# 1

# Fast Model Predictive Control and its Application to Energy Management of Hybrid Electric Vehicles

Sajjad Fekri and Francis Assadian
*Automotive Mechatronics Centre, Department of Automotive Engineering*
*School of Engineering, Cranfield University*
*UK*

## 1. Introduction

Modern day automotive engineers are required, among other objectives, to maximize fuel economy and to sustain a reasonably responsive car (i.e. maintain driveability) while still meeting increasingly stringent emission constraints mandated by the government. Towards this end, Hybrid Electric Vehicles (HEVs) have been introduced which typically combine two different sources of power, the traditional internal combustion engine (ICE) with one (or more) electric motors, mainly for optimising fuel efficiency and reducing Carbon Dioxide ($CO_2$) and greenhouse gases (GHG) (Fuhs, 2008).

Compared to the vehicles with conventional ICE, hybrid propulsion systems are potentially capable of improving fuel efficiency for a number of reasons: they are able to recover some portion of vehicle kinetic energy during braking and use this energy for charging the battery and hence, utilise the electric motor at a later point in time as required. Also, if the torque request (demanded by driver) is below a threshold torque, the ICE can be switched off as well as during vehicle stop for avoiding engine idling. These are in fact merely few representative advantages of the hybrid vehicles compared to those of conventional vehicles. There are also other benefits hybrid electric vehicles could offer in general, e.g. engine downsizing and utilising the electric motor/motors to make up for the lost torque. It turns out that the internal combustion engine of the hybrid electric vehicle can be potentially designed with a smaller size and weight which results in higher fuel efficiency and lower emissions (Steinmaurer & Del Re, 2005).

Hybrid electric vehicles have been received with great enthusiasm and attention in recent years (Anderson & Anderson, 2009). On the other hand, complexity of hybrid powertrain systems have been increased to meet end-user demands and to provide enhancements to fuel efficiency as well as meeting new emission standards (Husain, 2003).

The concept of sharing the requested power between the internal combustion engine and electric motor for traction during vehicle operation is referred to as "vehicle supervisory control" or "vehicle energy management" (Hofman & Druten, 2004). The latter term, employed throughout this chapter, is particularly referred to as a control allocation for delivering the required wheel torque to maximize the average fuel economy and sustain the battery state of charge (SoC) within a desired charging range (Fekri & Assadian, 2011).

The vehicle energy management development is a challenging practical control problem and a significant amount of research has been devoted to this field for full HEVs and Electric Vehicles (EVs) in the last decade (Cundev, 2010). To tackle this challenging problem, there are currently extensive academic and industrial research interests ongoing in the area of hybrid electric vehicles as these vehicles are expected to make considerable contributions to the environmentally conscious requirements in the production vehicle sector in the future – see (Baumann et al., 2000) and other references therein.
In this regard, we shall analysis and extend the study done by (Sciarretta & Guzzella, 2007) on the number of IEEE publications published between 1985 and 2010. Figure 1 depicts the number of publications recorded at the IEEE database[1] whose abstract contains at least one of the strings "hybrid vehicle" or "hybrid vehicles".
From Figure 1, it is obvious that the number of publications in the area of hybrid electric vehicles (HEVs) has been drastically increased during this period, from only 2 papers in 1985 to 552 papers in 2010. Recall that these are only publications of the IEEE database - there are many other publications than those of the IEEE including books, articles, conference papers, theses, filed patents, and technical reports which have not been taken into account in this study. Besides, a linear regression analysis of the IEEE publications shown in Figure 1 indicates that research in the field of hybrid vehicles has been accelerated remarkably since 2003. One may also predict that the number of publications in this area could be increased up to about 1000 articles in 2015, that is nearly twice as many as in 2010 - this is a clear evidence to acknowledge that HEVs research and development is expected to make considerable contributions to both academia and industry of production automotive sector in the future.

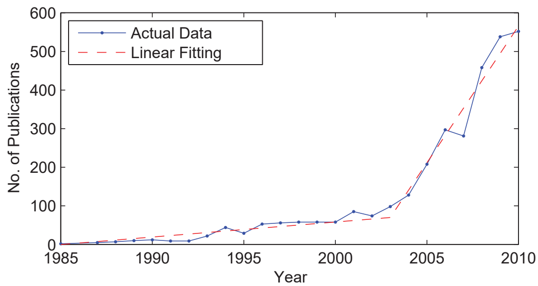



Fig. 1. Hybrid vehicle research trend based on the number of publications of the IEEE over the period 1985 to 2010.

Here are the facts and regulations which must be taken into consideration by automotive engineers:

- Due to the ever increasing stringent regulations on fuel consumption and emissions, there are tremendous mandates on Original Equipment Manufacturers (OEMs) to deliver fuel-efficient less-polluting vehicles at lower costs. Hence, the impact of advanced controls for the application of the hybrid vehicle powertrain controls has become extremely important (Fekri & Assadian, 2011).

[1] See http://ieeexplore.ieee.org for more information.

- It is essential to meet end-user demands for increasingly complex new vehicles towards improving vehicle performance and driveability (Cacciatori et al., 2006), while continuing to reduce costs and meeting new emission standards.
- There is a continuous increase in the gap between the theoretical control advancement and the control strategies being applied to the existing production vehicles. This gap is resulting on significant missed opportunities in addressing some fundamental functionalities, e.g. fuel economy, emissions, driveability, unification of control architecture and integration of the Automotive Mechatronics units on-board vehicle. It seems remarkably vital to address how to bridge this gap.
- Combined with ever-increasing computational power, fast online optimisation algorithms are now more affordable to be developed, tested and implemented in the future production vehicles.

There are a number of energy management methods proposed in the literature of hybrid vehicles to minimize fuel consumption and to reduce $CO_2$ emissions (Johnson et al., 2000). Among these energy management strategies, a number of heuristics techniques, say e.g. using rule-based or Fuzzy logic, have attempted to offer some improvements in the HEV energy efficiency (Cikanek & Bailey, 2002; Schouten et al., 2002) where the optimisation objective is, in a heuristic manner, a function of weighted fuel economy and driveability variables integrated with a performance index, to obtain a desired closed-loop system response. However, such heuristics based energy management approaches suffer from the fact that they guarantee neither an optimal result in real vehicle operational conditions nor a robust performance if system parameters deviate from their nominal operating points. Consequently, other strategies have emerged that are based on optimisation techniques to search for sub-optimal solutions. Most of these control techniques are based on programming concepts (such as linear programming, quadratic programming and dynamic programming) and optimal control concepts, to name but a few (Ramsbottom & Assadian, 2006; Ripaccioli et al., 2009; Sciarretta & Guzzella, 2007). Loosely speaking, these techniques do not offer a feasible casual solution, as the future driving cycle is assumed to be entirely known. Moreover, the required burdensome calculations of these approaches put a high demand on computational resources which prevent them to be implemented on-line in a straightforward manner. Nevertheless, their results could be used as a benchmark for the performance of other strategies, or to derive rules for rule-based strategies for heuristic based energy management of HEVs (Khayyam et al., 2010).

Two new HEV energy management concepts have been recently introduced in the literature. In the first approach, instead of considering one specific driving cycle for calculating an optimal control law, a set of driving cycles is considered resulting in the stochastic optimisation approach. A solution to this approach is calculated off-line and stored in a state-dependent lookup table. Similar approach in this course employs Explicit Model Predictive Control (Beccuti et al., 2007; Pena et al., 2006). In this design methodology, the entire control law is computed offline, where the online controller will be implemented as a lookup table, similar to the stochastic optimisation approach. The lookup table provides a quasi-static control law which is directly applicable to the on-line vehicle implementation. While this method has potential to perform well for systems with fewer states, inputs, constraints, and "sufficiently short" time-horizons (Wang & Boyd, 2008), it cannot be utilised in a wide variety of applications whose dynamics, cost function and/or constraints are time-varying due to e.g.

parametric uncertainties and/or unmeasurable exogenous disturbances. In other words, any lookup table based optimisation approach may end up with severe difficulties in covering a real-world driving situation with a set of individual driving cycle. A recent approach has endeavored to decouple the optimal solution from a driving cycle in a game-theoretic (GT) framework (Dextreit et al., 2008). In this approach, the effect of the time-varying parameters (namely drive cycle) is represented by the actions of the first player while the effect of the operating strategy (energy management) is modeled by the actions of the second player. The first player (drive cycle) wishes to maximize the performance index which reflects the optimisation objectives, say e.g. to minimise emission constraints and fuel consumption, while the second player aims to minimize this performance index. Solutions to these approaches are calculated off-line and stored in a state-dependent lookup tables. These look up tables provide a quasi-static control law which is directly suitable for on-line vehicle implementation. Similar to previous methods, the main drawbacks of the game-theoretic approach are the lack of robustness and due to quasi-static nature of this method, it cannot address vehicle deriveability requirements.

If only the present state of the vehicle is considered, optimisation of the operating points of the individual components can still be beneficial. Typically, the proposed methods define an optimisation criterion to minimise the vehicle fuel consumption and exhaust emissions (Kolmanovsky et al., 2002). A weighting factor can be included to prevent a drift in the battery from its nominal energy level and to guarantee a charge sustaining solution. This approach has been considered in the past, but it is still remained immensely difficult task to select a weighting factor that is mathematically sound (Rousseau et al., 2008). An alternative approach is to extend the objective function with a fuel equivalent term. This term includes the corresponding fuel use for the energy exchange with the battery in the optimisation criterion (Kessels, 2007).

Hybrid modeling tools have been recently developed to analyse and optimise a number of classes of hybrid systems. Among many other modeling tools developed to represent the hybrid systems, we shall refer to Mixed Logical Dynamical (MLD) (Bemporad & Morary, 1999), HYbrid Systems Description Language (HYSDEL) (Torrisi & Bemporad, 2004), and Piecewise Affine (PWA) models (Ripaccioli et al., 2009; Sontag, 1981), to name but a few. In addition, Hybrid Toolbox for MATLAB (Bemporad, 2004) is developed for modeling, simulation, and verifying hybrid dynamical models and also for designing hybrid model predictive controllers. Almost all of these hybrid tools, however, are only suitable for slow applications and can not attack the challenging fast real-time optimisation problems, e.g., for the use of practical hybrid electric vehicle energy management application.

Two fundamental drawbacks of aforementioned strategies are firstly their consideration of driveability being an afterthought and secondly the driveability issue is considered in an ad-hoc fashion as these approaches are not model-based dynamic. Applicable techniques such as game-theoretic based optimisation method utilise quasi-static models which are not sufficient to address driveability requirements (Dextreit et al., 2008).

Towards a feasible and tractable optimisation approach, there are a number of model-based energy management methods such as Model Predictive Controls (MPC). A recently developed package for the hybrid MPC design is referred to as Hybrid and Multi-Parametric Toolboxes (Narciso et al., 2008) which is based on the traditional model predictive control optimisation alternatives using generic optimisers. The main shortcoming of traditional model predictive control methods is that they can only be used in applications with "sufficiently slow" dynamics

(Wang & Boyd, 2008), and hence are not suitable for many practical applications including HEV energy management problem. For this reason the standard MPC algorithms have been retained away from modern production vehicles. In fact, a number of inherent hardware constraints and limitations integrated with the vehicle electronic control unit (ECU), such as processing speed and memory, have made on-line implementations of these traditional predictive algorithms almost impossible. In a number of applications, MPC is currently applied off-line to generate the required maps and then these maps are used on-line. However, generation and utilisation of maps defeat the original purpose of designing a dynamic compensator which maintains driveability. Therefore, there is a vital need of increased processing speed, with an appropriate memory size, so that an online computation of "fast MPC" control law could be implemented in real applications.

In this chapter, we shall describe a method for improving the speed of conventional model predictive control design, using online optimisation. The method proposed would be a complementary for offline methods, which provide a method for fast control computation for the problem of energy management of hybrid electric vehicles. We aim to design and develop a practical fast model predictive feedback controller (FMPC) to replace the current energy management design approaches as well as to address vehicle driveability issues. The proposed FMPC is derived based on the dynamic models of the plant and hence driveability requirements are taken into consideration as part of the controller design. In this development, we shall extend the previous studies carried out by Stephen Boyd and his colleagues at Stanford University, USA, on fast model predictive control algorithms. In this design, we are also able to address customising the robustness analysis in the presence of parametric uncertainties due to, e.g., a change in the dynamics of the plant, or lack of proper estimation of the vehicle load torque (plant disturbance).

In this chapter, we shall also follow and overview some of theoretical and practical aspects of the fast online model predictive control in applying to the practical problem of hybrid electric vehicle energy management along with representing some of simulation results. The novelty of this work is indeed in the design and development of the fast robust model predictive control concept with practical significance of addressing vehicle driveability and automotive actuator control constraints. It is hoped that the results of this work could make automotive engineers more enthusiastic and motivated to keep an eye on the development of state-of-the-art Fast Robust Model Predictive Control (FMPC) and its potential to attack a wide range of applications in the automotive control system designs.

In the remaining of this chapter, we will describe in detail the mathematical description, objectives and constraints along with the optimisation procedure of the proposed fast model predictive control. We shall also provide dynamical model of the hybrid electric vehicle (parallel, with diesel engine) to which the FMPC will be applied. Simulation results of the HEV energy management system will be demonstrated to highlight some of the concepts proposed in this chapter which will offer significant improvements in fuel efficiency over the base system.

## 2. Fast Model Predictive Control

The Model Predictive Control (MPC), referred also to as Receding Horizon Control (RHC), and its different variants have been successfully implemented in a wide range of practical applications in industry, economics, management and finance, to name a few (Camacho &

Bordons, 2004; Maciejowski, 2002). A main advantage of MPC algorithms, which has made these optimisation-based control system designs attractive to the industry, is their abilities to handle the constraints directly in the design procedure (Kwon & Han, 2005). These constraints may be imposed on any part of the system signals, such as states, outputs, inputs, and most importantly actuator control signals which play a key role in the closed-loop system behaviour (Tate & Boyd, 2001).

Although very efficient algorithms can currently be applied to some classes of practical problems, the computational time required for solving the optimisation problem in real-time is extremely high, in particularly for fast processes, such as energy management of hybrid electric vehicles. One method to implement a fast MPC is to compute the solution of a multiparametric quadratic or linear programming problem explicitly as a function of the initial state which could turn into a relatively easy-to-implement piecewise affine controller (Bemporad et al., 2002; Tondel et al., 2003). However, as the control action implemented online is in the form of a lookup table, it could exponentially grow with the horizon, state and input dimensions. This means that any form of explicit MPC could only be applied to small problems with few state dimensions (Milman & Davidson, 2003). Furthermore, due to there being off-line lookup table, explicit MPC cannot deal with applications whose dynamics, cost function and/or constraints are time-varying (Wang & Boyd, 2008). A non-feasible active set method was proposed in (Milman & Davidson, 2003) for solving the Quadratic Programming (QP) optimisation problem of the MPC. However, to bear further explanation, these studies have not addressed any comparison to the other earlier optimisation methods using primal-dual interior point methods (Bartlett et al., 2000; Rao et al., 1998). Another fast MPC strategy was introduced in (Wang & Boyd, 2010) which has tackled the problem of solving a block tridiagonal system of linear equations by coding a particular structure of the QPs arising in MPC applications (Vandenberghe & Boyd, 2004; Wright, 1997), and by solving the problem approximately. Starting from a given initial state and input trajectory, the fast MPC software package solves the optimization problem fast by exploiting its special structure. Due to using an interior-point search direction calculated at each step, any problem of any size (with any number of state dimension, input dimension, and horizon) could be tackled at every operational time step which in return will require only a limited number of steps. Therefore, the complexity of MPC is significantly reduced compared to the standard MPC algorithms. While this algorithm could be scaled in any problem size in principle, a drawback of this method is that it is a custom hand-coded algorithm, ie. the user should transform their problem into the standard form (Wang & Boyd, 2010; 2008) which might be very time-consuming. Moreover, one may require much optimisation expertise to generate a custom solver code. To overcome this shortcoming, a very recent research (Mattingley & Boyd, 2010a;b; 2009) has studied a development of an optimisation software package, referred to as CVXGEN, based on an earlier work by (Vandenberghe, 2010), which automates the conversion process, allowing practitioners to apply easily many class of convex optimisation problem conversions. CVXGEN is effectively a software tool which helps to specify one's problem in a higher level language, similar to other parser solvers such as SeDuMi or SDPT3 (Ling et al., 2008). The drawback of CVXGEN is that it is limited to optimization problems with up to around 4000 non-zero Karush-Kuhn-Tucker (KKT) matrix entries (Mattingley & Boyd, 2010b). In the next section, we will extend the work done by (Mattingley & Boyd, 2010b) and propose a new fast KKT solving approach, which alleviates the aforementioned limitation to some extent. We will implement our method on a hybrid electric vehicle energy management application in Section 4.

### 2.1 Quadratic Programming (QPs)

In convex QP problems, we typically minimize a convex quadratic objective function subject to linear (equality and/or inequality) constraints. Let us assume a convex quadratic generalisation of the standard form of the QP problem is

$$\begin{aligned} \min \quad & (1/2)x^T Qx + c^T x \\ \text{subject to} \quad & Gx \leq h, \\ & Ax = b. \end{aligned} \tag{1}$$

where $x \in R^n$ is the variable of the QP problem and $Q$ is a symmetric $n \times n$ positive semidefinite matrix.

An interior-point method, in comparison to other methods such as primal barrier method, is particularly appropriate for embedded optimization, since, with proper implementation and tuning, it can reliably solve to high accuracy in 5-25 iterations, without even a "warm start" (Wang & Boyd, 2010).

In order to obtain a cone quadratic program (QP) using the QP optimisation problem of Equation (1), it is expedient for the analysis and implementation of interior-point methods to include a slack variable $s$ and solve the equivalent QP

$$\begin{aligned} \min \quad & (1/2)x^T Qx + c^T x \\ \text{subject to} \quad & Gx + s = h, \\ & Ax = b, \\ & s \geq 0. \end{aligned} \tag{2}$$

where $x \in R^n$ and $s \in R^p$ are the variables of the cone QP problem.

The dual problem of Equation (3) can be simply derived by introducing an additional variable $\omega$: (Vandenberghe, 2010)

$$\begin{aligned} \max \quad & -(1/2)\omega^T Q\omega - h^T z - b^T y \\ \text{subject to} \quad & G^T z + A^T y + c + Q\omega = 0, \\ & z \geq 0. \end{aligned} \tag{3}$$

where $y \in R^m$ and $z \in R^p$ are the Lagrange multiplier vectors for the equality and the inequality constraints of (1), respectively.

The dual objective of (3) provides a lower bound on the primal objective, while the primal objective of (1) gives an upper bound on the dual (Vandenberghe & Boyd, 2004). The vector $x^* \in R^n$ is an optimal solution of Equation (1) if and only if there exist Lagrange multiplier vectors $z^* \in R^p$ and $y^* \in R^m$ for which the following necessity KKT conditions hold for $(x, y, z) = (x^*, y^*, z^*)$; see (Potra & Wright, 2000) and other references therein for more details.

$$F(x,y,z,s) = \begin{bmatrix} Qx + A^T y + G^T z + c \\ Ax - b \\ Gx + s - h \\ ZS\mathbf{e} \end{bmatrix} = 0, \qquad (4)$$
$$(s,z) \geq 0$$

where $S = diag(s_1, s_2, \ldots, s_n)$, $Z = diag(z_1, z_2, \ldots, z_n)$ and $\mathbf{e}$ is the unit column vector of size $n \times 1$.

The primal-dual algorithms are modifications of Newton's method applied to the KKT conditions $F(x,y,z,s) = 0$ for the nonlinear equation of Equation (4). Such modifications lead to appealing global convergence properties and superior practical performance. However, they might interfere with the best-known characteristic of the Newton's method, that is "fast asymptotic convergence" of Newton's method. In any case, it is possible to design algorithms which recover such an important property of fast convergence to some extent, while still maintaining the benefits of the modified algorithm (Wright, 1997). Also, it is worthwhile to emphasise that all primal-dual approaches typically generate the iterates $(x_k, y_k, z_k, s_k)$ while satisfying nonnegativity condition of Equation (4) strictly, i.e. $s_k > 0$ and $z_k > 0$. This particular property is in fact the origin of the generic term "interior-point" (Wright, 1997) which will be briefly discussed next.

### 2.2 Embedded QP convex optimisation

There are several numerical approaches to solve standard cone QP problems. One alternative which seems suitable to the literature of fast model predictive control is the path-following algorithm – see e.g. (Potra & Wright, 2000; Renegar & Overton, 2001) and other references therein.

In the path-following method, the current iterates are denoted by $(x_k, y_k, z_k, s_k)$ while the algorithm is started at initial values $(x_k, y_k, z_k, s_k) = (x_0, y_0, z_0, s_0)$ where $(s_0, z_0) > 0$. For most problems, however, a strictly feasible starting point might be extremely difficult to find. Although it is straightforward to find a strictly feasible starting point by reformulating the problem – see (Vandenberghe, 2010, §5.3)), such reformulation may introduce distortions that can potentially make the problem harder to solve due to an increased computational time to generate real-time control law which is not desired for a wide range of practical applications, e.g. the HEV energy management problem – see Section 4. In §2.4, we will describe one tractable approach to obtain such feasible starting points.

Similar to many other iterative algorithms in nonlinear programming and optimisation literature, the primal-dual interior-point methods are based on two fundamental concepts: First, they contain a procedure for determining the iteration step and secondly they are required to define a measure of the attraction of each point in the search space. The utilised Newton's method in fact forms a linearised model for $F(x,y,z,s)$ around the current iteration point and obtains the search direction $(\Delta x, \Delta y, \Delta z, \Delta s)$ by solving the following set of linear equations:

$$J(x,y,z,s) \begin{bmatrix} \Delta_x \\ \Delta_y \\ \Delta_z \\ \Delta_k \end{bmatrix} = -F(x,y,z,s) \qquad (5)$$

where $J$ is the Jacobian of $F$ at point $(x_k, y_k, z_k, s_k)$.
Let us assume that the current point is strictly feasible. In this case, a Newton "full step" will provide a direction at

$$\begin{bmatrix} Q & A^T & G^T & 0 \\ A & 0 & 0 & 0 \\ G & 0 & 0 & I \\ 0 & 0 & S & Z \end{bmatrix} \begin{bmatrix} \Delta x_k \\ \Delta y_k \\ \Delta z_k \\ \Delta s_k \end{bmatrix} = -F(x_k, y_k, z_k, s_k) \tag{6}$$

and the next starting point for the algorithm will be

$$(x_{k+1}, y_{k+1}, z_{k+1}, s_{k+1}) = (x_k + \Delta x_k, y_k + \Delta y_k, z_k + \Delta z_k, s_k + \Delta s_k)$$

However, the pure Newton's method, i.e. a full step along the above direction, could often violate the condition $(s, z) > 0$ – see (Renegar & Overton, 2001). To resolve this shortcoming, a line search along the Newton direction is in a way that the new iterate will be (Wright, 1997)

$$(x_k, y_k, z_k, s_k) + \alpha_k(\Delta x_k, \Delta y_k, \Delta z_k, \Delta s_k)$$

for some line search parameter $\alpha \in (0, 1]$. If $\alpha$ is to be selected by user, one could only take a small step ($\alpha \ll 1$) along the direction of Equation (6) before violating the condition $(s, z) > 0$. However, selecting such a small step is not desirable as this may not allow us to make much progress towards a sound solution to a broad range of practical problems which usually are in need of fast convenance by applying "sufficiently large" step sizes.
Following the works (Mattingley & Boyd, 2010b) and (Vandenberghe, 2010), we shall intend to modify the basic Newton's procedure by two scaling directions (i.e. affine scaling and combined centering & correction scaling). Loosely speaking, by using these two scaling directions, it is endeavoured to bias the search direction towards the interior of the nonnegative orthant $(s, z) > 0$ so as to move further along the direction before one of the components of $(s, z)$ becomes negative. In addition, these scaling directions keep the components of $(s, z)$ from moving "too close" to the boundary of the nonnegative orthant $(s, z) > 0$. Search directions computed from points that are close to the boundary tend to be distorted from which an inferior progress could be made along those points – see (Wright, 1997) for more details. Here, we shall list the scaling directions as follows.

### 2.3 Scaling iterations

We follow the works by (Vandenberghe, 2010, §5.3) and (Mattingley & Boyd, 2010b) with some modifications that reflect our notation and problem format. Starting at initial values $(\hat{x}, \hat{y}, \hat{z}, \hat{s}) = (x_0, y_0, z_0, s_0)$ where $s_0 > 0$, $z_0 > 0$, we consider the scaling iterations as summarised here.

- Step 1. Set $k = 0$.
- Step 2. Start the iteration loop at time step $k$.
- Step 3. Define the residuals for the three linear equations as:

$$\begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \hat{s} \end{bmatrix} + \begin{bmatrix} Q & A^T & G^T \\ A & 0 & 0 \\ G & 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix} + \begin{bmatrix} c \\ -b \\ -h \end{bmatrix}$$

- Step 4. Compute the optimality conditions:

$$\begin{bmatrix} 0 \\ 0 \\ s \end{bmatrix} = \begin{bmatrix} Q & A^T & G^T \\ -A & 0 & 0 \\ -G & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} c \\ b \\ h \end{bmatrix}, \quad (s,z) \geq 0.$$

- Step 5. If the optimality conditions obtained at Step 4 satisfy $\|(x,y,z,s) - (\hat{x},\hat{y},\hat{z},\hat{s})\|_\infty \leq \epsilon$, for some small positive $\epsilon > 0$, go to Step 13.
- Step 6. Solve the following linear equations to generate the affine direction (Mattingley & Boyd, 2010b):

$$\begin{bmatrix} Q & A^T & G^T & 0 \\ A & 0 & 0 & 0 \\ G & 0 & 0 & I \\ 0 & 0 & S & Z \end{bmatrix} \begin{bmatrix} \Delta x_k^{aff} \\ \Delta y_k^{aff} \\ \Delta z_k^{aff} \\ \Delta s_k^{aff} \end{bmatrix} = -F(x_k, y_k, z_k, s_k) \tag{7}$$

- Step 7. Compute the duality measure $\mu$, step size $\alpha \in (0,1]$, and centering parameter $\sigma \in [0,1]$

$$\mu = \frac{1}{n} \sum_{i=1}^{n} s_i z_i = \frac{z^T s}{n}$$
$$\sigma = \left( \frac{(s+\alpha_c \Delta s^{aff})^T (z+\alpha_c \Delta z^{aff})}{s^T z} \right)^3$$

and

$$\alpha_c = \sup\{\alpha \in [0,1] | (s + \alpha_c \Delta s^{aff}, z + \alpha_c \Delta z^{aff}) \geq 0\}$$

- Step 8. Solve the following linear equations for the combined centering-correction direction[2]:

$$\begin{bmatrix} Q & A^T & G^T & 0 \\ A & 0 & 0 & 0 \\ G & 0 & 0 & I \\ 0 & 0 & S & Z \end{bmatrix} \begin{bmatrix} \Delta x^{cc} \\ \Delta y^{cc} \\ \Delta z^{cc} \\ \Delta s^{cc} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \sigma \mu \mathbf{e} - \mathrm{diag}(\Delta s^{aff}) \Delta z^{aff} \end{bmatrix}$$

  This system is well defined if and only if the Jacobian matrix within is nonsingular (Peng et al., 2002, §6.3.1).
- Step 9. Combine the two affine and combined updates of the required direction as:

$$\Delta x = \Delta x^{aff} + \Delta x^{cc}$$
$$\Delta y = \Delta y^{aff} + \Delta y^{cc}$$
$$\Delta z = \Delta z^{aff} + \Delta z^{cc}$$
$$\Delta s = \Delta s^{aff} + \Delta s^{cc}$$

- Step 10. Find the appropriate step size to retain nonnegative orthant $(s,z) > 0$,

$$\alpha = \min\{1, 0.99 \sup(\alpha \geq 0 | (s + \alpha \Delta s, z + \alpha \Delta z) \geq 0)\}$$

[2] This is another variant of Mehrotra's predictor-corrector algorithm (Mehrotra, 1992), a primal-dual interior-point method, which yields more consistent performance on a wide variety of practical problems.

- Step 11. Update the primal and dual variables using:

$$\begin{bmatrix} x \\ y \\ z \\ s \end{bmatrix} := \begin{bmatrix} x \\ y \\ z \\ s \end{bmatrix} + \alpha \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta s \end{bmatrix}$$

- Step 12. Set $(\hat{x}, \hat{y}, \hat{z}, \hat{s}) = (x_k, y_k, z_k, s_k)$ and $k := k + 1$; Go to Step 2.
- Step 13. Stop the iteration and return the obtained QP optimal solution $(x, y, z, s)$.

The above iterations will modify the the search direction so that at any step the solutions are moved closer to feasibility as well as to centrality. It is also emphasised that most of the computational efforts required for a QP problem are due to solving the two matrix equalities in steps 6 and 8. Among many limiting factors which may make the above algorithm failed, floating-point division is perhaps the most critical problem of an online optimisation algorithm to be considered (Wang & Boyd, 2008). In words, stability of an optimisation-based control law (such as model predictive control) are significantly dependent on the risk of algorithm failures, and therefore it is vital to develop robust algorithms for solving these linear systems leading towards fast optimisation-based control designs, which is the focal point of our work. We should also stress that robustness of any algorithm must be taken into account at starring point. In particular, many practical problems are prone to make optimisation procedures failed at the startup. For instance, (possibly large) disparity between the initial states of the plant and the feedback controller might lead to large transient control signals which consequently could violate feasibility assumptions of the control law – this in turn may result into an unstable feedback loop. Therefore, the initialisation of the optimisation-based control law is significantly important and must be taken into consideration in advance. The warm start is also an alternative to resolve this shortcoming – see e.g. (Wang & Boyd, 2010). Here, we shall discuss a promising initialisation method for the solution of the linear systems which is integrated within the framework of our fast model predictive control algorithm.

### 2.4 Initialisation

We shall overview the initialisation procedure addressed in (Vandenberghe, 2010, §5.3) and (Mattingley & Boyd, 2010b). If primal and dual starting points $(\hat{x}, \hat{y}, \hat{z}, \hat{s})$ are not specified by the user, they are chosen as follows:

- Solve the linear equations (see §2.5 for a detailed solution of this linear system.)

$$\begin{bmatrix} Q & A^T & G^T \\ A & 0 & 0 \\ G & 0 & -I \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -c \\ b \\ h \end{bmatrix} \tag{8}$$

  to obtain optimality conditions for the primal-dual pair problems of

$$\begin{aligned} \min\ & (1/2)x^T Q x + c^T x + (1/2)\|s\|_2^2 \\ \text{subject to}\ & Gx + s = h \\ & Ax = b \end{aligned} \tag{9}$$

and

$$\begin{aligned} \max \quad & -(1/2)\omega^T Q\omega - h^T z - b^T y - (1/2)\|z\|_2^2 \\ \text{subject to} \quad & Q\omega + G^T z + A^T y + c = 0. \end{aligned} \tag{10}$$

- From the above, $\hat{x} = x, \hat{y} = y$ are found as the two initialisation points. The initial value of $\hat{s}$ is calculated from the residual $h - Gx = -z$, as

$$\hat{s} = \begin{cases} -z & \alpha_p < 0 \\ -z + (1 + \alpha_p)\mathbf{e} & \text{otherwise} \end{cases}$$

where $\alpha_p = \inf\{\alpha | -z + \alpha\mathbf{e} \geq 0\}$. Also, the initial value of $\hat{z}$ is computed as

$$\hat{z} = \begin{cases} z & \alpha_d < 0 \\ z + (1 + \alpha_d)\mathbf{e} & \text{otherwise} \end{cases}$$

where $\alpha_d = \inf\{\alpha | z + \alpha\mathbf{e} \geq 0\}$.

### 2.5 Fast KKT solution

As explained earlier, the most time-consuming parts of QP optimisation problem is due to solving the linear KKT systems of the format

$$\begin{bmatrix} Q & A^T & G^T & 0 \\ A & 0 & 0 & 0 \\ G & 0 & 0 & I \\ 0 & 0 & S & Z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ s \end{bmatrix} = \begin{bmatrix} r_x \\ r_y \\ r_z \\ r_s \end{bmatrix} \tag{11}$$

In Ref. (Mattingley & Boyd, 2010b) a numerical method has been introduced, using the permuted $LDL^T$ factorisation, to solve the KKT linear systems of Equation (11) in the compact form of $KX = R$ to find optimal variables of $X$. In the so-called "iterative refinement approach", see (Mattingley & Boyd, 2010b, §5.3)), the original KKTs is regularised by choosing a small $\epsilon > 0$ to ensure that such a factorisation exists and that it is numerically stable. However, since the solution of the modified KKT system is an approximation to the original KKT system, it could potentially affect both the affine and combined step sizes, as well as the feasibility conditions and the rate of global convergence. In words, such an approximation could introduce additional "hold-ups" to the QP problem at each time step which is not desirable for the purpose of the fast real-time optimisation applications, such the one considered in Section 4 as a case study.

In order to obtain "fast" and "reliable" solutions of Equation (11) at each iteration, it is significantly important to avoid any sort of calculation of unstructured (possibly sparse) matrix inverse, as well as to reduce the number of the KKT linear systems. Due to the particular structure of the original exact KKT system given in Equation (11), however, we shall employ a more reliable and stable interior-point solver of the convex QP optimisation problem, even for the KKT systems with sparse matrices. To this end, we start by eliminating the variable $s$ among the KKT linear systems. After some algebra, we will have

$$\begin{bmatrix} Q & A^T & G^T \\ A & 0 & 0 \\ G & 0 & -Z^{-1}S \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r_x \\ r_y \\ r_z - Z^{-1}r_s \end{bmatrix} \quad (12)$$

which reduces the number of original KKT systems solved per iteration by three. To calculate $s$, we could use $s = -Gx + r_z$.

The Cholesky factorisation method is the preferred KKT equation solver for linear and quadratic programs. However, due to the particular structure of $Z^{-1}S$, being a diagonal matrix, there is no longer a need to carry out the Cholesky factorization of the diagonal matrix of $Z^{-1}S$ given in Equation (12). In fact, $Z^{-1}S = W^T W$ with diagonal matrix $W = W^T$ will lead to $Z^{-1}S = W^2$. We can now obtain a reduced order of KKT system of Equation (12) with only two equations as

$$\begin{bmatrix} Q + G^T W^{-2} G & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} r_x + G^T W^{-2}(r_z - Z^{-1}r_s) \\ r_y \end{bmatrix} \quad (13)$$

From $x$ and $y$, the solution $z$ follows as $Wz = W^{-T}(Gx - r_z + Z^{-1}r_s)$. Recall that matrices $S$, $Z$ and $W$ are diagonal matrices, and hence calculation of inverses of $W^{-T} = W^{-1} = \mathrm{diag}(1/W(i,i))$ and $Z^{-1} = \mathrm{diag}(1/Z(i,i))$, $i = 1, 2, \ldots, n$ are straightforward and fast, even for large sparse problems.

Using the Cholesky factorization $Q + G^T W^{-2} G = LL^T$, the KKT solutions of Equation (13) are computed as follows – see also (Vandenberghe, 2010).

- Case 1. If $Q + G^T W^{-2} G$ is nonsingular, $y$ and $x$ are computed from the following equations, respectively:

$$\begin{aligned} AL^{-T}L^{-1}A^T y &= AL^{-T}L^{-1}\left(r_x + G^T W^{-2}(r_z - Z^{-1}r_s)\right) - r_y \\ LL^T x &= r_x + G^T W^{-2}(r_z - Z^{-1}r_s) - A^T y \end{aligned} \quad (14)$$

- Case 2. If $Q + G^T W^{-2} G$ is singular, the exact solutions of $y$ and $x$ are obtained respectively as

$$\begin{aligned} AL^{-T}L^{-1}A^T y &= AL^{-T}L^{-1}\left(r_x + G^T W^{-2}(r_z - Z^{-1}r_s) + A^T r_y\right) - r_y \\ LL^T x &= r_x + G^T W^{-2}(r_z - Z^{-1}r_s) + A^T(r_y - y) \end{aligned} \quad (15)$$

The above algorithm will provide the KKT linear systems to be solved reliably, and to precise accuracy, in a limited number of iterations. This, along with previous optimisation requirements addressed earlier, will help develop a fast reliable optimisation algorithm leading towards fast model predictive control which is briefly discussed in the subsequent section.

### 2.6 Tracking control problem using fast MPC

As we discussed earlier, standard MPC-based algorithms are great tools in the literature of feedback control system designs, mainly due to their abilities in handling constraints, e.g. actuator saturations, which are successfully taken into consideration in the design of an MPC. However, to provide an appropriate input control signal, MPC and its standard variants suffer from a major drawback due to having a desperate need of excessive computational time for solving the online minimization problems, at each sampling interval, particularly in the presence of large number of horizons, constraints, optimisation parameters or parametric uncertainties. This shortcoming is an outstanding motivation to look for some sort of efficient model predictive algorithms, to solve an integrated optimisation problems "sufficiently fast" in real time.

Any MPC tracking reference problem subject to constraints, being also the focal point of this section, results in the optimization of a quadratic programming (QP) problem as the objective function is quadratic – see §2.1.

The reference tracking control for a discrete-time linear dynamical system is described as the following quadratic optimisation problem:

$$\begin{aligned} \min \quad & \sum_{k=1}^{N} \left(y_k - y_k^{ref}\right)^T Q_y \left(y_k - y_k^{ref}\right) + u_k^T Q_u u_k \\ \text{subject to} \quad & \mathrm{x}_{k+1} = Ax_k + Bu_k + w_k \\ & \mathrm{y}_k = Cx_k + v_k \\ & \mathrm{x}_{\min} \leq x_k \leq \mathrm{x}_{\max} \\ & \mathrm{u}_{\min} \leq u_k \leq \mathrm{u}_{\max} \\ & \dot{\mathrm{u}}_{\min} \leq \dot{u}_k \leq \dot{\mathrm{u}}_{\max} \end{aligned} \tag{16}$$

where $Q_y$ and $Q_u$ are positive semidefinite weighting matrices for penalizing the tracking error and control effort, respectively; $A \in R^{n \times n}$, $B \in R^{n \times m}$, $C \in R^{p \times n}$ and $x_0 \in R^n$ are the discrete-time plant data, $w_k$ and $v_k$ are plant disturbance and measurement noise, respectively; $y_k^{ref}$ is the reference signal to be tracked at the plant output; $N$ is the horizon; The optimisation variables are the system state $x_k (k = 1, 2, \ldots, N)$ and input control signals $u_k (k = 0, 1, \ldots, N - 1)$.

Recall that here we only consider the linear time-invariant (LTI) systems. Nonetheless, the proposed method could be extended to the time-varying and/or nonlinear cases (Del Re et al., 2010). Also, regarding the fact that most of the physical plants are continuous-time, we shall consider a continuous-time linear dynamic system driven by stationary continuous-time white noise. To simulate such a continuous-time dynamics on a digital computer (or microprocessor) using an equivalent discrete-time dynamics, it is required to utilise equated linear discrete-time system and its noise statistics, so that both systems have identical statistical properties at the discrete time instants (Gelb, 1974, pp. 72-75). Moreover, any type of continuous-time plant dynamics could be transformed, with an appropriate sampling time, to the equivalent discrete-model in the format of the one shown in the subjective of Equation (16) – see (Grewal & Andrews, 1993, pp. 88-91).

The output tracking control system design in Equation (16) could be transformed to the standard quadratic programming problem illustrated in Equation (1). Therefore, we could use the fast KKT solutions following the initialisation. In Section 4, the proposed optimisation procedure of the fast model predictive control design approach will be implemented to the case study of the HEV energy management.

## 3. Hybrid diesel electric vehicle model

In Section 1, we briefly discussed the history of hybrid electric vehicles which, in some extent, could clarify the importance of our work carried out in the field of advanced energy management for the HEV applications. In this section, we will investigate how to model a simplified hybrid electric vehicle to replace the sophisticated nonlinear dynamic of the diesel internal combustion engine. We shall integrate this simplified HEV model, for the first time, with recent advances on fast model predictive control architecture described in Section 2 based on embedded convex optimisation.

Before describing the structure of our hybrid diesel electric vehicle, let us first overview a generic HEV structure. A representative configuration of an advanced $4x4$ parallel hybrid electric vehicle configuration is shown in Figure 2.

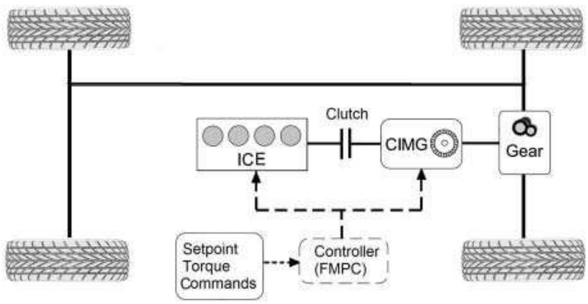



Fig. 2. Schematic structure of a parallel $4x4$ Hybrid Electric Vehicle (HEV). Low-level control components such as high voltage electric battery, electric rear axle drive etc are excluded in this high-level energy management configuration.

The hybrid electric vehicle structure shown in Figure 2 is equipped with a turbocharged diesel engine and a crankshaft integrated motor/generator (CIMG) which is directly mounted on the engine crankshaft. The CIMG is used for starting and assisting the engine in motoring-mode, and also for generating electric energy via charging a high-voltage battery (not shown in the figure). As our intention in this study is to investigate the "full-hybrid" mode, we shall assume that the integrated ICE-CIMG clutch is fully engaged and hence our descriptive HEV dynamical model (see §3.3) excludes a clutch dynamics as it is shown in Figure 2. Likewise, the gearbox is shown in Figure 2 but no gear setting was considered in our simplified HEV demonstration. This is due to the fact that our empirical diesel engine model is derived with engine speed range of $\omega = [1200, 2000]$rpm running at the first gear.

It is also worthwhile to emphasise that our design methodology on the development of the HEV energy management is a high-level design strategy. For this reason, most of the common low-level subsystems, integrated within the typical HEV dynamics, are not considered in the HEV configuration as shown in Figure 2. These low-level subsystems include CIMG low-level motor control, high voltage battery management, low level clutch control, low level transmission control, and electrical distribution including DC-DC converter, to name just a few. Furthermore, the dynamics of the engine model includes the average torque responses of both diesel engine and CIMG over all four cylinders, which are the quantities of interest.

For designing a well balanced feedback control law, control engineers should possess a good comprehension of the physics of the plant under investigation. One of the challenging aspect of any model based engine control development is the derivation of a simplified yet insightful model. For instance, the frequency range of the engine system, the nonlinearities associated with the internal engine processes (i.e. combustion process, heat distribution, air flow), the severe cross coupling, the inherent sampling nature of the four cycle internal

combustion engine, and the limited output sensor dynamic capabilities all contribute to make this modelling step a most arduous task (Lewis, 1980).
There are two main reasons to highlight the importance of simplified HEV dynamical models: First, it is not usually possible to obtain a detailed diesel engine data (or model) from the production vehicle manufacturer (Kaszynski & Sawodny, 2008). Secondly, obtaining a precise mathematical model of a HEV powertrain is a very challenging task particularly due to multi-energetic nature and switching dynamics of a powertrain.
For the above reasons, and for the ease of development of an advanced HEV energy management system, it is essential to obtain a straightforward and realistic model of the propulsion system to which an efficient control strategy, such as our proposed fast MPC design methodology, could be applied. Generally speaking, this model shall be used for the simulation of the overall vehicle motion (at longitudinal direction). Therefore, we do not intend to utilise any detailed model of the internal engine processes, but rather a high-level torque manager model that will generate control efforts based on a given set-point torque commands. Recall that this torque management structure could be easily adopted to other engine configurations in a straightforward manner.
As stated earlier, our developments towards a simplified hybrid model are based on a high fidelity simulation model of the overall diesel hybrid electric vehicle. This HEV dynamical model is modeled using two subsystems, a diesel internal combustion engine (ICE) and an armature-controller DC electric motor. The mathematical modeling of these two subsystems will be discussed in the remainder of this section.

### 3.1 Simplified diesel engine model

In this section, we shall present a simplified dynamical model of a turbo-charged diesel engine. This simplified model is based upon the nonlinear diesel engine dynamics and the fact that it must capture both the transient and steady-state dominant modes of the diesel engine during operational conditions.
The engine indicated torque $T_{ind}$ is assumed to be mapped from the delayed fueling input proportionally, and has limited bandwidth due to internal combustion dynamic effects, arising e.g. due to combustion and turbo lag. In a mathematical representation, we will have

$$T_{ind}(t) = \frac{1}{\tau(\omega)s + 1} T_B^{dem}(t - t_d(\omega)) \tag{17}$$

where $\tau$ is the speed-dependant time constant due to combustion lag, $t_d$ is the speed-dependant time-delay due to fueling course and $T_B^{dem}$ is the mapped fueling input representing the required ICE crankshaft (brake) torque.
Our simplified diesel engine model is empirically derived using a turbo-charged diesel engine at speed range of $\omega = [1200, 2000]$rpm with operational brake torque acting at $T_B = [50, 100]$NM. The speed-dependant fueling delay ($t_d$) and combustion lag ($\tau$) are given in Table 1.
Our diesel model also contains a speed-dependant torque loss $T_{Loss}$ arising due to friction torque, ancillary torque and pumping loss. Such a total torque loss is typically a nonlinear function of the engine speed. However, at the studied operating range of engine speed and brake torque, namely $\omega = [1200, 2000]$rpm and $T_B = [50, 100]$NM, the total engine torque loss is a linear function of $\omega$ modeled as $T_{Loss} = m\omega$ where $m = 0.12$ with $\omega$'s dimension in [rad/sec].

| $\omega$ | $T_B$ | $t_d$ | $\tau$ |
|---|---|---|---|
| 1200 | 50 | 100 | 144 |
| 1200 | 100 | 140 | 142 |
| 1600 | 50 | 84 | 140 |
| 1600 | 100 | 96 | 137 |
| 2000 | 50 | 80 | 140 |
| 2000 | 100 | 72 | 134 |

Table 1. Experimental results of fueling delay, $t_d$ [msecs], and combustion lag, $\tau$ [msecs], as functions of diesel engine speed [rpm] and brake torque [NM]. These results are captured by measuring the step response of the engine to a step change in the engine brake torque.

For the purpose of this study, we shall employ a 1-st order Pade approximation to model the fueling time-delay by a rational 1st-order LTI model of

$$e^{-t_d s} \cong \frac{-s + 2/t_d}{s + 2/t_d} \tag{18}$$

The simplified diesel engine model can now be described as the following state-space equations:

$$\begin{aligned} \dot{x}_1 &= -\tfrac{2}{t_d} x_1 + T_B^{dem} \\ \dot{x}_2 &= \tfrac{4}{t_d} x_1 - \tfrac{1}{\tau} x_2 - T_B^{dem} \\ T_{Loss} &= m\omega \\ T_B &= \tfrac{1}{\tau} x_2 - T_{Loss} \end{aligned} \tag{19}$$

where $x_1$ and $x_2$ are the states associated with the Pade approximation, and combustion lag dynamics, respectively.
The diesel dynamic shown in Equation (19) will be used in the overall configuration of the HEV dynamics.

### 3.2 Simplified CIMG Model

Assuming that the hybrid electric drivetrain includes an armature-controlled CIMG (DC motor), the applied voltage $v_a$ controls the motor torque ($T_M$) as well as the angular velocity $\omega$ of the shaft.
The mathematical dynamics of the CIMG could be represented as follows.

$$\begin{aligned} I_a &= \frac{1}{L_a s + R_a}(v_a^{dem} - v_{emf}) \qquad (20) \\ v_{emf} &= k_b \omega \\ T_M &= k_m I_a \end{aligned}$$

where $k_m$ and $k_b$ are torque and back emf constants, $v_a^{dem}$ is control effort as of armature voltage, $v_{emf}$ is the back emf voltage, $I_a$ is armature current, $L_a$ and $R_a$ are inductance and resistance of the armature, respectively.
Regarding the fact that the engine speed is synchronised with that of the CIMG in full-hybrid mode, the rotational dynamics of the driveline (of joint crankshaft and motor) is given as follows:

$$J\dot{\omega} + b\omega = T_B + T_M - T_L \tag{21}$$

where $\omega$ is the driveline speed, $J$ is the effective combined moment of rotational inertia of both engine crankshaft and motor rotor, $b$ is the effective joint damping coefficient, and $T_L$ is the vehicle load torque, which is representing the plant disturbance.
The armature-controlled CIMG model in Equation (20) along with the rotational dynamics of Equations (20) and (21) could be integrated within the following state-space modelling:

$$\begin{aligned} \dot{x}_3 &= v_a^{dem} - \frac{R_a}{L_a}x_3 - \frac{K_b}{J}x_4 \\ \dot{x}_4 &= \frac{1}{\tau}x_2 + \frac{K_b}{L_a}x_3 - \frac{b}{J}x_4 - T_{Loss} - T_L \\ \omega &= \frac{1}{J}x_4 \\ T_M &= \frac{K_b}{L_a}x_3 \end{aligned} \tag{22}$$

where $x_3$ and $x_4$ are the states associated with the armature circuit, and driveline rotational dynamics, respectively.
A simplified but realistic simulation model with detailed component representations of diesel engine and DC electric motor (CIMG) will be used as a basis for deriving the hybrid model as discussed in the subsequent section.

### 3.3 Simplified hybrid diesel electric vehicle model

Based on the state-space representation of both the diesel ICE and electric CIMG, given in Equation (19) and Equation (22), respectively, we can now build our simplified 4-state HEV model to demonstrate our proposed approach.
A schematic representation of the simplified parallel hybrid diesel electric vehicle model is shown in Figure 3.

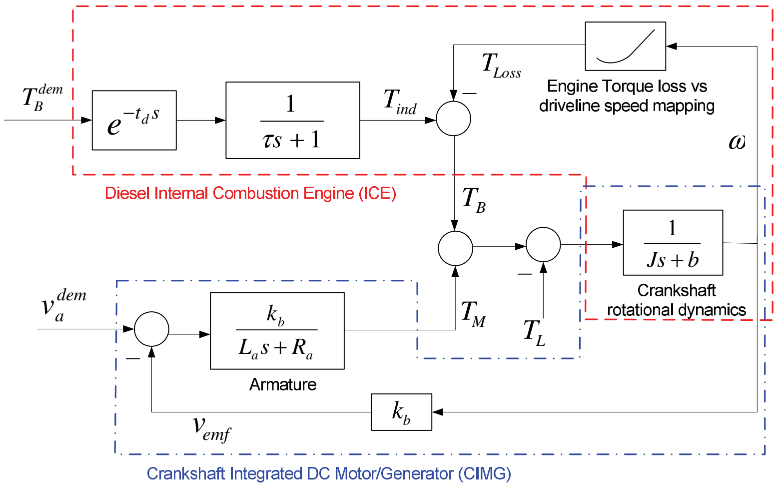



Fig. 3. Simplified model of the parallel Hybrid Diesel Electric Vehicle.

Recall that, as illustrated in Figure 2, the setpoint torque commands (indicated by $T_B^{req}$ and $T_M^{req}$) are provided to the controller by a high-level static optimisation algorithm, not discussed in this study – see (Dextreit et al., 2008) for more details. Also, in this figure the engine

brake torque and the CIMG torque are estimated feedback signals. However, the details of the estimation approach are not included here. For the sake of simplicity, in this work we shall assume that both engine and CIMG output torques are available to measure.
In addition, due to there being in "full hybrid" mode, it is assumed that the ICE-CIMG clutch is fully engaged and hence the clutch model is excluded from the main HEV dynamics - it was previously shown in Figure 2. Also, the gear setting is disregarded at this simplified model, as discussed earlier. Furthermore, the look-up mapping table of CIMG torque request vs armature voltage request ($v_a^{dem}$) is not shown in this model for the sake of simplicity.
The overall state-space equations of the simplified HEV model is represented by

$$\begin{aligned} \dot{x} &= \begin{bmatrix} -\frac{2}{t_d} & 0 & 0 & 0 \\ \frac{4}{t_d} & -\frac{1}{\tau} & 0 & 0 \\ 0 & 0 & -\frac{R_a}{L_a} & -\frac{K_b}{J} \\ 0 & \frac{1}{\tau} & \frac{K_b}{L_a} & -\frac{m+b}{J} \end{bmatrix} x + \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -1 \end{bmatrix} T_L \\ y &= \begin{bmatrix} 0 & \frac{1}{\tau} & 0 & -\frac{m}{J} \\ 0 & 0 & \frac{K_b}{L_a} & 0 \end{bmatrix} x \end{aligned} \tag{23}$$

where $x \in R^4$ is the state of the system obtained from Equations (19) and (22), $u = [T_B^{dem}\ v_a^{dem}]^T$ and $y = [T_B\ T_M]^T$ are control signals and HEV torque outputs, respectively.
The state-space equations of Equation (23) will be used in designing the proposed fast model predictive control described in Section 2. Some representative simulation results of HEV energy management case study will be shown in the next section to highlight some advances of our proposed embedded predictive control system.

## 4. Simulation results

In this section, we shall present our proposed Fast MPC algorithm described in Section 2 for the application of the simplified HEV energy management system discussed in Section 3. The problem addressed in the next subsection is to discuss required setpoint torque tracking problem with appropriate optimisation objective leading towards applying our fast MPC design to the HEV energy management problem as illustrated by some of our simulation results.

### 4.1 HEV energy management optimisation objective and control strategy

For the HEV energy management application subject to the objective function and constraints, HEV demanded torques are found at each time step by solving the optimisation problem of Equation (16) with the following data:

$$\begin{aligned} x_{min} &= [0,\ -56,\ -300,\ 0]^T \\ x_{max} &= [18,\ 56,\ 300,\ 360]^T \\ u_{min} &= [0,\ -380]^T \\ u_{max} &= [400,\ 380]^T \\ \dot{u}_{max} &= -\dot{u}_{min} = [0.5,\ 4]^T \end{aligned} \tag{24}$$

For our HEV setpoint tracking problem, based on Equation (16), $y_k = [T_B \;\; T_M]^T$ is the HEV torque outputs (ICE torque and CIMG torque, respectively), $y_k^{req} = [T_B^{req} \;\; T_M^{req}]^T$ is the tracking setpoint torques commands , $w_k \in R^4$ is the discretised vehicle load torque, $u_k = [T_B^{dem} \;\; v_a^{dem}]^T$ is the demanded HEV torques (control efforts) generated in real-time by the controller.
An equated LTI discrete-time system of the continuous-time state-space dynamics described in Equation (23) is obtained using a sampling interval $t_s$ (see Table 2). The plant initial condition $x_0 \in R^4$ is assumed zero in our simulations.
The parameters used in the proposed Fast MPC design together with other physical constants of the simplified HEV model are provided in Table 2.

| Parameter | Value | Unit |
|---|---|---|
| Sampling time ($t_s$) | 8 | msecs |
| ICE fueling delay ($t_d$) | 90 | msecs |
| ICE combustion lag ($\tau$) | 140 | msecs |
| Motor armature resistance ($R_a$) | 1 | Ohms |
| Motor armature inductance ($L_a$) | 0.3 | Henrys |
| Motor torque constant ($k_m$) | 0.25 | NM.Amp$^{-1}$ |
| Motor back emf constant ($k_b$) | 0.25 | Volts.secs.rad$^{-1}$ |
| Effective hybrid rotational inertia ($J$) | 0.6 | kg.m$^2$/s$^2$ |
| Effective hybrid rotational damping ($b$) | 0.125 | Nms |
| FMPC horizon ($N$) | 20 | - |
| Output penalising matrix ($Q_y$) | diag(400,200) | - |
| Control penalising matrix ($Q_u$) | diag(0.01,0.01) | - |

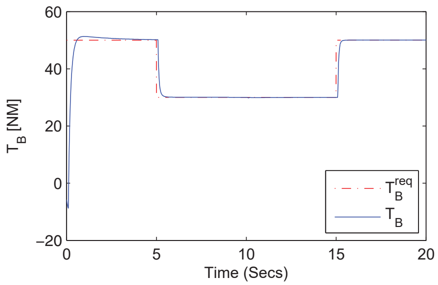
Table 2. Physical constants and FMPC design parameters in regard to the HEV model case study.

In the next subsection, the closed-loop behavior of the HEV energy management problem with our FMPC controller placed in the feedback loop has been evaluated based on the high-fidelity simplified model of the HEV described in Section 3.

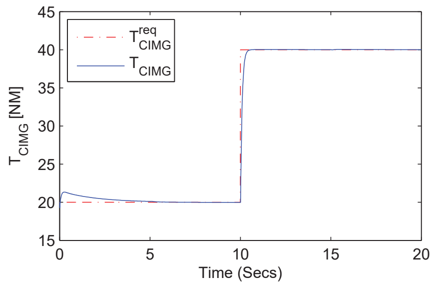
### 4.2 Simulation results

Our simulations have been carried out in Simulink and implemented in discrete-time using a zero-order hold with a sampling time of $t_s = 8$ msecs – see Table 2.
We shall emphasis that optimization based model predictive control (MPC) techniques, including the proposed fast MPC design methodology, require knowledge about future horizon (driving conditions in this case study). These future driving conditions in our case study include setpoint torque commands (requested by driver) and vehicle load torque. This fact will make implementation of all sort of optimisation based predictive control algorithms even more arduous to be applied in real time.
For the purpose of simulations, assuming that the future driving cycle (i.e. torque references and vehicle load) are entirely known could be perhaps an acceptable assumption. In our simulations, the future driving cycle is unknown whilst retaining constant for the whole horizon of $N$ samples. However, if the future driving cycle could be entirely known, the performance of the proposed FMPC would be superior than those shown here.

Figure 4 shows a typical simulation results for the period of 20 secs in tracking requested setpoint HEV torques. During this simulation period, the system is in hybrid mode as both ICE torque and CIMG torque are requested.




(a) Engine Brake Torque.




(b) CIMG Torque.

Fig. 4. Simulation results of the HEV torque setpoints and outputs using the proposed FMPC algorithm.

As shown in Figure 4, despite the fact that the HEV energy management is a coupled Two-Input Two-Output (TITO) dynamical system, both the diesel ICE and the DC electric motor have successfully tracked the requested torque setpoints. At times $t = 5$ secs and $t = 15$ secs , the TITO controller is requested for an increased and decreased ICE torques, respectively to which the fast MPC algorithm could precisely follow those commands, as illustrated in Figure 4(a). Similarly, there was an increased request for the CIMG torque (from 20 NM to 40 NM) at time $t = 10$ secs, and the controller has successfully delivered this torque request, as depicted in Figure 4(b).

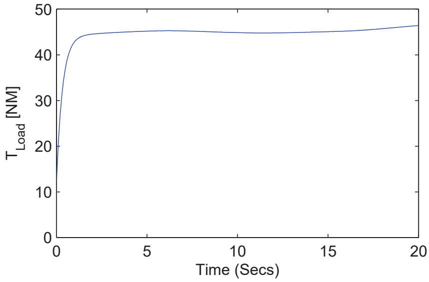
This is noted that our torque manager structure, as stated earlier, assumes that setpoint torque commands are provided by some sort of static optimisation algorithms. The designed FMPC is then enquired to optimise control efforts so as to track the requested torque references.

Figure 5 shows the load torque transient used in our simulations (being modeled as a plant disturbance), ICE torque loss and control efforts generated by the FMPC. We have assumed that plant disturbance (vehicle load) is known and available to controller. In reality, this might be an infeasible assumption where an estimation algorithm is required to estimate the vehicle load torque $w_k$ over the prediction horizon. Also, as mentioned earlier, the estimation of future driving conditions must be made online. Due to lack of space, however, we shall preclude addressing a detailed discussion in this course.

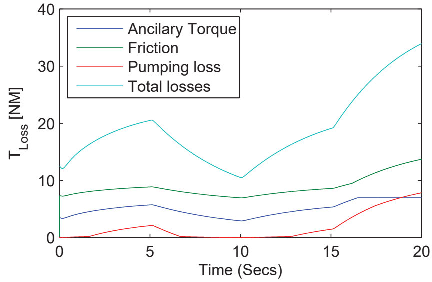
Figure 5(c) shows that the FMPC fully satisfies the required optimisation constraints as of Equation (24).

Figure 6 shows simulation results in regard to driveline speed and vehicle speed. It is worthwhile to point out that as illustrated in Figure 6(a), by requesting large torque commands, we have in fact violated our empirical HEV modeling assumption in that driveline speed must be limited to $\omega = [1200, 2000]$rpm. However, it can be seen that the FMPC can still successfully control the HEV energy endamagement dynamics in real-time. The vehicle speed shown in Figure 6(b) has been calculated using a dynamic model of the vehicle as a function of the driveline speed which is not discussed here.

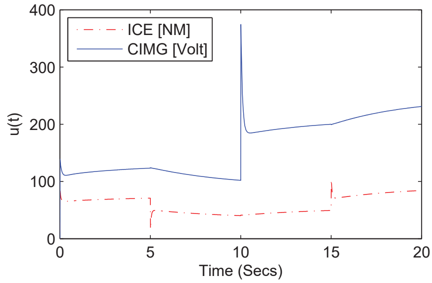
It is also important to mention that fueling delay and combustion lag are functions of engine speed and brake torque – see Table 1. However, in designing our fast MPC algorithm we




(a) Vehicle (load) Torque.

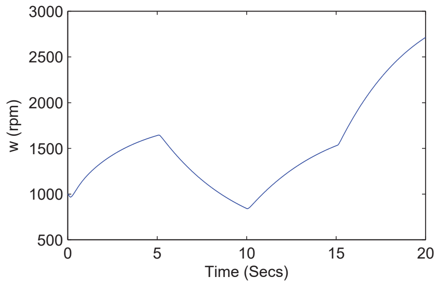



(b) Torque Loss.

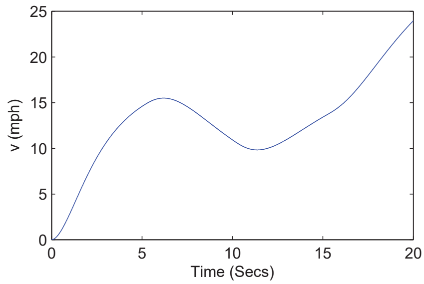



(c) Control signals.

Fig. 5. Simulation results of vehicle load, Torque loss, and Control efforts.




(a) Driveline Speed.




(b) Vehicle Speed.

Fig. 6. Simulation results of parallel diesel HEV driveline speed and vehicle speed.

require to utilise an LTI model of the HEV energy management plant. Towards this end, we use the numerical values of $\tau = 140$ msecs and $t_d = 90$ msecs, in our design to capture worst case of the ICE speed-dependant parameters. However, the simulation results are based on the actual time-varying speed-dependant parameters of the ICE, namely $\tau$ and $t_d$.
Regarding the real-time simulations in Simulink (fixed-step) using our Matlab custom S-function codes with a sampling time of $t_s$, the simulation time required for a single run of 20 secs was approximately 500 times faster than real-time running a Toshiba Portege laptop with an Intel(R) Core(TM) i5 processor, at 2.4GHz under Windows 7 Pro platform.

Without doubt, this shows a significant improvement on the computational capability of the control action that could potentially permit any sort of fast MPC algorithms to be run using inexpensive low-speed CPUs under possibly kilo Hertz control rates.

## 5. Conclusions

The aim of this chapter was to present a new Fast Model Predictive Control (FMPC) algorithm with an application for the energy management of hybrid electric vehicles (HEVs). The main goal of energy management in hybrid electric vehicles is to reduce the $CO_2$ emissions with enhanced fuel consumption for a hybrid powertrain control system. The applicability of conventional MPC in the energy management setting, however, has shown a main drawback of these algorithms where they currently cannot be implemented on-line due to the burdensome real-time numerical optimisation, arising due to e.g. hardware constraints and limitation of online calculations. The proposed FMPC design architecture could resolve such shortcomings of the standard MPC algorithms. In fact, such a custom method, is able to speed up the control action, by exploiting particular structure of the MPC problem, much faster than that of the conventional MPC methods. Moreover, our proposed FMPC design methodology does not explicitly utilise any knowledge in regard to the future driving cycle. Simulation results illustrated that FMPC could be a very promising on-line control design algorithm and could play a key role in a wide variety of challenging complex automotive applications in the future.

## 6. Acknowledgment

This work was supported by EPSRC, UK, under framework "Low Carbon Vehicles Integrated Delivery Programme", Ref. EP/H050337/1.
We would like to thank Dr. Jacob Mattingley and Yang Wang, from Stanford University, for their valuable comments and discussions which helped us in preparation of an earlier version of our simulation results.

## 7. References

Anderson, C.D. & Anderson, J. (2010). *Electric and Hybrid Cars: A History*, 2nd Edition, McFarland & Co Inc.

Bartlett, R.A.; Wachter, A. & Biegler, L.T. (2000). Active set vs. interior point strategies for model predictive control, *Proc. of the American Control Conf.*, Vol. 6, pp. 4229-4233.

Baumann, B.M.; Washington, G.; Glenn, B.C. & Rizzoni, G. (2000). Mechatronic design and control of hybrid electric vehicles, *IEEE/ASME Trans. On Mechatronics*, 5(1): 58-72.

Beccuti, A.G.; Papafotiou, G.; Frasca, R. & Morari, M.; . (2007). Explicit Hybrid Model Predictive Control of the dc-dc Boost Converter, *IEEE Power Electronics Specialists Conference, PESC 2007*, Orlando, Florida, USA, pp. 2503-2509

Bemporad, A. (2004). *Hybrid Toolbox - User's Guide*.
URL: *http://www.ing.unitn.it/ bemporad/hybrid/toolbox*

Bemporad, A.; Morari, M; Dua, V. & Pistikopoulos, E.N. (2002). The explicit linear quadratic regulator for constrained systems, *Automatica* 38: 3-20.

Bemporad, A. & Morary, M. (1999). Control of systems integrating logic, dynamics, and constraints, *Automatica* 35: 407-427.

Cacciatori, E.; Vaughan, N.D. & Marco, J. (2006 ). Evaluating the Impact of Driveability Requirements on the Performance of an Energy Management Control Architecture for a Hybrid Electric Vehicle, *The 2nd International IEE Conference on Automotive Electronics*, IEE Savoy Place, UK.

Camacho, E.F. & Bordons C. (2004). *Model Predictive Control*, Springer-Verlag.

Cikanek, S.R. & Bailey, K.E. (2002). Regenerative braking system for a hybrid electric vehicle, *In Proc. of the American Control Conf.*, Anchorage, AK, Vol. 4, pp. 3129-3134.

Cundev, D. (2010). *Hybrid-electric Cars and Control Systems: Control strategy of car hybrid system and its experimental confirmation*, LAP LAMBERT Academic Publishing.

De la Pena, D.M.; Bemporad, A. & Filippi, C. (2006). Robust explicit MPC based on approximate multiparametric convex programming, *IEEE Transactions on Automatic Control* 51: 1399-1403.

Del Re, L.; Glielmo, L.; Guardiola, C. & Kolmanovsky, I. (Eds.) (2010). *Automotive Model Predictive Control: Models, Methods and Applications*, 1st edn, Springer.

Dextreit, C.; Assadian, F.; Kolmanovsky, I.; Mahtani, J. & Burnham, K. (2008). Hybrid electric vehicle energy management using game theory, *SAE World Congress & Exhibition*, SAE Paper No. 2008-01-1317.

Fekri, S. & Assadian, F. (2011). Application of Robust Multivariable Control on Energy Management of Hybrid Electric Vehicles, *International Journal of Vehicle Design (IJVD), Special Issue on "Automotive Mechatronics: Innovations in Design, Analysis and Implementation"*, March 2011, under review.

Fuhs, A. (2008). *Hybrid Vehicles*, 1st edn, CRC Press.

Gelb, A. (1974). *Applied Optimal Estimation*, MIT Press, MA, USA.

Grewal, M.S. & Andrews, A.P. (1993). *Kalman Filtering: Theory and Practice*, Prentice-Hall.

Hofman, T. & Druten, R.V. (2004). Energy analysis of hybrid vehicle powertrains, *In Proc. Of the IEEE Int. Symp. On Vehicular Power and Propulsion*, Paris, France.

Husain, I. (2003). *Electric and Hybrid Vehicles: Design Fundamentals*, 1st edn, CRC Press.

Johnson, V.H; Wipke, K.B. & Rausen, D.J. (2000). HEV control strategy for real-time optimization of fuel economy and emissions, *In Proc. of the Future Car Congress*, Washington DC, USA.

Kaszynski, M. & Sawodny, O. (2008). Modeling and identification of a built-in turbocharged diesel engine using standardized on-board measurement signals, *Proc. of IEEE International Conference on Control Applications (CCA 2008)*, pp. 7-12.

Kessels, J. (2007). *Energy Management for Automotive Power Nets*, PhD thesis, Faculteit Electrical Engineering, Technische Universiteit Eindhoven, The Netherlands.

Khayyam, H.; Kouzani, A.; Nahavandi, S.; Marano, V. & Rizzoni, G. (2010). *Energy Management*, InTech, chapter Intelligent Energy Management in Hybrid Electric Vehicles, pp. 147-175.

Kolmanovsky, I.; Siverguina, I. & Lygoe, B. (2002b). Optimization of powertrain operating policy for feasibility assessment and calibration: Stochastic dynamic programming approach, *In Proc. of the American Control Conf.*, Anchorage, AK, pp. 1425-1430.

Kwon, W.H. & Han, S. (2005). *Receding Horizon Control*, Springer-Verlag.

Lewis, J.B. (1980). *Automotive engine control: A linear quadratic approach*, PhD thesis, Dept. of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, MA, USA.

Ling, K.V.; Wu, B.F. & Maciejowski, J.M. (2008). Embedded model predictive control (MPC) using a FPGA, *Proc. 17th IFAC World Congress*, pp. 15250-15255.

Maciejowski, J.M. (2002). *Predictive Control with Constraints*, Prentice-Hal.

Mattingley, J. & Boyd, S. (2010). Real-time convex optimization in signal processing, *IEEE Signal Processing Magazine* 27: 50-61.

Mattingley, J. & Boyd, S. (2010). CVXGEN: A code generator for embedded convex optimization, *working manuscript*.

Mattingley & Boyd (2009). *Automatic Code Generation for Real-Time Convex Optimization*, Cambridge University, chapter Convex Optimization in Signal Processing and Communications.

Mehrotra, S. (1992). On the implementation of a primal-dual interior point method, *SIAM Journal on Optimization* 2: 575-601.

Milman, R. & Davidson, E.J. (2008). A fast MPC algorithm using nonfeasible active set methods, *Journal of Optimization Theory and Applications* 139: 591-616.

Milman, R. & Davidson, E.J. (2003). Fast computation of the quadratic programming subproblem in MPC, *Proc. of the American Control Conf.*, pp. 4723-4729.

Narciso, D.A.C.; Faisca, N.P. & Pistikopoulos, E.N. (2008). A framework for multi-parametric programming and control Ů an overview, *Proc. of IEEE International Engineering Management Conference, IEMC Europe 2008*, pp. 1-5.

Peng, J.; Roos, C. & Terlaky, T. (2002). *Self-Regularity: A New Paradigm for Primal-Dual Interior-Point Algorithms*, Princeton University Press.

Potra P.A. & Wright S.J. (2000). Interior-point methods, *Journal of Computational and Applied Mathematics* 124: 281-302.

Ramsbottom, M. & Assadian, F. (2006). Use of Approximate Dynamic Programming for the Control of a Mild Hybrid , *Proceedings of WMG Hybrid Conference*, University of Warwick, Warwick, United Kingdom, December 12-13.

Rao, C.V.; Wright, S.J. & Rawlings, J.B. (1998). Applications of interior-point methods to model predictive control, *Journal of Optimization, Theory and Applications* 99: 723-757.

Renegar, J. & Overton, M. (2001). *A Mathematical View of Interior-point Methods in Convex Optimization*, Society for Industrial & Applied Mathematics, USA.

Ripaccioli, G.; Bemporad, A.; Assadian, F.; Dextreit, C.; Di Cairano, S. & Kolmanovsky I.V. (2009). chapter Hybrid Modeling, Identification, and Predictive Control: An Application to Hybrid Electric Vehicle Energy Management, R. Majumdar and P. Tabuada (Eds.), *Proc. of the 12th International Conference on Hybrid Systems Computation and Control (HSCC 2009)*, Springer-Verlag Berlin Heidelberg, pp. 321Ű335.

Rousseau, G.; Sinoquet, D.; Sciarretta, A & Yohan Milhau. (2008). Design Optimisation and Optimal Control for Hybrid Vehicles, *In Proc. of the International Conference on Engineering Optimization*, EngOpt 2008, Rio de Janeiro, Brazil, June 2008.

Schouten, N.J.; Salman, M.A. & Kheir, N.A. (2002). Fuzzy logic control for parallel hybrid vehicles, *IEEE Trans. on Control Systems Technology* 10(3): 460-468.

Sciarretta, A. & Guzzella, L. (2007). Control of hybrid electric vehicles, *IEEE Control Systems Magazine* pp. 60-70.

Sontag, E. (1981). Nonlinear regulation: The piecewise linear approach, *IEEE Trans. on Automatic Control* 26(2): 346-358.

Steinmaurer, G. & Del Re, L. (2005). Optimal energy management for mild hybrid operation of vehicles with an integrated starter generator, *In Proc. of the SAE World Congress*, Detroit, Michigan.

Tate, E. & Boyd, S. (2001). Towards finding the ultimate limits of performance for hybrid electric vehicles, *In Proc. of the SAE Future Transportation Technology Conf.*, Costa Mesa, CA.

Tondel, P.; Johansen, T.A. & Bemporad, A. (2003). An algorithm for multi-parametric quadratic programming and explicit MPC solutions, *Automatica*, 39(3): 489-497.

Torrisi, F.D. & Bemporad, A. (2004). Hysdel - a tool for generating computational hybrid models, *IEEE Trans. on Control Systems Technology* 12(2): 235-249.

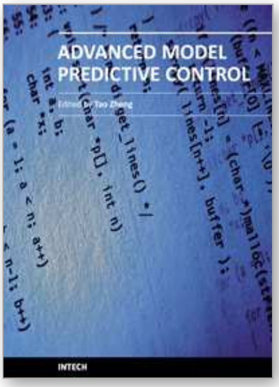
Vandenberghe, L. (2010). The CVXOPT linear and quadratic cone program solvers. *working manuscript*.

Vandenberghe, L. & Boyd S. (2004). *Convex Optimization*, Cambridge University Press.

Wang, Y. & Boyd, S. (2010). Fast model predictive control using online optimization, *J. Optim. Theory Appl* 18: 267-278.

Wang, Y. & Boyd, S. (2008). Fast model predictive control using online optimization, *Proceedings of the 17th IFAC World Congress*, Seoul, Korea, pp. 6974-6979.

Wright, S.J. (1997). *Primal-dual interior-point methods*, SIAM.




**Advanced Model Predictive Control**

Edited by Dr. Tao ZHENG

ISBN 978-953-307-298-2
Hard cover, 418 pages
**Publisher** InTech
**Published online** 24, June, 2011
**Published in print edition** June, 2011

Model Predictive Control (MPC) refers to a class of control algorithms in which a dynamic process model is used to predict and optimize process performance. From lower request of modeling accuracy and robustness to complicated process plants, MPC has been widely accepted in many practical fields. As the guide for researchers and engineers all over the world concerned with the latest developments of MPC, the purpose of "Advanced Model Predictive Control" is to show the readers the recent achievements in this area. The first part of this exciting book will help you comprehend the frontiers in theoretical research of MPC, such as Fast MPC, Nonlinear MPC, Distributed MPC, Multi-Dimensional MPC and Fuzzy-Neural MPC. In the second part, several excellent applications of MPC in modern industry are proposed and efficient commercial software for MPC is introduced. Because of its special industrial origin, we believe that MPC will remain energetic in the future.

# Fast model predictive control and its application to energy management of hybrid electric vehicles

Fekri, Sajjad

2011-06-24



Fekri S. Assadian F. (2011) Chapter 1: Fast model predictive control and its application to energy management of hybrid electric vehicles. In: Advanced Model Predictive Control. IntechOpen, July 2011, pp. 3-28

https://doi.org/10.5772/685