The boundary conditions for these equations depend on the conditions of the problem (final time and end position), in general form it is:

$$\left[\mathcal{L} - \dot{x}^T(t) \left(\frac{\partial \mathcal{L}}{\partial \dot{x}} \right) \right]_{*t_f} \delta t_f + \left(\frac{\partial \mathcal{L}}{\partial \dot{x}} \right)^T_{*t_f} \delta x_f = 0 \quad (7)$$

For the formulation of the problem of Boltza, ie the function of terminal cost, the Hamiltonian is used or function Pontryagin $\mathcal{H} = V(x(t), u(t), t) + \lambda^T f(x(t), u(t), t)$, and this is what constitutes the principle of Pontryagin [5].

Then with $\left(\frac{\partial \mathcal{H}}{\partial u} \right)_* = 0$, an expression for $u^*(t)$ is obtained in terms of optimal functions for $x^*(t)$ and $\lambda^*(t)$. The latter are the set of differential equations defined by:

$$\dot{x}^*(t) = \left(\frac{\partial \mathcal{H}}{\partial \lambda} \right)_*$$

$$\dot{\lambda}^*(t) = - \left(\frac{\partial \mathcal{H}}{\partial x} \right)_* \quad (8)$$

With boundary conditions rewritten in terms of Hamiltonian. Along with the initial condition this formulation leads to a TPBVP (two point boundary value problem). The second variational Hamiltonian, provides information of the character endpoint: minimum or maximum. Note that the introduction of the Hamiltonian rewrites the optimization problem of a functional in an optimization problem of a function with respect to $u(t)$. Which implies that $\mathcal{H}(x^*(t), \lambda^*(t), u^*(t), t) \leq \mathcal{H}(x^*(t), \lambda^*(t), u(t), t)$ [5].

For example using the procedure Pontryagin a plant defined as in 2 and 3 using the quadratic form of the performance index, the control relationship is obtained:

$$\frac{\partial \mathcal{H}}{\partial u} = 0 \rightarrow Ru^*(t) + B^T \lambda^*(t) = 0$$

$$u^*(t) = -R^{-1} B^T \lambda^*(t) \quad (9)$$

And the Hamiltonian system:

$$\begin{bmatrix} \dot{x}^* \\ \dot{\lambda}^* \end{bmatrix} = \begin{bmatrix} A & -E \\ Q & -A^T \end{bmatrix} \begin{bmatrix} x^* \\ \lambda^* \end{bmatrix}$$

$$E = BR^{-1}B^T \quad (10)$$

With the boundary conditions for the initial and final state one TVBVP is obtained. This relationship generates an optimal control law in open loop, in the case of state feedback system, the control law is rewritten as:

$$\begin{aligned} u^*(t) &= -R^{-1}B^T P x^*(t) \\ \lambda^*(t) &= P x^*(t) \end{aligned} \quad (11)$$

P is a transformation matrix must satisfy the call Riccati difference equation:

$$P(t) = -P(t)A - A^T \dot{P}(t) - Q + P(t)EP(t) \quad (12)$$

With E as defined above. Rewriting the final condition of $P(t)$, this can be solved backward in time.

Another way, the optimal control problem can be solved using in principle of optimality resulting in the so-called dynamic programming Bellman [6, 7], whose idea is to reformulate a problem in an optimization problem multi-stage discrete, such in each sampling period optimal decision is made. If each sub-stage is optimal, then the whole state trajectory, formed by these sub-stages is optimal. In mathematical terms in optimal functional is written:

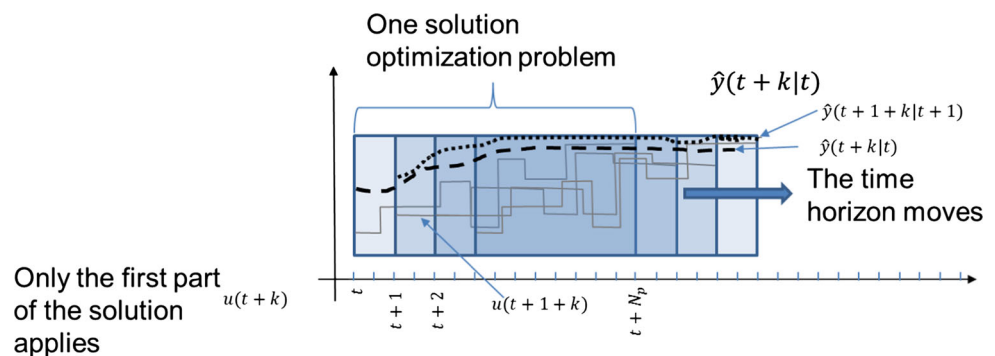
$$J_k^*(x(k)) = \min_{u(k)} [V[x(k), u(k)] + J_{k+1}^*(x^*(k+1))] \quad (13)$$

With k as the parameterization of discrete time. An iterative development of the solution in discrete time, each sub-stage, resulting in an analytical expression, for example, for linear plant as a function of Kalman gain and the matrix Riccati, with the same results are obtained with the principle of Pontryagin. Direct application of the principle of dynamic programming in continuous time systems leads to the Hamilton-Jacobi-Bellman equation:

$$\begin{aligned} J_t^* + \mathcal{H}(x^*(t), J_t^*, u^*(t), t) &= 0 \\ J_t^* &= \frac{\partial J^*(x^*(t), t)}{\partial t} \\ J_x^* &= \frac{\partial J^*(x^*(t), t)}{\partial x^*} \end{aligned} \quad (14)$$

Once resolved, and obtained J^* , the gradient is calculated, and optimal control law from $\left(\frac{\partial \mathcal{H}}{\partial u}\right)_* = 0$.

Fig. 4 Functional diagram of a controller MPC



The above methods pose a elegant solution, but it is not easy to obtain relations 6 for real-world problems, and in cases where it is easy to obtain, the solution is an exemplary task, just as occurs for the Hamilton–Jacobi–Bellman equation. However it should be noted that the principle of optimality of Bellman, can then support the use of other optimization strategies, as the active set, and the principle of Pontryagin clearly defines the optimum conditions for dynamic optimization problems. In what follows, the industrial usual scheme for optimal control problems of dynamic processes, synthesized as predictive control model MPC (Model Predictive Control) is established.

2.2 Formulation MPC

With regard to industrial control systems, which must be solved in real time, mechanisms and theories optimization run on systems discrete computation and also in line such that the information of the plant and optimization are treated as discretized. MPC scheme based on discrete models of the plant or process control is then formulated.

Considering the optimization problem with performance index 3 and $t_f = \infty$, consider implementing your solution as a controller infinite horizon. The design of this controller can be performed repeatedly solving optimal control problems finite time so that the horizon is sliding to meet the infinite horizon. At each sampling instant resolves a problem of optimal open-loop control with a finite horizon, using the measurements to define the current state. In Fig. 4, a schematic of the process shown, the shaded areas shows slipping a window in time, containing the horizon optimization provided by the MPC controller. The first window shows the process given in the current sampling period, the following exemplifies a subsequent sampling period.

At the current time t , the controller makes up, based on the model and measurements of the state, a problem of optimal open-loop control with a final time for states equal to a prediction horizon N_p , and input variables equal N_c a period control. N_p and N_c are given in terms of sam-

pling instants k are parameters and controller setting. The optimization problem formed is solved by a method and solution, which has the form of a constant path pieces, only the first piece is applied to the actuators of the process, as highlighted in the graph in $u(t)$, ie in the interval $[t, t + 1]$. A sampling period then updated with state information, a new problem is formulated with the same size in the horizons but taken from $t + 1$, that is with a displacement of a sampling period. Note from the graph that the output paths differ from one sampling instant to the next in the controller window, this exemplifies the work of the controller, which for an instant sampling behavior of the plant is predicted for a prediction horizon, taking into account current measurements using the process model and control path obtained, in the next instant as the prediction differs from the actual behavior, the prediction is readjusted again.

So $\hat{y}(t + k|t)$, in the figure, does not refer to process measurements, such as $u(t + k)$ does not over process inputs. The notation $(t + k|t)$ denotes the value of the variable which accompanies in the future time $t + k$ predicted or calculated at time t . It is valid to consider that the time t , always be the reference point such that as the horizons are fixed for a given parameter, we can without loss of generality do $t = 0$, and with this in mind we establish a general practice formulation controller MPC [8]:

$$\min_u J = x_N^T F x_N + \sum_{k=0}^{N_p} [\hat{y}(k) - r(k)]^T Q [\hat{y}(k) - r(k)] + u(k)^T R u(k) \quad (15a)$$

s.a.

$$\begin{aligned} \hat{x}(k+1) &= A\hat{x}(k) + Bu(k), \quad k = 0, 1, 2, \dots, N_p \\ y(k) &= C\hat{x}(k) \\ x(0) &= x(k) \end{aligned} \quad (15b)$$

$$\left. \begin{aligned} X_{min} &\leq \hat{x}(k) \leq X_{max} \\ U_{min} &\leq u(k) \leq U_{max} \\ Y_{min} &\leq \hat{y}(k) \leq Y_{max} \end{aligned} \right\} \quad \forall k$$

$$x_N \in X_F \quad (15c)$$

The standard MPC controller then solves a dynamic optimization problem with linear quadratic constraints that fits the active set algorithm. Convexity properties are secured by determining the weight matrices F , Q and R . To keep the small error and the square of positive error, Q must be positive semidefinite, the cost of control should be a positive amount implying that R must be positive definite, and the terminal cost involves obtaining at the end a desired position for the state, to get close to that position, F must be positive definite. For the final application, these matrices allow the engineer to tune the final answer control, weighing conflicting criteria and performance requirements.

The restriction $x(0) = x(k)$ indicates that the model is solved with current measurements as the initial value of the differential equation, this implies a feedback even though the formulation is an optimal control problem in open loop [10]. The matrix C represents the constant technological relations between the system outputs and states, if the outputs are directly the set of state variables, then C , can be an identity matrix, whereby the output is directly the state, the case of the linear model is always constant, then the output is an amplified representation of the state, is valid then, in the objective function replace \hat{y} by \hat{y} .

Note in the Eq. 15b, which for each subsequent period to N_p an equality constraint is built, then a larger N_p implies better prediction, but more processing demand [20]. 15c represents the valid bounds for all instants predicted prediction, are features of the operating functional realities as saturation of actuators and safety limits for states and outputs, they then make the feasible region of solutions for the problem formulated, and the region X_F , is a subset of the feasible region where the terminal state falls. Note that the optimization problem must be run online, which will require greater resources of the control device when the dynamics of the process so requires.

2.3 NMPC strategies

The MPC formulation described, is suitably arranged for linear dynamic, such that the plant which is the equality constraints form a set of linear constraints, adequately solved by algorithms quadratic programming: a quadratic objective function convex and a set of linear constraints. The issue is that the objective seeks to optimize a process operation with non-linear dynamics expressed through a clear non-linear model. In the hope of obtaining the benefits of the rigorous theoretical support and the broad experience of MPC, several NMPC (Nonlinear Model Predictive Control) formulations have been explored [9–11]. The initial interest, in the framework of the project of optimization of internal combustion engines, is to evaluate the capacity to execute the non-linear control algorithms in the engine optimal operation. Two of these NMPC formulations have been evaluated: one involves programmable gain techniques used to commute an MPC from a family of MPC controllers, based on the operation point [12–14]. The second one uses NLP algorithms directly to solve the problem formulated on each sampling period using the combustion engine non-linear model [15–17].

2.3.1 NMPC supported on MMPC

MMPC (Multiple Models Predictive Controls) is the term coined [11] for a technique that involves more than one MPC controller designed for several operating points of a non-linear plant. The non-linear plant is modeled as a bank of

linear models [12] for several operating points distributed across the range of control interest. More models imply more accuracy from the multi-model system, but with an obvious computational increment. The bank of n linear models has the general form:

$$\begin{aligned} x(k+1) &= {}^i A x(k) + {}^i B u(k), \quad i = 1, 2, \dots, n \\ y(k) &= {}^i C x(k) \end{aligned} \quad (16)$$

Matrices ${}^i A$, ${}^i B$ and ${}^i C$, are constants but different for each left index i belonging to an operating point. An operation point is given by $\phi \in \Phi$.

Φ , is the complete operation space or at least the space of control interest, and $\phi = h(x, u, y)$, with h as a linearization of the non-linear model. The space of interest is partitioned into disjunctive operating regimes as $\Phi_1 \cup \Phi_2 \cup \dots \cup \Phi_n = \Phi$, where each 16 model is valid for an operation regime. Local linear models are combined into a global non-linear model by a convex combination of weighting functions $w_1(\phi) \dots w_n(\phi)$:

$$M = \sum_{i=1}^n w_i(\phi) M_i, \quad w_i(\phi) \in [0, 1] \quad (17)$$

M_i , denotes the linear model i as in representation 16, and a usual form for w is the normalized Gaussian function. The definition of operating points is on the planning variables, one or more control inputs and one or more outputs where the system is balanced. The MMPC scheme presented by the literature [18] is provided in Fig. 5.

In each control period, the deviation between the models and plant outputs is weighted by the weighting function, so that the closest model at the current operating point has a greater impact on the MPC controller. Substantial improvement has been evident concerning the MMPC strategy, as the

introduction of LPV (Linear Parameters Varying) to reduce the computational cost of the model family and to synthesize the use of the online model into one parameter model that exclusively depends on measurements [13].

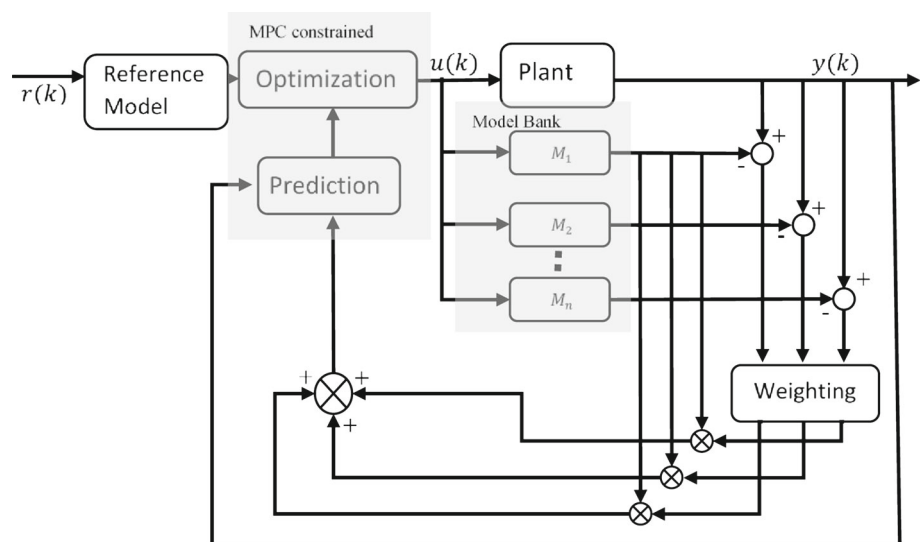
Complexity lies in the fact that finding such function is not a trivial task. Going on a different way and using the methodology of commuted controllers system, the different controllers designed are embedded in the computing system, and according to the planning variables the appropriate controller is interconnected to the plant [14].

Bearing in mind the goal of developing the NMPC controller into an embedded computing system, the solution for n models in each control period is not precisely optimal due to the storage and memory limitations. The same happens to n controllers in a commutated control system. Instead, an MMPC scheme is evaluated but the difference is that n models are not obtained off-line to then be programmed in the hardware, but the current model is obtained based on the operating point from the non-linear model of the process, supporting itself on the characteristics of parallel processing of the computing devices used. The linear model, which defines the solution of the current controller, is updated according to a shooting scheme of the linearization operation, implemented according to the given description [4] (see Fig. 6). In the implementation section, the shooting methods experienced are described, and the experimental results are reported, without the theoretical support development, which is not carried out in this paper and is left for future work.

2.3.2 NMPC with SQP

Notice that despite working in the non-linear space of the control process, the control scheme previously developed is still a linear MPC solution since the controller that solves the optimality uses a linear model at a given instant. The NMPC

Fig. 5 Scheme of MMPC of controller implementation



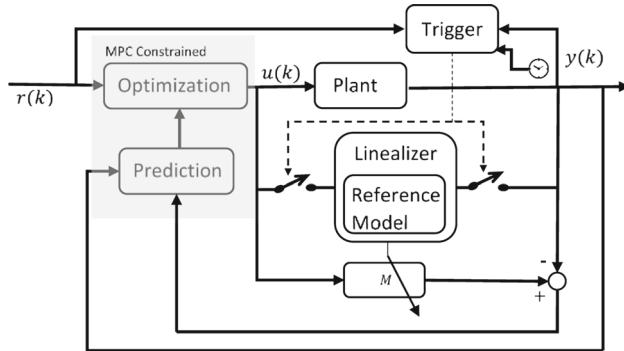


Fig. 6 MMPC scheme with online model update

term is not widely coined for this type of schemes, rather the use of non-linear models and programming algorithms is common for the same.

SQP algorithm is an appropriate option within NLP that has adapted well to the solution of optimal control problems. Just as the MPC standard, in which the linear model is used to predict the system response from an initial state up to the prediction horizon, the nonlinear model is used with the same objective for the NMPC standard, the difference lies in that the solution of future states with the linear model can be found through methods established for the solution of systems of algebra linear equations, as well as for engines of solution of ODE differential equations with mature algorithms in the numerical solution of differential equations like Runge-Kutta or Euler which are used for the non-linear model case.

Joint operations of the ODE and NLP solvers might involve a higher processing load, especially considering the infinite dimension of the optimal control problem formulation. Therefore, for a treatable solution for this with the NLP algorithms, the optimal control problem is turned into a non-linear programming problem of finite dimension [9], parameterizing the inputs and states by a finite number of parameters and approximating the differential equations during optimization. For the aforementioned, 3 methods or strategies called direct methods [19,20] are reported in literature: single shooting, also called sequential or feasible path method; multiple shooting and collocation, the latter also referred to as simultaneous methods.

In the sequential method, only control variables are discretized on the time horizon, and the model equations are solved by the ODE solver on each NLP engine iteration, so that only the control paths are considered as optimization variables or degrees of freedom. For each cost evaluation in the solution of the mathematical problem, the differential equations and cost are numerically integrated using the current estimate of parameterization vector of the inputs that come from the optimizer, the name of the sequential method comes from it, since the optimization and simulation steps are executed one after the other leading to a feasible state path.

In simultaneous methods, the solution of differential equations and optimization are executed simultaneously. For that purpose, the states, i.e., differential equations, are also discretized and parameterized and entered into the optimization problem as additional constraints, meaning that they are transformed into algebraic equality constraints [21]. With multiple shooting, the horizon is divided by several subintervals with control parameterization. Differential equations and the cost are integrated independently during each iteration of optimization, based on the current control estimate. Continuity of solutions between subintervals is guaranteed by imposing additional consistency constraints. In other words, when constraints need to be evaluated, the model is treated as a problem of initial value on each subinterval, and the continuous solution of the complete interval is obtained by using the final values of a subinterval as the initial conditions of the next one. Then, the initial values of each subinterval are treated as variables of optimization [16].

Due to the implementation ease by linking ready-to-use trustable and efficient ODE and NLP codes, sequential schemes have been efficiently used on chemical processes applications [22] with the obvious disadvantage of the repetitive numerical integration of the model and that they do not guarantee handling naturally unstable systems. The multiple shooting method was developed to solve this latter problem of the unstable systems, since the time partition and the solutions of the models on each partition allow the introduction of more state variables information in the formulation and capturing of unstable modes. However, as the mesh gets finer, the resulting NLP problem generally increases its size, and it also requires ODE solvers that are efficient enough to solve the multiplicity of the models produced.

Finite elements collocation avoids the use of the ODE solver, and hence, its problems, taking advantage of the dispersion and structure problem. Obviously, the proper selection of the discretization problem, with the right length of finite elements and the preservation of a stable and accurate solution [19] should be kept in mind. As is the multiple shooting, the resulting finite NLP is increased in size. An important, but not excluding, characteristic of the simultaneous methods is their capacity to handle instabilities, and this naturally requires some additional effort, and in terms of which scheme needs to be implemented of the scheme optimal controller of the combustion engine and concerning the scheme to be implemented for the optimal controller of the combustion engine it seems that a sequential scheme is enough, since most of the engine's operational functional space is naturally stable, only frontier areas show some degree of instability [4]. Additionally, the possibility of using ODE and NLP solvers for already available embedded systems accelerate the design and implementation of the optimal controller in real time. Performance of an NMPC sequential

scheme is then evaluated using ready-to-use ODE and NLP solvers.

Based on the substantiation of sequential methods in the single shooting, the Bolza-type, non-linear optimal, continuous-time control problem, described by 3, is reformulated into an NLP problem of finite dimension discretizing it to be solved through numerical methods. To do so, the first step is to divide the time interval $[t_0, t_f]$ by N intervals of the same size $[t_k, t_{k+1}]$ with $k = 0, 1, \dots, N-1$.

$$\begin{aligned} \min J &= S(x(t_f), t_f) + \int_{t_0}^{t_f} V(x(t), u(t), t) dt \\ \text{s.t.} \quad & \frac{dx}{dt} = f(x(t), y(t), u(t), t) \\ & X_{\min} \leq x(t) \leq X_{\max} \\ & U_{\min} \leq u(t) \leq U_{\max} \\ & Y_{\min} \leq y(t) \leq Y_{\max} \\ & x(t_0) = x_0 \end{aligned} \quad (18)$$

Then, $u(t)$ is parameterized into a continuous piecewise function or finite-dimension control vector. Even though there are several forms of parameterization, this is the most common one [23], since it directly fits the use of zero-order hold in analog-to-digital converter hardware. Then, $u(t)$ is redefined as $u(t) = u_k \forall t_k \leq t \leq t_{k+1}$ and vector p as the set of $u(k)$ values in the horizon optimization $p = \{u_k\}_{k=0}^{N-1}$. Schematization of this will be provided in Fig. 7.

The consequence of using the ODE solver for the initial value problem that involves the model:

$$\begin{aligned} \frac{dx}{dt} &= f(x(t), y(t), u_k, t), \quad x(t_0) = x_0, \\ t &\in [t_0, t_f] \end{aligned} \quad (19)$$

Leads to a discretized state vector $x_k = x(t_k)$ for $k = 0, 1, \dots, N-1$. Problem 18 is redefined in an NLP problem of finite dimension using this parameterization:

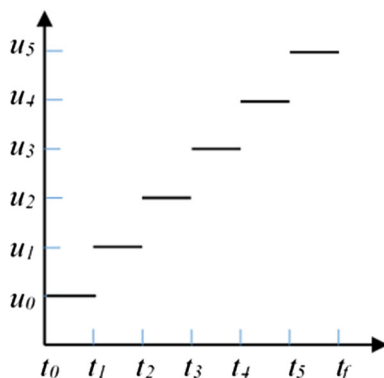


Fig. 7 Parameterized control signal

$$\begin{aligned} \min J &= S(x_N, t_N) + \sum_{k=0}^{N-1} V_k(x_k, u_k) \\ \text{s.t.} \quad & x_{k+1} = f_k(x_k, y_k, u_k), \quad k = 0, 1, \dots, N-1 \\ & \left. \begin{aligned} X_{\min} &\leq x_k \leq X_{\max} \\ U_{\min} &\leq u_k \leq U_{\max} \\ Y_{\min} &\leq y_k \leq Y_{\max} \end{aligned} \right\} \quad k = 0, 1, \dots, N-1 \end{aligned} \quad (20)$$

Additionally, the use of the model simulation eliminates its equations from the optimization problem [21], including the objective function dynamics. We have:

$$\begin{aligned} \min J &= S(x_N, t_N) + \sum_{k=0}^{N-1} V_k(x_k, u_k) : x_{k+1} \\ &= f_k(x_k, y_k, u_k) \end{aligned} \quad (21)$$

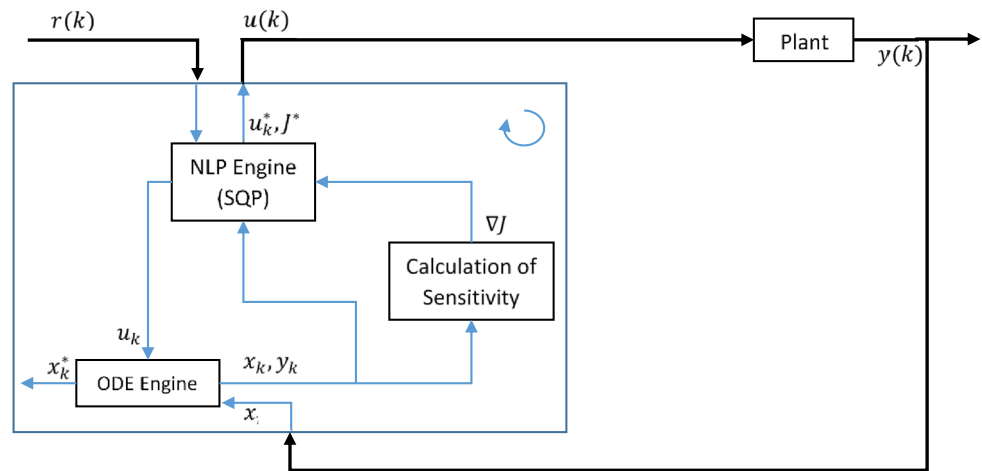
The resulting problem formed by the objective function and path constraints is a discrete time problem with dimension N that can be solved through a SQP algorithm. The work in conjunction with the ODE solver delivers the information about the dynamics of the plant to the NLP solver with the SQP algorithm that aims at an optimal p vector. The single shooting working scheme is shown in Fig. 8. For every NLP iteration, a simulation on ODE is carried out from initial conditions to the prediction horizon. The state profile is analyzed for convergence and from there to the NLP block, which calculates the control vector again. After convergence, the optimal control vector is obtained from the NLP block and the optimal state vector is obtained from the ODE block.

It is observed that the accuracy of the solution mainly depends on the model solutions in the ODE block, and the correct estimate of the initial values. The calculation of sensitivity block determines the convergence assessment criteria of the SQP algorithm and obtains the gradient of the objective function in relation to the changes in the control vector, $\nabla_{u_k} J$, $k = 0, 1, \dots, N-1$. To do so, the definitions of sensitivity of the state x_k and sensitivity of the control vector are used as in [23]:

$$\begin{aligned} S_{x_k}(t) &= \left\{ \frac{\partial x(t)}{\partial x_k} : x(t_k) = x_k \right\} \\ S_{u_k}(t) &= \left\{ \frac{\partial x(t)}{\partial u_k} : x(t_k) = x_k \right\} \end{aligned} \quad (22)$$

These functions show the influence of a slight variation of x_k, u_k in the state of the system in the time t , with $t \geq t_k$, the interest is then to calculate the influence they might have in the future state $S_{x_k}(t_{k+1}), S_{u_k}(t_{k+1})$. For $t \geq t_k$, the corresponding definitions of sensitivities respectively satisfy the following differential equations:

Fig. 8 Single shooting scheme with SQP algorithm



$$\begin{aligned} \dot{S}_{x_k}(t) &= \frac{\partial f}{\partial x}(x(t), u_k) S_{x_k}, \quad \text{para } k = 0, 1, \dots, N-1 \\ S_{x_k}(0) &= I \end{aligned} \quad (23)$$

$$\begin{aligned} \dot{S}_{u_k}(t) &= \frac{\partial f}{\partial x}(x(t), u_k) S_{u_k} + \frac{\partial f}{\partial u_k}(x(t), u_k), \\ \text{para } k &= 0, 1, \dots, N-1 \\ S_{u_k}(0) &= 0 \end{aligned} \quad (24)$$

Gradients of the objective function are also possible to calculate from the sensitivities $\frac{\partial V_k}{\partial x_k}$ and $\frac{\partial V_k}{\partial u_k}$, as:

$$\begin{aligned} \frac{\partial V_k}{\partial x_k} &= \int_{t_k}^{t_{k+1}} \frac{\partial V}{\partial x}(x(t), u_k) S_{x_k}(t) dt \\ \frac{\partial V_k}{\partial u_k} &= \int_{t_k}^{t_{k+1}} \left(\frac{\partial V}{\partial x}(x(t), u_k) S_{u_k}(t) + \frac{\partial V}{\partial u}(x(t), u_k) \right) dt \end{aligned} \quad (25)$$

2.4 Considerations for stability and robustness

Numerous efforts have been made in relation to the performance of the NMPC controller since its very beginning, but only a few of them have had clear practical applicability. In an effort to provide some evaluative comments on the performance of the 2-HIL systems used in this report, mature reviews made by Bemporad and Morari about the robust MPC [24] and Allgöwer [9] are briefly reviewed.

The length of the prediction and control horizons are relevant in terms of stability. An elongated prediction means more information for the controller and it is perhaps more desirable, but it implies a higher processing load because the size of the problem increases. On the other hand, using short-length horizons can lead to control loss especially in unstable systems, and the real paths of the state and control

signal will differ from the predicted ones even under ideal scenarios. Consequently, close-loop optimality and stability are not guaranteed, since for the nominal case, closed-loop stability with infinite horizon can be reached due to the fact that the paths calculated at a given time by the solution to the NMPC problem are equal to the close-loop system paths [9], which also implies close-loop, stability, giving birth to the idea of making the prediction finite horizons long enough so that stability is accomplished.

It is shown in [25] that for the linear MPC, supported by the closed-loop asymptotic stability theorems for an infinite horizon, an optimal finite horizon value exists so that the optimal input sequence and the value of the objective function are also optimal for the infinite horizon. This implies that the close-loop MPC control system with optimal finite horizon is asymptotically stable. This result emerges from the premise that there is always an instant from which input or state constraints would not be violated. In the selection of such instant as prediction horizon, optimality will remain. It is then suggested, in practice, to make the prediction time long enough compared to the dynamics of the system. Slightly higher to the time the system takes to return to its stable state after disruption. The foregoing remarks have also been dealt for the NMPC case [26].

Other techniques, with quite mature substantiation, guarantee the NMPC closed-loop stability by adding fictitious constraints and properly using the terminal cost of the objective function. For example, a simple option is adding an equality constraint in the state to zero final time. However, this creates feasibility problems for short prediction horizons and additionally the exact solution would never end [7]. Therefore, such constraint could be relaxed leading the final state to a terminal region and using the terminal cost as an appropriate penalizing function that can be chosen off line to reinforce the closed-loop stability. However, finding them is not an easy task since the terminal cost is a system local Lyapunov function, under the local control law in the terminal

region. If the linearized system is stabilizable and the cost function is quadratic, a linear feedback law can be used and the terminal cost can be approximated as a quadratic form $x^T P x$. The problem at hand fits these assumptions; the linearized model is naturally stable for the planned operation range and the quadratic cost function. The following is a procedure developed in [27] and summarized by [9], to obtain both elements off-line:

- Solve the linear control problem on a linearization of the model equations to obtain a locally stabilizing linear feedback of the state $u = Kx$.
- Choose a constant $\kappa \in [0, \infty)$ such that $\kappa < -\lambda_{\max}(A_K)$ and solve the following Lyapunov equation $(A_K + \kappa I)^T P + P(A_K + \kappa I) = -(Q + K^T R K)$, and with this, a matrix P defined positive and symmetric, $A_K = A + BK$ is obtained.
- Find the biggest possible term α_1 term to define a $\Omega_1 := \{x \in \mathbb{R}^n | x^T P x < \alpha_1\}$ region with Kx that belongs to the limited set of inputs to the system and the x state that belongs to such a region that is a subset of the feasible region.
- Then, find the highest possible value for $\alpha \in [0, \alpha_1)$ to specify the terminal region $\Omega := \{x \in \mathbb{R}^n | x^T P x < \alpha\}$ so that the optimal value of $\max_x \{x^T P \varphi(x) - \kappa \cdot x^T P x | x^T P x \leq \alpha\}$ is not positive, being $\varphi(x) := f(x, Kx) - A_K x$.

This procedure applies if the linear equivalent in the origin is stable. With these values the open-loop optimal paths in each sampling instant approximate the optimal solution to the problem formulated with infinite horizon. It is noteworthy that the $u = Kx$ law is never applied. This is only used to find the stability terms. This method has been called QIH-NMPC quasi infinite horizon and perhaps the most important conclusion of the authors is that if the terminal penalty function is such that the corresponding local control law is a good approximation to the control law resulting from the problem with infinite horizon in an origin-centered neighborhood, the performance of both problems can be made equal by even using short prediction horizons, and thus the closed-loop stability will not be corrupted.

The solution to the problem of stability has been addressed and at some degree, satisfactorily but only for the nominal case. From a real point of view, the case is not completely satisfactory since it is assertive to mention that the model is neither exact nor the system is exempt from being disturbed. The issue that remains is whether the closed-loop stability is still not facing disruptions and/or modeling errors. In the same Algöwer's [9] review, three robust schemes for NMPC are explicitly mentioned, besides the inherent robustness of closed-loop control systems against reduced disturbances: one of them seeks to keep the standard NMPC set up, only

changing the objective function to maximize the worst disturbance case in a given sequence, leading to a problem called min-max [28]; a second one uses the standard problem of robust control, formulated in a way similar to the MPC and with mobile horizons, modifying the NMPC cost functions in the form of H_∞ ; and another one that results from the remark that only in sampling instants corrective action is applied, and intermediate disruptions are not rejected afterwards. From here, not optimizing the input, a sequence of local control laws is suggested instead, that is to say, a vector of parameterizations of feedback controllers.

The min-max formulation of the objective function has been the most frequent basis to add robustness to the NMPC, which usually improves the performance in the worst scenario caused by the parameters' uncertainty, but that shows poor results for the nominal case. To overcome this difficulty, minimization of the weighted sum of the nominal value or the expected terminal cost has been used with variance around the nominal or expected value of the terminal cost with variance around the nominal or expected value, caused by the parameters' uncertainty. Naggy and Braat [29] use a Kalman's extended filter with this method to fight noise and uncertainty, as well as several NMPC robust schemes based on EKF, but using a sequential strategy of integration and correction, instead of executing prediction by integrating the model equations on the complete prediction horizon. The initial values for the state equations are obtained from the EKF estimator. They report a two-factor improvement on batch processes. Explicitly, the objective function involves a vector of uncertain parameters, as well as the constraints. The vector of uncertain parameters belongs to a set characterized by a probability density function.

Several authors indicate the inherent robustness of the nominal NMPC. Panocchia [30] establishes the properties of the robust stability for a sub-optimal non-linear MPC, and extends it to the optimal case, considering that the optimal controller is a particular case of sub-optimality. Robust exponential stability is shown concerning arbitrary and small disruptions and errors in the estimate and measurement, from nominal stability evidence in the linear MPC using the cost function as the Lyapunov function. The optimal solution is not attainable for non-linear systems in general, but if optimization provides a sub-optimal feasible solution that only improves cost from a weak start, the balance asymptotic stability remains and the sub-optimal cost function is a Lyapunov candidate exponential continuous function. Assuming the optimization problem remains feasible for all sampling periods, there is a Lyapunov function for the closed-loop system that implies robustness for disruptions that are small enough. In general, for linear systems with a quadratic cost, the MPC controller is a Lyapunov continuous function for the closed-loop system, since feedback of the optimal state is continuous and, therefore, achieves inherent robustness.

The general requirement for robustness using a sub-optimal cost is that the cost function does not increase compared to a sequence of definite weak start, which is reachable and easy to apply.

Along the same line of inherent robustness, handling feedback delays has led to a slight NMPC reformulation. Feedback delays have a subsequent loss of performance and stability. In the NMPC formulation based on SQP, the use of sensitivities for the single shooting cycle was reviewed. Again, Biegler and others [31] use the NLP calculation of sensitivities to promote an NMPC one step forward, thus reducing the online computational load of calculation online and, therefore, recovering nominal stability, and also leading the capacity to reject disruptions when the NMPC-SQP inherent robustness face hard non-linear and strong disruptions. By creating approximate solutions around a solution of the updated reference in a continuous way, to face the disruptions and inaccuracies of the model, the controller takes advantage of the parametric properties of the open-loop control problem formulation and it approximates to the optimal true solution using the NLP sensitivity concepts, which help to establish optimality levels according to the levels established for uncertainty. Note that it is similar to a sub-optimal condition. The sensitivity approximation is used for the weak start of the following problem. It is possible to calculate Lipschitz constants that implicitly will take into account the system's non-linearity and disruptions. With this, the authors establish inherent robustness from the input to the stability concept.

3 Implementation

The first evaluations on the effectiveness of the NMPC control schemes described above have been carried out, using a 2-HIL platform (Hardware in The Loop). Constructive and implementations details are reported in [32]. The platform has a real-time processing device programmed with the structure of Hendricks [2] model and with the parameters validated in [3], a similar device runs the NLP algorithms, the NMPC control structures and the parallel sub-processes of the machine discrete events. In the loop of the model-engine with the controller, the physical actuators have been inserted: an ignition coil and its spark plugs, injectors and injector rails measuring of injected flow, motorized throttle valve of acceleration with a sensor of TPS position. The dynamics of the pressure sensors, speed, Lambda and temperature have been properly implemented to an FPGA (Field Programmable Gate Array) level in a device that acts as an engine. The 2-HIL schemes are shown in Fig. 9.

The central block marked as human-machine interface constitutes the channel of interaction with the platform, Fig. 10 shows an image with its possibilities. In the upper right area, the non-linear control scheme to be evaluated is

selected, and the relevant parameters are available to evaluate in each one. In the case of MMPC, the option to select the model update mechanism, and the parameters of each mechanism such as: update period, update delay, minimum error and speed change threshold, among others; in the case of the SQP scheme, the tolerance of the objective function, the tolerance of the parameters and the maximum number of calls of the algorithm can be controlled, so that the optimization can be made more relaxed or more demanding.

In order to define the overall optimization strategy, the “Optimization Fronts” zone has standardized controls, to weigh interest in the objective function, such that the user can perform the most powerful, or more economical, or more environmental controller, or a combination between all fronts with different degrees of belonging. This results in the calculation of the weighting matrices, and in the relaxation in the constraints of some variables, which are updated in the contiguous zone “Algorithms Settings”, however from these controls can also be configured optimization parameters: matrices for weighting the objective function, control and prediction horizons, and configuring the constraints of some decision variables.

The “Testing Control” zone implements the necessary functions to evaluate the adjustments and configurations: establishes a channel to change the motor speed reference (by performing the throttle functions, the throttle valve is controlled by the algorithm), and controls to establish different scenarios and evaluate the performance and robustness of NMPC control schemes. Like, add measurement noise to all or individually to each of the measured variables, generate disturbances on control variables or inputs, and define uncertainty of the model, this can be done on all parameters or individually to each parameter. Configurable scenarios then turn out to be quite broad: changes in engine load, fluctuations in air density, clogging or malfunction of injectors, poor valve function, fuel quality changes, sensor malfunction and noise inherent in the measurement, changes in ambient temperature and atmospheric pressure, etc.

Also provided is a configuration zone of modeling parameters, which allows slightly changing the characteristics of the modeled engine and the operating conditions. Finally, there are visualization zones where the evolution of the optimization algorithm on the built surface according to the adjustments and configurations and the behavior of the relevant variables: velocity, air / fuel ratio, injected fuel and accelerator position are observed.

3.1 NMPC controller based on the MMPC scheme

The implemented and evaluated MMPC scheme corresponds to Fig. 6. Unlike the reported MMPC schemes, the models are not obtained in advance, but in line with the structure execution. The issue to be faced is the time when the model

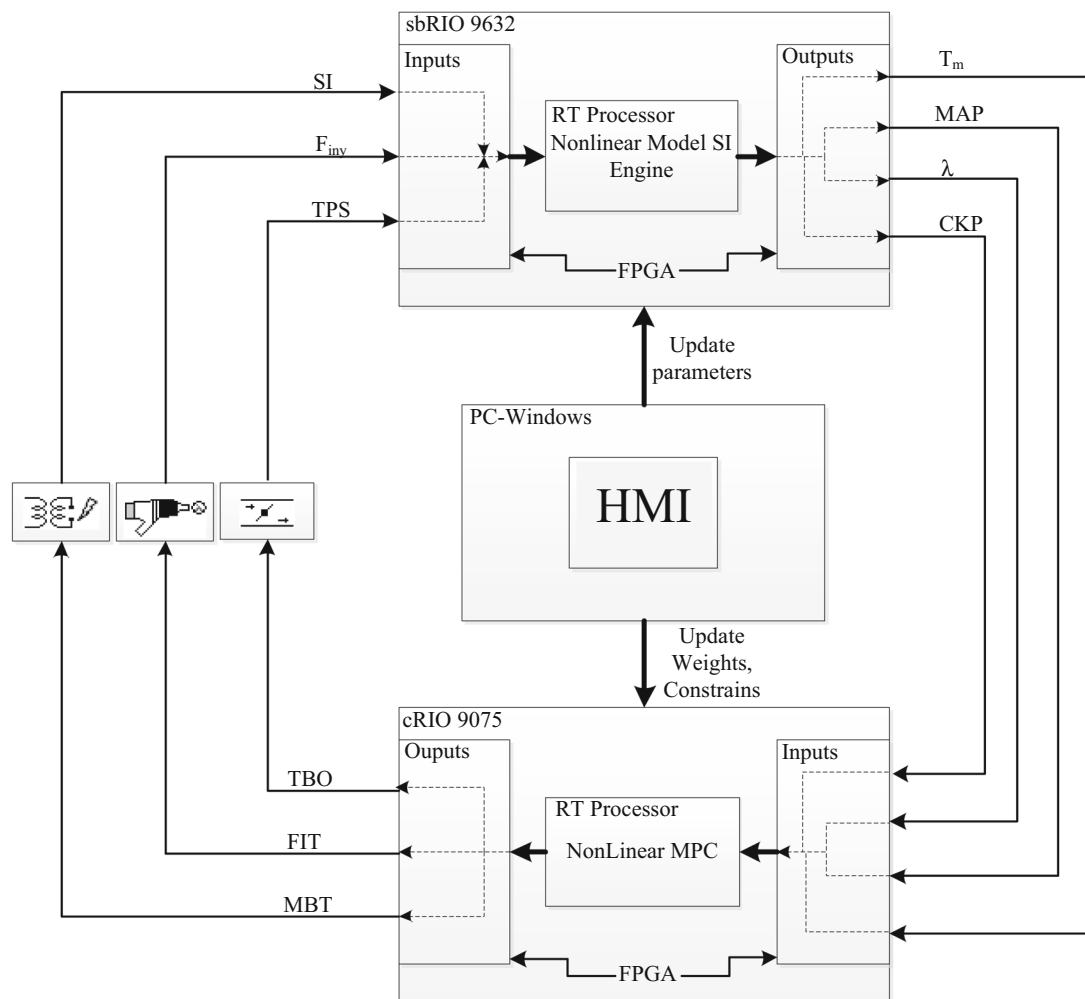


Fig. 9 HIL scheme for NMPC controllers test

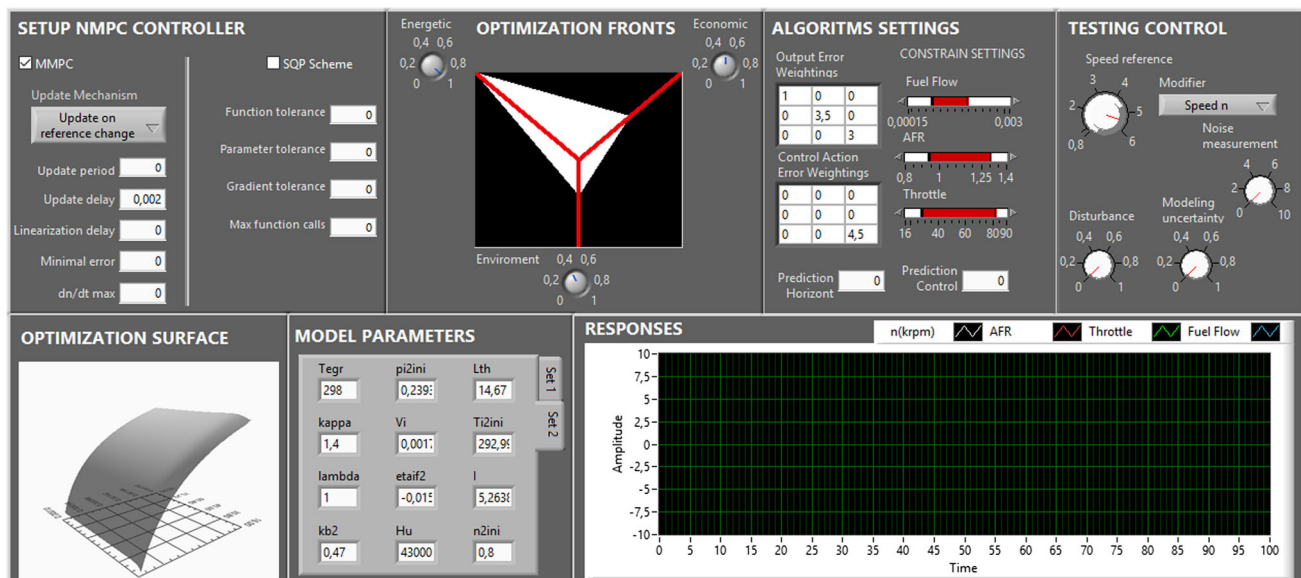


Fig. 10 HIL platform interaction

should be updated. This report explains the results of the subsequent strategies without a rigorous theoretical analysis of the potential impact on the controller performance.

Even so, it is appropriate to show the inference that the bias of the model used by the MMPC controller has on the system performance. Figure 11 shows the MPC performance using a linear model calculated at a 0.55 krpm speed. Relatively small changes have been made to the speed reference. At a reference of 570 and 580 rpm, the error in a stable state can be considered void. Having differences of 100 rpm on the model operation point, some errors begin to be noticeable, over 2 rpm at a 600-rpm reference.

Even though this difference could be considered insignificant, it is clear that is increasing. For example in Fig. 12, the reference has been taken to 1000 rpm, the error in a stable state is close to 75 rpm, however 18 seconds later (second 64 on the graph) the model has been updated to the current point of operation and the offset is completely reduced.

3.1.1 Update at regular intervals

In reference to Fig. 6, the Shooting scheme is implemented as a real time chronometer with adjustable start timing. Regardless of the changes in reference, and the sampling and control

period, it invokes the linearization processes by taking the motor current outputs as well as the inputs calculated by the controller to calculate a new linear model. With the new model and the current control parameterization, a new MPC control is implemented in the next period available. The process is again repeated when the current timing is finished. The operation is conducted in parallel regardless of the control actions regularization, a reason for which the process is asynchronous. Figure 13 shows performance by using a period of updating of 1 s. An acceptable overshoot is observed, with random magnitude, and stability is accomplished in a time slightly longer than the response with the factory ECU, reported in [3] and the MPC response without the updating effect, as shown in the next figures.

The effect of reducing the updating period to 1 s is dramatic for the system stability. With a 500 ms period, the response in Fig. 14 shows initial oscillations compared to a change of reference with initial shoots of higher magnitude rather than using 1 s, even though the stability is maintained. A period of 100 ms (see Fig. 15) imposes oscillations of progressive amplitude, even coming from a stable state until the control system becomes unstable.

Figures 16, 17, 18 and 19 are the responses when increasing the updating periods over 1.5, 2, 3 and 6 s respectively.

Fig. 11 Not updated MPC controller

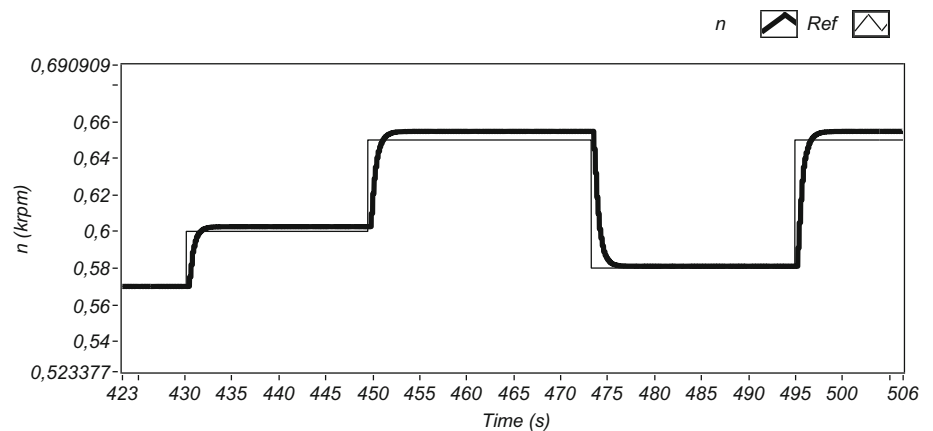


Fig. 12 MPC with eventual updating

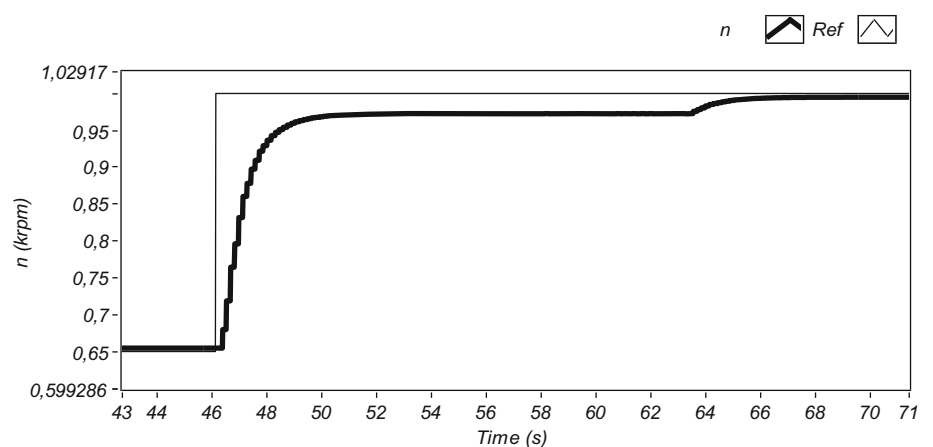
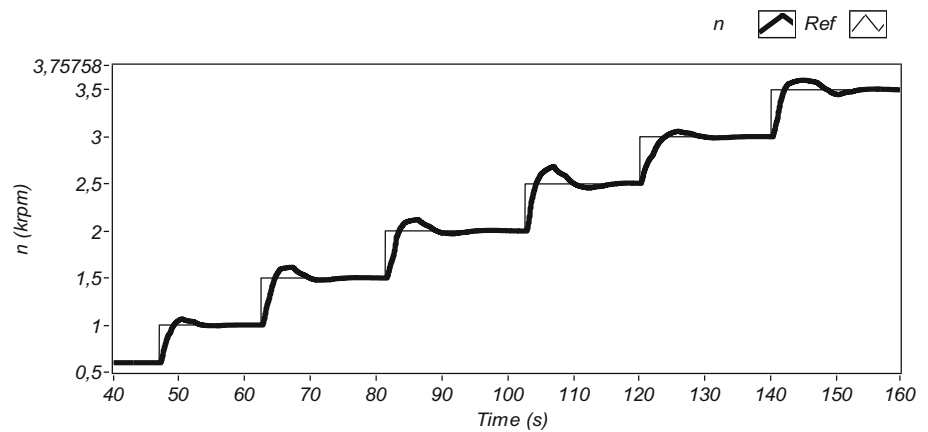
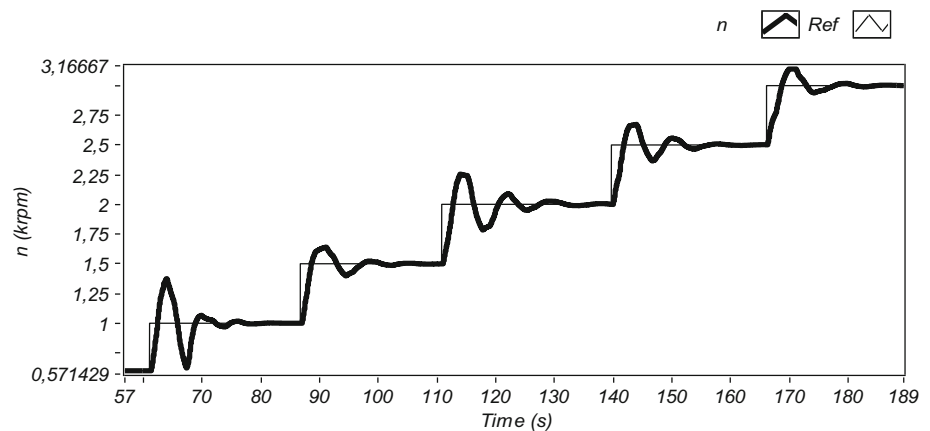
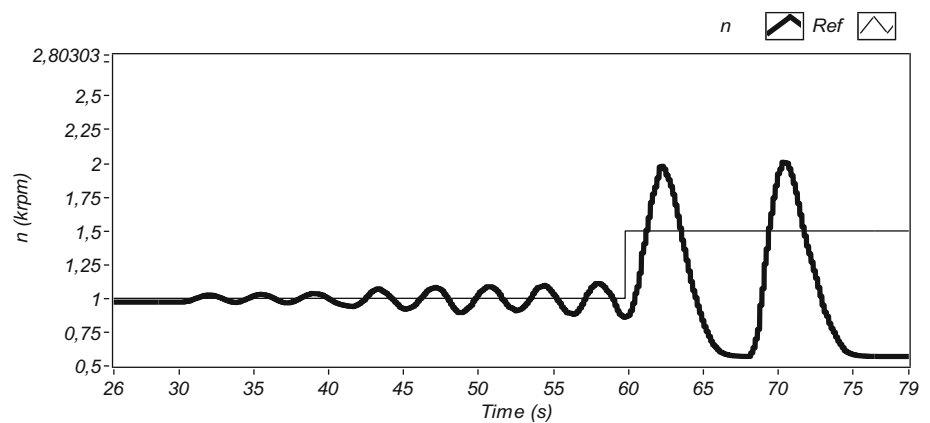


Fig. 13 MMPC with every second updates**Fig. 14** MMPC with updates every 0.5 s**Fig. 15** MMPC with updates every 0.1 s

Overshoot decrease is clearly seen in updating periods of 1.5 and 2 s, but the stabilizing time also increases and this is more noticeable with longer updating periods such as 3 s.

In Fig. 18 you can observe points within the transitional period in which the response takes a new boost towards the reference. These are instants where a model update has taken place. Furthermore, in Fig. 19, an updating period longer than the stabilization time of the plant has been used presenting a negative effect that although it finally takes the system towards the reference, the speed fluctuations can be unac-

ceptable depending on the parameters of application where the combustion engine is used.

However, the issue on the possibility of finding an appropriate moment for the model update during the transitional part arises so that continuity of such boosting points is soft enough to be noticed. The asynchronous timing scheme does not show an appropriate scenario for this since updating points are independent not only from the control period, but also most importantly from the reference changing point.

Fig. 16 MMPC with updates every 1.5 s

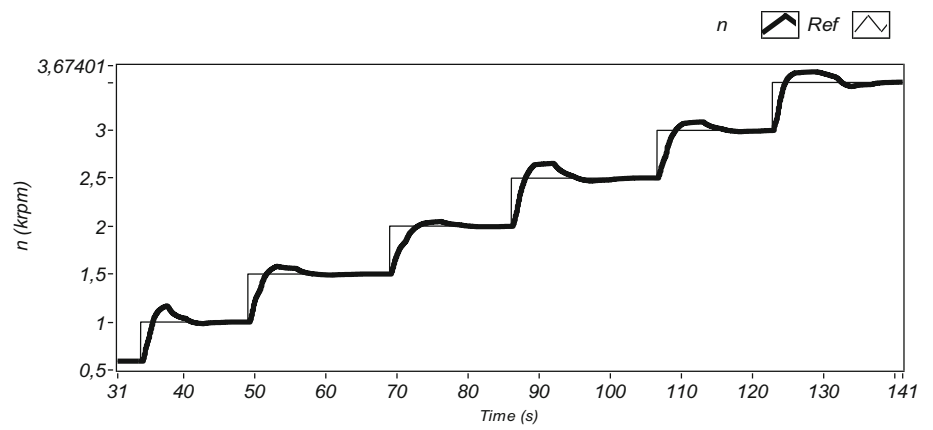


Fig. 17 MMPC with updates every 2 s

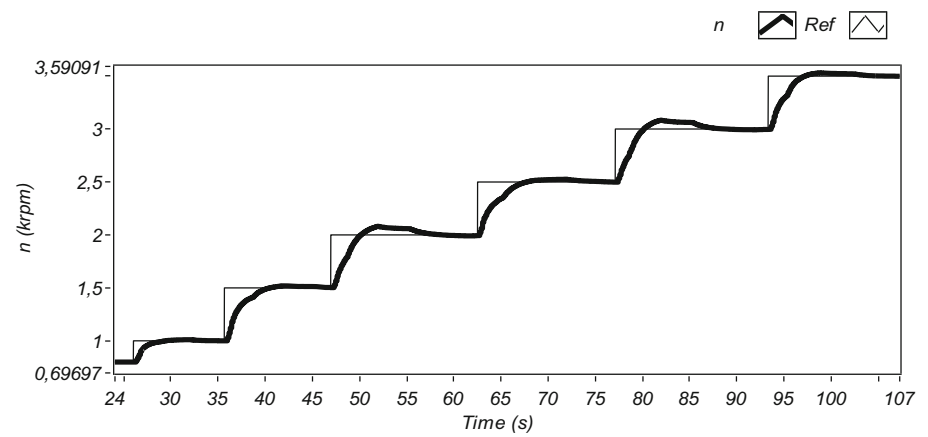


Fig. 18 MMPC with updates every 3 s

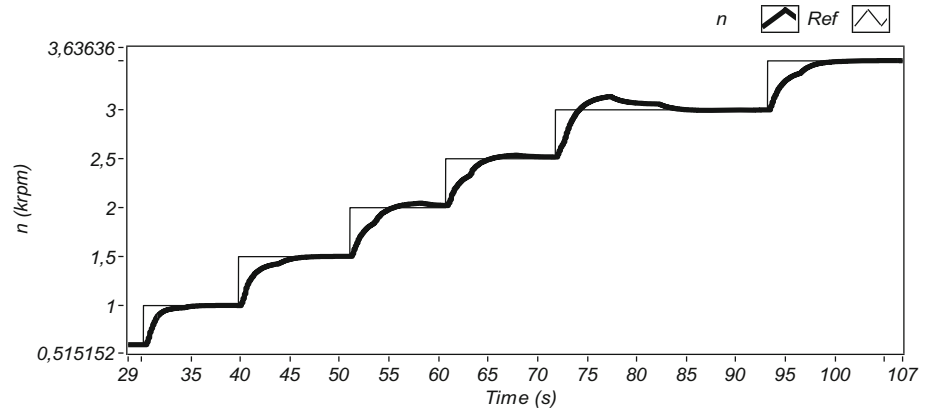


Fig. 19 MMPC with updates every 6 s

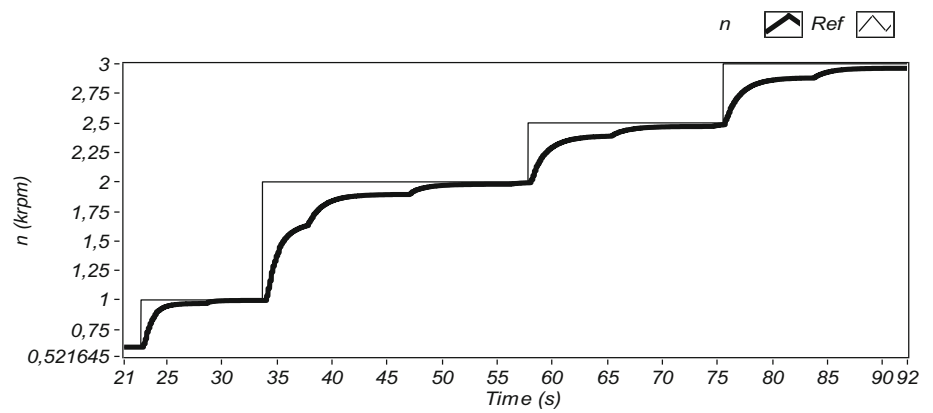


Fig. 20 MMPC with update delayed 1 s to the change of reference

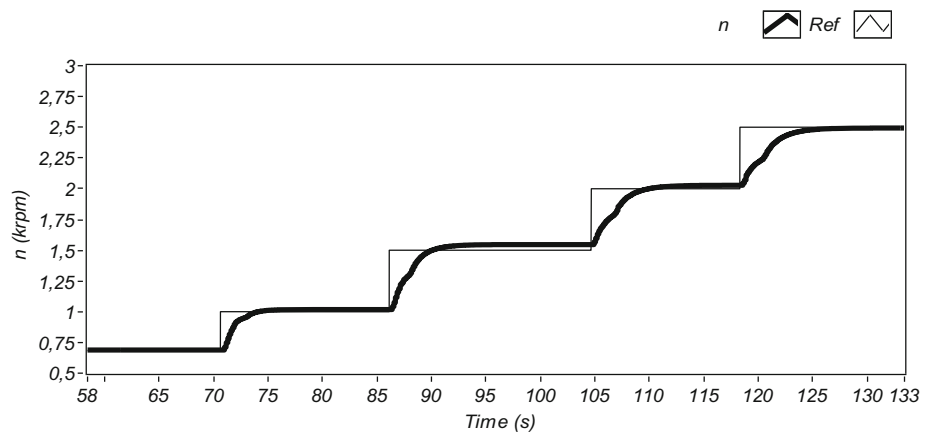


Fig. 21 MMPC with update delayed 0.5 s to the change of reference

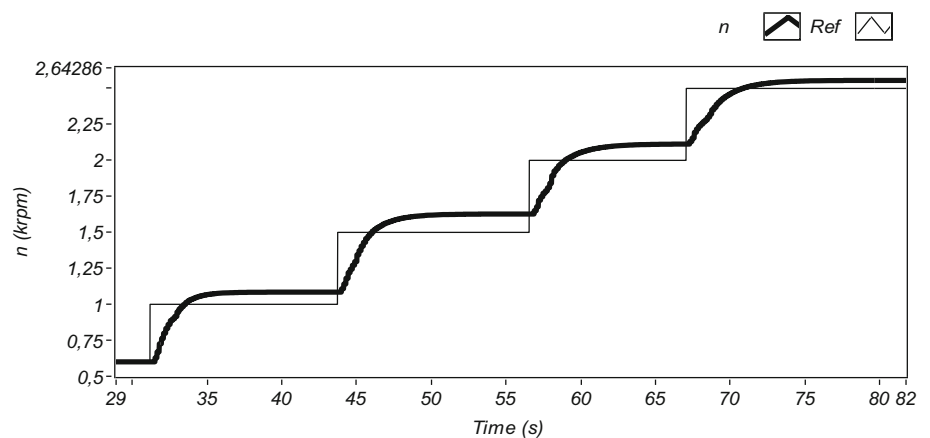
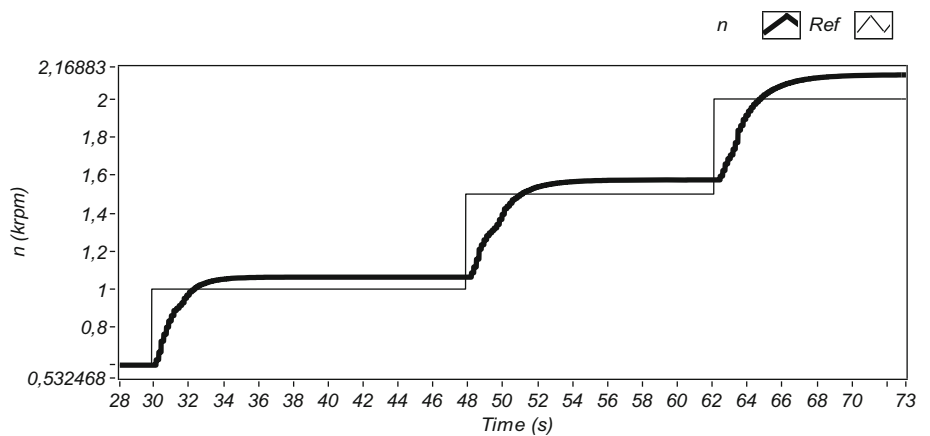


Fig. 22 MMPC with update delayed 0.1 s to the change of reference



3.1.2 Update with delay to the change of reference

In line with the aforementioned, this scheme supervises a change in the reference to start a timing that, once completed, invokes the process of updating of the model used. Then, the operation tunes in with the dynamics of the speed reference. A comparison among Figs. 20, 21 and 22 clearly shows the improvement in the response invoking the update at given

points in the transitional response. Figure 20 uses an update delay of 1 s. The use of delays lower than 1 s resulted in performance deterioration for reference monitoring as shown in Figs. 20 and 21.

The use of delays over 1 s resulted in longer stabilization periods and a loss of softness in the transitional stage, as shown in Figs. 23, 24 and 25 with delays of 1.5, 1.1 and 0.9 s respectively,

Fig. 23 MMPC with update delayed 1.5 s to the change of reference

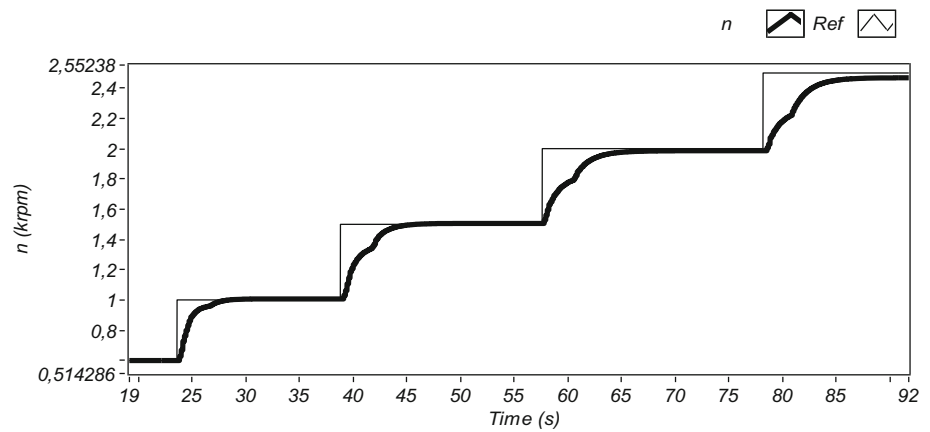


Fig. 24 MMPC with update delayed 1.1 s to the change of reference

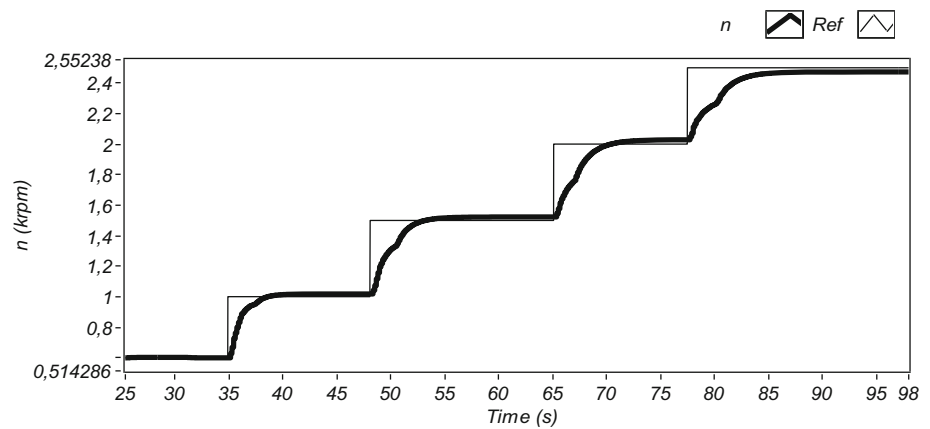
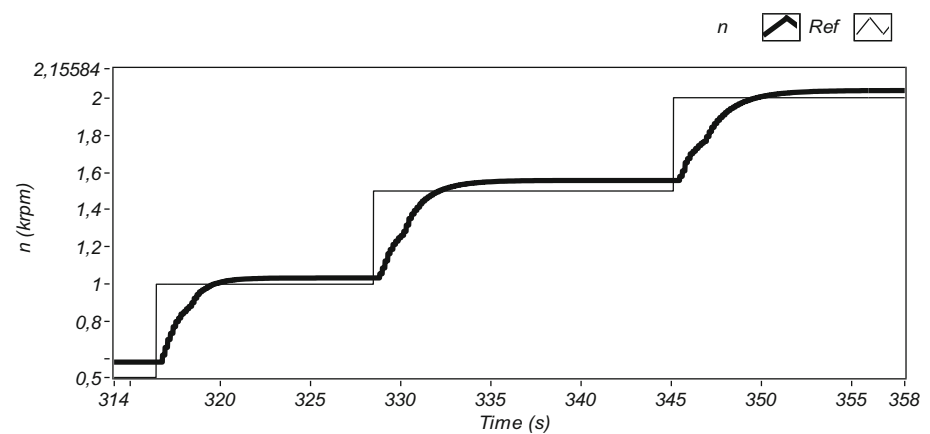


Fig. 25 MMPC with update delayed 0.9 s to the change of reference



3.1.3 Update through the measurement of change in the response

This scheme supervises the change of speed in the response and error dynamics in order to estimate the instants when the response is reaching the stabilizing point and the future error of permanent state, then correct it assuming that the error is due to the non-correspondence of the model. It then results in three adjustment parameters: e_{min} that defines the smallest error in an admissible steady state, and when detecting an error above this value, the supervision of the acceleration is activated. A margin of this incline $\frac{dn}{dt}_{max}$ allows determin-

ing a point in the dynamics of the plant to take a point of linearization. This parameter can be used to activate the linearization when it reaches the stable state; however, if e_{min} is defined for a relatively large margin, it can result in the deterioration of the engine operation softness in relation to final applications.

Therefore, a more adequate selection, under the condition of a wide error margin, is a point in the dynamics that will allow estimating that the stable state will soon be reached and a linearization delay Δt_l that works as a predicting parameter of the moment in which the stable state will be reached. Figure 26 outlines this process.

Figure 27 shows results using the aforementioned mechanism. This response shows a guarantee in error reduction in a stable state and a relatively soft response with minimal update load. Observe that the model update process can happen more than once in a transitional period. However, the results were one or two triggers of the linearization process for the tests conducted.

The test parameters were: $e_{min} = 0.01$, $\frac{dn}{dt}_{max} = 0.05$, $\Delta t_l = 30$ Ts. Ts refers to sampling periods. A soft response is achieved at the beginning of the operation with only one updating process; as the reference changes, an increasing overshoot is perceived; with the need of an additional updating process for correction. A wider change of

reference shows better correction reducing the overshoot for the same area of operation, which gives rise to the hypothesis that the effect is due to a cumulative error in the model between updating processes.

The updating mechanism is an element needed to modify the MPC algorithm into an NMPC algorithm. To evaluate other aspects of the NMPC in the control of the combustion engine, the use of a unique updating mechanism has been defined in this report. A rigorous analysis of the updating mechanisms has not been reported here; however, a qualitative conclusion allows defining the mechanisms' best behavior in tests. The most important aspects to conclude are defined by the capacity of reference monitoring, softness in the control or bouncing reduction in the model updates, and low processing load for updates. Table 1 shows a qualitative comparison in these aspects.

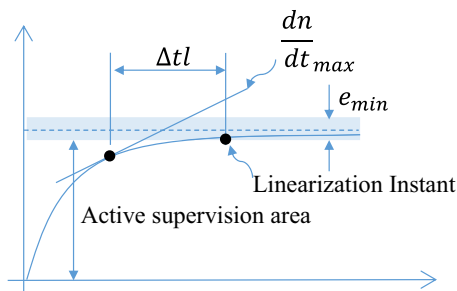


Fig. 26 Linearization scheme measuring the response

Fig. 27 MMPC with update supervising the response

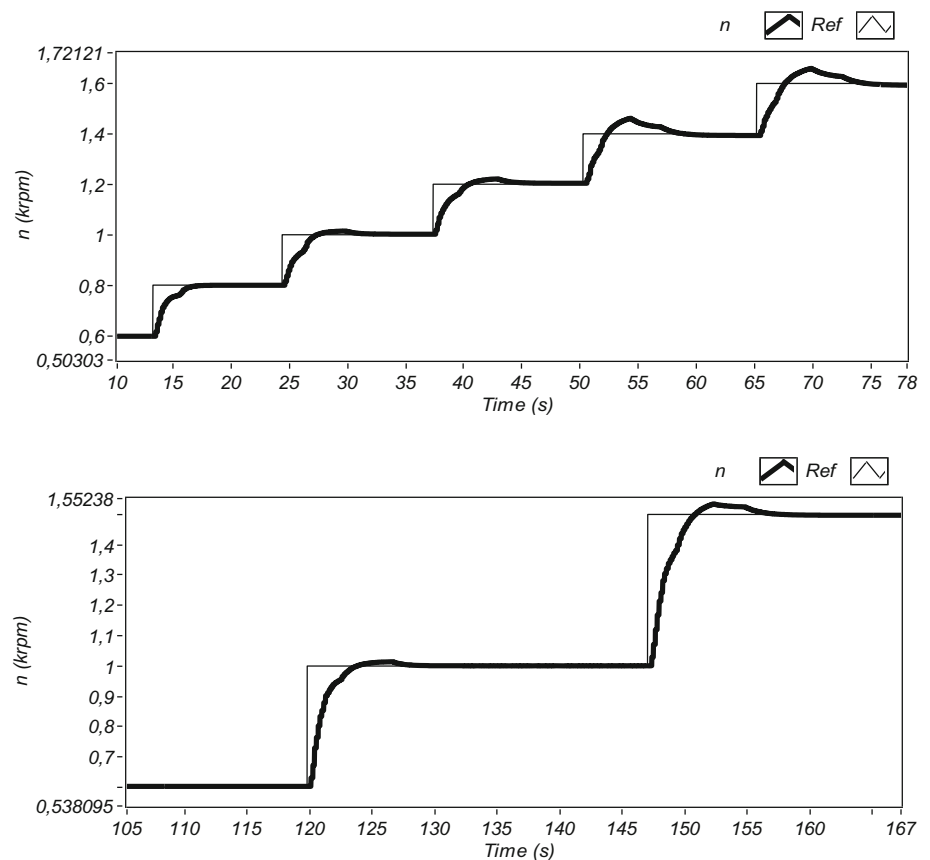
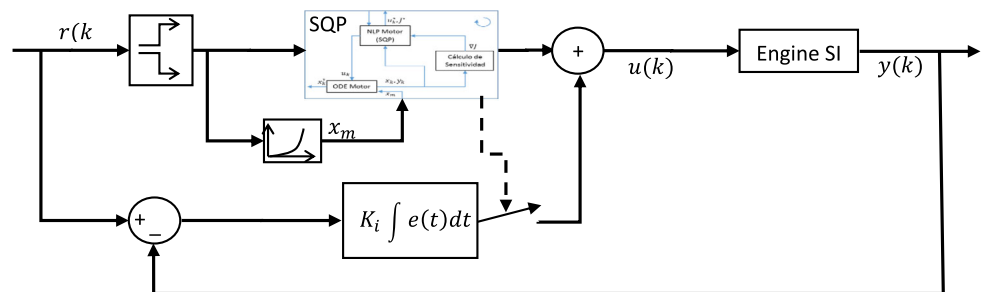


Table 1 Qualitative comparison of updating mechanisms

Mechanism	Monitoring reference	Softness of control	Processing load
Periodic update	Good	Fair	High
Change of reference update	Fair	Good	Low
Change in the dynamics and error in stable state update	Good	Good	Low

Fig. 28 Implementation of NMPC with SQP scheme

The success in the convergence of the Local SQP algorithm depends to a large extent on a good estimate of the initial search values, as was established in [4]. The MPC control structure uses current measurement values of the state to define the initial values to formulate the next sampling period. However, for the MPC the search surface is quadratic and with linear constraints in a close neighborhood with the operation point, whereas for the NLP algorithm the surface is not quadratic, even less so for the linear constraints, and also the search space refers to the whole range of operation where the variables are defined for the problem.

The result then is that, depending on the initial point; convergence can take more or less time, which is prohibitive for real-time control systems. In order to comply with the necessary determinism, a modification of the usual procedure that the MPC uses to formulate the optimization problem on each sampling period has been suggested by using an estimation function of the optimal initial point instead of measuring the state. The function is dependent on the speed reference and has been obtained experimentally. This has guaranteed convergence times below the time of response of the plant. Besides considering the free-of-disruptions system and the correct correspondence between the model and the plant, a single NLP problem is formulated and solved when the reference changes. This lessens the processing load that would take place by solving an NLP problem in each sampling period.

However, the NMPC needs to face the problems inherent to uncertainty and disruptions by becoming robust. According to literature, a robust NMPC can be built using the right weighting of the terminal cost function and adding terminal constraints, using prediction horizons long enough, and also combining it with compensators, among others, as mentioned in the section Robust Performance Considerations section. The last alternative to evaluate disruptions, mod-

eling uncertainty and error elimination of the stable state for speed reference monitoring has been considered by developing a strategy similar to the one stated in [29], but without eliminating the state.

Figure 28 shows a scheme of the NMPC with SQP implementation adding a compensator to confront disruptions, variations of modeling parameters, and noise measurement. A two-direction flank detector for the speed reference determines a call to the SQP algorithm. Based on the current reference, an experimental function obtains the initial state used by the algorithm to start the optimal search. Once the algorithm converges, it updates the state of the actuators and remains in this latter state until a new change of reference takes place. This is essentially an open-loop control action. An integral action in parallel works as a compensator of the disruptions that are not corrected by the open-loop NMPC. It only works on the updating of control actions by the SQP namely being inhibited before the SQP algorithm convergence. The final performance of the scheme implemented on the 2-HIL platform presented in [32] is shown in the result section.

4 Results

4.1 Quality testing: MMPC with online updating

In order to verify the whole functionality of the NMPC algorithm based on the operating point linearization, operating scenarios have been established which seek to discover all the real possibilities by covering the total operating range, verifying constraints and changing the optimization objectives through the use of weight matrices of the objective function. In the following, the tests with the best results with the remarks on the appropriate adjustments are presented.

Figure 29 shows the monitoring of the speed reference in the lower half of the operating area with a prediction horizon of 10 Ts and a control horizon of 1 Ts, and operation limits of $0.5 < n \text{ (krpm)} < 4$, and $0.7 < AFR < 1.3$.

In this graph, the term AFR refers to the normalized λ air-fuel relationship. Only the speed error has been weighted, which defines an intentionality of exclusively energetic optimization; however, note the respect for λ levels, which keeps emissions within the limits of emissions. The controller uses the third updating mechanism given in Table 1. Figure 30 shows the behavior

of the control action, levels for actuators are kept in $16.5 < \alpha \text{ (}^\circ\text{)} < 60$ and $0.00015 < m_{iny} \text{ (kg/ms)} < 0.003$.

The response in the upper areas operation (Figs. 31, 32), shows the controller's compliance with the highest level of speed, when the last observable change of reference is established to 4.5 krpm. λ response is also kept within the specified range, although bordering the lower limit, which indicates fuel-rich mixtures and is evidence of the high consumption. The controller has to keep the speed at the specified limit, as shown at this precise event.

Fig. 29 Reference Monitoring in low areas of operation

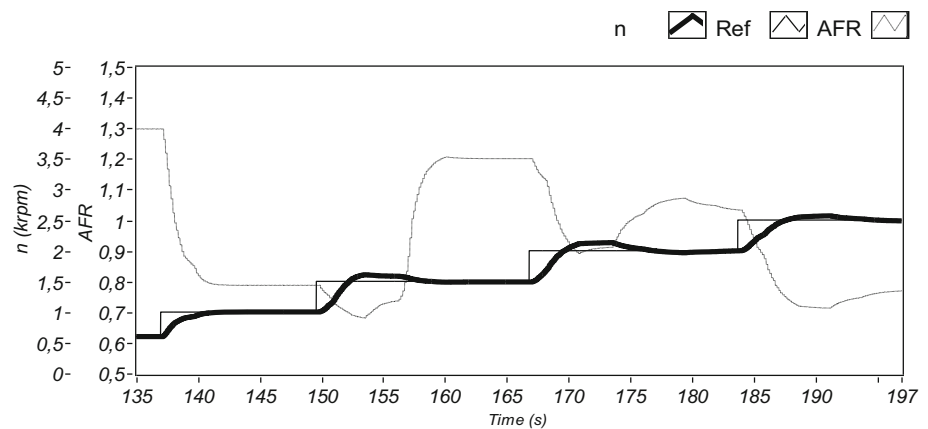


Fig. 30 Behavior of the controls actions

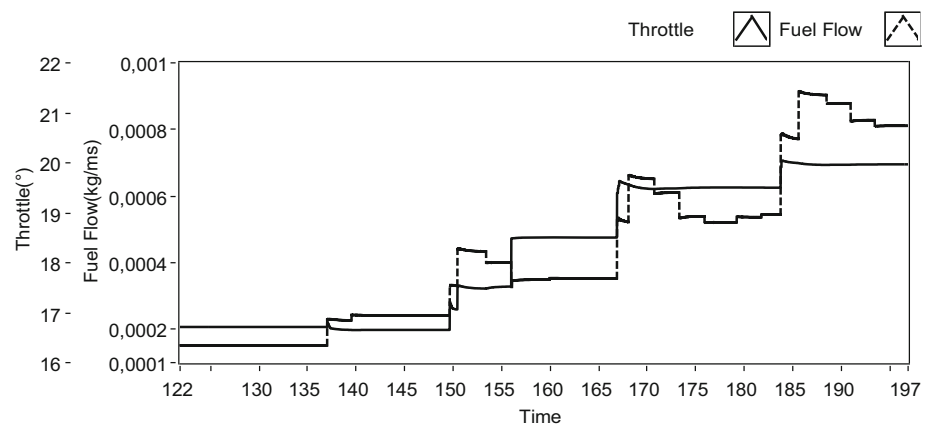
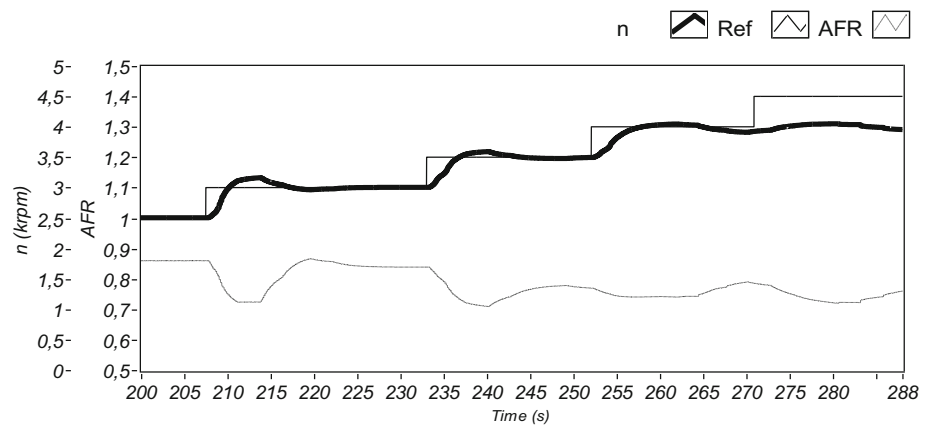


Fig. 31 Reference monitoring in high areas of operation



For the following test, the weighting matrix of the output error was modified and the speed reference and λ monitoring were both equally weighted, implying with this a simultaneous objective for keeping optimization with a fixed reference of 1 $\lambda_{Ref} = 1$ from both power and environmental points of view. An increase in the initial overshoot to the change in the speed reference is observed; however, it is kept below 50 rpm (see Fig. 33).

The minimal error under a stable state is kept and the monitoring to λ fixed reference is successful and stable.

Movements from the reference occur only when there is model updating, and these movements do not overpass 0.15 deviation. Likewise, the levels of the control action are kept as evidenced in Fig. 34.

Suggesting a merely environmental objective involves only weighting the monitoring error of λ_{Ref} , but turning off the speed reference monitoring is impractical, only done in cases like in idling, as shown in the test presented in Fig. 35. The optimal controller has been initiated in time 30, succeeding in establishing AFR in its reference, but the speed

Fig. 32 Control actions in high areas of operation

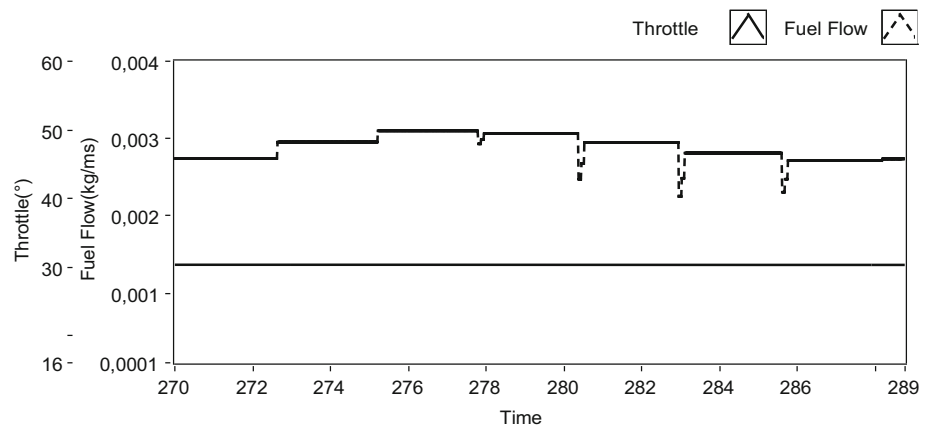


Fig. 33 Results compared with simultaneous objectives operation

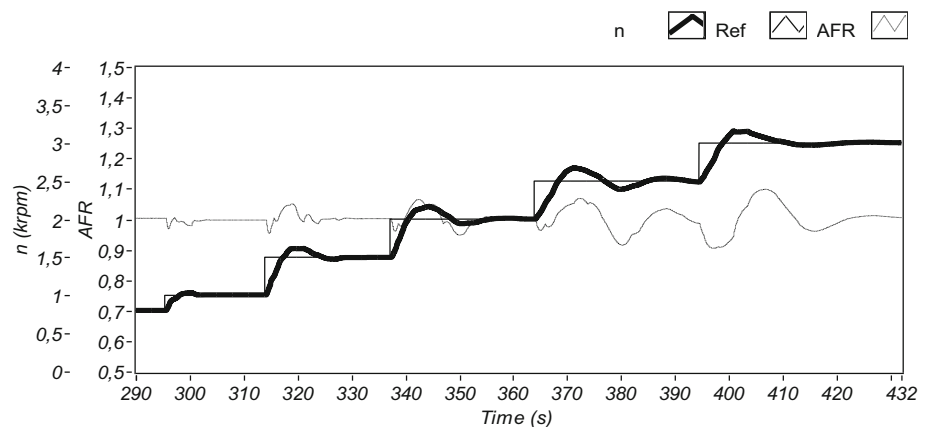
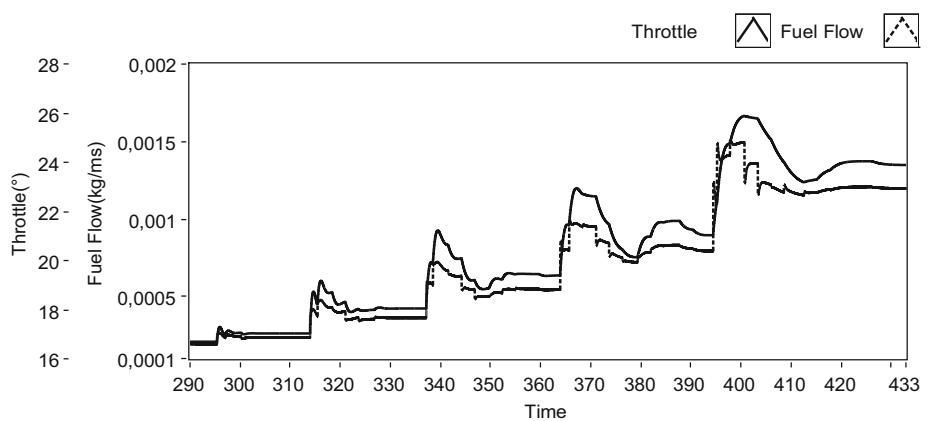


Fig. 34 Control actions with simultaneous objectives operation



is established at the typical idling speed which differs from the given reference, and for the next changes of speed reference, the engine speed is kept the same. On the other hand, when the environmental objective should be imposed over the power objective then the weighting matrix could be such that the weight of the speed monitoring error is lower than the Lambda monitoring. Relative magnitude between these weights defines the quality of the environmental objective. For example, Fig. 36 shows the test of the optimal controller assigning a weight of 50 to Lambda reference monitoring and 1 to the speed reference monitoring.

A more stable and softer behavior is observed in Lambda in relation to the simultaneous objective test, monitoring Lambda's reference, and it also monitors the speed reference although with lower quality (higher overshoot) and with a consumption of resources higher than that observed in the graph of control actions to the right, where the fuel flow can be measured above 0.002 kg/ms, which does not happen in the simultaneous objective experiment.

Another scenario that targets an economic objective with the reduction of consumed fuel in the engine operation is conducted by adding weights to the matrix R of the objective

Fig. 35 Results with merely environmental objective operation

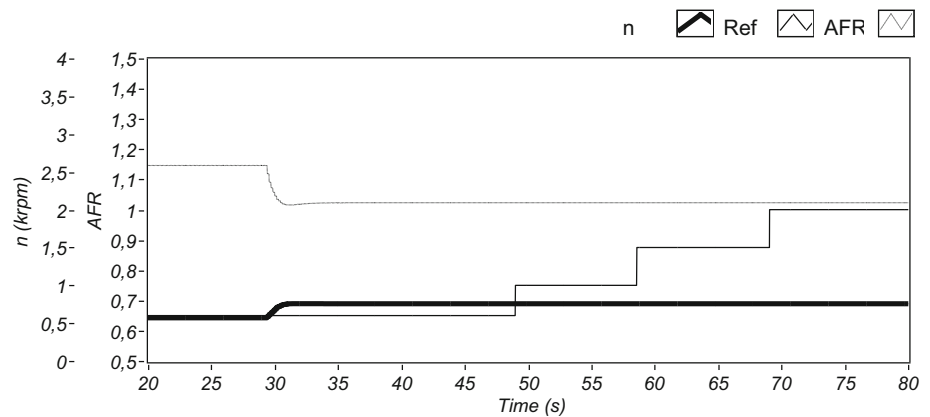


Fig. 36 Controller with environmental objective

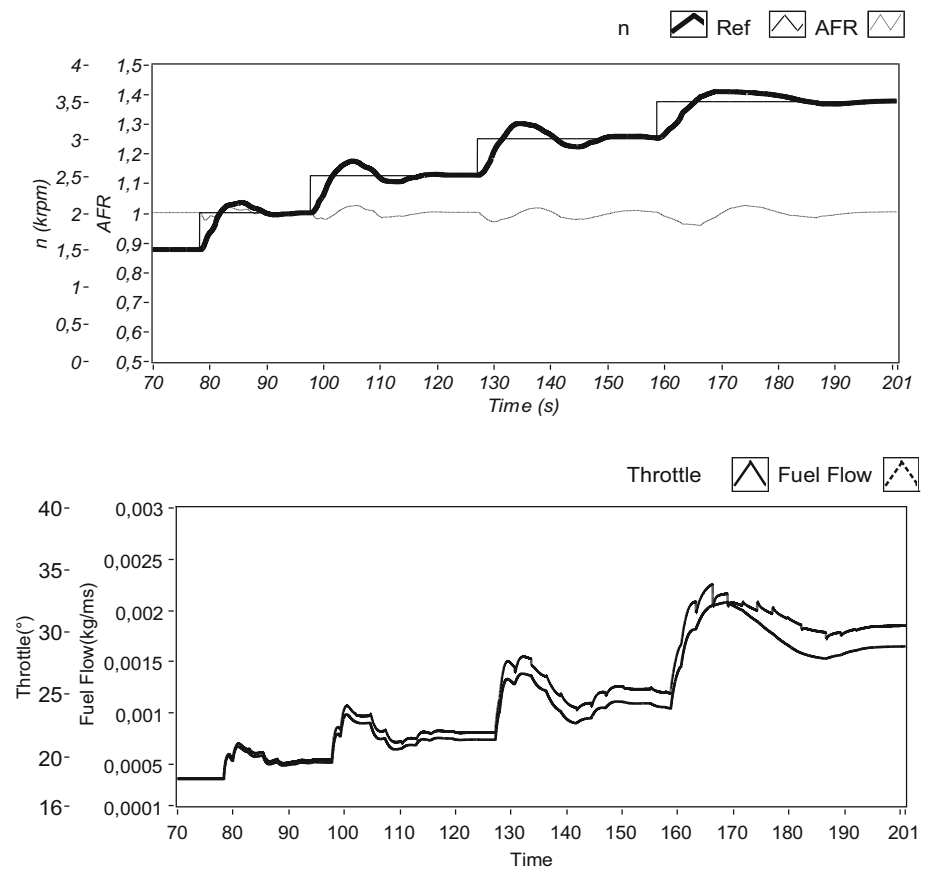
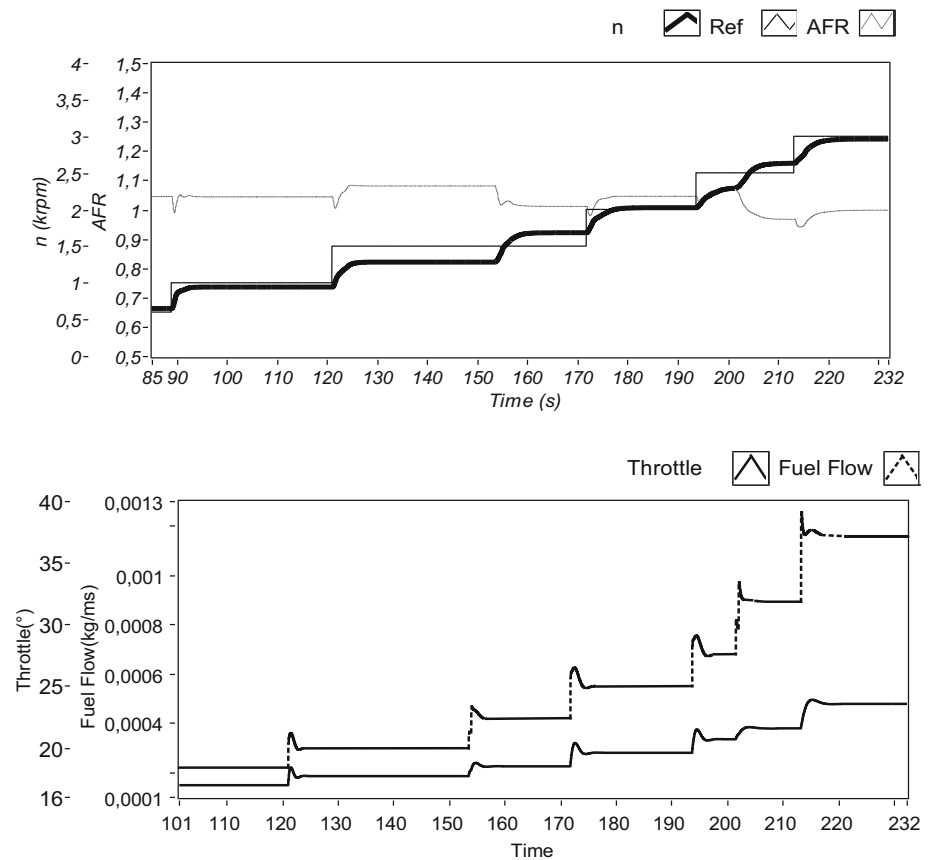


Fig. 37 Controller with economic objective



function that weighs the actions of the optimal controller. By assigning a weight of 1 to the opening of the throttle valve (required to avoid turning the control action off, although it is not financially significant) and 5 to the fuel injected, the responses shown in Fig. 37 have been obtained. The weights in the speed reference monitoring and λ_{Ref} have been both kept to 1.

The error in the stable state in speed worsens notoriously, even though there are operating areas with good behavior. The relation fuel-air ratio is kept within a very narrow margin (± 0.1) around the reference. The fuel flow is evidently kept low, below 0.0013 kg/ms, lower than the consumption of the previous scenarios presented. Likewise, transitions of the control action become softer and with fewer changes, which results in a more relaxed commuting operation in the injectors and positioning of the throttle valve.

4.2 Quality testing: MMPC with SQP

The results of the control system implemented with the embedded linearization strategy that provides the SQP algorithm and the single shooting method are presented under the same test scenarios. Figure 38 shows the behavior by designing an energy objective weighting the speed reference monitoring and avoiding λ reference monitoring. Effective-

ness in the objective is proved and the suggested levels are kept for the other variables.

Delays in the engine response as well as a stable state error in the upper area of the operation are evident. These delays are caused specifically by the time the algorithm takes to converge. The speed of response depends on the natural dynamics of the engine, since the transient response is free or in open loop, which is also the reason for the softness found.

Figure 39 shows performance for a power and environmental simultaneous objective tested in the range of operation of the engine. This test has shown better results for monitoring of the speed reference, less initial delay, and the error of λ is kept below 0.1. The effort of control increases slightly in terms of the throttle angle, but this is irrelevant concerning consumption of resources.

A merely environmental objective generates satisfactory results also from the financial point of view, as might be seen in control actions (see Fig. 40) where fuel consumption reached levels lower than 0.002. λ is narrowly kept around the normalized reference, and the speed error strongly increases, being optimal only in operation areas around 3 krpm.

An economic objective by minimizing the fuel, but at some point with speed reference monitoring, did not result in substantial improvement in fuel consumption. The fuel flow has reached more than 0.0025 kg/ms, which is the same as for

Fig. 38 Controller NMPC with SQP, power objective

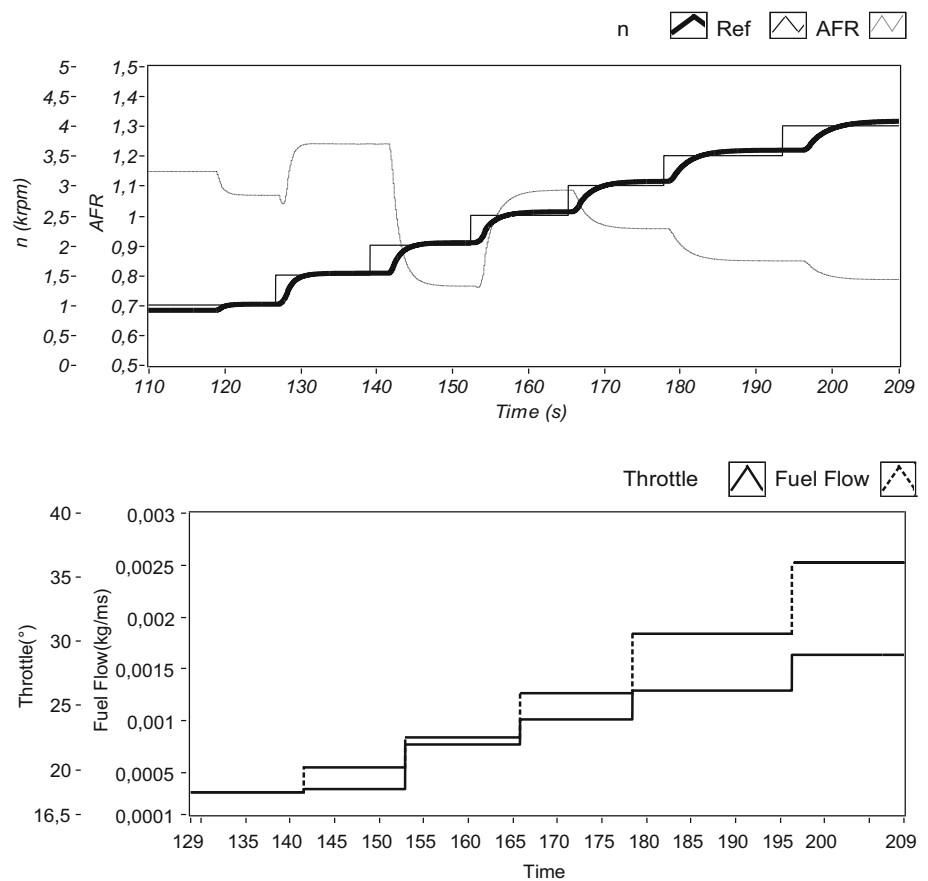


Fig. 39 Controller NMPC with SQP, simultaneous objectives

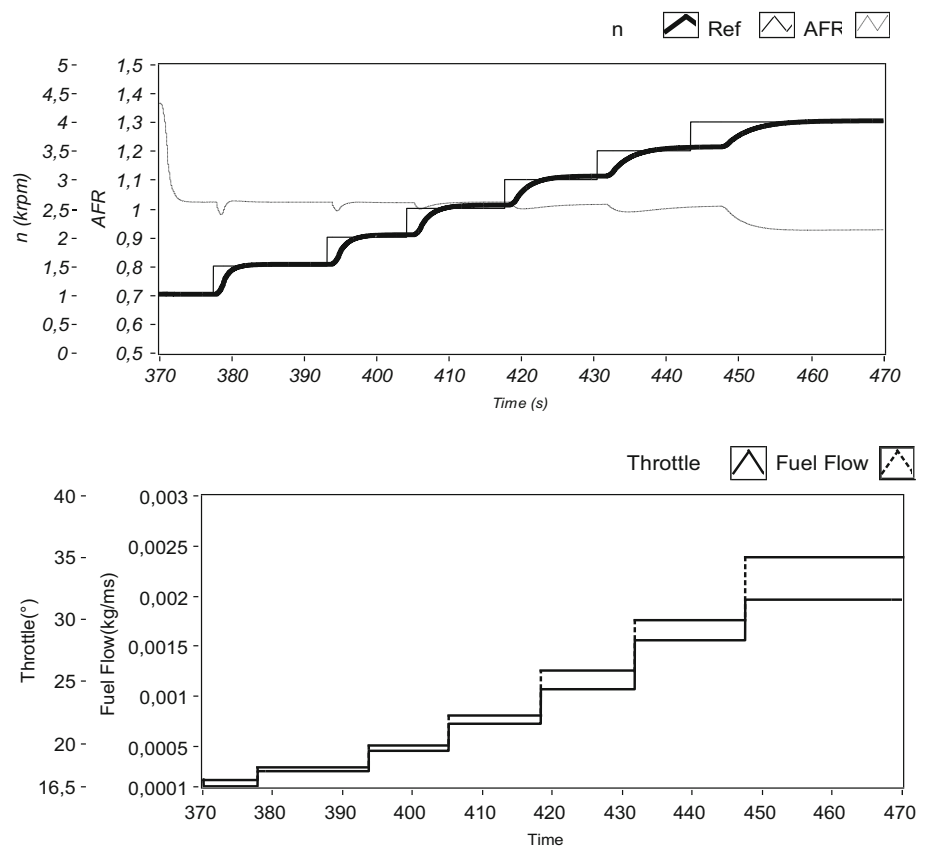
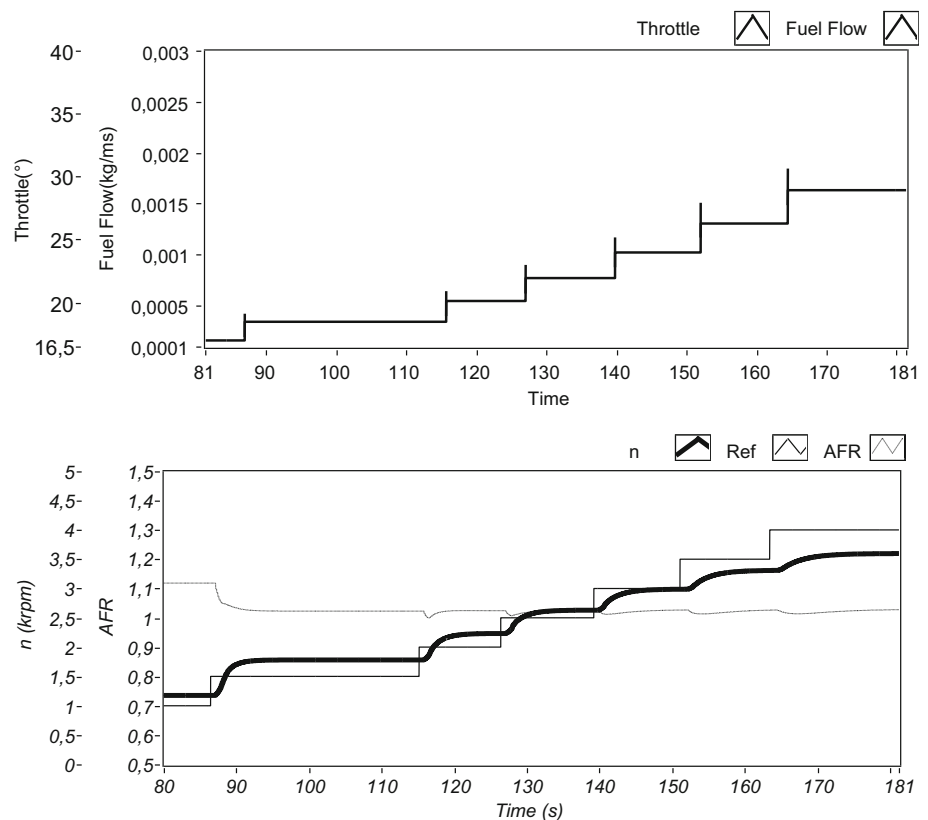


Fig. 40 Controller NMPC with SQP, environmental objective



tests in which weighting for the injected flow was zero. The levels of the air-fuel ratio were not kept and an error of stable state in the speed monitoring occurred. These results are shown in Fig. 41.

4.3 Stability tests: MMPC with online update

In discussions presented in literature on stability and reviewed in previous sections of this report the inherent stability of the MPC controller has been relied on with the reliability of a prediction horizon long enough to consider the natural dynamics of the plant. Since the plant is naturally stable, this simple method has allowed to experimentally keeping the stability of the system in closed-loop. Different prediction horizons have been applied in order to test the stability of the NMPC on the combustion engine. The results are shown in the following graphics (Fig. 41).

Regarding the views given in [25], a finite prediction horizon implies stability, but from a practical perspective, a problem of infinite dimension does not have a solution. The opposite could then imply instability due to poor prediction, which generates incorrect control actions, similar to the responses obtained in Figs. 42 and 43, with prediction horizons of 2 and 5 Ts, respectively.

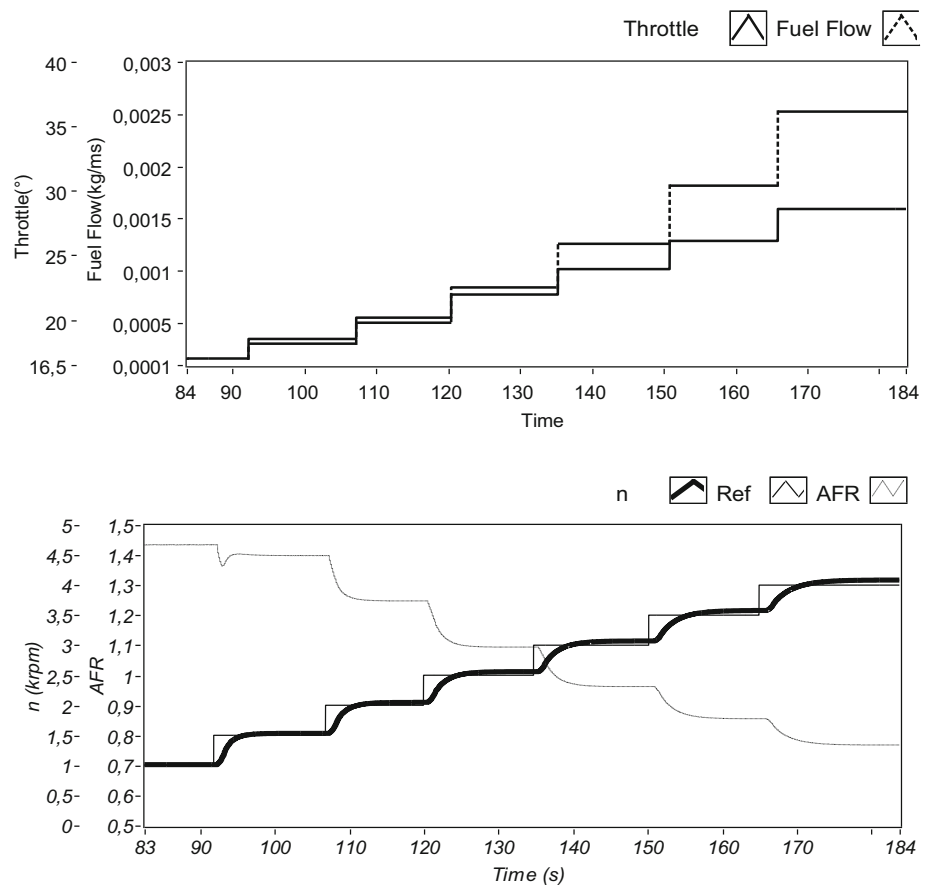
An apparent stability remains in the first scenario for speed, but variations in the air-fuel ratio are inappropriate and characterized by unexpected changes in the fuel flow

control signal. By broadening both the prediction and control horizons, the result was total stability of the process. To keep the control horizon to 1 Ts and increase it to 10Ts results in undeniable improvement of stability (see Fig. 44). An increase to 100Ts has not only improved stability, but also the optimization performance indexes and softness of the controlled variables and their control actions as seen in Fig. 45.

4.4 Stability tests: NMPC with SQP

With the single shooting strategy, the differential equations solver solves the model starting from the initial conditions until a future time given at each SQP algorithm iteration. A very broad prediction horizon guarantees higher observability of the system behavior against the tests taken by the SQP algorithms for the control actions, which results in higher accuracy of the optimal control, but it also implies a superior convergence time. A very short horizon can lead to biased control actions.

Figure 46 above and below show the horizon inference on the quality of the control system response. An equal weighting that corresponds to the dual power and environmental objective has been used for the three tests. Values over 20 s (Fig. 47) resulted in convergence times which are inappropriate for the engine operation. The horizon experimentally selected as appropriate has been that of 20 s.

Fig. 41 Controller NMPC with SQP, economic objective

4.5 Robust performance tests

To carry out a qualitative experimental analysis of the controller robustness, test scenarios adding disruptions, modeling parameter deviation and noise on the sensor signals have been considered. An inherent robustness of the MMPC with linearization is found in relation to the updating mechanisms of the model. Measurement of the stable state error, which can imply non-correspondence of the current model with the plant, allows the subsequent updating of the model to a less biased model against the current one. This adds robust behavior to the control loop. However, uncertainty in the parameters of the non-linear model also adds to the non-correspondence of the model calculated by the linearization algorithm. In the following, the results of both types of controller against the various definitions that add uncertainty are presented.

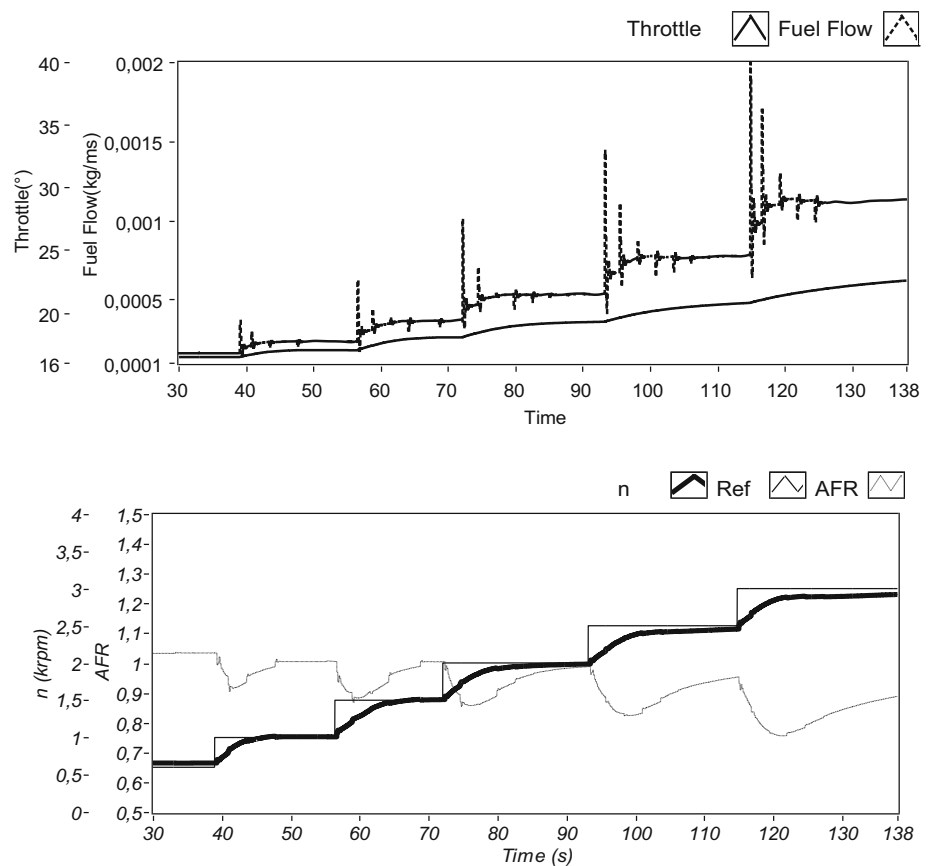
4.6 Adding disruptions

The possible disruptions regarded in the combustion operation system, which are within the limits the controller absorbs, can be generated inside the process as sudden changes of load in the crankshaft, or also as disruptions in

the actuating or operating signals of the actuators, which results in sudden changes in the fuel flow and the throttle valve position. In a final application, the load in the engine is not constant. In fact, if the engine is a vehicle's drive, variations in the rhythm of the paths represent a disruption to the load torque, the same as a change in the number of passengers or the material the vehicle carries. Sudden effects on the fuel pump pressure appear occurring due to changes in the battery voltage, which can vary when a connected electrical element is turned on or off. However, the result of a drop or increase in pressure translates into a disruption to the fuel flow (this is how it has been handled in this report). Such flow disruption can also occur due to obstruction in the line, due to dirt in the injectors and other elements of the fuel system. Similarly, the electrical fluctuations can affect the performance of the positioning engine of the throttle representing a sudden change in the opening angle.

In the model, the load has been represented as a varying counter-torque described by the load-power on the basis of the speed $P_L (kW) = k_L n^3$, where k_L is the constant of the load experimentally obtained. Therefore, the variation of k_L , has been used for the tests of load fluctuations on the axle. Figure 48 shows the effect on the variation of k_L in

Fig. 42 Controller MMPC with $N_p = 2$, $N_C = 1$



relation to the nominal load of the model with $k_L = 0.47$. Loads with coefficients lower than 1 are values very close to working under vacuum conditions and loads with coefficients higher than 2 represent a very big load for the engine in use, whose maximum delivered power is given by the specifications in 55 kW. Moving a load whose coefficient is 2 implies a power consumption of 50 kW to 3 krpm as shown in the graph.

Accordingly, disruptions have been limited in terms of k_L , checking for different speeds of operation due to the power changes required by this variation.

The robustness tests for the MMPC with an online update are then shown in the following Figs. 49, 50, 51, 52:

All tests show satisfactory behavior of the controller to recover the stability and the desired position of the state. It should be noted that at 4 krpm, stability recovery became much slower than in the other tests and close to finding instability.

Changes to a $k_L = 0.1$ y a $k_L = 2$ on the NMPC with SQP were made as shown in Fig. 53. The effect of reducing the load below the nominal has had a higher impact than when it is increased. In this case, the corrective effect is provided by the compensator and not directly by the algorithm. However, the resulting modifications in the variables are minimal by keeping optimality based on practical conclusions. Note, for

example, that λ reference monitoring improves after disruption has been corrected.

Disruptions in the opening angle of the throttle have been considered by measuring the controller robust performance against complete openings and sudden complete closures of the valve. Figure 54 shows the effects of disrupting the control action with the opening to the maximum level of 60°, and then with the closure to the minimal level of 16.5° with the MMPC controller. It is observed that the most disrupting effect corresponds to the closure and specifically on Lambda, and stability is soon recovered. A similar effect, slightly more disrupting is noticed in the NMPC controller with SQP, as shown in Fig. 55.

Disruptions in the control action in the fuel flow have also been considered within the operation limits of the actuators and levels in the control actions, so sudden movements up to the maximum flow of 0.003 kg/ms and the minimum possible of 0.00015 kg/ms were generated. Figure 56 shows in a similar way to the previous test how the most disturbing effect is the flow reduction, in which the variables get out of control. Movement towards the maximum flow is soon recoverable by the controller for both variables.

A delimitation of the flow disruption up to 0.0003 kg/ms has resulted in a satisfactory recovery of the engine operation; however, it has been observed that delimiting the disruption

Fig. 43 Controller MMPC
with $N_p = 5$, $N_C = 2$

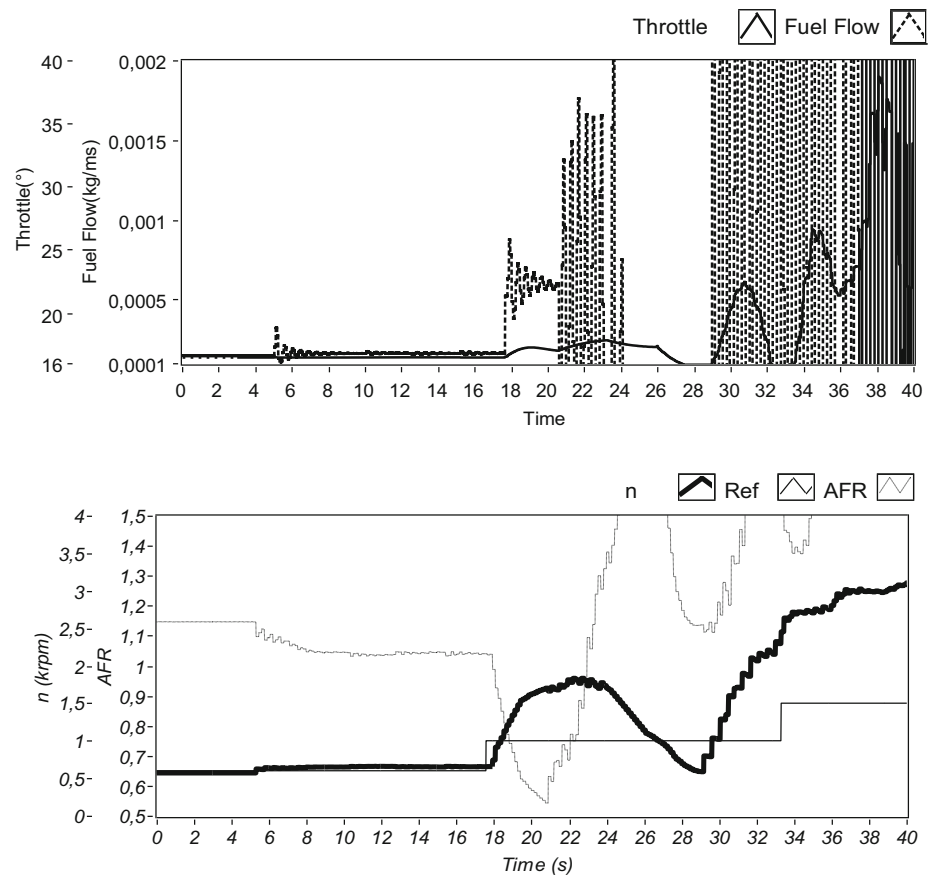


Fig. 44 Controller MMPC
with $N_p = 5$, $N_C = 2$

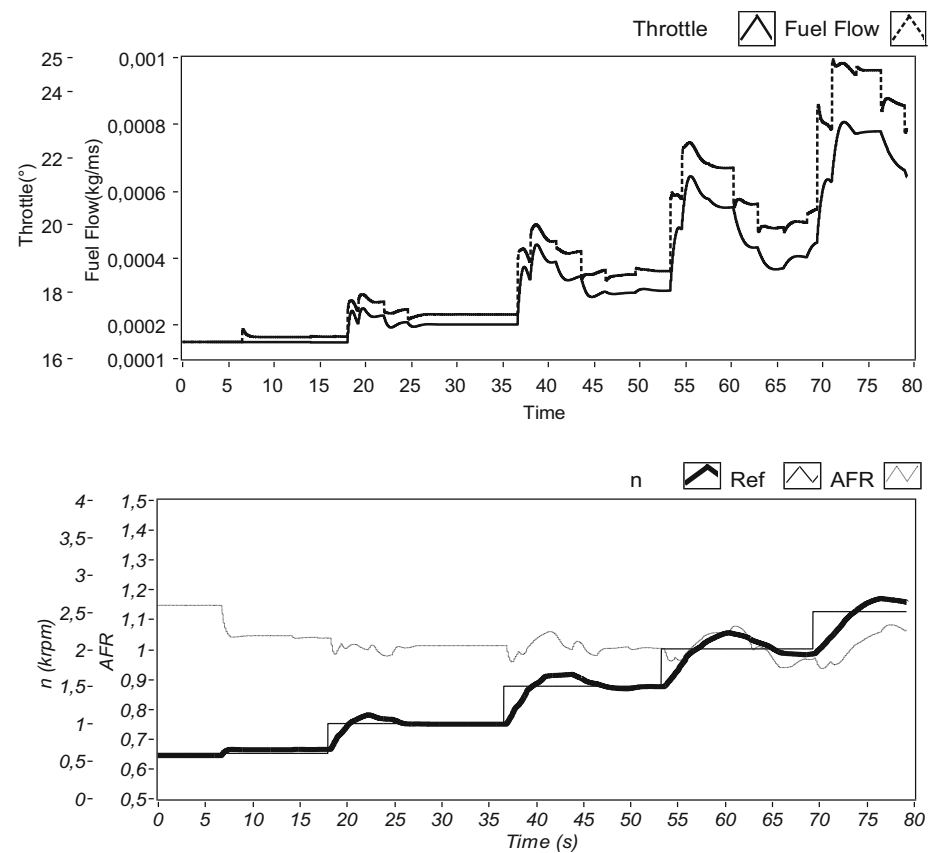


Fig. 45 Controller MMPC with $N_p = 100$, $N_C = 1$

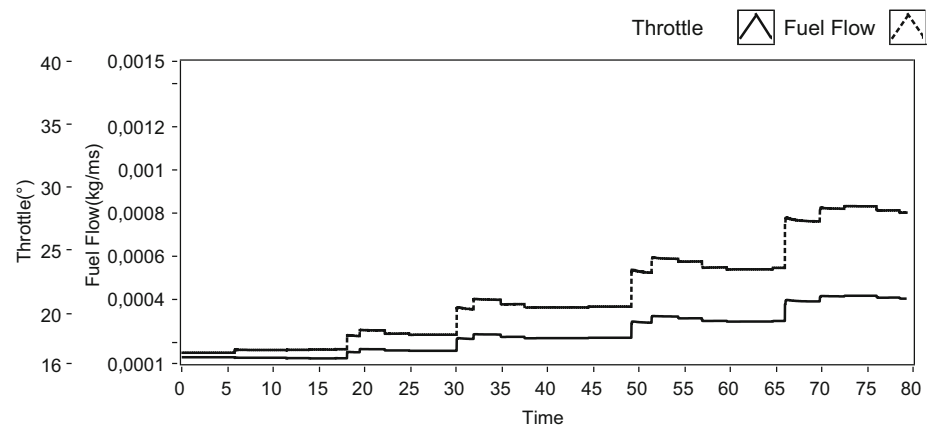


Fig. 46 NMPC with $N_p = 10$ s and $N_p = 5$ s

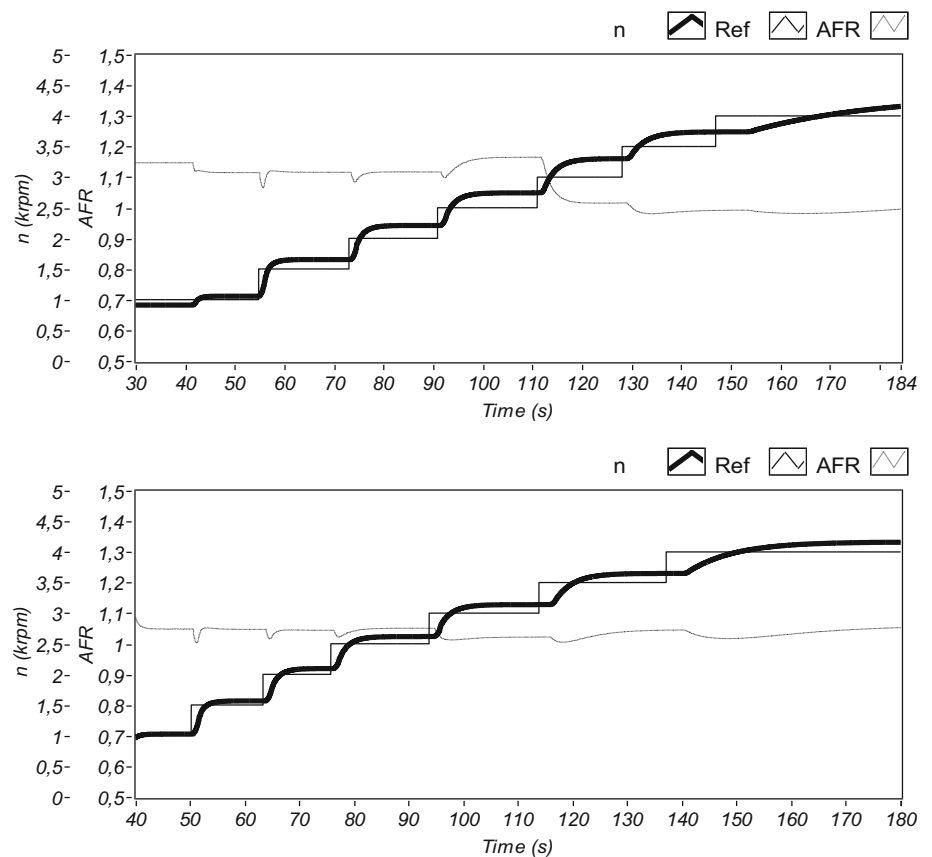


Fig. 47 Controller NMPC with $N_p = 20$ s

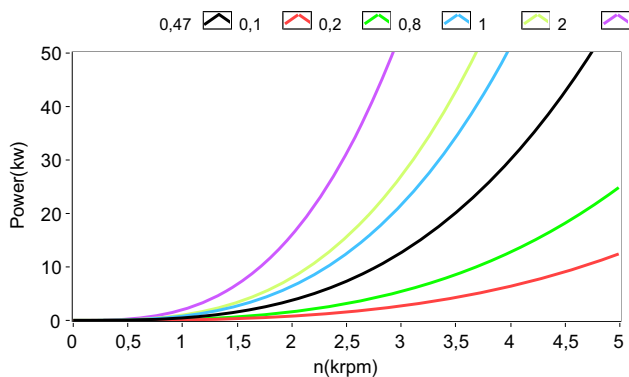
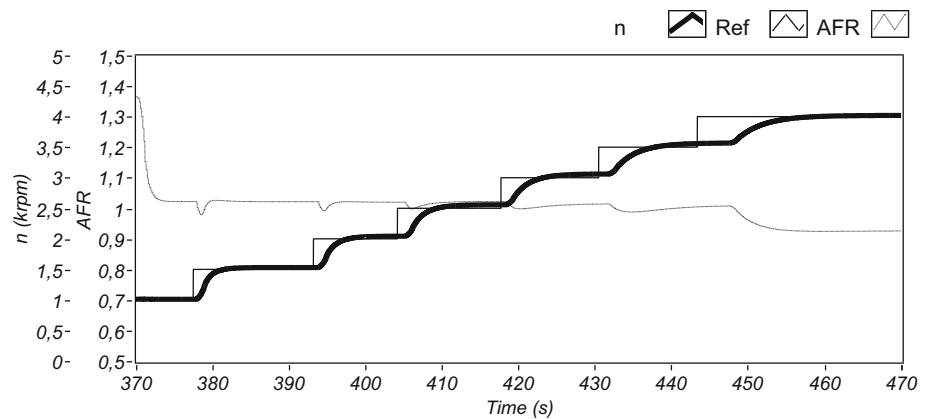
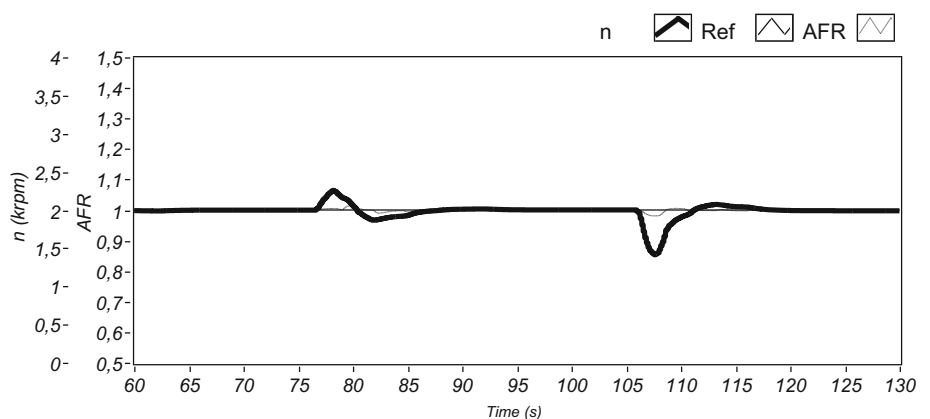


Fig. 48 Behavior of the power load P_L with k_L variation

has depended on the point of operation, especially for the Lambda control recovery. The definition of a relative level of 0.0002 kg/ms below the flow control action has shown better recovery behavior. Figure 57 shows this effect.

The disruptive effect is again more noticeable in Lambda, but also stronger in the low-speed operation areas than in the high areas. This is mainly due to the noticeable gradient in the system behavior in this area of low operation. Broadening of the level has resulted in instability of this area specifically.

Fig. 49 MMPC against load disruptions, $k_L = 0.1$ y $k_L = 2$ in 2 krpm



Tests in the NMPC with SQP have shown similar behavior (see Fig. 58) with slight increase in λ broadening oscillations.

4.7 Deviation of modeling parameters

A significant part of the modeling parameters are represented by the coefficients obtained for experimental curves [2] that describe several processes of the engine dynamics. The variability analysis of the total of coefficients and robustness tests for these parameters is out of the scope of this report, specifically due to its extension. However, since these coefficients are constant and have been validated in [3], it has been assumed that they do not provide modeling uncertainty.

On the other hand, the result of parameters such as environmental temperature T_{amb} , atmospheric pressure P_{atm} , and the fuel heat capacity H_u , not only affect the experimental polynomial expressions, but also other parts of the model. In addition, even though they are considered parameters, it is clear that they can vary depending on the engine location, the variation of environmental conditions and the qualities of the fuels used. The possible variability of other parameters such as displaced volume, Reynolds number, inertia, etc. has been assumed as not significant since they are constructive aspects and/or standardized universal constants.

Fig. 50 MMPC against load disruptions, $k_L = 0.1$ y $k_L = 2$ in 1 krpm

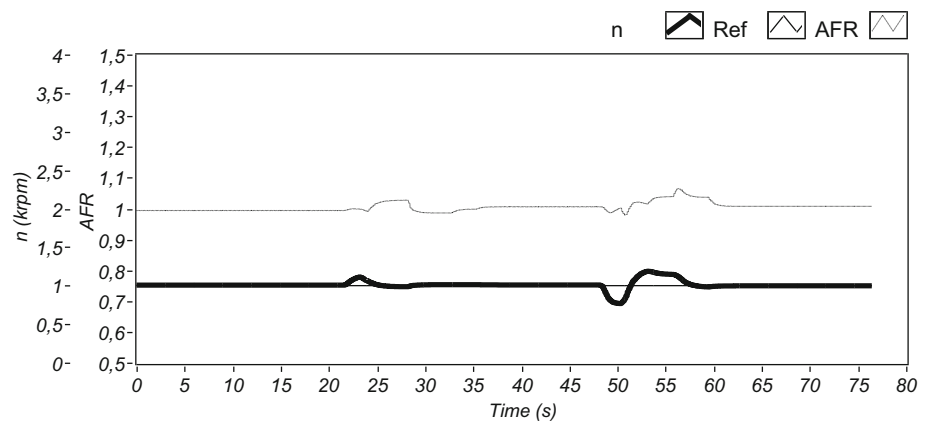


Fig. 51 MMPC against load disruptions, $k_L = 0.1$ y $k_L = 2$ in 3 krpm

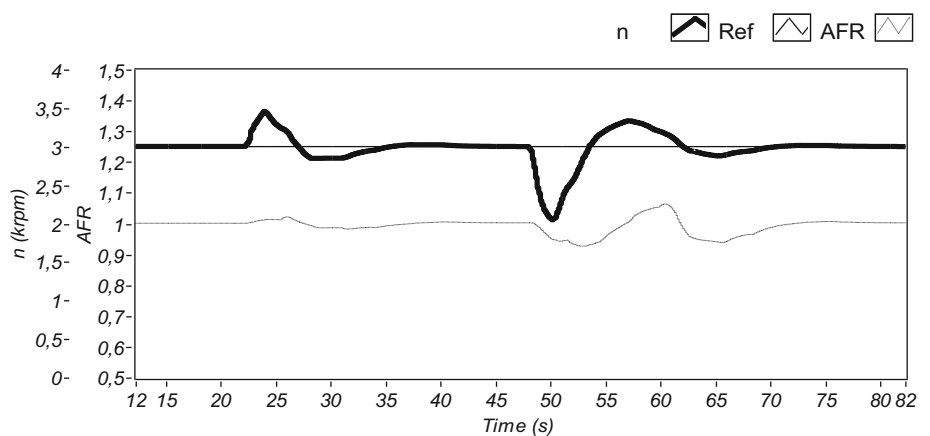
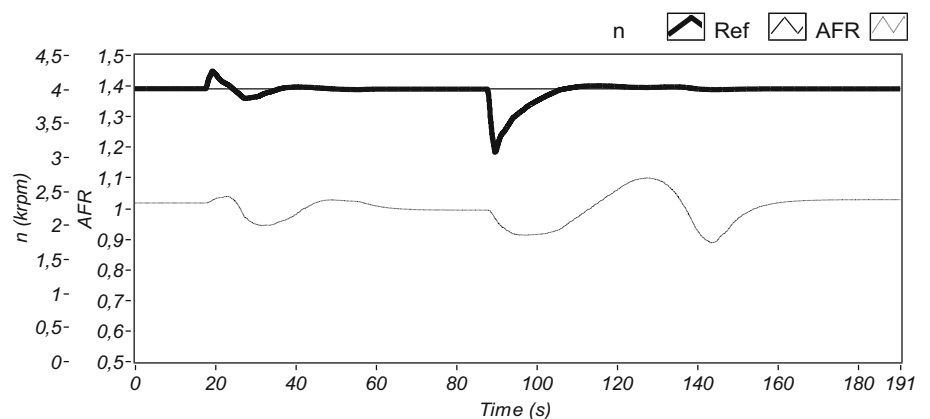


Fig. 52 MMPC against load disruptions, $k_L = 0.1$ y $k_L = 2$ in 4 krpm



The practical variation of the atmospheric pressure has been assumed from a value of 0.65 bars in La Paz, Bolivia, which is one of the highest cities of the world with usual industrial activity, up to the sea level pressure in 1.01 bar. Considering recordings of extreme temperatures in cities with industrial activity, the levels of the environment temperature have been defined as $233.15^\circ\text{K} < T_{amb} < 323.15^\circ\text{K}$; and to check robustness against the deviation of the fuel's lower specific heat capacity, checkups have been considered taking into account the variation of the heat capacities of the two qualities of the gasoline-type fuels found in Colombia

[33]. This means 42,078.73 kJ/kg for the regular type and 42,751.98 kJ/kg for the extra type.

The nominal value of the model for the atmospheric pressure has been taken as the sea level pressure. Figure 59 shows a change of parameter up to the specified limit value of 0.65 bars. The performance of the speed control remains intact, and the Lambda control acquires a permanent error without being destabilized.

The correction of this stable state error happens through a model update, and this implies that the appropriate mechanism to face the model update corresponds to the update

Fig. 53 Disruptions of k_L to 0.1 and to 2 in NMPC with SQP

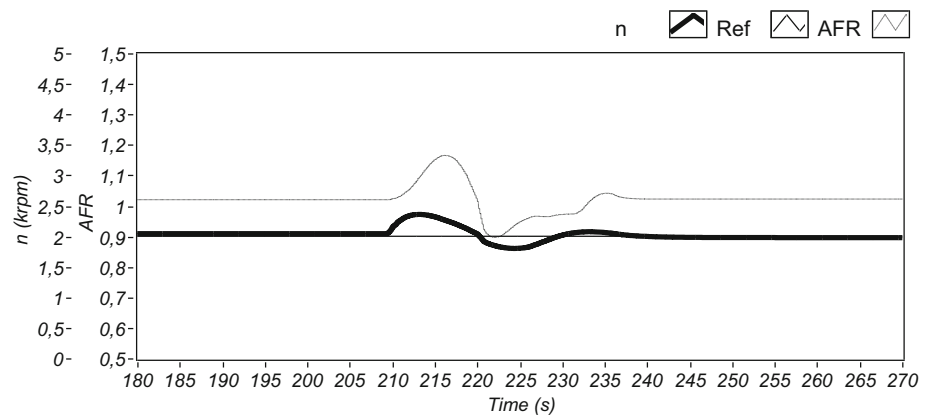
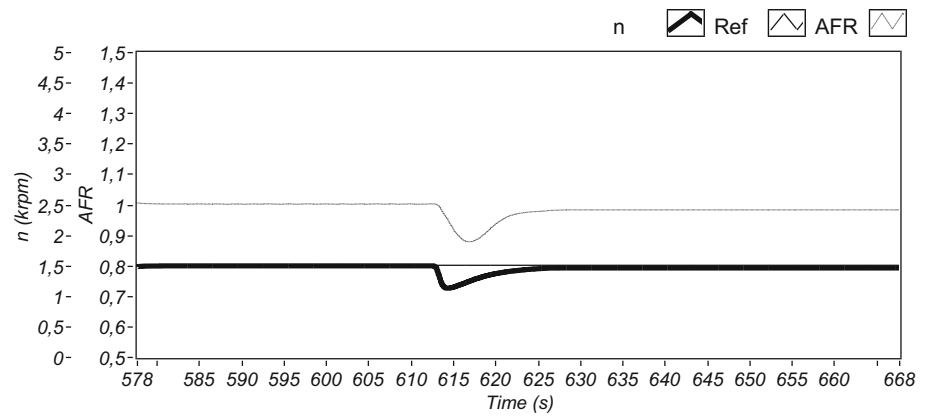


Fig. 54 MMPC against throttle opening disruptions, $\alpha = 60^\circ$ y $\alpha = 16.5^\circ$ 4 krpm

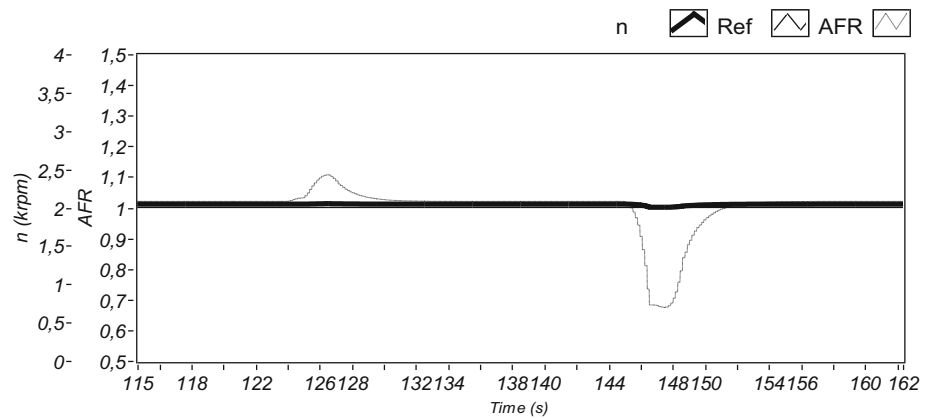


Fig. 55 NMPC/SQP against throttle opening disruptions, $\alpha = 60^\circ$ y $\alpha = 16.5^\circ$ 4 krpm

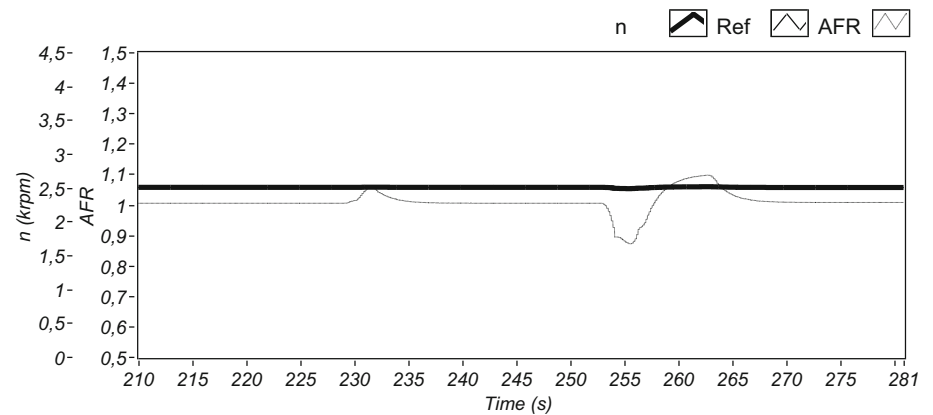


Fig. 56 Disruptions of m_{iny} in MMPC with update

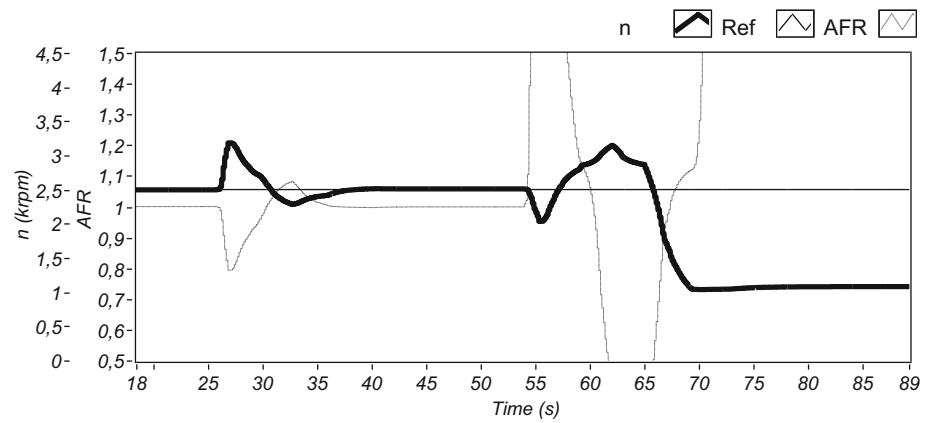


Fig. 57 Disruptions of m_{iny} with relative level in MMPC with update

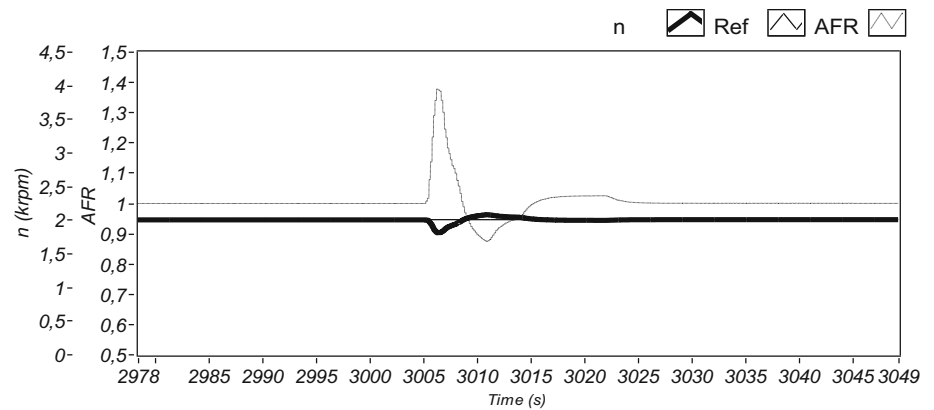
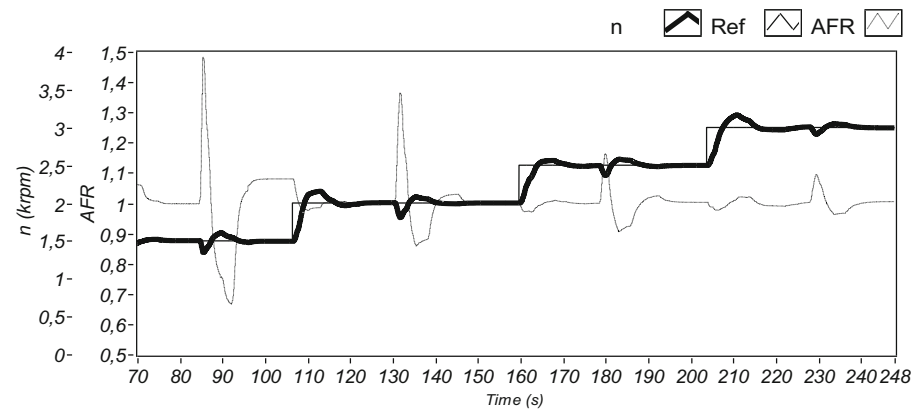


Fig. 58 Disruptions of m_{iny} in NMPC with SQP

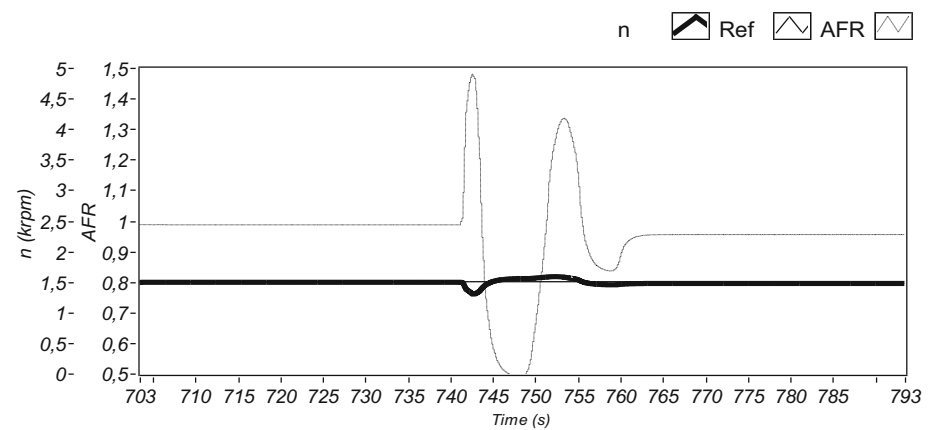


Fig. 59 Deviations of P_{atm} to 0.65 bar in MMPC and NMPC with SQP

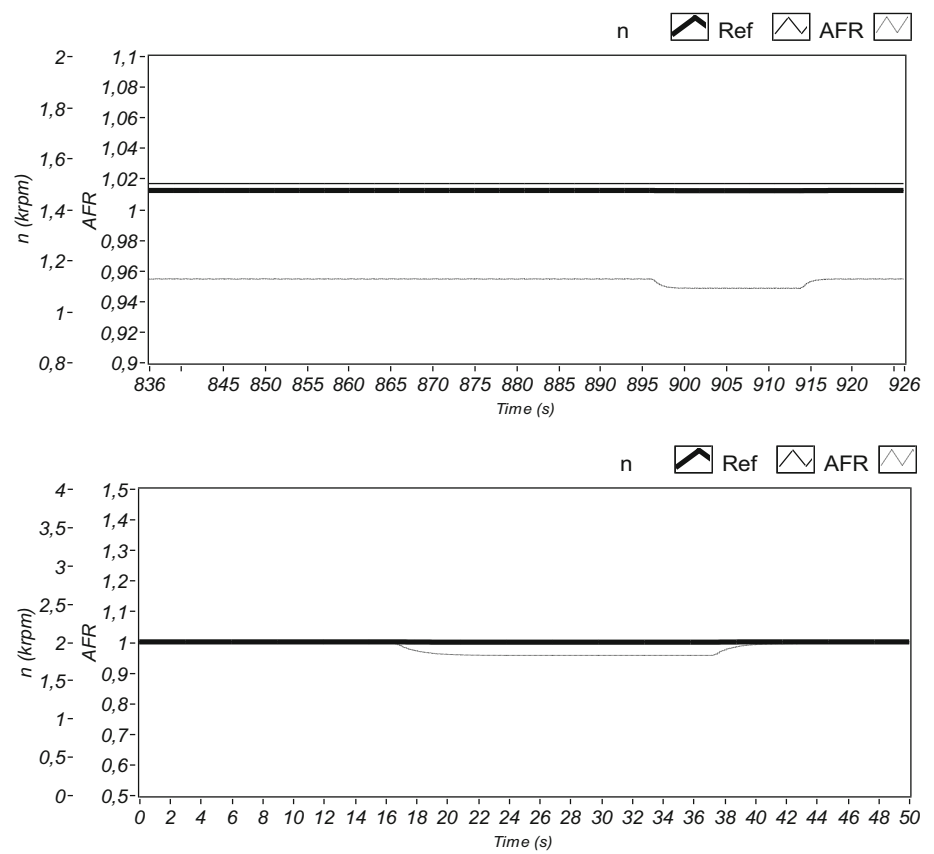


Fig. 60 Deviations of T_{amb} to 233.15 °K and to 323.15 °K in MMPC and NMPC with SQP

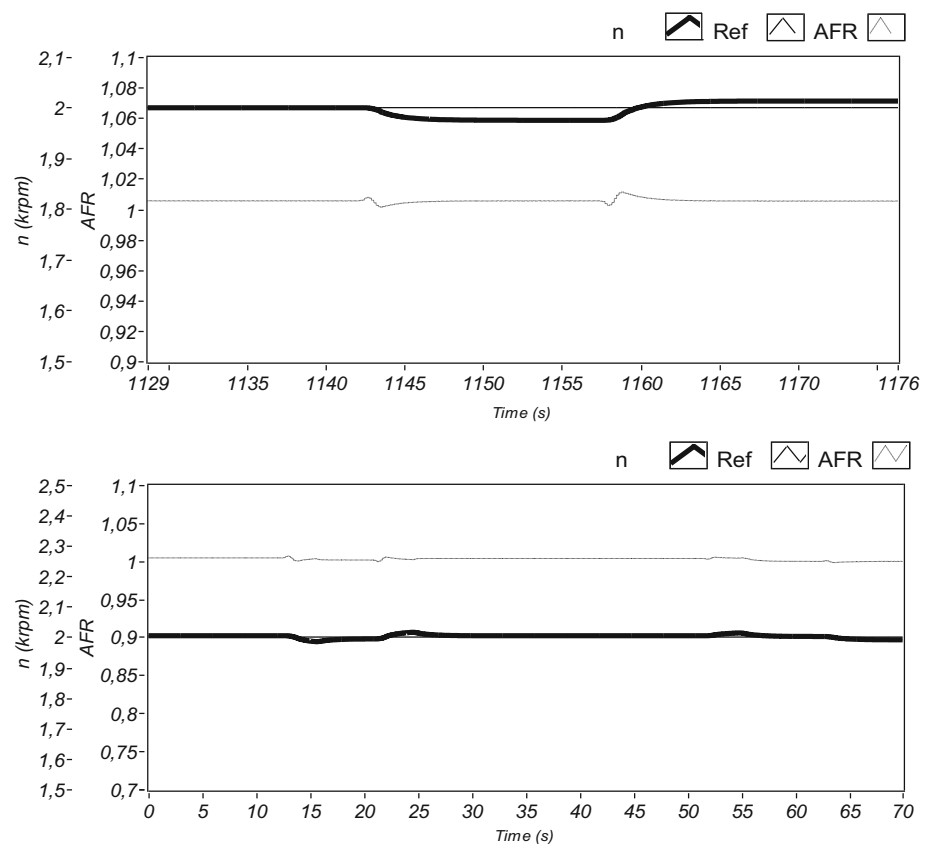


Fig. 61 Deviations of H_u in MMPC and NMPC with SQP

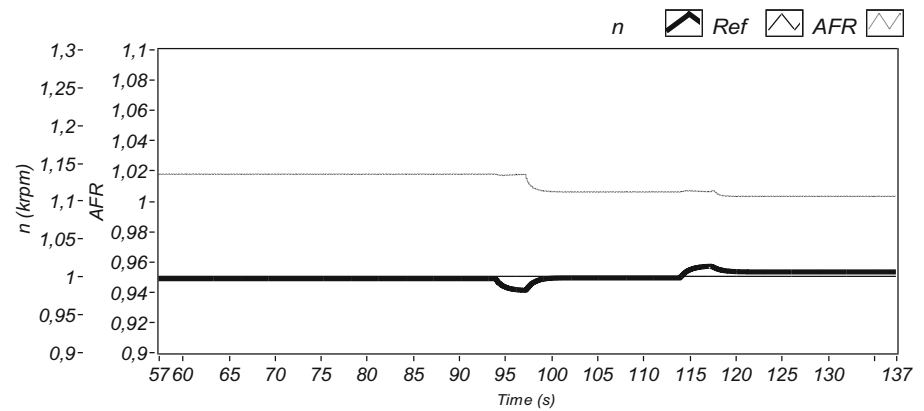
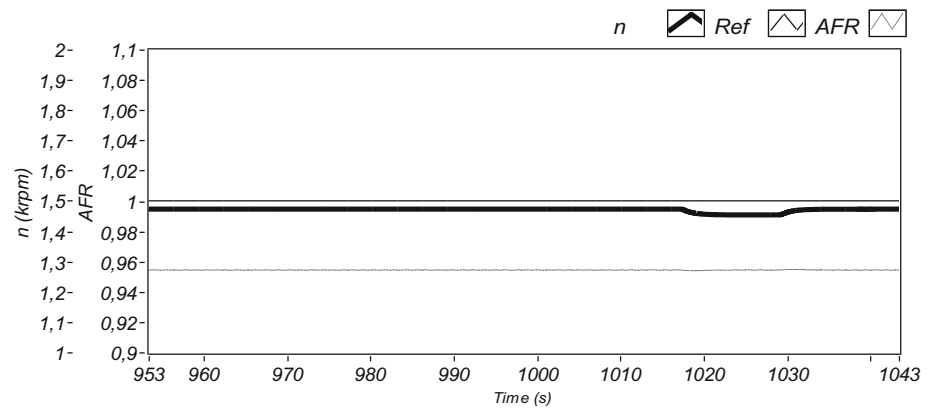
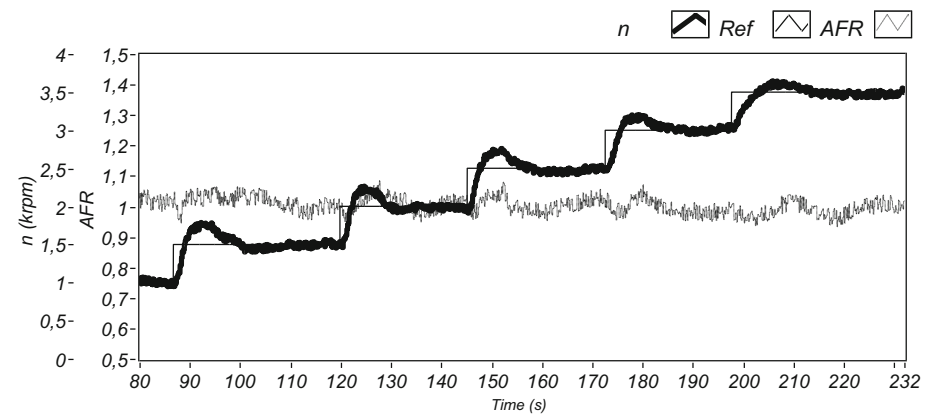
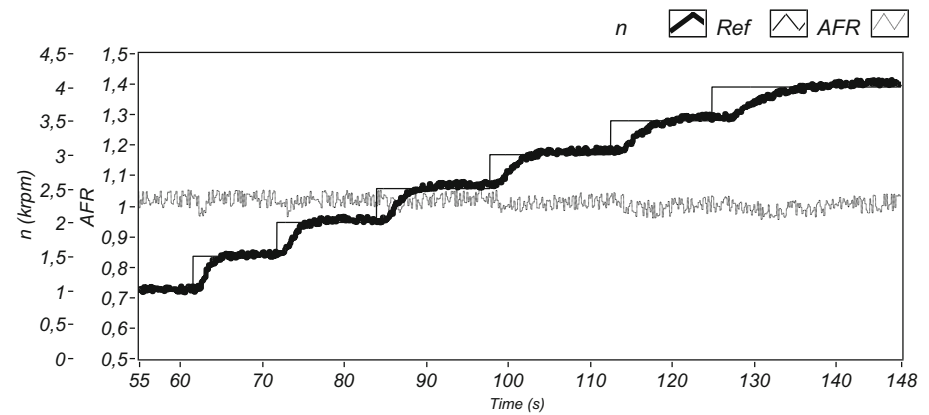


Fig. 62 Performance of controllers MMPC and NMPC with SQP controllers against noisy



in regular intervals, since without error in the speed control the third mechanism in table 3 does not invoke an update. An asynchronous update has been invoked some time later the parameter shifting, which serves to correct the error. The effect on the NMPC controller (in the bottom) is the same, only the Lambda reference monitoring is affected, but at a deviation that is barely noticeable.

Figure 60 at the top shows the results obtained when the nominal environmental temperature established in the model in 293.15 °K is deviated to possible working extremes. The effect is nearly unnoticeable; therefore, the figure has a scale adjustment for better visualization. At instant 12 the environmental temperature has changed to 233.15 °K, in instant 22 it returns to the nominal, and in instant 53 it is taken to 323.15 °K. The figure in the bottom shows the tests using the SQP algorithm inside the NMPC controller.

Similarly, the tests against the changes of the calorific value of gas assumed from a practical context have been satisfactory. Figure 61 shows the effect of changing the calorific value in instant 93–42,078.73 kJ/kg and then next in instant 113–43,000 kJ/kg in the MMPC (above) y en NMPC with SQP (bottom). The graph has been enlarged 10× in order to see the effect.

4.8 Measurement noise

Sources of white noise of uniform distribution have been added to the output variables to evaluate the controller's performance against the sources of noise. The parameters and type of noise have been experimentally determined on the signals captured from the respective sensors of the data acquisition system reported in [34] and they have been added as such without the filtering processes of the measurement system. The amplitudes represented to the engineering units determined for each output are: 0.01 for P_m , 1 for T_m , 0.05 for n , and 0.03 for λ . Figure 62 shows the performance of both NMPC controllers assuming the noise measurement without a filtering system. The response is kept in the correct reference monitoring with the stability achieved without noise in the measurement. (MMPC at the top, NMPC with SQP at the bottom).

5 Conclusions

An experimental process to verify the functionality and performance of non-linear optimal controllers to be embedded inside the control and operation structure of an automotive electronic control unit has been carried out. Hardware architecture in the loop used with two real-time processing units, one for the engine model and the other for the non-

linear controller, supports the validity of the results. Also, the use of compatible COTS components simplifies its development.

Two NMPC control schemes have been evaluated by carrying out stability, robustness and response time tests. The first one is a linear MPC with an online model update; this, in essence, is a non-linear controller given the non-linear model of an automobile engine. For the model update, several update strategies in parallel with the MPC controller work were experimented. These strategies were influential for the stability and performance of the control loop. From a qualitative point of view (processing load, reference monitoring and control smoothness), it was defined that the best strategy to update the model is through the observation of the change and error dynamics in stable state of the engine speed.

With such update strategy, tests were done changing the optimization objectives towards points of reduction of emissions, or fuel saving or keeping the speed reference and also simultaneously, all this being under the appropriate weighting of variables in the objective function. Tests with energy optimization show correct reference monitoring, besides respecting narrow levels of environmental emissions along the whole range of operation. Tests with simultaneous optimization show the monitoring capacity of the references, for both speed and emissions, slightly punishing with overshooting of the speed response. A joint economic objective punishes the speed reference monitoring to possibly permissible levels for some applications, keeping optimality in the emissions and consumption, and respecting the fuel and air consumption levels.

The other scheme corresponds to the use of the SQP algorithm, and the use of an ODE solver for the engine model, and also an integrator to compensate for the stable state errors. The results were also satisfactory, achieving similar operation for independent and simultaneous objectives, with softer movements in the control signals.

Stability tests on both controllers showed improvement on the performance by increasing the prediction horizon up to 100 sampling periods in the first scheme, and of 20 seconds for the second scheme.

The system robustness has been verified by introducing practical changes of the load in the axle, momentary opening of the throttle and movements of the injected flow that even though the result was an immediate rejection from the scheme with updating, and both the results are satisfactory. Likewise, admissible deviations in the modeling parameters (environmental temperature, atmospheric pressure, gas heat capacity) and white noise for the sensor signals have been entered and the results of stability, performance outcomes and optimality criteria are kept in both control schemes.

References

- Kirk, D.E.: Optimal Control Theory: An Introduction. Courier Corporation, New York (2004)
- Hendricks, E., Sorenson, S.: Mean value modelling of spark ignition engines. SAE Technical Paper 900616 (1990). doi:[10.4271/900616](https://doi.org/10.4271/900616)
- Hernández, L.M.: Modelo Matemático de Tipo Valor Medio de un Motor de Combustión Interna y Estimación de Estados usando Técnicas Bayesianas (Master Tesis). Universidad Eafit, Medellín, Colombia (2012). <http://hdl.handle.net/10784/1291>. Accessed 10 Nov 2013
- Chica, J.A.V., Torres, A.G.D., Acosta, D.A.: NMPC controller applied to the operation of an internal combustion engine: formulation and solution of the optimization problem in real time. Int. J. Interact. Des. Manuf. (2016). doi:[10.1007/s12008-016-0343-2](https://doi.org/10.1007/s12008-016-0343-2)
- Naidu, S.D.: Optimal Control Systems. Idaho State University, Pocatello, Idaho/CRC Press, USA (2002)
- Bellman, R.E., Kalaba, R.E.: Dynamic Programming and Modern Control Theory. Academic Press, New York (1965)
- Bellman, R.: Dynamic Programming. Princeton University Press, Princeton (1957)
- National Instruments Corp: LabVIEW Control Design User Manual. Austin, Texas (2009)
- Algöwer, F., Findeisen, R., Nagi, Z.K.: Nonlinear model predictive control: from theory to application. J. Chin. Inst. Chem. Eng. **35**(3), 299–315 (2004)
- Albersmeyer, J., Beigel, D., Kirches, C., Wirsching, L., Bock, H.G., Schlöder, J.P.: Fast nonlinear model predictive control with an application in automotive engineering. In: Magni, L., et al. (eds.) Nonlinear Model Predictive Control. Towards New Challenging Applications, pp. 471–480. Springer, Berlin (2009)
- Kuure-Kinsey, M., Bequette, B.W.: Multiple model predictive control of nonlinear systems. In: Magni, L., Raimondo, D.M., Allgower, F. (eds.) Nonlinear Model Predictive Control. Lecture Notes in Control and Information Sciences, vol. 384, pp. 153–165. Springer, Berlin (2009)
- Foss, B.A., Cong, S.B.: Nonlinear MPC based on multi-model for distillation columns. In: Proceedings of the 14th World Congress of IFAC, Beijing, P.R. China, pp. 337–342 (1999)
- Chisci, L., Falugi, P., Zappa, G.: Gain-scheduling MPC of nonlinear systems. Int. J. Robust Nonlinear Control **13**, 295–308 (2003). doi:[10.1002/rnc.819](https://doi.org/10.1002/rnc.819)
- The MathWorks, Inc. Multiple MPC controllers simulate. Switching between multiple MPC controllers. <http://www.mathworks.com/help/mpc/ref/multiplempccontrollers.html> (2014)
- Tenny, M.J., Wright, S.J., Rawlings, J.B.: Nonlinear model predictive control via feasibility-perturbed sequential quadratic programming. Comput. Optim. Appl. **28**(1), 87–121 (2004)
- Barclay, A., Gill, P.E., Rosen, J.B.: SQP methods and their applications to numerical optimal control. In: Variational Calculus, Optimal Control and Applications International Series of Numerical Mathematics, vol. 124, pp. 207–222 (1998)
- Büskens, C., Maurer, H.: SQP methods for solving optimal control problems with control and state constraints: adjoints variables, sensitivity analysis and real-time control. J. Comput. Appl. Math. **120**, 85–108 (2000)
- Bequette, B.W.: Multiple model predictive control (MMP) for nonlinear systems and improved disturbance rejection. In: Process Modeling and Optimization for Energy and Sustainability PASI. Angra dos Reis, RJ, Brazil, July 19–29 (2011)
- Biegler Lorentz, T.: Nonlinear Programming. Concepts, Algorithms and Applications to Chemical Process. SIAM, Carnegie Mellon University, Pittsburgh, Pennsylvania (2010)
- Ferreau, J.: An Online Active Set Strategy for Fast Solutions of Parametric Quadratic Programs with Applications to predictive engine Control (Tesis Magistral). Universität Heidelberg, Alemania (2006)
- Aburajabaltamimi, J.: Development of Efficient Algorithms for Model Predictive Control of Fast Systems (Tesis Doctoral). Palestine Polytechnic University, Hebro (2011)
- Balsa Canto, E.: Algoritmos Eficientes para la Optimización Dinámica de Procesos Distribuidos (Tesis Doctoral). Universidad de Vigo, Vigo (2001)
- Borroux, D.: Nonlinear Model Predictive Control for an Artificial Pancreas (Tesis Magistral). Technical University of Denmark, Kongens Lyngby (2009)
- Bemporad, A., Morari, M.: Robust model predictive control: a survey. In: Robustness in Identification and Control. Lecture Notes in Control and Information Sciences, vol. 245, 1999, pp. 207–226 (2007)
- Ferreau, H. J.: An Online Active Set Strategy for Fast Solution of Parametric Quadratic Programs with Applications to Predictive Engine Control. (Diplom Thesis), University of Heidelberg (2006)
- De Nicolao, G., Magni, L., Scattolini, R.: Stabilizing receding-horizon control of nonlinear time varying systems. IEEE Trans. Autom. Control **AC-3**(7), 1030–1036 (1998)
- Chen, H., Allgöwer, F.: A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. Automatica **34**(10), 1205–1218 (1998)
- Magni, L., De Nicolao, G., Scattolini, R., Allgöwer, F.: Robust receding horizon control for nonlinear discrete time systems. In: Proceedings 15th IFAC World Congress, 2002 (2001)
- Nagy, Z.K., Braatz, R.D.: Robust nonlinear model predictive control of batch processes. AIChE J. **49**, 1776–1786 (2003). doi:[10.1002/aic.690490715](https://doi.org/10.1002/aic.690490715)
- Pannocchia, G., Rawlings, J.B., Wright, S.J.: Is Suboptimal Nonlinear MPC Inherently Robust 18th IFAC World Congress Milano (Italy) August 28–September 2, 2011, pp. 7981–7986. doi:[10.3182/20110828-6-IT-1002.02848](https://doi.org/10.3182/20110828-6-IT-1002.02848)
- Zavala, Victor M., Biegler, Lorenz T.: The advanced-step NMPC controller: optimality, stability and robustness. Automatica **45**(1), 86–93 (2009). ISSN 0005-1098. doi:[10.1016/j.automatica.2008.06.011](https://doi.org/10.1016/j.automatica.2008.06.011)
- Valencia, A., Diaz, A.: Simulation NMPC in 2-HIL to design ECU. Int. J. Interact. Des. Manuf. (IJIDeM). (2015). doi:[10.1007/s12008-015-0265-4](https://doi.org/10.1007/s12008-015-0265-4)
- de Planeación, Unidad, Energética, Minero: Balance Energético Nacional 1975–2006. Ministerio de Minas y Energía, República de Colombia (2007)
- Díaz, A., Hernández, M.: Banco de Pruebas Automatizado para Uso en Modelación y Control de un Motor de Combustión Interna. In: Ninth LACCEI Latin American and Caribbean Conference (LACCEI'2011), Engineering for a Smart Planet, Innovation, Information Technology and Computational Tools for Sustainable Development, Medellín, Colombia, 3–5 Aug 2011