Model Predictive Control for Real Time Engine Applications

Filip Axelhed, Camilla Bodin, Sebastian Engberg Fors, Marcus Grip, Esaias Jerrelind, Carolina Pahkasalo, Khalid Pirgul & André Sollander

Abstract

This automatic control CDIO-project was performed during fall 2019 in a collaboration between LiU and Volvo Cars AB. The goal was to control an internal combustion engine using a model predictive controller in real time. During the project a simulation environment was developed and the controller was implemented in an engine laboratory.

Simulation Environment

A simulation environment was used to test different controllers on different models before implementation in engine laboratory. The structure of the simulation environment is illustrated in the following figure.

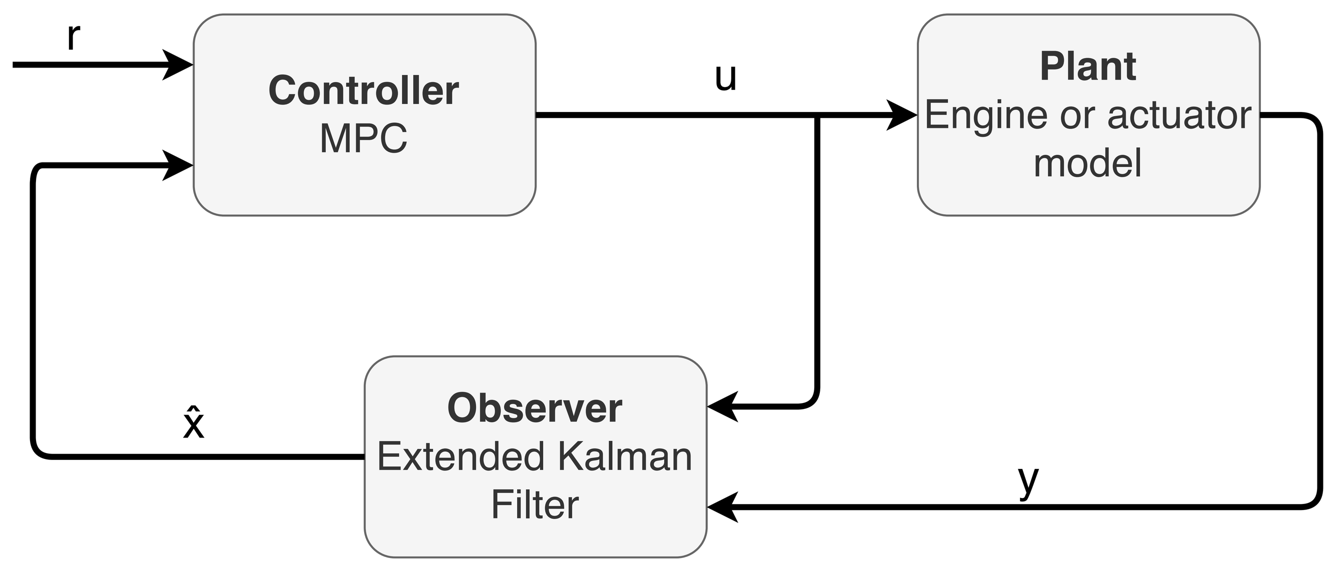

Figure 1: Illustration of the simulation environment.

Actuators and models

The actuators that have been the focus in this project are the throttle and the variable valve timing (VVT). These two actuators have been modeled to make it possible to control them. Only the throttle resulted in a model that could be controlled successfully. The model used for the mass flow over the throttle is described as

$$\dot{m}_{at} = \frac{p_{bef,thr}}{\sqrt{RT_{bef,thr}}} A_{eff}(\alpha_{th}) \Psi_{cv}(\Pi)$$

The model was complemented with the Ohata model described as

$$\Psi_{cv}(\Pi) = \sqrt{\frac{\gamma+1}{2\gamma}(1-\Pi_{lim})(\Pi_{lim} + \frac{\gamma-1}{\gamma+1})}$$

Control system

The control system consisted of an MPC that used a linearized and discretized model of the controlled system. The linearization was done using Taylor expansion and the discretization using the Euler method. The final MPC design had reference tracking and integration of error, resulting in the following cost function:

$$J_N(x(k)) = \sum_{j=0}^{N-1} ||z(k+j) - r(k+j)||^2_{Q_1} + ||\Delta u(k+j)||^2_{Q_2}$$

Where $u = u_0 + \Delta u$ and u_0 is the linearization point. Three states were used in the MPC;

$$x = [p_{im}, \alpha, I_e]$$

Where the third state is used for integration of error. This state is the difference between reference and output, which is controlled to zero to get rid of steady state error. In this project $z = Mx = [p_{im}, I_e]^T$ was used.

For the MPC to find the optimal solution a QP-solver is needed. The QPsolver used in this project was qpOASES, an open source solver which supports real time code generation. It was important that all control elements could be code generated to make it work in the engine laboratory. The following figure illustrates the structure of the controller.

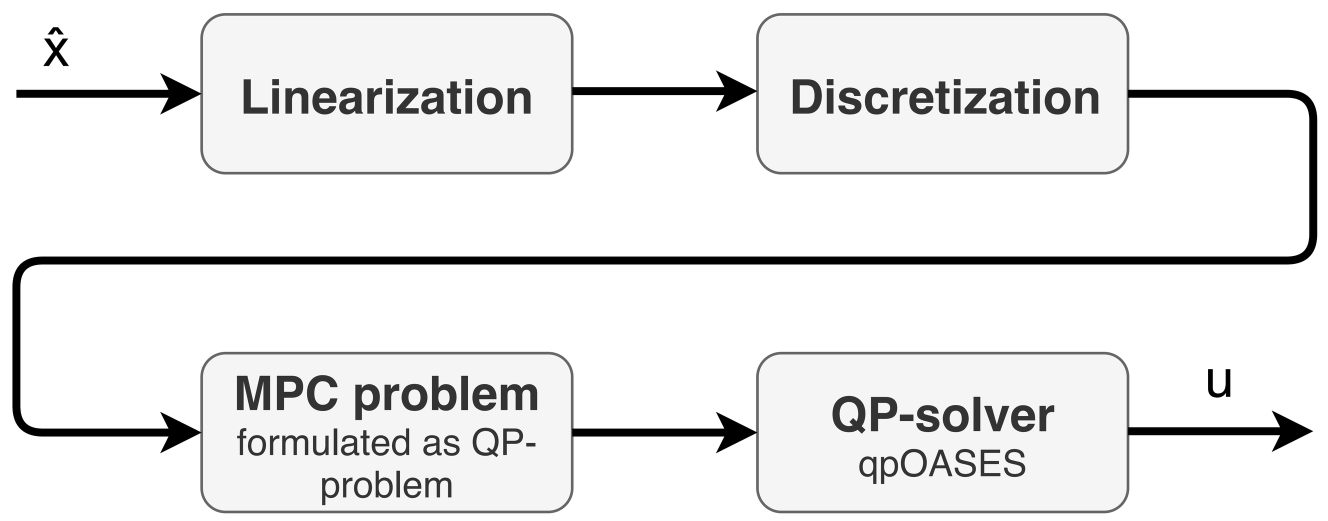

Figure 2: Illustration of the structure of the controller.

Furthermore, an observer was crucial for the controller to work in the engine laboratory and for the full engine model in the simulation environment. An Extended Kalman Filter was used to estimate the states p_{im} and α.

Results and Conclusion

The MPC performed well in the simulation environment, where the steady state error was eliminated and the reference tracking satisfactory, as shown below.

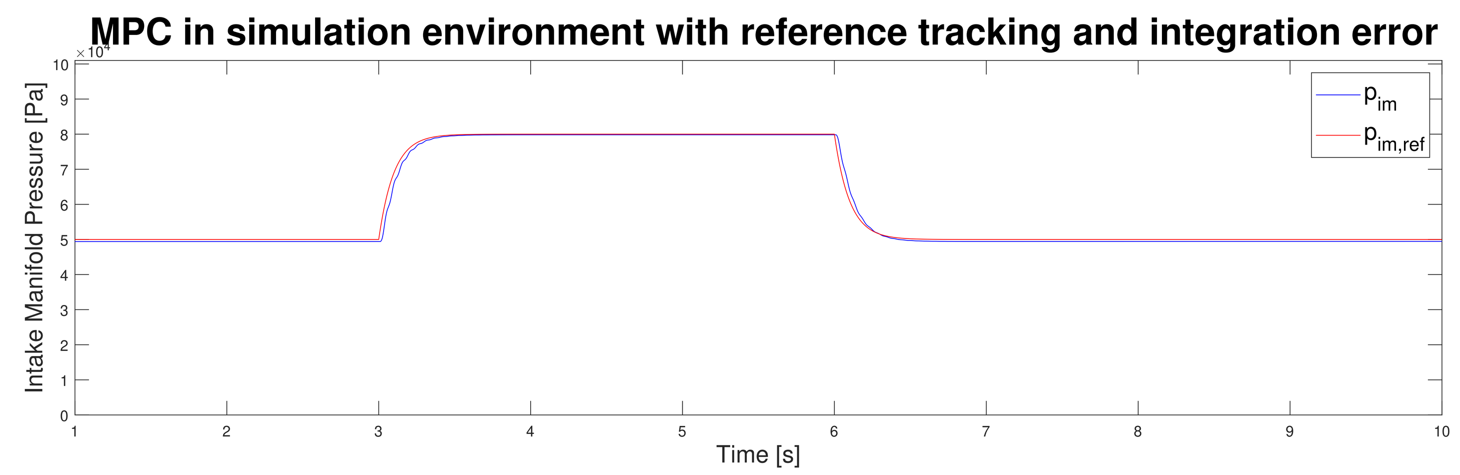

Figure 3: Results of MPC with reference tracking and integration of error in the simulation environment.

The controller in the engine laboratory also performed well. See the figure below.

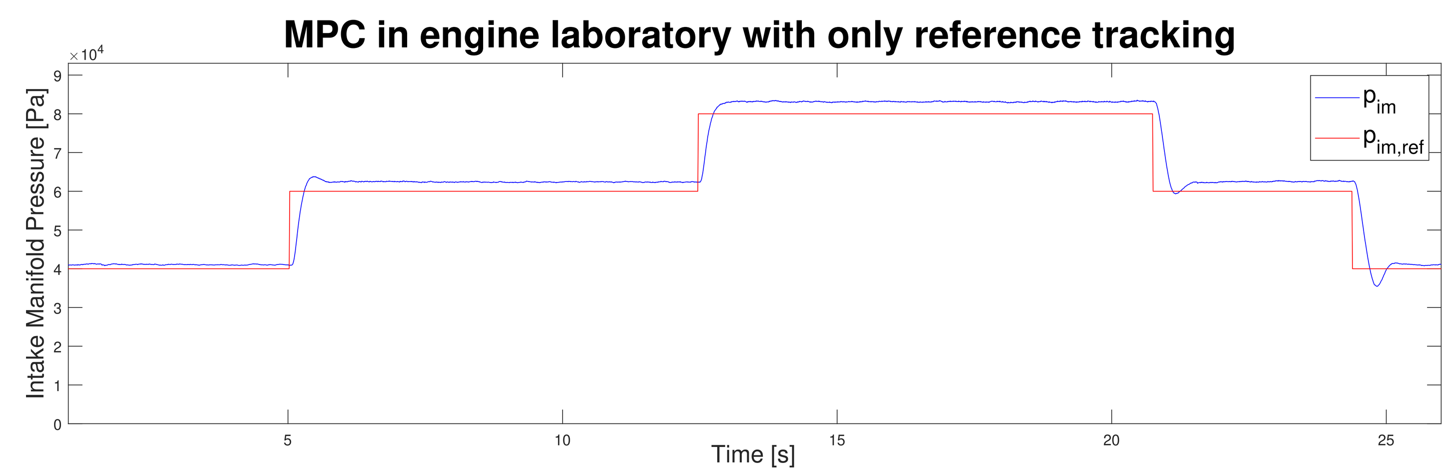

Figure 4: Results using the MPC in the engine laboratory.

Future work

Possible future work could be to change the control signal to air flow. Also, a MIMO MPC could be produced with models for VVT and wastegate.

Acknowledgements

The project group would like to express greatest gratitude towards the mentor Robin Holmbom for guidance and commitment throughout the project. A big thanks to Lars Eriksson and Volvo Cars.

LINKÖPING UNIVERSITY
Division of Automatic Control