IEEE ISIE 2005, June 20-23, 2005, Dubrovnik, Croatia

Explicit Hybrid Optimal Control of Direct Injection Stratified Charge Engines

N. Giorgetti[1], A. Bemporad[1], I.V. Kolmanovsky[2] and D. Hrovat[2]

[1]Dipartimento di Ingegneria dell'Informazione, University of Siena, Siena, Italy.
E-mail: {giorgetti,bemporad}@dii.unisi.it

[2]Ford Research and Advanced Engineering, Dearborn, Michigan.
E-mail: {ikolmano,dhrovat}@ford.com.

Abstract— **This paper illustrates the application of hybrid modeling and receding horizon optimal control techniques to the problem of air-to-fuel ratio and torque management in advanced technology gasoline direct injection stratified charge (DISC) engines. A DISC engine represents an example of a constrained hybrid system, because it can operate in two discrete modes (stratified and homogeneous) and because the mode-dependent constraints on the air-to-fuel ratio and on the spark timing need to be enforced. The paper extends the prior work by the authors [1] and reports the development of an explicit controller which implements the optimal solution using piecewise affine functions of the state, thereby avoiding the need for on-line optimization. Strategies to simplify the explicit controller by reducing the number of regions in its characterization are discussed.**

I. Introduction

In direct-injection stratified charge (DISC) engines the operating mode changes between stratified combustion (with non-homogeneous fuel-air mixture across the cylinder) and homogeneous combustion (with homogeneous fuel-air mixture across the cylinder). This combustion mode changes is effected by changing the fuel injection timing between late and early. Due to the reduction in pumping losses in the stratified mode, the fuel economy can be improved, but the stratified operation can only be sustained in a restricted part of the engine operating range.

In DISC engines the torque and air-fuel ratio need to be seamlessly and optimally controlled through coordination of throttle, spark timing and fuelling with the combustion mode selection. In addition, the mode-dependent state and control constraints on the ranges of the air-to-fuel ratio and spark timing need to be enforced.

The existing approaches to this control problem rely on logic-based switching applied to a family of low level controllers, see [2], [3] and references therein. The construction of the switching logic and low level controllers in these references is, to a large extent, DISC engine-specific. From the standpoint of reducing time-to-market, more systematic control design procedures which can be effortlessly applied to various (DISC and non-DISC) engine and powertrain configurations with multiple operating modes and constraints are of significant interest.

The hybrid modelling and the Receding Horizon (RH) optimal control framework discussed in this paper provides, in principle, a systematic control design procedure of this kind. In it, the receding horizon controller is first tuned in simulations on Mixed Logic Dynamical (MLD) characterization of the system dynamics. The tuning process involves adjusting the horizon and the weights in the cost function (which act as knobs with direct influence on shaping the closed-loop response) until the desired performance is achieved. Then the equivalent explicit piecewise linear form of the receding horizon controller is computed off-line by using a multi-parametric solver. If the explicit form of the receding horizon controller has a sufficiently small number of regions, it may be suitable for implementation in the automotive micro-controllers, which, in comparison to the regular PCs, have limited computational capabilities.

In the prior work [1], we already developed a receding horizon controller for the DISC engine model. This controller utilized on-line optimization to seamlessly coordinate throttle, fuel, spark timing and combustion mode selection so that to effect torque and air-to-fuel ratio tracking while enforcing pointwise-in-time constraints on the air-to-fuel ratio and spark timing. In [1] we have also shown that the transient response can be shaped by changing the weights in the cost function; in this way the controller can switch between torque tracking as the primary objective and air-to-fuel ratio tracking as the primary objective, depending on the engine operating conditions.

In the present paper we discuss the explicit form of the receding horizon controller and measures taken to reduce the number of the regions in its representation. They include the use of a 2-norm cost function instead of an ∞-norm one (as in [1]) and the treatment of the reference commands as states with constant dynamics and not the inputs.

The paper is organized as follows. A mathematical model of the DISC engine is summarized in Section II. The hybrid modeling and the optimal control strategy are discussed in Sections III and IV, respectively. In Section V we report the simulation results. The implementation of the control law in the explicit piecewise affine form is discussed in Section VI. Finally, Section VII contains some concluding remarks.

II. Nonlinear model

The controller development and the simulation results are based on a control-oriented, mean-value DISC engine model developed in [4], [5]. For simplicity, the model is restricted to zero exhaust gas recirculation rate and engine speed of 2000 rpm. The intake manifold pressure and mass flow rates into the intake manifold are related by the following equation

$$\dot{p}_m = c_m \left(W_{th} - W_{cyl}\right) = c_m \left(W_{th} - k_{cyl} \cdot p_m\right), \tag{1}$$

where p_m [kPa] is the intake manifold pressure; $c_m = \frac{RT_m}{V_m}$, T_m [K] is the intake temperature, R is the difference of specific heats for air [kJ/kg/K], and V_m is the intake manifold volume [m^3]; W_{th} is the mass flow rate through the electronic throttle [kg/sec] which is a nonlinear function of p_m obtained from a standard orifice flow representation; $W_{cyl} = k_{cyl} p_m$ is the mass flow rate of air into the engine cylinders [kg/sec], and k_{cyl} is a pumping coefficient that depends on the engine speed and intake temperature. The equation (1) for the manifold filling dynamics represents the differentiated ideal gas law under isothermal conditions.

The in-cylinder air-to-fuel ratio is defined as

$$\lambda = \frac{W_{cyl}}{W_f} = \frac{k_{cyl} p_m}{W_f}, \tag{2}$$

where W_f is the mass flow rate of fuel into the engine cylinders [kg/sec].

The engine brake torque is a sum of three terms,

$$\tau = \tau_{mfr} + \tau_{pump} + \tau_{ind}, \tag{3}$$

where τ_{mfr} [Nm] and τ_{pump} [Nm] are the mechanical friction torque and the pumping torque, respectively, and are modelled by affine functions of p_m which also depend on the engine speed. The τ_{ind} [Nm] is the indicated torque,

$$\tau_{ind} = (\theta_a + \theta_b(\delta - \delta_{mbt})^2) W_f, \tag{4}$$

where θ_a, θ_b, δ_{mbt} are affine functions of λ that depend on the spark timing δ and the combustion mode ρ ($\rho = 0$ corresponds to the stratified mode, while $\rho = 1$ corresponds to the homogeneous mode). The binary character of ρ is the main source of "hybridness" in the DISC engine model.

The goal of this paper is to design a control law that generates the inputs W_{th}, W_f, δ, ρ as a function of the measurements or estimates of p_m, τ and λ so that the latter follow some desired reference trajectories $p_{m_{ref}}$, τ_{ref} and λ_{ref}. In addition, constraints must be satisfied which enforce engine operation within a feasible range. Because of the presence of the binary input ρ, we will solve the control problem within a hybrid systems framework. To this end, we need a hybrid model of the DISC engine.

III. Hybrid Model for Control

Hybrid systems provide a unified framework for describing processes evolving according to continuous dynamics, discrete dynamics, and logic rules [6]–[12]. The interest in hybrid systems is mainly motivated by the large variety of practical situations where physical processes interact with digital controllers, as for instance in embedded systems. Several modeling formalisms have been developed to describe hybrid systems [13], [14], among them the class of Mixed Logical Dynamical (MLD) systems [15]. Examples of real-world applications that can be naturally modeled within the MLD framework have been reported in [15]–[17]. The language HYSDEL (HYbrid Systems DEscription Language) was developed in [18] to obtain MLD models from a high level textual description of the hybrid dynamics, and it will be exemplified in this paper. The model described in Section II is transformed into an equivalent discrete-time hybrid model through the following steps:

1) Linearization and time-discretization. We define two operating points, one for the stratified mode: $\tau^d(0) = 40$ Nm, $\lambda^d(0) = 35$, $\delta^d(0) = 16$ deg, $p_m^d(0) = 74.86$ kPa, and the other one for the homogeneous mode: $\tau^d(1) = 40$ Nm, $\lambda^d(1) = 14$, $\delta^d(1) = 14$ deg, $p_m^d(1) = 35.52$ kPa with $W_f^d(\rho) = \frac{k_{cyl} p_m^d(\rho)}{\lambda^d(\rho)}$. We linearize the model (1), (2), (3) around these two points. The resulting linear models are then discretized in time using the exact sampling. Denoting by $T = 0.01$ sec the sampling period and by t the current time instant, Equation (1) becomes

$$\begin{aligned} p_m(t+1) = e^{-Tc_m k_{cyl}} p_m(t) + \\ \tfrac{1}{k_{cyl}}(1 - e^{-Tc_m k_{cyl}}) W_{th}(t), \end{aligned} \tag{5}$$

while Equations (2) and (3) become

$$\lambda(t) = \lambda^d(\rho) + \frac{k_{cyl}}{W_f^d(\rho)} \tilde{p}_m(t) \quad - \frac{k_{cyl} p_m^d(\rho)}{W_f^d(\rho)^2} \tilde{W}_f(t), \tag{6}$$

and

$$\begin{aligned} \tau(t) \;\; = \;\; & \tau^d(\rho) + \left.\frac{\partial \tau}{\partial p_m}\right|_{d(\rho)} \tilde{p}_m + \left.\frac{\partial \tau}{\partial W_f}\right|_{d(\rho)} \tilde{W}_f \\ + \;\; & \left.\frac{\partial \tau}{\partial \delta}\right|_{d(\rho)} \tilde{\delta} + \left.\frac{\partial \tau}{\partial \lambda}\right|_{d(\rho)} \tilde{\lambda}, \end{aligned} \tag{7}$$

where $[\cdot]^d$ denotes the operating point for variable $[\cdot]$, $\tilde{[\cdot]} = [\cdot] - [\cdot]^d$, (ρ) denotes the dependence on the mode ρ of the engine, and the notation $\big|_{d(\rho)}$ denotes the value at $p_m^d(\rho)$, $\lambda^d(\rho)$, $W_f^d(\rho)$, $\delta^d(\rho)$.

2) Integrators. The model is augmented by two integrators to obtain zero offsets in steady-state

$$\epsilon_\tau(t+1) = \epsilon_\tau(t) + T \cdot (\tau_{ref} - \tau), \tag{8a}$$

$$\epsilon_\lambda(t+1) = \epsilon_\lambda(t) + T \cdot (\lambda_{ref} - \lambda), \tag{8b}$$

where $[\cdot]_{ref}$ represents the reference value for variable $[\cdot]$. In particular, this augmentation of the integrators ensured zero offsets in τ and λ from τ_{ref} and λ_{ref} on the nonlinear simulation model, despite some model mismatch between it and the MLD design model.

The hybrid model that will be used later for control design has p_m, ϵ_τ, ϵ_λ as state variables, τ, λ as output variables and W_{th}, W_f, δ, ρ as manipulated variables. The references τ_{ref}, λ_{ref} are passed to the hybrid model as "states" with constant dynamics. This represents the simplest way to introduce references into HYSDEL and reduce the number of free parameters in the computation of explicit controller as we will further discuss in Section VI.

3) Constraints. Constraints are added to guarantee the correct operation of the engine:

a. A constraint on the air-to-fuel ratio. It is due to engine roughness and misfiring at air-to-fuel ratios that are too lean, and increases in hydrocarbon and smoke emissions at air-to-fuel ratios that are too rich. The constraint takes the form

$$\lambda_{min}(\rho) \leq \lambda(t) \leq \lambda_{max}(\rho). \tag{9}$$

Note that the limits $\lambda_{min}(\rho)$, $\lambda_{max}(\rho)$ depend on the combustion mode ρ.

b. A constraint on the mass flow rate through the electronic throttle, $0 \leq W_{th} \leq K$, where K is assumed to be a constant representing the physical limits of the throttle[1]. We subsequently treat W_{th} as a control input and assume that the throttle position is backtracked to provide the specified value of W_{th}.

c. A constraint on the spark timing $\delta(t)$ to avoid excessive engine roughness:

$$0 \leq \delta(t) \leq \delta_{mbt}(\lambda, \rho), \tag{10}$$

where $\delta_{mbt}(\lambda, \rho)$ is modeled as a piecewise affine function of λ and ρ, i.e., $\delta_{mbt}(\lambda, \rho) = k_1\lambda + k_2\rho + k_3$.

The above dynamic equations and constraints have been modeled in HYSDEL . The corresponding list is reported in the Appendix. The HYSDEL compiler translates difference equations and constraints into the MLD system,

$$x(t+1) = Ax(t) + B_1u(t) + B_2\gamma(t) + B_3z(t), \tag{11a}$$
$$y(t) = Cx(t) + D_1u(t) + D_2\gamma(t) + D_3z(t), \tag{11b}$$
$$E_2\gamma(t) + E_3z(t) \leq E_1u(t) + E_4x(t) + E_5. \tag{11c}$$

In our case, $x = [p_m\ \epsilon_\tau\ \epsilon_\lambda\ \tau_{ref}\ \lambda_{ref}] \in \mathbb{R}^5$, $y = [\lambda\ \tau]' \in \mathbb{R}^2$, $u = [W_{th}\ W_f\ \delta\ \rho]' \in \mathbb{R}^3 \times \{0, 1\}$, where W_{th}, W_f, δ, ρ are the manipulated variables and τ_{ref}, λ_{ref} are the references for the integrators, and γ, z are auxiliary variables introduced for the translation of the constraints and dynamics into (11). In general, γ and z are, respectively, a binary and a real auxiliary vector whose values are determined uniquely by the inequalities (11c) once $x(t)$ and $u(t)$ are fixed [15]. In our case the binary vector γ is empty, as no additional Boolean variables are needed to describe the hybrid dynamics of the DISC engine, and $z \in \mathbb{R}^5$.

[1]In a more complete approach to this problem K may be represented as a piecewise affine function of p_m that approximates the orifice equation; this more elaborate representation for K can also be handled with our design approach.

IV. Online RH Controller

Receding horizon control has found many industrial applications and it has been successfully applied to hybrid dynamical systems [15], [16]. In this section we show how we can derive a RH controller for the DISC engine. In the RH philosophy at each instant of time a finite horizon open loop optimization problem is solved, by assuming the current state as the initial condition. The optimization results in a control sequence and only the first element of this sequence is applied to the hybrid system. This process is iteratively repeated at each subsequent time instant thereby providing a feedback mechanism for disturbance rejection and reference tracking. The optimal control problem is defined as:

$$\min_{\xi} J(\xi, x(t)) \triangleq \sum_{k=0}^{N-1} \|Ru_k\|_2 + \|Qy_k\|_2 + \sum_{k=1}^{N} \|Sx_k\|_2, \tag{12a}$$

$$\text{subj. to} \begin{cases} x_0 & = & x(t), \\ x_{k+1} & = & Ax_k + B_1u_k + B_2\gamma_k + B_3z_k, \\ y_k & = & Cx_k + D_1u_k + D_2\gamma_k + D_3z_k, \\ E_2\gamma_k & + & E_3z_k \leq E_1u_k + E_4x_k + E_5, \end{cases} \tag{12b}$$

where N is the control horizon, $x(t)$ is the state of the MLD system at time t, $\xi \triangleq [u_0', \gamma_0', z_0', \ldots, u_{N-1}', \gamma_{N-1}', z_{N-1}']'$ is the optimization vector, Q, R and S are weighting matrices, and $\|\cdot\|_2$ is the 2-norm (i.e., $\|Qy\|_2 = y'Qy$). In our case,

$$y_k = [\tau_k - \tau_{ref},\ \lambda_k - \lambda_{ref}]', \tag{13a}$$
$$u_k = [W_{th,k} - W_{th,ref},\ W_{f,k} - W_{f,ref},\ \delta_k - \delta_{ref},\ \rho_k - \rho_{ref}]', \tag{13b}$$
$$x_k = [p_{m,k} - p_{m,ref},\ \epsilon_\tau,\ \epsilon_\lambda]', \tag{13c}$$

and

$$Q = \begin{pmatrix} q_\tau & 0 \\ 0 & q_\lambda \end{pmatrix}, R = \begin{pmatrix} r_{W_{th}} & 0 & 0 & 0 \\ 0 & r_{W_f} & 0 & 0 \\ 0 & 0 & r_\delta & 0 \\ 0 & 0 & 0 & r_\rho \end{pmatrix},$$
$$S = \begin{pmatrix} s_{p_m} & 0 & 0 \\ 0 & s_{\epsilon_\tau} & 0 \\ 0 & 0 & s_{\epsilon_\lambda} \end{pmatrix}. \tag{14}$$

The reference values for $W_{th,ref}$, $W_{f,ref}$, and $p_{m,ref}$ are calculated to match in steady-state the values of τ_{ref}, λ_{ref}, δ_{ref} and ρ_{ref} according to the specified feed-forward maps.

In (12) we assume that possible physical and/or logical constraints on the variables of the hybrid system are already included in the mixed-integer linear constraints of the MLD model, as they can be conveniently modeled through the language HYSDEL . Problem (12) can be translated into a mixed integer quadratic program (MIQP), i.e., into the minimization of a quadratic cost function subject to linear constraints, where some of the variables are constrained to be binary, see [19] for details.

Since the RH controller based on the optimal control problem (12) can not be directly implemented on standard

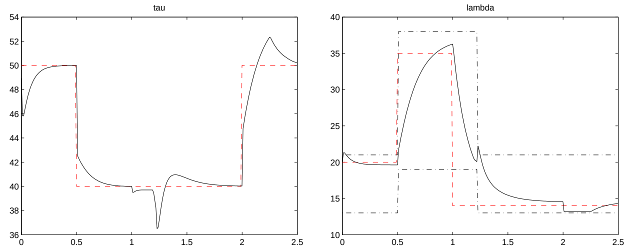

Fig. 1. Left: Time history of engine brake torque $\tau(t)$ (dashed line: desired value, solid line: response of the nonlinear model). Right: Time history of air-to-fuel ratio $\lambda(t)$ (dashed line: desired value, solid line: response of the nonlinear model, dash-dot line: A/F bounds).

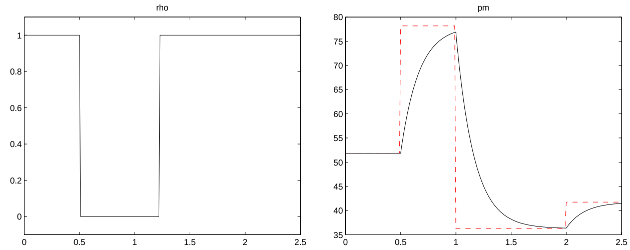

Fig. 2. Left: Time history of combustion mode $\rho(t)$. Right: Time history of intake manifold pressure $p_m(t)$ (dashed line: desired value, solid line: response of the nonlinear model).

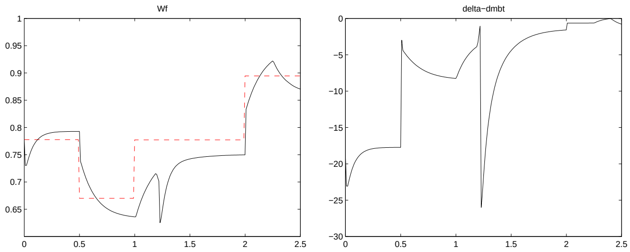

Fig. 3. Left: Time history of fuel mass flow rate $W_f(t)$ (dashed line: desired value, solid line: response of the controller). Right: Time history of spark retard from MBT, $\delta(t) - \delta_{MBT}(t)$.

automotive control hardware, as it would require an MIQP to be solved on-line, the design of the controller is performed in two steps. First, the RH controller is tuned in simulation using MIQP solvers, until the desired performance is achieved. Then, for implementation purposes, the explicit piecewise affine form of the RH law (see Section VI) is computed off-line by using a multiparametric mixed integer quadratic programming solver [20]. The value of the resulting piecewise affine control function is identical to the one which would be calculated by the RH controller designed in the first phase, but the on-line complexity is reduced to the simple function evaluation instead of on-line optimization.

V. Simulations

The closed-loop behavior of the DISC engine under RH control has been evaluated through the simulations on the nonlinear model (1)–(4). We selected the control horizon as $N = 1$, and we set the weights as

$$\begin{aligned} &q_\tau = 10, \qquad q_\lambda = 0.01, \\ &r_{W_{th}} = 1, \qquad r_{W_f} = 1, \qquad r_\delta = 1, \qquad r_\rho = 0.1, \\ &s_{p_m} = 0.1, \qquad s_{\epsilon_\tau} = 10^4, \qquad s_{\epsilon_\lambda} = 10. \end{aligned}$$

Note that we selected q_τ much greater than q_λ and set s_{ϵ_τ} to a large value to emphasize torque tracking as the primary objective. The weight r_ρ was set to a sufficiently small value to leave enough freedom to choose the best mode at each time instant, yet not too small in order to avoid chattering behavior.

We have assumed the following mode-dependent bounds on the air-to-fuel ratio:

$$(\rho = 0) \begin{cases} \lambda_{max} = 38 \\ \lambda_{min} = 19 \end{cases}, \quad (\rho = 1) \begin{cases} \lambda_{max} = 21 \\ \lambda_{min} = 13 \end{cases}.$$

The resulting closed-loop responses are shown in Figures 1–3.

The simulation scenario starts in the homogeneous combustion mode. A step reduction in the torque command occurs at $t = 0.5$ sec, in response to which the controller changes the combustion mode from homogeneous to stratified synergistically with the adjustment of throttle, spark and fuel rate, thereby realizing fuel economy benefits of stratified operation. At $t = 1$ sec the air-to-fuel ratio command changes to the value of 14 (e.g., in response to a request to purge the Lean NOx Trap), which causes the controller to switch the combustion mode back to homogeneous synergistically with the adjustment of throttle, spark and fuel.

It took approximately 8 sec to complete the simulations on a PC Intel Centrino 1.2 GHz running the Hybrid Toolbox for Matlab [19] and the MIQP solver of CPLEX [21], that is about 32 msec per time step and slower than the real time. Therefore, the receding horizon optimal controller which uses the on-line optimization cannot be directly implemented on the automotive hardware, both because of the excessive CPU requirements and software complexity. This problem is dealt with in the next section.

VI. Explicit RH Controller

We are interested in an explicit representation [22], $u(t) = f(\theta(t))$, of the receding horizon control law (12), with $u = [W_{th}\ W_f\ \delta\ \rho]'$ and $\theta = [p_m\ \epsilon_\tau\ \epsilon_\lambda\ \tau_{\rm ref}\ \lambda_{\rm ref}\ p_{m,{\rm ref}}\ W_{th,{\rm ref}}\ W_{f,{\rm ref}}\ \delta_{\rm ref}]'$. Such an explicit controller can be obtained by using the Hybrid Toolbox for Matlab [19] and it can be viewed as a collection of affine gains over (possibly overlapping) polyhedral partitions (regions) of the set of parameters.

For a control horizon $N = 1$ we started with an initial control design containing 490 regions and succeeded in reducing this number to 75 using the following design steps. Firstly, we switched to a 2-norm cost function in (12a) instead of the ∞-norm cost function (which was used by us earlier in [1]). We have found that the 2-norm cost function provided better on-line performance and also resulted in a smaller number of regions because of the reduction in the number of constraints in the multi-parametric problem. Secondly, we implemented the controller in a semi-explicit form. Specifically, the Hybrid

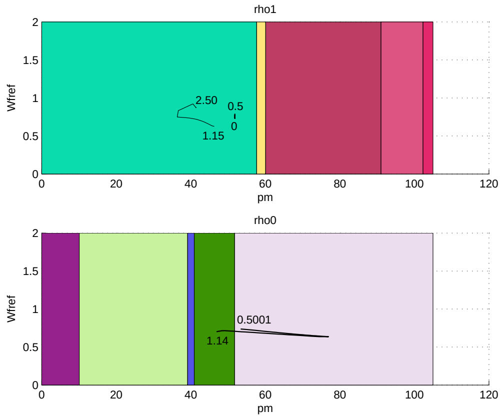

Fig. 4. Cross-sections of the controller regions by p_m-$W_{f,\mathrm{ref}}$ plane and the simulated closed-loop trajectory with time stamps. Upper subplot: $\rho = 1$, lower subplot: $\rho = 0$.

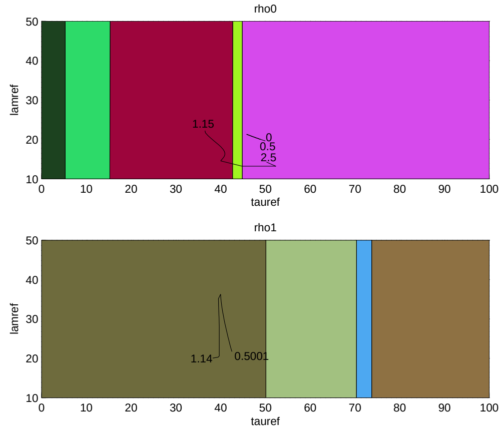

Fig. 5. Cross-sections of the controller regions by τ_{ref}-λ_{ref} plane and the simulated closed-loop trajectory with time stamps. Upper subplot: $\rho = 0$, lower subplot: $\rho = 1$.

Toolbox was applied to produce explicit representations of two control laws for the continuously-valued inputs W_{th}, W_f and δ separately for $\rho = 0$ and $\rho = 1$. The optimal value for ρ was then selected on-line based on the comparison of the value functions for $\rho = 0$ and $\rho = 1$. Finally, we chose to treat the references of the integrators as states and not inputs; this reduced the number of free parameters and led also to the reduction in the number of regions.

In Figure 4 we illustrate the cross-sections of the explicit controller regions for $\rho = 0$ and $\rho = 1$ cases by the p_m-$W_{f,ref}$ plane assuming that $\epsilon_\tau = 0$, $\epsilon_\lambda = 0$, $\tau_{ref} = 50$ Nm, $\lambda_{ref} = 20$, $p_{m,\mathrm{ref}} = 51.84$ kPa, $W_{th,\mathrm{ref}} = 15.55$, and $\delta_{ref} = 16$. In Figure 4 we also show the trajectory of $(p_m(t), W_{f,ref}(t))$ corresponding to the simulations in Figures 1–3. Note that as ρ changes the trajectory migrates from the upper plot to the lower plot and back at 0.5 sec and 1.15 sec, respectively. In Figure 5 we illustrate the cross-sections of the explicit controller regions for $\rho = 0$ and $\rho = 1$ cases by the τ_{ref}-λ_{ref} plane, assuming that $p_m = 51.84$ kPa, $\epsilon_\tau = 0$, $\epsilon_\lambda = 0$, $p_{m,\mathrm{ref}} = 51.84$ kPa, $W_{th,\mathrm{ref}} = 15.55$, $W_{f,\mathrm{ref}} = 0.78$ and $\delta_{ref} = 16$. The $(\tau(t), \lambda(t))$ trajectory corresponding to the simulations in Figures 1–3 is also shown.

While the on-line solution of the RH optimal control problem and its explicit off-line solution provide the same result, the explicit controller requires a lower computational effort. The simulations, in fact, take just 0.52 sec, which is about 2 msec per time step whereas the sampling period is $T = 10$ msec. The Hybrid Toolbox for Matlab [19] provides an option for automatically generating the controller C-code which can then be used for the embedded implementation in the production microcontroller. This approach was pursued[2], requiring 43 Kb of ROM (this is a total size of code and constants) and execution time of 3 msec to calculate $u(t)$.

If we consider a control horizon $N = 2$ the number of regions increases to 5637. In Figure 6 we report the simulations for this control horizon. The closed loop behavior with $N = 2$ is very similar to the case $N = 1$ which suggests that the control horizon $N = 1$ is adequate.

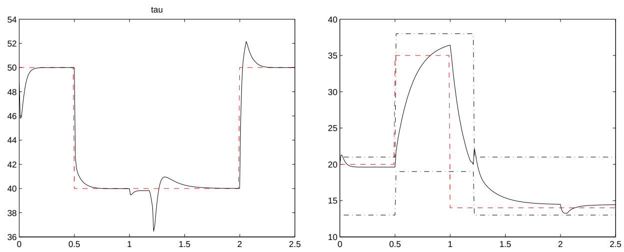

Fig. 6. Left: Time history of engine brake torque $\tau(t)$ for $N = 2$ (dashed line: desired value, solid line: response of the nonlinear model). Right: Time history of air-to-fuel ratio $\lambda(t)$ for $N = 2$ (dashed line: desired value, solid line: response of the nonlinear model, dash-dot line: A/F bounds).

VII. Conclusions

In this paper we described an approach to implementing a receding horizon (RH) hybrid optimal controller for the DISC engine. The controller simultaneously manipulates discrete and continuous control inputs of the engine to effect torque and air-to-fuel ratio tracking and to enforce pointwise-in-time state and control constraints on the air-to-fuel ratio and spark timing. The explicit implementation of the RH controller, in the form of a piecewise affine control law computed off-line, avoids the need for on-line optimization altogether. Still, a large number of regions in the explicit controller implementation may make it prohibitive for the memory and chronometrics constrained automotive micro-controllers. Strategies to reduce the number of regions in the polyhedral partition of the explicit controller have been discussed in the paper and their application has been shown to reduce the number of regions by factor of more than 6.

VIII. Appendix: HYSDEL model of DISC engine

```
/*
```

[2]The authors wish to acknowledge Mr. Craig Cox of Ford Motor Company for performing the production microcontroller assessment.

```
 HYSDEL model of open-loop DISC engine.

 Model: Courtesy of Ford Research Laboratory
        (D. Hrovat and I. Kolmanovsky)

 Purpose: Used to generate MPC controller

 (C) 2003-2005 N. Giorgetti, A. Bemporad.
*/

SYSTEM hysdisc{
   INTERFACE{
      STATE{
         REAL pm       [1, 101.325];
         REAL xtau     [-1e3, 1e3];
         REAL xlam     [-1e3, 1e3];
         REAL taud     [0,    100];
         REAL lamd     [10,    60];}
      OUTPUT {
         REAL lambda;
         REAL tau;}
      INPUT{
         REAL Wth      [0,38.5218];
         REAL Wf       [0,      2];
         REAL delta    [0,     40];
         BOOL rho;}
      PARAMETER{
         REAL Ts;
         REAL pm1, pm2;
         REAL l01, l02, l0c;
         REAL l11,l12,l1c;
         REAL t01,t02,t03,t04,t05;
         REAL t11,t12,t13,t14,t15;}
   }
   IMPLEMENTATION{
      AUX{
          REAL lam;
          REAL taul;
          REAL lmin,lmax;
          REAL dmbt;}
      DA{
         lam={ IF rho THEN l11*pm+l12*Wf+l1c
               ELSE   l01*pm+l02*Wf+l0c      };
         taul={IF rho THEN t11*pm+t12*Wf+
         t13*delta+t14*lam+t15
               ELSE   t01*pm+t02*Wf+
               t03*delta+t04*lam+t05 };
         lmin ={IF rho THEN 13 ELSE 19};
         lmax ={IF rho THEN 21 ELSE 38};
         dmbt ={IF rho THEN -28.74+3.1845*lam
                ELSE 14.0877+0.2810*lam};}
      CONTINUOUS{
         pm=pm1*pm+pm2*Wth;
         xtau=xtau+Ts*(taud-taul);
         xlam=xlam+Ts*(lamd-lam);
         taud=taud;
         lamd=lamd;}
      OUTPUT {
         lambda=lam;
         tau=taul;}
      MUST{
         lmin-lam      <=0;
         lam-lmax      <=0;
         delta-dmbt    <=0;}
   }
}
```

REFERENCES

[1] A. Bemporad, N. Giorgetti, I. Kolmanovsky, and D. Hrovat. A hybrid system approach to modeling and optimal control of disc engines. In *Proc. of IEEE Conf. on Decision and Control*, pages 1582–1587, Las Vegas, Nevada, 2002.

[2] J.Sun, I.V. Kolmanovsky, J. Dixon, and M. Boesch. Control of disi engines: Analytical and experimental investigations. In *Proc. of 3rd IFAC Workshop on Advances in Automotive Control*, pages 249–254, Karsruhe, Germany, 2001.

[3] M.V. Druzhinina, I.V. Kolmanovsky, and J. Sun. Hybrid control of a gasoline direct injection engine. In *Proc. of IEEE Conference on Decision and Control*, Phoenix, Arizona, 1999.

[4] J.Sun, I.V. Kolmanovsky, D. Brehob, J. Cook, J. Buckland, and M. Haghgooie. Modelling and control problems for gasoline direct injection engines. In *Proc. of IEEE Conf. Control Applications*, pages 471–477, Hawaii, 1999.

[5] I.V. Kolmanovsky, M.V. Druzhinina, and J. Sun. Nonlinear torque and air-to-fuel ratio controller for direct injection stratified charge gasoline engines. In *Proc. of AVEC 2000, 5-th International Symposium on Advanced Vehicle Control*, Ann Arbor, Michigan, 2000.

[6] P.J. Antsaklis. A brief introduction to the theory and applications of hybrid systems. *Proc. IEEE, Special Issue on Hybrid Systems: Theory and Applications*, vol. 88, pages 879–886, 2000.

[7] J. Lygeros, C. Tomlin, and S. Sastry. Controllers for reachability specifications for hybrid systems. *Automatica*, vol. 35, pages 349–370, 1999.

[8] M.S. Branicky. *Studies in hybrid systems: modeling, analysis, and control*. PhD thesis, LIDS-TH 2304, Massachusetts Institute of Technology, Cambridge, MA, 1995.

[9] M.S. Branicky and S.K. Mitter. Algorithms for optimal hybrid control. In *Proc. of IEEE Conf. on Decision and Control*, New Orleans, USA, 1995.

[10] K. Gokbayrak and C.G. Cassandras. A hierarchical decomposition method for optimal control of hybrid systems. In *Proc. of IEEE Conf. on Decision and Control*, pages 1816–1821, Phoenix, AZ, 1999.

[11] S. Hedlund and A. Rantzer. Optimal control of hybrid systems. In *Proc. of IEEE Conf. on Decision and Control*, pages 3972–3976, Phoenix, AZ, 1999.

[12] C.C. Pantelides, M.P. Avraam, and N. Shah. Optimization of hybrid dynamic processes. In *Proc. of American Contr. Conf.*, 2000.

[13] W.P.M.H. Heemels, B. De Schutter, and A. Bemporad. Equivalence of hybrid dynamical models. *Automatica*, vol. 37, pages 1085–1091, July 2001.

[14] G. Labinaz, M.M. Bayoumi, and K. Rudie. A survey of modeling and control of hybrid systems. In *Proc. of 13th IFAC World Congress*, 1996.

[15] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, vol. 35, pages 407–427, 1999.

[16] F. Borrelli, A. Bemporad, M. Fodor, and D. Hrovat. A hybrid approach to traction control. In *Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science. Springer Verlag, 2001.

[17] A. Bemporad, F.D. Torrisi, and M. Morari. Discrete-time hybrid modeling and verification of the batch evaporator process benchmark. *European Journal of Control*, vol. 7, pages 382–399, 2001.

[18] F.D. Torrisi and A. Bemporad. Hysdel - a tool for generating computational hybrid models. *IEEE Trans. Contr. Systems Technology*, vol. 12, pages 235–249, March 2004.

[19] A. Bemporad. Hybrid toolbox for real time applications. Technical report, University of Siena, 2004.

[20] A. Bemporad, M. Morari, V. Dua, and E.N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, vol. 38, pages 3–20, 2002.

[21] ILOG, Inc. *CPLEX 9.0 User Manual*. Gentilly Cedex, France, 2003.

[22] F. Borrelli, M. Baotic, A. Bemporad, and M. Morari. Constrained optimal control of discrete-time linear hybrid systems. In *Proc. of American Control Conference*, 2003.