Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Continuation/GMRES Method based Nonlinear Model Predictive Control for IC Engines ⋆

Mingxin Kang * **Tielong Shen** * **Xiaohong Jiao** **

* *Department of Engineering and Applied Sciences, Sophia University, Tokyo, 102-8554, Japan. (e-mail: kangmx; tetu-sin@sophia.ac.jp)*
** *College of Electrical Engineering, Yanshan University, Qinhuangdao 066004, P. R. China. (e-mail: jiaoxh@ysu.edu.cn)*

Abstract: This paper presents a model-based receding horizon optimal control algorithm for the engine speed tracking control. A mean-value model including the air intake dynamics and the rotational dynamics is exploited in the tracking controller design, which is calibrated based on the physical rules combined with curve fitting techniques. Based on this mean-value model, the dynamical model of the speed tracking error is derived for any given speed command. The design problem is reduced to the receding horizon control problem under the constraint of the tracking error dynamics. The online computational algorithm based on C/GMRES approach is adopted to solve this nonlinear receding optimal problem. Finally, simulation and experiments results demonstrate the satisfactory tracking performance by using the proposed controller.

Keywords: model predictive control; speed tracking control; gasoline engine; receding horizon control

1. INTRODUCTION

Modern gasoline engine is a sophisticated control system since more and more subsystems and control algorithms have been integrated into it to meet the stringent emission regulatory and legislation requirement. Engine speed control is an important issue and it has been widely applied in the several aspects such as idling control (Cairano [2012], Xu [2013]), start-up control (Zhang [2010]), coordinate control with accessories (Stotsky [2000]). Meanwhile, engine speed control is also implemented through a wide operation range to meet the growing number of efficiency and emission requirements. Because of the nonlinearity and time-varying properties of gasoline engines, a challenge of such speed control issue is how to achieve the accurate and fast speed tracking over the wide operating range, especially in the transient process.

In recent years, model predictive control (MPC) has been widely applied in the automotive industry, due to its capability of optimal control for multi-variable systems with constraints on plant and actutors (Hrovat [2012]). The concept of MPC is to solve optimal control problem over a finite time horizon by minimizing the given performance index repeatedly and in real time. However, as for a nonlinear control system, e.g., engine control system, the online solution of nonlinear model predictive control (NMPC) is always a challenge since it has to handle the time-varying variables and also requires powerful computational capability for engine electronic control unit (ECU). Some studies investigated solution methods of NMPC have been reported (Bemporad [2002], Wang [2010], Ohtsuka [2004]). Actually, in view of the requirement of transient control in most engineering practice, a fast control action from the proposed NMPC is crucial for the control performance.

The purpose of this paper is to develop a speed tracking control scheme based on the NMPC method, which is valid in a wide engine operation range. A fast NMPC computational algorithm, which is called continuation and generalized minimum residual method (C/GMRES) (Ohtsuka [2004]), is adopted to provide the optimal control action. In order to reduce the speed tracking error, a dynamic error system based on the mean-value model is formulated in the controller design. The control effects will be evaluated at both simulation level and practical bench tests level.

The paper is organized as following structure. In Section 2, the mean-value model of the engine dynamics is reviewed briefly, which is employed to the control law design. The error dynamics model for engine speed tracking problem is designed and the specific optimal problem formulation is proposed as well. In Section 3, the online computational algorithm for nonlinear optimal problem is introduced. The final control law based on the intake air mass flow rate is further proposed in this section. In Section 4, the simulation and validation experiments will be conducted and the results demonstrate the proposed NMPC controller can obtain the good tracking performance. Finally, the conclusions will be drawn in Section 5.

⋆ The work of the authors was supported by the Toyota Motor Corporation. Corresponding author: Tielong Shen(tetu-sin@sophia.ac.jp).

2. PRELIMINARIES

2.1 Dynamical model

The general structure of simplified engine system is shown in Fig.1. In this research, the engine speed controller checks rotation speed of the crankshaft (ω) and also intake manifold pressure signal (p_m) in real time. And then, the proper throttle command (ϕ) will be calculated and delivered to the actuator.

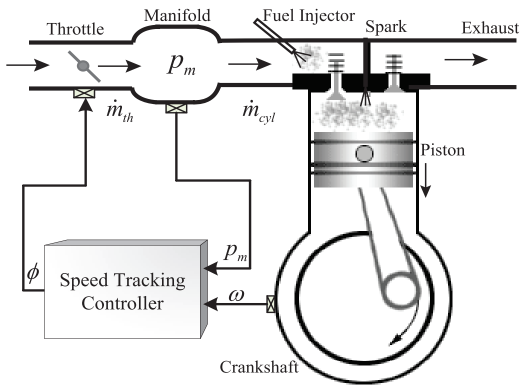


Fig. 1. The general structure of engine speed controller

Typically, the engine speed is determined by the output torque from crankshaft and external load. The physical process of the torque generation in cylinders can be regarded as the function with the intake manifold pressure and engine speed. Therefore, the rotation dynamics of the gasoline engine is expressed as follows:

$$J\dot{\omega} = g_1(\omega)p_m + g_2(\omega) - d \cdot \omega - \tau_l \tag{1}$$

where, J is the rotational inertia of the engine crankshaft; p_m is the intake manifold pressure; $g_1(\omega)$ and $g_2(\omega)$ are the time-varying parameters corresponding to the engine speed and they can be calibrated by the engine tests in steady conditions. d is the friction torque coefficient, and τ_l denotes the load torque working on the crankshaft.

For the 4-stroke gasoline engine, the mean-valve model (MVM) of air system has been well studied (Guzzella [2010]), which reflects the input-output behavior of system with reasonable precision but low computational complexity. According to the theory of MVM, the dynamical model of the intake manifold pressure can be written as the following form:

$$\dot{p}_m = a_0(\dot{m}_{th} - f_0(\omega, p_m) \cdot p_m \cdot \omega) \tag{2}$$

where, a_0 is regarded as a constant; $\dot{m}_{th}$ denotes the air mass flow rate through the throttle valve; $f_0(\omega, p_m)$ is the model parameter with respect to the engine speed and intake manifold pressure.

In the above physical models, the unknown model parameters can be obtained by implementing the recursive least square (RLS) identification method and curve fitting techniques. The detailed work has been summarized in (Li [2013]).

2.2 Tracking error dynamics

Actually, the control effects of the NMPC scheme is always constrained by the model precision. As for the complicated engine system with time-varying and nonlinear characteristics, the model parameters obtained by the RLS estimation algorithm unavoidably have identification errors. Most tests shows that there occurs the identification error for air intake system between the model outputs and the actual response, especially in the steady process (see Li [2013]). Moreover, for the optimal tracking problem, the tracking error dynamic model is necessary for the control design. Therefore, a new control-oriented model based on the speed tracking error is designed in this study.

For given desired speed response ω_d and $\dot{\omega}_d$, the tracking error is defined as follows:

$$e_1 = \omega - \omega_d \tag{3}$$

$$e_2 = \frac{1}{J}g_1(\omega)p_m - \frac{1}{J}d \cdot \omega_d - \dot{\omega}_d \tag{4}$$

where, e_1 is the error between the desired speed command ω_d and actual value.; e_2 is related with the dynamical characteristic of the desired command.

Based on the proposed definitions of the tracking errors, the error dynamic model used in the NMPC algorithm design can be formulated in the following forms:

$$\begin{cases} \dot{e}_1 = e_2 + \dfrac{1}{J}\{g_2(\omega_d + e_1) - d \cdot e_1 - \tau_l\} \\ \dot{e}_2 = v \end{cases} \tag{5}$$

In equation (5), v is regarded as the optimal control variable calculated by the NMPC scheme. It is deduced from the derivative of filtered tracking error, and its physical forms can be specifically expressed as follows:

$$\begin{aligned} v = \frac{1}{J}g_1(\omega)a_0(\dot{m}_{th} - f_0(\omega, p_m)p_m\omega) + \frac{1}{J^2}p_m f_1(\omega) \\ [g_1(\omega)p_m + g_2(\omega) - d\omega - \tau_l] - \frac{1}{J}d\dot{\omega}_d - \ddot{\omega}_d \end{aligned} \tag{6}$$

where, $f_1(\omega)$ is the time-varying parameters related with engine speed, which is deduced from $f_1(\omega) = \dot{g}_1(\omega)$.

In view of the boundedness of the physical system, the control variable v is actually constrained in a certain range:

$$v_{\min} \leq v \leq v_{\max} \tag{7}$$

2.3 NMPC problem

Based on the control oriented models mentioned above, NMPC controller is designed in order to achieve the speed tracking control over a wide operating range. It just solves out a sequence of control actions over a finite control horizon by optimizing the certain performance function but only implementing the first element on the plant.

According to the proposed error dynamic model, the NMPC problem is described as follows: To find an optimal control variable v^*, such that to minimize the following performance index over a certain predictive horizon ($\tau \in [t, t+T], t \to \infty$),

$$J(v) = r_1 e_1^2(t+T) + \int_t^{t+T} [r_2 e_1^2(\tau) + r_3 v^2(\tau)]d\tau \tag{8}$$

subject to the tracking error dynamic model (5) and the control constraints (7). In (8), r_1, r_2 and r_3 are the weighing coefficients.

3. CONTROL SCHEME

In order to solve this receding horizon optimal control, we applied the C/GMRES method, which is a fast computational algorithm for nonlinear optimal problem based on the pontryagin minimum principle (PMP). The essence of algorithm is to deduce a sequence of the optimal control actions by solving a group of matrix equations iteratively. While, only the first element of the optimal control sequence is implemented on the control plant finally. This section will give a brief review about C/GMRES method.

3.1 C/GMRES method

Considering the discrete-time property of the solution algorithm, the general description of receding horizon optimal problem with the discrete-time form is proposed first. The discrete form of the dynamical model can be deduced by the forward difference method:

$$x(t+1) = x(t) + f(x(t), u(t))\Delta\tau$$

where, $x(t)$ denote the state variables at t time, $u(t)$ is the input vector, $\Delta\tau$ is the discrete time.

For receding horizon optimal problem, suppose that the predictive horizon is divided into N-step. We define the following symbols for convenience:

$$x_t(k) = x(t+k), \ \ u_t(k) = u(t+k), (0 \le k \le N, t \to \infty)$$

Then, the discrete-time MPC problem can be summarized as follows: For given the initial conditions $x_t(0) = x_t$, find an optimal control action u^* to minimize the following performance index ($k = 0, 1, \cdots, N-1, \Delta\tau = \frac{T}{N}$.):

$$J = \Phi(x_t(N), N) + \sum_{k=0}^{N-1} L[x_t(k), x_d(k), u_t(k)]\Delta\tau \quad (9)$$

subject to

$$\begin{aligned} &x_t(k+1) = x_t(k) + f(x_t(k), u_t(k))\Delta\tau \\ &x_t(0) = x_t \\ &C(x_t(k), u_t(k)) = 0 \\ &u_{\min} \le u_t(k) \le u_{\max} \end{aligned}$$

The above inequality constraints can be converted to equality constraints by introducing a dummy input $u_t^{'}$, and defining new inputs vector as $\tilde{u}_t = [u_t, u_t^{'}]$. Then, the inequality constraints can be written in the following form: $C_1(u_t(k), u_t^{'}(k)) = 0$.

Let H denote the Hamiltonian defined by

$$\begin{aligned} H = &L[x_t(k), x_d(k), u_t(k)]\Delta\tau - ru_t^{'}(k) + \\ &\lambda_t^{\mathrm{T}}(k+1)[x_t(k) + f(x_t(k), u_t(k))\Delta\tau] + \\ &\mu_t^{\mathrm{T}}(k)C(x_t(k), u_t(k)) + \mu_{t1}^{\mathrm{T}}(k)C_1(u_t(k), u_t^{'}(k)) \end{aligned}$$

According to the PMP, the optimal solution of the problem must satisfy the following necessary conditions:

$$\begin{aligned} &x_t^*(k+1) = x_t^*(k) + f(x_t^*(k), u_t^*(k))\Delta\tau && (10) \\ &\lambda_t^*(k) = \lambda_t^*(k+1) + H_x^T(x_t^*(k), x_d^*(k), \tilde{u}_t^*(k), \\ &\qquad \lambda_t^*(k+1), \mu_t^*(k), \mu_{t1}^*(k))\Delta\tau && (11) \\ &H_{\tilde{u}_t}^T(x_t^*(k), x_d^*(k), u_t^*(k), u_t^{'*}(k), \lambda_t^*(k+1), \\ &\qquad \mu_t^*(k), \mu_{t1}^*(k)) = 0 && (12) \\ &\lambda_N^* = \Phi_{x_N}(x_t^*(N)) && (13) \\ &x_t^*(0) = x_t && (14) \\ &C(x_t^*(k), u_t^*(k)) = 0 && (15) \\ &C_1(u_t^*(k), u_t^{'*}(k)) = 0 && (16) \end{aligned}$$

In order to solve the above equations, we define a vector of multipliers as $\tilde{\mu}_t = [\mu_t, \mu_{t1}]$, and write all the inputs and multipliers over the predictive horizon steps in a vector as follows:

$$U_t = [\tilde{u}_t^{*\mathrm{T}}(0), \tilde{\mu}_t^{*\mathrm{T}}(0), \cdots, \tilde{u}_t^{*\mathrm{T}}(N-1), \tilde{\mu}_t^{*\mathrm{T}}(N-1)]^{\mathrm{T}}$$

Meanwhile, the equations (12), (15) and (16) can be regarded as one matrix equation with N-dimensions defined as follows:

$$F(U_t, x_t, t) = \begin{bmatrix} H_{\tilde{u}_t}^T(x_t^*(0), x_d^*(0), \tilde{u}_t^*(0), \lambda_t^*(1), \tilde{\mu}_t^*(0)) \\ C(x_t^*(0), u_t^*(0)) \\ C_1(u_t^*(0), u_t^{'*}(0)) \\ \vdots \\ H_{\tilde{u}_t}^T(x_t^*(N-1), x_d^*(N-1), \tilde{u}_t^*(N-1), \\ \lambda_t^*(N), \tilde{\mu}_t^*(N-1)) \\ C(x_t^*(N-1), u_t^*(N-1)) \\ C_1(u_t^*(N-1), u_t^{'*}(N-1)) \end{bmatrix} = 0 \quad (17)$$

For a given U_t and x_t, $x_t^*(k)$ can be calculated recursively by equations (10) and (14); and then $\lambda_t^*(k), (k = 0, .., k)$ can be reverse calculated from $N \to 0$ by equations (11) and (13). Herein, the C/GMRES method is adopted to solve the matrix equation (17), which is detailedly described in (Ohtsuka [2004]).

In C/GMRES method, instead of solving equation (17) itself at each time, we just choose the proper initial value $U_t(0)$ and take the time derivative of the equation (17) in account. Specifically, the computational process is expressed as following equations,

$$\begin{aligned} &F(U_t(0), x_t(0), 0) = 0 \\ &\dot{F}(U_t, x_t, t) = -\zeta F(U_t, x_t, t) \end{aligned}$$

where, $\zeta > 0$. Furthermore, the above equations can be written as following forms if the F_{U_t} is nonsingular,

$$\dot{U}_t = F_{U_t}^{-1}(-\zeta F - F_{x_t}\dot{x}_t - F_t) \quad (18)$$

where, the Jacobians F_{U_t}, F_{x_t} and F_t are obtained by forward difference approximations and GMRES method in view of the computational load. As an iterative method for numerical solution of the nonsymmetric system of linear equations, GMRES can approximate the solution by the vector in a Krylov subspace with minimal residual. It can obtain the approximate solution with a fewer iterative calculations (Saad [2003]). Based on the GMRES, $\dot{U}_t$ can be obtained and U_t is accordingly calculated by integrating $\dot{U}_t$ in real time.

3.2 Control structure for speed control

Based on the above algorithm, the optimal control law can be obtained and it is expressed as optimal air mass flow rate $\dot{m}_{th}^*$ as following form,

$$\begin{aligned}\dot{m}_{th}^* = J\{v^* + \frac{d}{J}\dot{\omega}_d + \ddot{\omega}_d - \frac{1}{J^2}p_m f_1(\omega)\\ [g_1(\omega)p_m + g_2(\omega) - d\omega - \tau_L]\}/g_1(\omega)a\\ +f_0(\omega, p_m)p_m\omega\end{aligned} \tag{19}$$

where, v^* is the optimal solution calculated by the NMPC controller.

However, it should be noted that the air mass flow rate is not the direct manipulate variable in practical engine system, it has to be converted to the throttle angle command. In this study, a PI controller is introduced to achieve the throttle adjustment. The expression of PI controller can be written as follows:

$$\phi = k_{p1}(\dot{m}_{th}^* - \dot{m}_{th}) + k_{i1}\int(\dot{m}_{th}^* - \dot{m}_{th})dt \tag{20}$$

where, ϕ is the throttle opening angle, k_{p1} and k_{i1} are proportional gain and integral gain, respectively; $\dot{m}_{th}$ is the actual air mass flow rate. The diagram of the proposed control scheme is illustrated in Fig.2.

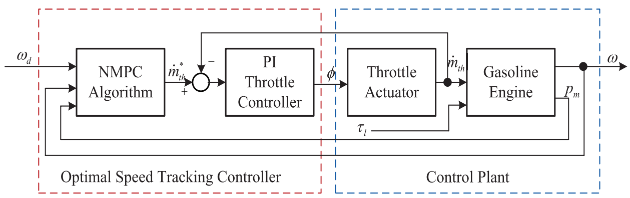


Fig. 2. The block diagram of the control scheme

4. SIMULATION AND EXPERIMENTS

In this study, both simulation and experiments have been conducted to verify the effectiveness of the proposed N-MPC controller. Simulation was done in the Simulink platform based on an estimated engine model (Li [2013]). While, the experiments were performed on a 3.5L -V6 gasoline engine test bench (supported by Toyota Motor Inc.). For comparison, a PID control scheme was also designed to achieve the engine speed tracking.

4.1 Simulation results

In the simulation tests, the predictive horizon is set as $5s$ and the control period is $0.01s$. Several typical test commands are implemented to verify the speed tracking performance over a wide operating range and under the different external load disturbances.

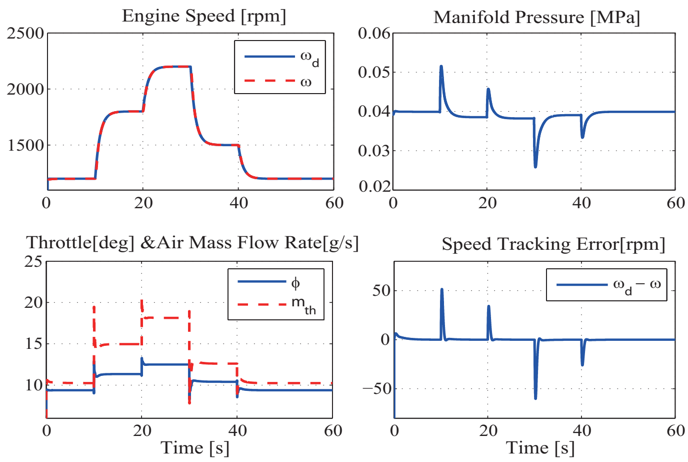


Fig. 3. Simulation results with step command tests

Fig. 3 shows the speed response with a sequence of step commands. In this simulation, the load torque is given as 50 Nm, the desired speed command changes from 1200 rpm to the 2200 rpm. The actual speed response is shown in the left top figure. Besides, as the important state variables, the dynamics of intake manifold pressure and air mass flow rate are also illustrated in Fig. 3. The right bottom figure shows the speed tracking error within a margin of 60 rpm during the transient process. The simulation results prove that the proposed control scheme obtains the good tracking performance.

Fig.4 shows another control results with a sinusoidal command inputs. The load torque is set as 70 Nm in this simulation. It is obviously seen from the right bottom figure that the speed tracking error is generally limited under 10 rpm. Meanwhile, these simulation results also verify the availability of the NMPC scheme in speed tracking control. Similar to the simulation tests, the experiments on the gasoline engine test bench have been further conducted.

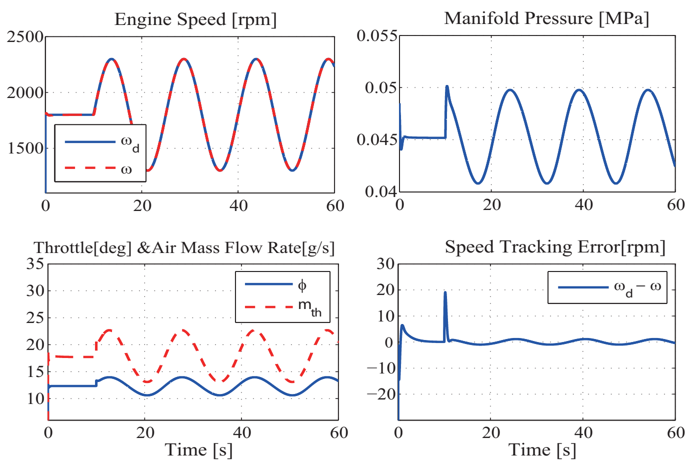


Fig. 4. Simulation results with sine command tests

4.2 Experimental results

The gasoline engine based test bench has been prepared to verify the proposed controller, as shown in Fig. 5. In this test bench, the engine is coupled with a low-inertia dynamometer, which can emulate the change of the external load as fast as possible.

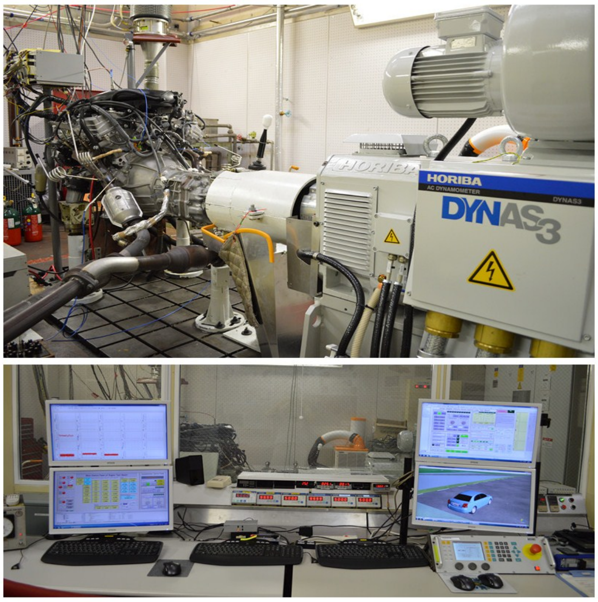
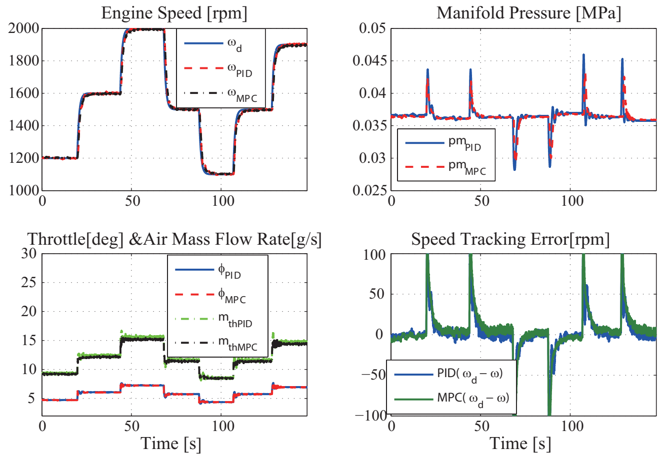


Fig. 5. The gasoline engine test platform

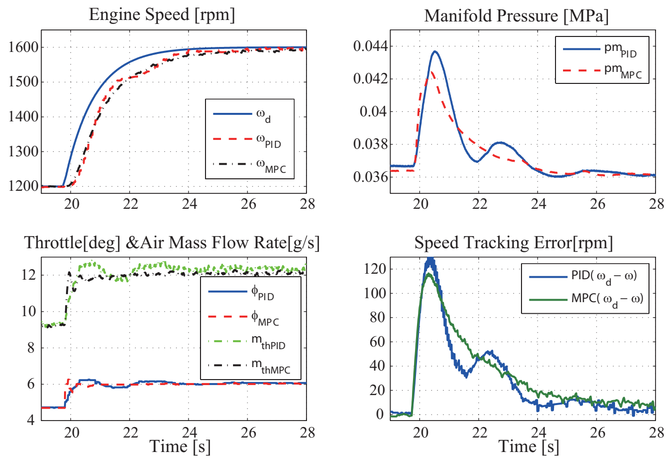
In order to control the actuators inside the engine, a dSPACE processor, which is regarded as the top-level controller, is installed and connected to the engine ECU. The engine ECU provides the free interface and completes the data exchange with dSPACE via CAN-bus. Herein, the proposed NMPC control algorithm can be complied into the dSPACE board in C-code version. The final control command will be performed by actuators with a very short signal transmission delay, which can be ignored.

Meanwhile, some necessary signals including intake manifold pressure, air mass flow rate, and engine speed can be obtained by the additional sensors. The NMPC controller calculates the optimal control variable by using these signals and update the state variables at every calculation step. In the experiments, the control period of NMPC controller is $0.01s$, the predictive horizon is $5s$ and predictive step length is $0.025s$. That is, the predictive steps are 200 but only the first-step control action is applied to the actual throttle actuator. Besides, PID control parameters are chosen as $k_P = 0.01, k_I = 0.006, k_D = 0.002$.

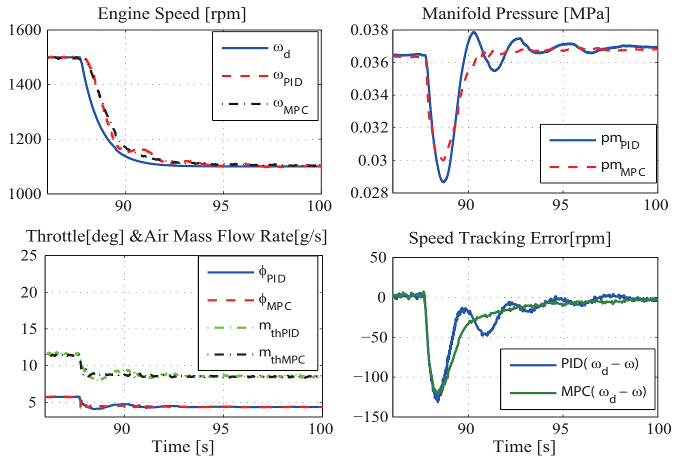
Fig.6 and Fig.7 show the experiment results with different test command. Here, the dynamometer works on the torque control mode and keeps the load torque constant, 40 Nm. In Fig.6(a), the desired speed performs along the step change from 1200 rpm to 2000 rpm. The actual speed responses using both MPC and PID controllers show the good tracking performance and the tracking errors are constrained under 200 rpm in the transient process, while only 10 rpm at most during the steady conditions using MPC controller. The throttle angle is also controlled well by the PI scheme. However, the PID control easily induces the overshoot and has a longer adjustment time. The detailed control effects are illustrated in Fig.6(b) and Fig.6(c).



(a) The overall control effect for step test



(b) Drawing of partial enlargement for step test

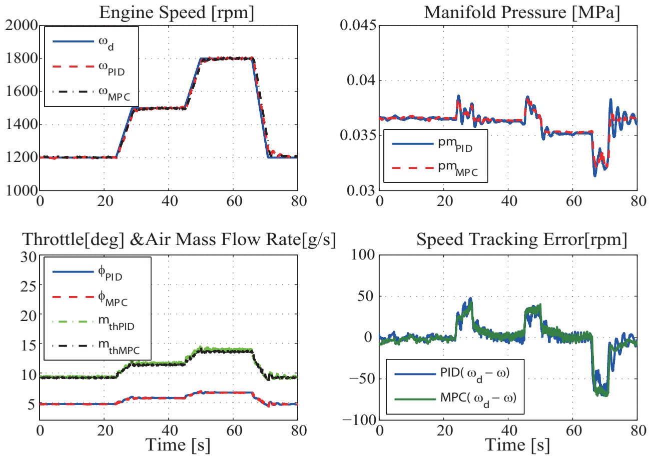


(c) Drawing of partial enlargement for step test

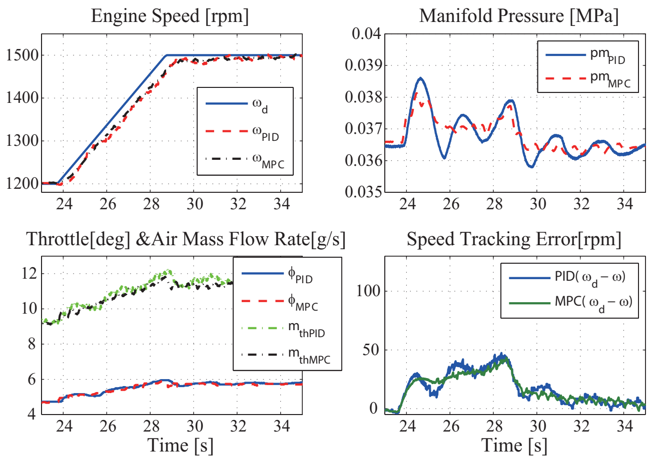
Fig. 6. Experimental results with step command test

The ramp test is also conducted, as shown in Fig.7. In this test, the actual response of engine speed is more smooth and it obtains a smaller tracking error in comparison with the step speed command. In comparison with PID control, the control effects are nearly the same using both controllers.

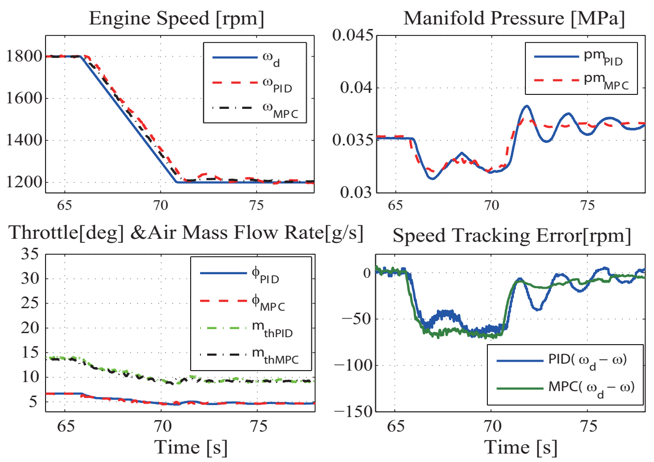
According to the experimental results, it can be concluded that the proposed NMPC controller can keep a good tracking performance both in transient state and steady



(a) The overall control effect for ramp test



(b) Drawing of partial enlargement for ramp test



(c) Drawing of partial enlargement for ramp test

Fig. 7. Experimental results with ramp command test

state operating conditions. The error dynamics system formulated in the controller is beneficial for handling tracking problems.

5. CONCLUSIONS

The model-based engine speed tracking control has been studied in this paper. A nonlinear receding horizon optimal controller based on the error dynamical system is designed to achieve the speed tracking. The simulation and experimental results verify the good control performance for the proposed NMPC controller. The engine speed can be controlled well at a wide operate range and different external load torques. The NMPC control scheme obtains a good robustness and effectiveness in the engine speed tracking problem. Besides, in comparison with PID control, the proposed NMPC controller has a shorter adjustment time and a better transient performance. The future work will use throttle opening as the control input of NMPC problem instead of air mass flow rate. Disturbance and uncertainty will also be considered.

ACKNOWLEDGEMENTS

The authors would like to express their thanks for the technical supports from the Toyota Motor Corporation. Besides, the authors also would like to thank Fei Li (YanShan Univ., China) for her work on the experiments and her assistance in the program design.

REFERENCES

S. Di. Cairano, D. Yanakiev, A. Bemporad, I. V. Kolmanovsky, D. Hrovat. Model predictive idle speed control: design, analysis, and experimental evaluation, *IEEE Transactions on Control Systems Technology*, **20**-1, pp.84-97 ,2012.

F. Xu, H. Chen, X. Gong, and Y. Hu. Engine idle speed control using nonlinear model predictive control. *7th IFAC Symposium on Advances in Automotive Control*, Tokyo, Japan, Sep. 4-7, 2013.

J. Zhang, T. Shen, R. Marino. Model-based cold-start speed control scheme for spark ignition engines. *Control Engineering Practice*, **18**-11, pages 1285–1294, 2010.

A. Stotsky, B. Egardt, S. Eriksson. Variables structure control of engine idle speed with estimation of unmeasurable disturbances. *Journal of Dynamic Systems, Measurement, and Control*, **122**-4, pages 599–603, 2000.

D. Hrovat, S. Di. Cairano, H. E. Tseng, I. V. Kolmanovsky. The development of model predictive control in automotive industry: a survey. *IEEE International Conference on Control Applications (CCA)*, pages 295–302, Dubrovnik, 3-5 Oct. 2012.

A. Bemporad, F. Borrelli, M. Morari. Model predictive control based on linear programming-the explicit solution. *IEEE Transactions on Automatic Control*, volume 47, issue 12, pages 1974–1985, 2002.

Y. Wang, S. Boyd. Fast model predictive control using online optimization. *IEEE Transactions on Control Systems Technology*, volume 18, issue 2, pages 267–278, 2010.

T. Ohtsuka. A continuation/GMRES method for fast computation of nonlinear receding horizon control. *Automatica*, volume 40,issue 4, pages 563–574, 2004.

Lino Guzzella, Christopher H. Onder. Introduction to Modeling and Control of Internal Combustion Engine System. *Springer*, 2010.

F. Li, T. Shen, X. Jiao. Model-based design approach for gasoline engine control Part I: modeling and validation. *Chinese Control Conference*, Xi'an, China, pages 7774–7779, 26-28, July, 2013.

Y. Saad. Iterative Methods for Sparse Linear Systems, 2nd edition, *Society for Industrial and Applied Mathematics*, April 30, 2003.