Model Predictive Powertrain Control: an Application to Idle Speed Regulation

S. Di Cairano, D. Yanakiev, A. Bemporad, I.V. Kolmanovsky, D. Hrovat

Abstract Model Predictive Control (MPC) can enable powertrain systems to satisfy more stringent vehicle requirements. To illustrate this, we consider an application of MPC to idle speed regulation in spark ignition engines. Improved idle speed regulation can translate into improved fuel economy, while improper control can lead to engine stalls. From a control point of view, idle speed regulation is challenging, since the plant is subject to time delay and constraints. In this paper, we first obtain a control-oriented model where ancillary states are added to account for delay and performance specifications. Then the MPC optimization problem is defined. The MPC feedback law is synthesized as a piecewise affine function, suitable for implementation in automotive microcontrollers. The obtained design has been extensively tested in a vehicle under different operating conditions. Finally, we show how competing requirements can be met by a switched MPC controller.

1 Introduction

As the requirements for automotive vehicles become more stringent and customers expect improved driveability and better fuel economy, advanced powertrain control strategies become an appealing tool to be exploited in meeting these objectives. In this paper we consider the use of Model Predictive Control (MPC) [14] as a control strategy to achieve high controller performance. MPC presents several advantages with respect to standard control strategies, such as the capability of explicitly enforcing constraints and the optimization of a user defined performance objective,

S. Di Cairano, D. Yanakiev, I.V. Kolmanovsky, and D. Hrovat
Powertrain Control R&A, Ford Motor Company, Dearborn, MI,
e-mail: sdicaira,dyanakie,ikolmano,dhrovat@ford.com

A. Bemporad
Dipartimento di Ingegneria dell'Informazione, Università di Siena, Italy,
e-mail: bemporad@dii.unisi.it

while maintaining the inherent robustness of feedback control. Explicit model predictive control techniques [2] can be used to synthesize the controller in the form of a piecewise affine function, which is feasible for implementation microcontrollers that have limited memory and computational power.

In this paper we demonstrate the design, synthesis, and experimental validation of a model predictive controller for Idle Speed Control (ISC) [8–10]. ISC is of continuing importance in spark ignited engines, since a higher performance controller may allow for lower engine idle speed or for operating with lower spark reserve with no risk of engine stalls. Lower engine idle speed or lower spark reserve reduce fuel consumption [8]. Several techniques have been considered to control idle speed, including state feedback [10], speed gradient with backstepping [11], observer-based control [3, 6], output feedback stabilization [7], hybrid controllers [1], and adaptive control [15], see also the references therein.

In this paper we design a controller that regulates engine speed at the indicated idle setpoint by controlling the airflow in the engine manifold via an electronic throttle. The challenges are the delay between the change in the commanded airflow and the change in the engine torque, and the constraints on the system input (limits on the throttle position) and output (allowed range for the engine speed error).

The paper is organized as follows. In Section 2 we introduce a nonlinear engine model used for idle speed control and for simulation, and in Section 3 the MPC controller, based on a linearized model, is designed. After the controller is tested in closed-loop with the nonlinear model, the complexity is assessed and the control law is synthesized in Section 4. Experimental validation is discussed in Section 5. The conclusions are summarized in Section 6

2 Engine model for idle speed control

Engine rotational dynamics are based on Newton's second law,

$$\dot{N} = \frac{1}{J}\frac{30}{\pi}(M_{\mathrm{e}} - M_{\mathrm{L}}), \tag{1}$$

where N is the engine speed measured in rpm, M_{e} [Nm] is the net engine brake torque, and M_{L} [Nm] is the load torque applied to the crankshaft.

In port-fuel injection engines, the torque cannot be instantaneously changed, and engine torque production dynamics have to be considered,

$$M_{\mathrm{e}}(t) = M_{\mathrm{e},\delta}(t - t_d) - M_{\mathrm{fr}}(t) - M_{\mathrm{pmp}}(t), \tag{2}$$

where $M_{\mathrm{e},\delta}(t - t_d)$ [Nm] is the indicated torque delayed by the intake-to-torque production time delay, t_d [s], $M_{\mathrm{fr}}(t)$ are the engine friction torque losses, and $M_{\mathrm{pmp}}(t)$ are the pumping torque losses. In Equation (2), constant spark retard is assumed, while near idle $M_{\mathrm{fr}}(t) + M_{\mathrm{pmp}}(t)$ can be approximated as constant. The delay t_d is about 360 degrees of the crankshaft revolution,

$$t_d(t) = \frac{60}{N(t)}. \tag{3}$$

Additional dynamics are associated with manifold filling. Under constant gas temperature assumption, the intake manifold pressure dynamics are

$$\dot{p}_{\mathrm{im}} = \frac{RT_{\mathrm{im}}}{V_{\mathrm{im}}}(W_{\mathrm{th}} - W_{\mathrm{cyl}}), \tag{4}$$

where W_{th} [kg/s] is the mass flow rate through the electronic throttle and W_{cyl} [kg/s] is the mean value of the mass flow rate into the engine cylinders. In (4), T_{im} [K] is the intake manifold temperature, V_{im} [m^3] is the intake manifold volume and R is the ideal gas constant in kJ/kg/K. Assuming stoichiometric air-to-fuel ratio, the engine indicated torque is approximately proportional to the cylinder air charge, m_{cyl} [kg],

$$M_{\mathrm{e},\delta}(t) = \gamma_1 m_{\mathrm{cyl}}(t) = \gamma_1 \frac{W_{\mathrm{cyl}}(t)}{N(t)}. \tag{5}$$

where γ_1 is an engine dependent parameter. The cylinder flow is a function of the intake manifold pressure, p_{im} [Pa] and engine speed, N,

$$W_{\mathrm{cyl}}(t) = \frac{\gamma_2}{\gamma_1} p_{\mathrm{im}}(t) N(t) + \gamma_0, \tag{6}$$

and γ_0, γ_2 are constant parameters. Near idle, the flow through the throttle is choked and one can approximate

$$W_{\mathrm{th}}(t) = \gamma_3 \vartheta(t), \tag{7}$$

where ϑ [deg] is the throttle position and γ_3 is an engine dependent constant. From (5), (6) it follows that $M_{\mathrm{e},\delta}(t) = \gamma_2 p_{\mathrm{im}}(t) + \frac{\gamma_1}{N(t)}\gamma_0$. Differentiating this and using (4), (6), and (7)

$$\dot{M}_{\mathrm{e},\delta} = -\gamma_2 \frac{RT_{\mathrm{im}}}{V_{\mathrm{im}}} \frac{N}{\gamma_1} M_{\mathrm{e},\delta} + \gamma_2 \frac{RT_{\mathrm{im}}}{V_{\mathrm{im}}} \gamma_3 \vartheta - \frac{\gamma_0 \cdot \gamma_1}{N^2} \dot{N}. \tag{8}$$

Neglecting the (typically small) term $\frac{\gamma_0 \cdot \gamma_1}{N^2} \dot{N}$ it follows that

$$\dot{M}_{\mathrm{e},\delta} = -\gamma_2 \frac{RT_{\mathrm{im}}}{V_1} \frac{N}{\gamma_1} M_{\mathrm{e},\delta} + \gamma_2 \frac{RT_{\mathrm{im}}}{V_{\mathrm{im}}} \gamma_3 \vartheta. \tag{9}$$

Thus, the complete engine model near idle is defined by (1), (2), (3) and (9).

3 Control-oriented model and controller design

From Section 2 it follows that the engine dynamics are described by a second order system (1), (2), (9) with time delay (3). Around idle they can be conveniently approximated as a second order system subject to input delay [9],

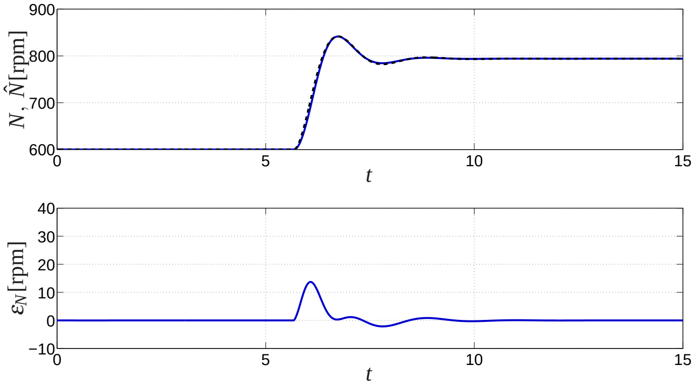


Fig. 1 Upper plot: step response of the nonlinear model (solid) and of the linearized model (dashed). Lower plot: error $\varepsilon_N(t) = \hat{N}(t) - N(t)$

$$Y_p(s) = G(s)e^{-st_d}U(s), \qquad G(s) = k\frac{1}{\frac{s^2}{\omega^2} + 2\frac{\zeta}{\omega}s + 1}, \tag{10}$$

where t_d can be approximated as constant by (3) where $N(t) = \overline{N}$, and $\overline{N}$ is the idle speed setpoint. Model (10) represents the deviations of engine speed from idle speed setpoint, $y_p(t) = N(t) - \overline{N}$, in response to variations in throttle position form nominal position, $u(t) = \vartheta(t) - \overline{\vartheta}$, where $\overline{\vartheta}$ is the nominal throttle position to compensate for $M_{\text{fr}}(t) + M_{\text{pmp}}(t)$ and for the nominal load torque $\overline{M}_L$ at idle.

Model (10) has the advantage of being easily identifiable from both simulation and/or experimental data. The coefficients in (10) can be computed in the time domain from step-response data. In particular, we can identify the linearized model from an available high-fidelity nonlinear simulation model based on (1), (2), (3) and (9) where the coefficients are obtained from experimental data. Although direct identification from data is possible, we prefer to use a nonlinear simulation model first, because it allows for testing the closed-loop behavior in simulation. Through simulations we obtain a qualitative understanding of the effects of the tuning knobs on the closed-loop behavior much more rapidly than by experiments. This understanding is fundamental to reduce the calibration time in the vehicle, so that the overall controller calibration procedure cost is reduced.

Figure 1 shows the results of the identification where the parameters in (10) are obtained from the relations $L \approx \exp(-\pi\zeta(1-\zeta^2)^{-1/2})$, $\omega_d = \omega(1-\zeta^2)^{1/2}$, $t_M = \pi\omega_d^{-1}$, and where L is the normalized overshoot, t_M is the oscillation period and ω_d is the damped system frequency. The evolution of the engine speed as predicted by the linearized model, $\hat{N}(t) = y_p(t) + \overline{N}$, is compared to the engine speed evolution for the nonlinear model, $N(t)$, in Figure 1. The plot of the error $\varepsilon_N(t) = \hat{N}(t) - N(t)$ indicates that a satisfactory fit is obtained.

The identified transfer function $G(s)$ is then converted to discrete-time state space form with sampling $T_s = 30$ms,

$$x_f(k+1) = A_f x_f(k) + Bu_\delta(k), \quad x_f \in \mathbb{R}^2, \tag{11a}$$

$$y_f(k) = C_f x_f(k), \tag{11b}$$

where u_δ is the delay-free input. For idle speed $\overline{N} = 600$rpm, $t_d = 100$ms$\approx 4T_s$. The discrete-time model of a signal $u(\cdot)$ delayed by $n_\delta \in \mathbb{Z}_{0+}$ steps is

$$\begin{aligned} &x_\delta(k+1) = A_\delta x_\delta(k) + B_\delta u(k), \quad x_\delta \in \mathbb{R}^{n_\delta}, \\ &u_\delta(k) = C_\delta x_\delta(k), \\ &A_\delta = \begin{bmatrix} 0 & \cdots & 0 \\ & & \vdots \\ I_{n_\delta - 1} & & \\ & & 0 \end{bmatrix}, \quad \begin{aligned} B_\delta &= [1\ 0 \cdots 0]^T, \\ C_\delta &= [0 \cdots 0\ 1]. \end{aligned} \end{aligned} \tag{12}$$

By cascading (11) and (12), the complete discrete-time model of (10) is obtained,

$$x_p(k+1) = A_p x_p(k) + B_p u(k), \quad x_p = \left[\begin{smallmatrix} x_f \\ x_\delta \end{smallmatrix}\right], \ x_p \in \mathbb{R}^6, \tag{13a}$$

$$y_p(k) = C_p x_p(k). \tag{13b}$$

The plant model (13) has to be extended with ancillary states that are used to enforce specifications. In idle speed control, steady-state errors due to changes in load torque caused, for instance, by power steering or air-conditioning system, or due to errors in the scheduled airflow feedforward have to be removed. In order to achieve offset free constant disturbance rejection one can introduce *integral action* [12] by adding the dynamics $q(k+1) = q(k) + T_s y_p(k)$, where $q \in \mathbb{R}$ is the discrete-time integral of the output. When integral action is added to (13), the resulting model is

$$x(k+1) = Ax(k) + Bu(k), \quad x = \left[\begin{smallmatrix} x_p \\ q \end{smallmatrix}\right], \ x \in \mathbb{R}^7, \tag{14a}$$

$$y(k) = Cx(k), \tag{14b}$$

$$A = \left[\begin{smallmatrix} A_p & 0 \\ T_s C_p & 1 \end{smallmatrix}\right], \ B = \left[\begin{smallmatrix} B_p \\ 0 \quad 0 \end{smallmatrix}\right], \ C = \left[\begin{smallmatrix} C_p & 0 \\ 0 & 1 \end{smallmatrix}\right]. \tag{14c}$$

Model (14) will be used as the MPC prediction model. In order to complete the specification of the plant, the constraints on systems inputs and states have to be defined. Limits on the engine speed, in rpm, are (conservatively) defined to avoid engine stalls and engine flares,

$$-150 \leq y_p \leq 150\ . \tag{15}$$

Additional limits in the throttle angle are imposed,

$$0 \leq u + u_{FF} \leq 8, \tag{16}$$

where u_{FF} is the scheduled feedforward term in nominal conditions at idle, and ideally $u_{FF} = \overline{\vartheta}$. Note that the full (feedforward+feedback) throttle input is $\vartheta(t) = u_{FF} + u(t)$. Finally, the cost function for the finite horizon optimal control problem in the MPC strategy is specified. Since a non-zero input u can be required at steady state to reject disturbances, weighting the input increment in the cost function is more suitable,

$$J\big(\mathbf{y}(k),\mathbf{u}(k),u(k-1)\big)=\sum_{i=0}^{h-1}(y(i|k)-r_y)^TQ(y(i|k)-r_y)+\Delta u(i|k)S\Delta u(i|k) \quad (17)$$

where $h\in\mathbb{Z}_+$ is the prediction horizon, $\mathbf{y}(k)=\big(y(0|k),\dots,y(h-1|k)\big)$ and $\mathbf{u}(k)=\big(u(0|k),\dots,u(h-1|k)\big)$ are the output and input sequences predicted at step k, respectively, r_y is the output setpoint, $\Delta u(i|k)=u(i|k)-u(i|k-1)$, and $u(-1|k)=u(k-1)$. Here, the weight matrices Q and S are both positive definite.

Collecting (14), (15), (16), (17), the optimal control problem is formulated as

$$\begin{aligned}
\min_{\sigma,\mathbf{u}(k)}\quad & \rho\sigma^2+\sum_{i=0}^{h-1}(y(i|k)-r_y)^TQ(y(i|k)-r_y)+\Delta u(i|k)S\Delta u(i|k) && \text{(18a)}\\
\text{s.t.}\quad & x(i+1|k)=Ax(i|k)+Bu(i|k),\ i=0,\dots,h-1, && \text{(18b)}\\
& y(i|k)=Cx(i|k),\ i=0,\dots,h-1, && \text{(18c)}\\
& u_{\min}\le u(i|k)\le u_{\max},\ i=0,\dots,h-1, && \text{(18d)}\\
& y_{\min}-\sigma\mathbf{1}\le y(i|k)\le y_{\max}+\sigma\mathbf{1},\ i=0,\dots,h_c-1, && \text{(18e)}\\
& \sigma\ge 0,\quad u(i|k)=u(h_u-1|k),\ i=h_u,\dots,h-1, && \text{(18f)}\\
& u(-1|k)=u(k-1),\ x(0|k)=\hat{x}(k), && \text{(18g)}
\end{aligned}$$

where $\mathbf{1}$ is a vector entirely composed of 1, $\hat{x}(k)$ is the state estimate at time k, $h_c\le h$ is the constraint horizon, and $h_u\le h$ is the control horizon. For the idle speed MPC controller, in (18),

$$y_{\min}=\begin{bmatrix}-150\\-\infty\end{bmatrix},\ y_{\max}=\begin{bmatrix}150\\+\infty\end{bmatrix},\ u_{\min}=-u_{FF},\ u_{\max}=8-u_{FF},$$

and (18e) enforce *soft output constraints*. Soft constraints can be violated at the price of a penalty, modelled by the optimization variable $\sigma\in\mathbb{R}_{0+}$, weighted by the positive constant ρ, which must be at least two orders of magnitude larger than the other weights. Soft output constraints are used to avoid (18) becoming unfeasible due to large unmeasured disturbances.

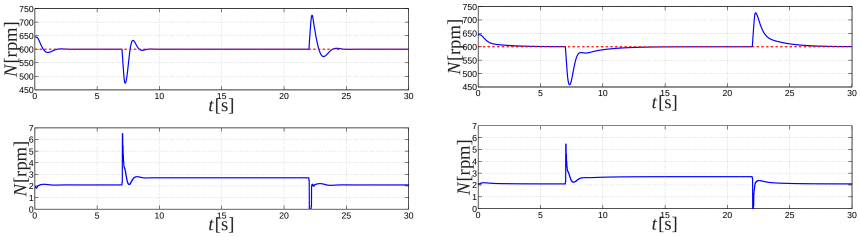
At each step $k\in\mathbb{Z}_{0+}$, the MPC controller reads the measurements and computes the state estimate $\hat{x}(k)$, then solves the quadratic programming problem (18) to obtain the optimal input sequence $\mathbf{u}^*(k)$, and applies the input $u(k)=u^*(0|k)$. For the idle speed controller, a Kalman filter has been used for state estimation.

Closed-loop simulation tests were performed in SIMULINK® with the MPC in closed-loop with the nonlinear engine simulation model (1), (2), (3) and (9). In the simulations, $\bar{N}=600$rpm, $\bar{\vartheta}=2.09$, $h=30$, $h_u=2$, $h_c=3$, $r_y=[0\ 0]^T$.

The feedforward is $u_{\text{FF}}=2.5$, slightly different from $\bar{\vartheta}$ in order to test the regulation capabilities of the controller and the rejection of feedforward input errors.

At $t=0$s the controller has to regulate the system to 600rpm. Then, a disturbance rejection test is performed by introducing an unmeasured torque disturbance load

$\tilde{M}_{\mathrm{L}} = 20$Nm, simulating, for instance, the power steering motor load, starting at $t = 7$s, and ending at $t = 22$s.



(a) Oscillating closed-loop dynamics.

(b) Overdamped closed-loop dynamics.

Fig. 2 Simulation of the MPC idle speed controller in closed-loop with the nonlinear engine model

Figure 2 shows the resulting closed-loop behavior for two different tunings of the cost function weights. In Figure 2(a) the closed-loop dynamics exhibit overshoot, while in Figure 2(b) they are overdamped.

4 Controller synthesis and refinement

The controller designed in Section 3 needs to undergo further steps before being tested on the vehicle. First, it must be synthesized in a way that allows implementation in the automotive microcontrollers. Then, it must be refined to work properly with an experimental engine.

4.1 Feedback law synthesis and functional assessment

Solving the optimal control problem (18) in conventional automotive microcontrollers may still be too computationally demanding. Also, the CPU operations and the memory requirements of the optimization algorithms are difficult to predict. To overcome these difficulties, the MPC controller is explicitly synthesized [2]. The explicit MPC control law of problem (18) is a piecewise affine state feedback, $u(k) = \varphi_{\mathrm{MPC}}(x(k), r_y(k), u(k-1))$, where $\varphi_{\mathrm{MPC}}(x, r_y, u_{-1}) = F_i^x x + F_i^r r_y + F_i^u u_{-1} + g_i$ if $(x, r_y, u_{-1}) \in \mathscr{P}_i$, $i = 1, \dots, s$, and $\{\mathscr{P}_i\}_{i=1}^{s}$ is a polyhedral partition of the state-reference-previous input space.

For the idle speed control optimization problem (18), $horz = 30$, $horz_c = 3$, $horz_u = 2$, and as a consequence there are 4 input constraints and 6 output constraints enforced along the horizon. The obtained explicit controller is composed of 35 regions, and the parameters in function $\varphi(\cdot)$ are 7 states, 2 references, 1 previous input. The *worst case* number of operations for both region search and command computation that have to be executed at each control cycle is less than 5000, which amount to less than $2 \cdot 10^5$ operations per second. Hence, according to [13], the explicit controller uses less than 1% of the CPU in the *worst case*. It shall be noted that at idle the microcontroller is, in general, underloaded, due to the low rate of engine event triggered tasks. The controller will require less than 6KB of data memory storage.

Finally, since the explicit feedback law is computed, local stability of the closed-loop system can be simply evaluated. For the controller shown in Figure 2(b) the largest eigenvalue absolute value is $\|\lambda\|_{\max} = 0.97$, confirming that the closed-loop system is locally stable. Piecewise quadratic Lyapunov functions can be constructed using LMI to confirm global stability.

4.2 Prediction model refinement

The controller designed in Section 3 is based on a simulation model. Although this allows for controller design, evaluation, and functional assessment without the need of experimental data, the controller benefits from a refinement to efficiently operate with a specific engine. The refinement involves two steps: (i) the control-oriented engine model (10) is identified again, this time using experimental data, and, (ii) the weights of the cost function (17) are re-tuned. The results of step (i) in the experimental V6 4.6L engine are shown in Figure 3, where the identified engine model is validated with respect to the experimental data. The model error is larger than the one in Figure 1. However, since MPC is a feedback strategy, the resulting model uncertainty does not present an issue. The experimental data can be used to evaluate the measurements and process noise, and the modelling errors. This allows to appropriately tune the Kalman filter for estimating the state of model (14) from available measurements, the engine speed in our case.

5 Experimental validation

The refined controller was experimentally validated in nominal and non-nominal conditions. A nominal test involving power-steering torque load rejection at nominal idle ($\bar{N} = 600$rpm) with the transmission in *Neutral* is shown in Figure 4. The MPC controller (solid line) is compared with a tuned baseline PID-based controller. The MPC significantly outperforms the baseline controller. The main reason is the explicit constraint enforcement, and the delay model, which allows to implement a more aggressive control action (by increasing the output regulation weight Q) without losing stability.

The controller performance was also evaluated at non-nominal conditions, with the transmission in *Drive*, a decreased desired idle speed ($\bar{N} = 525$rpm), and, as a

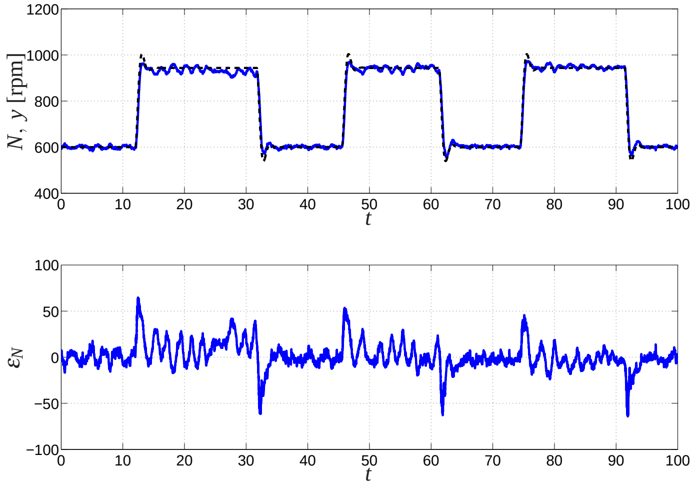


Fig. 3 Upper plot: step response of the nonlinear model (solid) and of the linearized model (dashed). Lower plot: error $\varepsilon_N(t) = \hat{N}(t) - N(t)$

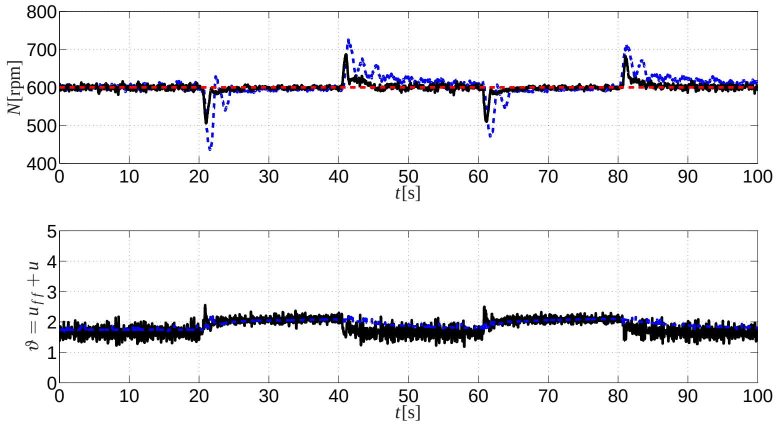


Fig. 4 Power steering load rejection test. MPC controller (solid) and baseline (dashed).

consequence, a larger delay. Also, due to the transmission load, the engine dynamics are slightly different. In the test (see Figure 5) two disturbances are introduced and then removed almost simultaneously, power-steering load and air conditioning load. Due to the recognized large increase in the load, the scheduled setpoint for idle speed changes during the experiment. The MPC controller outperforms again the baseline controller, both in terms of disturbance rejection and time-varying reference tracking.

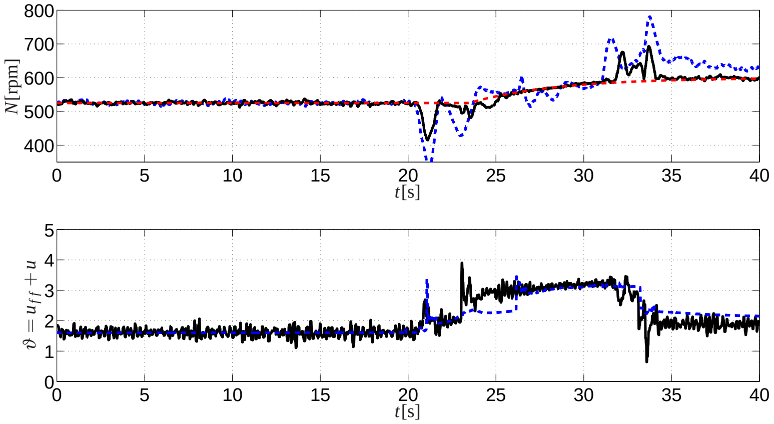


Fig. 5 Simultaneous power steering and air conditioning load rejection test. MPC controller (solid) and baseline (dashed).

Finally, in Figure 6 a cold-start test is shown, where the idle speed setpoint is initially very high (about 1250rpm), and then slowly approaches the nominal value (600rpm). The feedback controller is activated when the setpoint reaches 1000rpm. During the tests, several disturbances are introduced, including power steering load, air conditioning load, and multiple repetitions of these. Even in this test, the controller shows a very satisfactory performance.

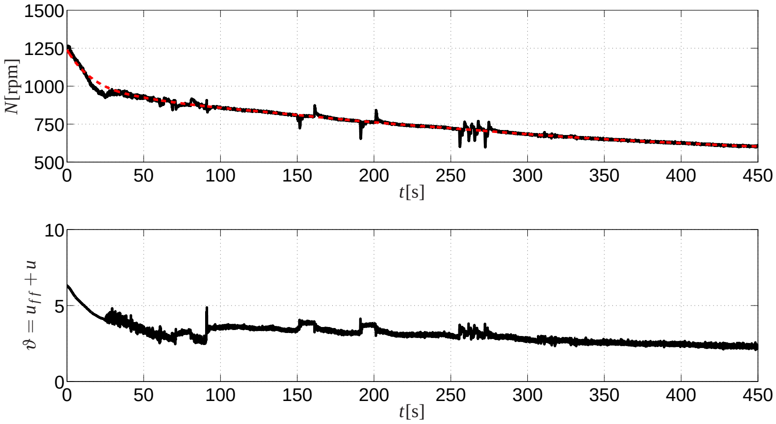


Fig. 6 Cold start test of the MPC controller. Rejection of power steering and air conditioning disturbances.

The tests in Figures 4–6 show that the MPC achieves higher performance by acting more aggressively on the control input. As a consequence, the controller may be unnecessarily sensitive around the setpoint due to measurement noise. In order to maintain high performance load rejection and reduce the sensitivity around the setpoint, two MPC controller can be scheduled, $\varphi_1(x(k), r_y(k), u(k-1))$ with low sensitivity, and $\varphi_2(x(k), r_y(k), u(k-1))$ with aggressive action. In particular, φ_1 can be tuned to have the same sensitivity as the baseline controller by using the techniques presented in [4]. A switched control law is applied,

$$u(k) = \text{ if } |y_p| < y_\ell \text{ then } \varphi_1(x(k), r_y(k), u(k-1)) \text{ else } \varphi_2(x(k), r_y(k), u(k-1)), \tag{19}$$

where y_ℓ is the switching threshold that is selected by analyzing the measurement and the process noise. The results for such a controller in a test similar to the one in Figure 4 are shown in Figure 7, where we see that the MPC still outperforms the baseline, but now around the setpoint the controller action is more cautious.

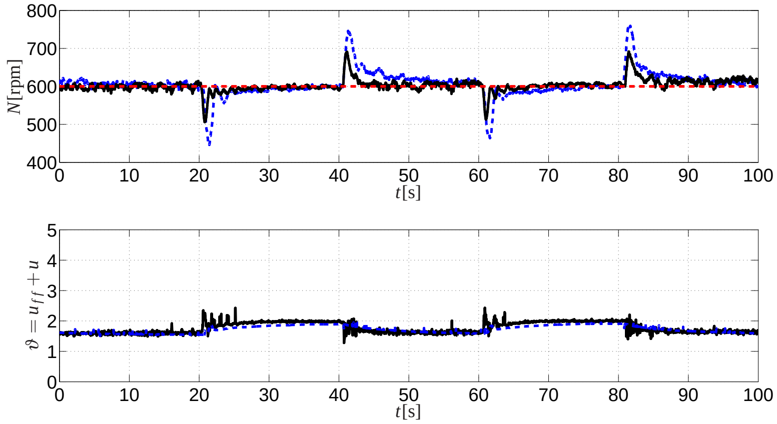


Fig. 7 Power steering rejection test. Switched MPC controller (solid) and baseline (dashed).

6 Conclusions

We have presented the design, synthesis, and validation of a model predictive controller for idle speed regulation. The MPC controller provides higher performance in terms of disturbance rejection and convergence rate than a tuned baseline PID-based controller, and it is synthesized in a form that is suitable for implementation in standard automotive microcontrollers. The predictive features of the MPC strategy better account for intake-to-torque production time-delay, while performance objective and soft output constraints result in faster, nonlinear compensation of disturbances than the conventional baseline controller. As we have shown for the idle speed regulation problem, the MPC solution is well within the capability of automotive microcontrollers, with a CPU load smaller than 1%. An idle speed MPC controller commanding both throttle and spark delay is under investigation, and preliminary results are shown in [5].

References

1. Balluchi, A., Di Natale, F., Sangiovanni-Vincentelli, A., van Schuppen, J.: Synthesis for idle speed control of an automotive engine. In: R. Alur, G. Pappas (eds.) Hybrid Systems: Computation and Control, *Lecture Notes in Computer Science*, vol. 2993, pp. 126–141. Springer-Verlag (2004)
2. Bemporad, A., Morari, M., Dua, V., Pistikopoulos, E.: The explicit linear quadratic regulator for constrained systems. Automatica **38**(1), 3–20 (2002)
3. Bengea, S., Li, X., De Carlo, R.: Combined controller-observer design for uncertain time delay systems with applications to engine idle speed control. J. of Dynamic Sys., Meas. and Control **126**, 772–780 (2004)
4. Di Cairano, S., Bemporad, A.: Model predictive controller matching: Can MPC enjoy small signal properties of my favorite linear controller? In: Proc. European Control Conf. Budapest, Hungary (2009). To appear
5. Di Cairano, S., Yanakiev, D., Bemporad, A., Kolmanovsky, I., Hrovat, D.: An MPC design flow for automotive control and applications to idle speed regulation. pp. 5686–5691. Cancun, Mexico (2008)
6. Gibson, A., Kolmanovsky, I., Hrovat, D.: Application of disturbance observers to automotive engine idle speed control for fuel economy improvement. In: Proc. American Contr. Conf., pp. 1197–1202. Minneapolis, MN (2006)
7. Glielmo, L., Santini, S., Cascella, I.: Idle speed control through output feedback stabilization for finite time delay systems. In: Proc. American Contr. Conf., pp. 45–49. Chicago, IL (2000)
8. Guzzella, L., Onder, C.: Introduction to modeling and control of internal combustion engine systems. Springer (2004)
9. Hrovat, D., Sun, J.: Models and control methodologies for IC engine idle speed control design. Control Engineering Practice **5**(8), 1093 – 1100 (1997)
10. Kiencke, U., Nielsen, L.: Automotive control systems for engine, driveline, and vehicle. Springer (2000)
11. Kolmanovsky, I., Yanakiev, D.: Speed gradient control of nonlinear systems and its applications to automotive engine control. Journal of SICE **47**(3) (2008)
12. Kwakernaak, H., Sivan, R.: Linear optimal control systems. Wiley-Interscience, New York (1972)
13. Magner, S., Cooper, S., Jankovic, M.: Engine control for multiple combustion optimization devices. In: Proc. SAE World Congress, pp. Paper 26–21–0003. Detroit, MI (2006)
14. Rawlings, J.: Tutorial overview of model predictive control. IEEE Control Systems Magazine pp. 38–52 (2000)
15. Yildiz, Y., Annaswamy, A., Yanakiev, D., Kolmanovsky, I.: Adaptive idle speed control for internal combustion engines. In: Proc. American Contr. Conf., pp. 3700–3705. New York, NY (2007)