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

Dynamic Programming for Integrated Emission Management in Diesel Engines

J. van Schijndel * **M.C.F. Donkers** ** **F.P.T. Willems** *,***
W.P.M.H. Heemels *

* *Dept. Mech. Eng., Eindhoven University of Technology, Netherlands*
** *Dept. Elec. Eng., Eindhoven University of Technology, Netherlands*
*** *TNO Automotive, Powertrains, Netherlands*
j.v.schijndel@student.tue.nl,{m.c.f.donkers,f.p.t.willems,m.heemels}@tue.nl

Abstract: Integrated Emission Management (IEM) is a supervisory control strategy that aims at minimizing the operational costs of diesel engines with an aftertreatment system, while satisfying emission constraints imposed by legislation. In previous work on IEM, a suboptimal real-time implementable solution was proposed, which was based on Pontryagin's Minimum Principle (PMP). In this paper, we compute the optimal solution using Dynamic Programming (DP). As the emission legislation imposes a terminal state constraint, standard DP algorithms are sensitive to numerical errors that appear close to the boundary of the backward reachable sets. To avoid these numerical errors, we propose Boundary Surface Dynamic Programming (BSDP), which is an extension to Boundary Line Dynamic Programming and uses an approximation of the backward reachable sets. We also make an approximation of the forward reachable sets to reduce the grid size over time. Using a simulation study of a cold-start World Harmonized Transient Cycle for a Euro VI engine, we show that BSDP results in the best approximation of the optimal cost, when compared to existing DP methods, and that the real-time implementable solution only deviates 0.16 [%] from the optimal cost obtained using BSDP.

1. INTRODUCTION

To meet modern emission legislation limits, several new technologies were introduced in heavy-duty trucks in the last two decades. This has led to an increased complexity of the truck's engine and aftertreatment system. As a result, fuel consumption has remained almost constant over this period of time [ACEA, 2011]. Traditional supervisory control strategies are often based on heuristic rules, which do not lead to optimal solutions. Integrated Emission Management (IEM) is a supervisory control strategy that can increase fuel efficiency by using ideas from optimal control theory to maximize the synergy between the engine and the Engine Aftertreatment System (EAS).

Some results on emission management can be found in the literature [Serrao et al., 2013]. In IEM as described in [Cloudt and Willems, 2011], the objective is to minimize operational costs, while satisfying emission legislation constraints. This is achieved by finding a balance between using Exhaust Gas Recirculation (EGR) to reduce engine-out emissions and using the EAS. The IEM strategy proposed in [Cloudt and Willems, 2011] uses ideas from Energy Management Systems (EMS) for hybrid electric vehicles, see, e.g., [de Jager et al., 2013] and references therein. In the work on EMS, Dynamic Programming (DP) is often used to find the optimal solution. As this optimal solution is inherently noncausal and requires the drive cycle to be known a priori, suboptimal real-time implementable solutions have been proposed in the form of an Equivalent Cost Minimization Strategy (ECMS). ECMS is based on Pontryagin's Minimum Principle (PMP) and aims at approximating the optimal solution. Although the real-time implementable solution to the IEM problem of [Cloudt and Willems, 2011] is based on PMP and has a resemblance to ECMS, a comparison of this real-time solution with the optimal solution obtained through DP has never been made before.

This paper presents the optimal solution to the IEM problem for the diesel engine of a Euro VI heavy-duty truck completing a cold-start World Harmonized Transient Cycle (WHTC). A cold start is considered, since this cycle is more challenging from a thermal management and emissions point of view. The main contribution of this paper is a comparison of the suboptimal real-time implementable strategy with the optimal solutions obtained through DP. As emission legislation poses a constraint on the terminal state, the standard DP algorithm is sensitive to numerical errors [Sundström et al., 2010]. These numerical errors are caused by interpolating between finite and infinite costs and can be resolved by including an approximation of the backward reachable sets in the DP algorithm. As the work of [Sundström et al., 2010] can only be applied to scalar-state systems, the second contribution of this paper is the extension of the aforecited work towards the particular higher-order system of IEM. Moreover, we will include an approximation of the forward reachable sets in the newly proposed DP method to reduce the grid size over time. The contributions of this paper will be demonstrated by a simulation study of a cold-start WHTC. This provides insight into how the heuristic and optimal solutions to the IEM problem cope with the drive cycle and shows that

the novel extension to the DP algorithm outperforms the existing DP algorithms.

2. PROBLEM DESCRIPTION

The objective of this paper is, loosely speaking, to *minimize engine operational costs* over a drive cycle while *satisfying emission constraints* imposed by legislation. Before being able to formalize the objective of IEM, which we will do at the end of this section, we will first discuss the controller structure and present the necessary models.

2.1 System Description and Control Structure

In this paper, we consider a typical Euro VI heavy-duty powertrain, consisting of a 6 cylinder, 12.9 liter, 375 kW engine and an Engine Aftertreatment System (EAS). The engine is equipped with a cooled high pressure Exhaust Gas Recirculation (EGR) system and a Variable Turbine Geometry (VTG) with charge air cooler. The EAS consists of a Diesel Oxidation Catalyst (DOC), a Diesel Particulate Filter (DPF), a Selective Catalytic Reduction catalyst (SCR) and an Ammonia Oxidation catalyst (AMOX).

A block diagram of the control structure can be found in Fig. 1. The amount of fuel fed to the engine is taken so that the requested power is delivered. The autonomous AdBlue dosing strategy, which controls the SCR system, aims at maximising NO_x conversion given a certain amount of NH_3-slip. The air management controls the position of the EGR valve and the geometry of the VTG. The supervisory control strategy determines the desired EGR and VTG mass flows, which result in a tradeoff between emissions and fuel economy. We focus on tailpipe NO_x emissions, because present technology is capable of reducing other emissions such that they comply with the Euro VI legislation.

2.2 Engine and Aftertreatment Models

The engine model is static and based on [Wahlström and Eriksson, 2011]. It uses mass and energy balances combined with empirical relations. The model predicts the fuel mass flow $\dot{m}_f$, the total exhaust gas mass flow $\dot{m}_{\text{exh}}$, the engine-out NO_x mass flow $\dot{m}_{\text{NO}_x,\text{eo}}$ and the exhaust gas temperature T_{exh} as function of desired torque τ_d [Nm] for a specific desired rotation speed ω_d [rads^{-1}], and as function of the mass flow through the EGR and VTG, denoted by u_1 [kgs^{-1}] and u_2 [kgs^{-1}], respectively. The supervisory controller determines u_1 [kgs^{-1}] and u_2 [kgs^{-1}], where the allowable u_1 and u_2 depends on the combination of τ_d and ω_d. It is assumed that the air management and the fuel control perfectly track their setpoints given by the IEM strategy. The experimental results of [Willems et al., 2013] suggest that this is a valid assumption.

In earlier work on IEM, the EAS is modelled using a third-order dynamic model. In this paper, we reduce this model to a second-order model, because it simplifies the computations for DP later, while still allowing us to compare the real-time implementable strategy with the optimal solution obtained through DP. Note that the methods presented below can just as well be applied to the third-order model, albeit at the cost of a higher

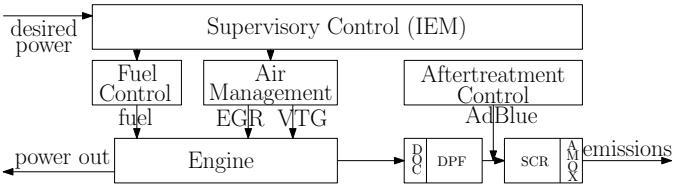


Fig. 1. Block diagram of the powertrain.

computational complexity. To obtain the second-order model, we describe the temperatures for the DOC and SCR in [Willems et al., 2013] by one single temperature T_{EAS} [K] that represents both catalytic converters. To achieve this, the heat capacity of the entire system is taken as the sum of the heat capacities of the individual components, resulting in

$$\dot{x} = f(x,u,t) := \begin{bmatrix} k_1\dot{m}_{\text{exh}}(T_{\text{exh}} - x_1) + k_2(T_{\text{amb}} - x_1) \\ \dot{m}_{\text{NO}_x,\text{eo}}(1 - \eta_{\text{SCR}}(x_1, \dot{m}_{\text{exh}})) \end{bmatrix}, \quad (1)$$

where $x = [x_1 \;\; x_2]^\top = [T_{\text{EAS}} \;\; m_{\text{NO}_x,\text{tp}}]^\top$, T_{exh} [K] and T_{amb} [K] are the temperature of the engine-out exhaust gas and the ambient, respectively, $\dot{m}_{\text{exh}}$ [kgs^{-1}] is the total mass flow of the engine-out exhaust gas, and $m_{\text{NO}_x,\text{tp}}$ [kg] is the cumulative amount of tailpipe NO_x emissions. The SCR efficiency η_{SCR} is a nonlinear function that depends on the EAS temperature and the exhaust gas mass flow $\dot{m}_{\text{exh}}$. The heat transfer coefficients of the EAS are $k_1 = 3.56 \cdot 10^{-2}$ [kg^{-1}] and $k_2 = 5.34 \cdot 10^{-4}$ [s^{-1}].

2.3 Optimal Control Problem Formulation

We can now formulate IEM as an optimal control problem, in which the objective is to minimize the total fluid costs, while staying within the constraints on tailpipe NO_x emissions. This problem can be expressed as:

$$J_c(x(t_s)) = \min_{u_1,u_2} \int_{t_s}^{t_f} \pi_f \dot{m}_f(u,t) + \pi_A \dot{m}_A(x_1,u,t)\,\mathrm{d}t, \quad (2)$$

subject to (1) and

$$\underline{x}_2 \le x_2(t_f) \le \overline{x}_2, \quad (3)$$

where $\pi_f = 1.34$ [€/kg] and $\pi_A = 0.50$ [€/kg] are the price of fuel and AdBlue, respectively, $\dot{m}_f$ is the fuel mass flow of the engine model, and $\underline{x}_2$ [kg] and $\overline{x}_2$ [kg] are the terminal state constraints on tailpipe NO_x emissions, where typically $\underline{x}_2 = 0$ [kg].

The AdBlue mass flow $\dot{m}_A$ in (2) results from the assumption that the reaction with NO_x is stoichiometric and that the SCR control system aims at maximizing NO_x conversion. It is given by $\dot{m}_A = k_3\dot{m}_{\text{NO}_x,\text{eo}}\eta_{\text{SCR}}(x_1, \dot{m}_{\text{exh}})$, in which $k_3 = 2.007$ [-] is the stoichiometric ratio between AdBlue and NO_x, and $\dot{m}_{\text{NO}_x,\text{eo}}$ and η_{SCR} are given as before.

3. DYNAMIC PROGRAMMING

Dynamic Programming (DP) is a way to find the optimal solution to an optimal control problem [Bertsekas, 2005]. To solve the optimal control problem given by (2) subject to (1) and (3) using DP, we first approximate the problem in discrete time using the forward Euler method with step size $h = 1$ [s], resulting in

$$J(x[0]) = \min_{\{u[k]\}_{k=0}^{N-1} \in \mathcal{U}^t} \sum_{k=0}^{N-1} G(x[k],u[k],k) + G(x[N],N), \quad (4)$$

subject to

$$x[k+1] = F(x[k], u[k], k) := x[k] + f(x[k], u[k], k), \quad (5)$$

with $f(x, u, k)$ as in (1), and

$$L(x[N]) \leq 0. \quad (6)$$

We use the forward Euler method because of its simplicity and because of the fact that we have relatively slow dynamics. In (4), the running cost is given by

$$G(x, u, k) = \pi_f \dot{m}_f(u, k) + \pi_A \dot{m}_A(x_1, u, k), \quad (7)$$

for $k = \{0, 1, \ldots, N-1\}$, the terminal cost by $G(x, N) = 0$, the terminal constraint by

$$L(x) = [x_2 - \overline{x}_2 \quad \underline{x}_2 - x_2]^\top, \quad (8)$$

and $\mathcal{U}^t = \mathcal{U}[0] \times \ldots \times \mathcal{U}[N-1]$.

The DP solution to the discrete-time optimal control problem (4)-(6) can be found, see, e.g., [Bertsekas, 2005], by solving the backwards recursion

$$J(x, k) = \min_{u \in \mathcal{U}[k]} \{G(x, u, k) + J(F(x, u, k), k+1)\}, \quad (9)$$

subject to $J(x, N) = G(x, N)$ for all $k \in \{0, 1, \ldots, N-1\}$, and for all $x \in \mathcal{X}^b[N]$, where

$$\mathcal{X}^b[N] = \{x \in \mathbb{R}^n \mid L(x) \leq 0\}. \quad (10)$$

The function $J(x, k)$ is often called the cost-to-go function.

We will now discuss two DP methods, in which we perform a spatial discretization to produce finite sets $\mathcal{X}^o[k]$ and $\mathcal{U}^o[k]$, where $\mathcal{X}^o[k] \subset \mathcal{X}[k]$, $\mathcal{U}^o[k] \subset \mathcal{U}[k]$. We use linear interpolation and extrapolation to obtain values for $x[k] \in \mathcal{X}[k]$ that are not contained in $\mathcal{X}^o[k]$. The accuracy of the solution of the DP problem can be increased by increasing the number of elements in $\mathcal{X}^o[k]$ and $\mathcal{U}^o[k]$ in a well-distributed way, albeit at the cost of a higher computational complexity. An important difference between the DP methods discussed below is how the set $\mathcal{X}[k]$, $k \in \{0, 1, \ldots, N\}$, is chosen.

3.1 Conventional Dynamic Programming

In Conventional DP (CDP), the terminal state constraint (6) is taken into account by making the terminal cost of the infeasible area, i.e., where $L(x(N)) > 0$, large. This means that we can solve the DP problem by initializing

$$J(x, N) = \begin{cases} G(x, N), & \text{if } L(x) \leq 0, \\ \alpha, & \text{if } L(x) > 0, \end{cases} \quad (11)$$

for $x \in \mathcal{X}^o[N]$, where $\alpha \in \mathbb{R}$ is chosen sufficiently large. Then, for $k = N-1$ to 0, $x \in \mathcal{X}^o[k]$, we compute

$$J(x, k) = \min_{u \in \mathcal{U}^o[k]} \{G(x, u, k) + J(F(x, u, k), k+1)\}. \quad (12)$$

In CDP, a time-invariant set $\mathcal{X}^I$ is chosen that has its points well distributed over a sufficiently large part of the state space $\mathbb{R}^n$, so that all relevant points are included. We take $\mathcal{X}[k] := \mathcal{X}^I$ for all $k \in \{0, 1, \ldots, N-1\}$.

CDP is the most straightforward way to solve a DP problem with terminal state constraints. However, the fact that we grid and approximate infeasible points in the state space by a large but finite number α causes the algorithm to suffer from numerical errors due to interpolation at the boundary [Sundström et al., 2010]. As illustrated in Fig. 2, the true minimal cost is achieved somewhere in between x^{i-1} and x^i, but due to interpolation, the minimum appears to occur at x^i. In this figure, the dotted line illustrates the interpolation when doing conventional DP, while the dashed line illustrates the real cost-to-go and could be captured by a grid point (cross) on the boundary of the feasible area. To minimize the numerical errors using CDP, we have to make sure that α is not too high, yet high enough to avoid it from affecting the solution to (12).

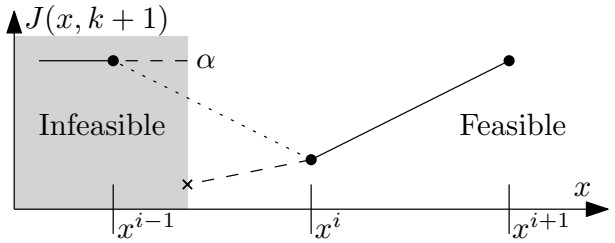


Fig. 2. Possible interpolation error at the boundary in 1D.

A way to find a proper (and possibly time and position dependant) value for this parameter α, thereby reducing the interpolation errors, is proposed in the Level-Set Dynamic Programming (LSDP) method of [Elbert et al., 2013]. In this method, the feasible sets are described by a function $\mathcal{I}(x, k)$ that satisfies $\mathcal{I}(x, k) \leq 0$ if x is backward reachable at time step k and $\mathcal{I}(x, k) > 0$ if it is not. In doing so, a smooth transition between the optimal costs of feasible and infeasible points is achieved. This leads to an implicit (and approximate) solution to the interpolation errors near the boundary of the backward reachable sets (i.e., the feasible regions), as was shown in Fig. 2. We will propose an explicit (and approximate) solution to this problem by extending [Sundström et al., 2010] towards the higher-order system of the previous section.

3.2 Boundary-Surface Dynamic Programming

We will now propose Boundary-Surface Dynamic Programming (BSDP), which can be seen as an extension to Boundary-Line Dynamic Programming (BLDP) [Sundström et al., 2010] and which can only be applied to scalar-state systems. The main idea of BSDP, as well as BLDP of [Sundström et al., 2010], is to include grid points that are on the boundary of the backward reachable sets to avoid the aforementioned interpolation errors. In this context, the backwards reachable set $\mathcal{X}^b[k]$ is, loosely speaking, defined as the set of $x[k]$ at time $k \in \{0, 1, \ldots, N-1\}$ for which there exist a solution to (5) that satisfies $x[N] \in \mathcal{X}^b[N]$, with $\mathcal{X}^b[N]$ as in (10). In the BSDP method that we propose, we also compute the forward reachable sets $\mathcal{X}^f[k]$, which can be defined as the set of $x[k]$ at time $k \in \{1, 2, \ldots, N\}$ which can be reached from the set of initial conditions $\mathcal{X}^f[0]$. Computations of these two sets will, in principle, allow us to place grid points $x \in \mathcal{X}^o[k]$, $k \in \{0, 1, \ldots, N-1\}$, exactly on the boundary of the intersection of the forward and backward reachable sets, thereby having no grid points x outside the reachable sets.

To compute the forward and backward reachable sets, we propose a heuristic algorithm that takes into account the specific structure of the IEM problem in $\mathbb{R}^2$, i.e., $n = 2$, of Section 2. We assume that the dynamics (5) are boundary preserving in the sense that the boundary of a set that is mapped using the dynamics (5) is equal to the mapping of the boundary of that same set. This can be guaranteed under certain homeomorphism conditions on F as in (5).

To compute approximations $\mathcal{X}^f[k]$, $k \in \{0,1,\ldots,N\}$, of the forward reachable sets for a given convex set of initial conditions $\mathcal{X}^f[0] \subset \mathbb{R}^2$, the boundary of the forward reachable set $\partial\mathcal{X}^f[k]$ is approximated by computing

$$\underline{x}_1[k] = \min\left\{x_1 \in \mathbb{R} \mid x \in \mathcal{X}^f[k]\right\}, \tag{13a}$$

$$\overline{x}_1[k] = \max\left\{x_1 \in \mathbb{R} \mid x \in \mathcal{X}^f[k]\right\}, \tag{13b}$$

and defining

$$\tilde{x}_1^i[k] = \tfrac{i-1}{s-1}\,\underline{x}_1[k] + \tfrac{s-i}{s-1}\,\overline{x}_1[k], \tag{14}$$

for $i \in \{1,\ldots,s\}$, with some $s \in \mathbb{N}$, allowing us to compute

$$\underline{x}_2^i[k] = \min\left\{x_2 \in \mathbb{R} \,\middle|\, \begin{bmatrix}\tilde{x}_1^i[k]\\ x_2\end{bmatrix} \in \mathcal{X}^f[k]\right\}, \tag{15a}$$

$$\overline{x}_2^i[k] = \max\left\{x_2 \in \mathbb{R} \,\middle|\, \begin{bmatrix}\tilde{x}_1^i[k]\\ x_2\end{bmatrix} \in \mathcal{X}^f[k]\right\}, \tag{15b}$$

to obtain the following approximation of the set $\partial\mathcal{X}^f[k]$

$$\hat{\mathcal{X}}^f[k] := \bigcup_{i=1}^{s} \left\{ \begin{bmatrix}\tilde{x}_1^i[k]\\ \underline{x}_2^i[k]\end{bmatrix}, \begin{bmatrix}\tilde{x}_1^i[k]\\ \overline{x}_2^i[k]\end{bmatrix} \right\}. \tag{16}$$

The final step is to compute

$$\mathcal{X}^f[k+1] := \mathrm{co}\{F(x,u,k) \in \mathbb{R}^n \,|\, x \in \hat{\mathcal{X}}^f[k], u \in \mathcal{U}^o[k]\}, \tag{17}$$

where the notation $\mathrm{co}\mathcal{X}$ denotes the convex hull of a set $\mathcal{X}$. Indeed, (17) computes an approximation of the forward reachable set as it maps all the values x can attain at time k to all the values x can attain at time $k+1$. Note that the points on the boundary of the set $\mathcal{X}^f[k]$ are found in a simple way by using the convex hull operation in (17) and that by increasing $s \in \mathbb{N}$, we are able to obtain a more accurate approximation of the true forward reachable set. Note also that by taking the convex hull we will typically create an overapproximation of the forward reachable set, meaning that we capture at least all forward reachable points, but possibly some points that are not forward reachable. However, the convex hull operation significantly reduces the computational complexity and, as it turns out, will lead, for the system under study, to a good approximation of the true forward reachable sets. To find all forward reachable sets, we solve (13)-(17) recursively for $k \in \{0,1,\ldots,N-1\}$.

To compute the backward reachable sets, we discuss an extension to [Sundström et al., 2010] towards a second-order system with a special structure. The extension of the ideas presented in this paper to more general systems are currently under investigation. A requirement in this paper is that the terminal state constraint $\mathcal{X}^b[N]$ given by the mapping L as in (8) can be written as $\mathcal{X}^b[N] = \{x \in \mathbb{R}^2 \mid \underline{r}(x_1) \le x_2 \le \overline{r}(x_1)\}$ for some functions $\underline{r}, \overline{r} : \mathbb{R} \to \mathbb{R}$. To avoid unnecessary computations, we directly incorporate the overapproximation $\mathcal{X}^f[k]$ of the forward reachable sets. Therefore we use (14), so that for a given $\mathcal{X}^b[N]$, the boundary of the backward reachable set $\partial\mathcal{X}^b[k]$ is recursively approximated by computing for $i \in \{1,\ldots,s\}$

$$\overline{x}_2^i[k] = \max_{u \in \mathcal{U}^o[k]} \{x_2 \in \mathbb{R} \,|\, F([\tilde{x}_1^i[k]\ x_2]^\top, u, k) \in \tilde{\mathcal{X}}^b[k+1]\}, \tag{18a}$$

$$\underline{x}_2^i[k] = \min_{u \in \mathcal{U}^o[k]} \{x_2 \in \mathbb{R} \,|\, F([\tilde{x}_1^i[k]\ x_2]^\top, u, k) \in \tilde{\mathcal{X}}^b[k+1]\}. \tag{18b}$$

The approximation of the backward reachable set is then computed with

$$\mathcal{X}^b[k] := \bigcup_{i=1}^{s-1} \mathrm{co}\big(\{\tilde{x}_1^i[k]\} \times [\underline{x}_2^i[k], \overline{x}_2^i[k]] \bigcup \{\tilde{x}_1^{i+1}[k]\} \times [\underline{x}_2^{i+1}[k], \overline{x}_2^{i+1}[k]]\big). \tag{19}$$

To find all backward reachable sets, we solve (18)-(19) recursively for $k \in \{0,1,\ldots,N-1\}$.

The approximations of the forward and backward reachable sets can be used to place all grid points on the boundary and in the interior of the intersection of these two sets, i.e., $x \in \mathcal{X}^o[k] \subset \mathcal{X}[k] := \tilde{\mathcal{X}}^f[k] \cap \tilde{\mathcal{X}}^b[k]$ for all $k \in \{0,1,\ldots,N-1\}$. For our system, we can find these sets with relatively low computational effort, which gives increased accuracy for only a slightly increased computational time.

4. REAL-TIME IMPLEMENTABLE STRATEGY

Let us now present a suboptimal real-time implementable strategy that is based on [Cloudt and Willems, 2011]. This methodology has resemblance to ECMS and is based on Pontryagin's Minimum Principle (PMP). PMP says that the optimal solution $u^*[k]$ along an optimal trajectory $x^*[k]$ necessarily satisfies

$$u^*[k] = \arg\min_{u \in \mathcal{U}^o[k]} H(x^*[k], u, k), \tag{20a}$$

$$\lambda[k] = \tfrac{\partial H(x^*[k], u^*[k], k)}{\partial x^*[k]}, \tag{20b}$$

$k \in \{0,1,\ldots,N-1\}$, subject to some boundary values, see, e.g., [Bertsekas, 2005]. In this expression, the Hamiltonian is given by $H(x,u,k) = G(x,u,k) + \lambda^\top[k+1]F(x,u,k)$, where $G(x,u,k)$ is the running cost as in (7), $\lambda[k]$, $k \in \{0,1,\ldots,N\}$, are the costates and $F(x,u,k)$ are the state dynamics as in (5).

In principle, the solution to the IEM problem can be obtained by numerically solving (20), but we will not pursue this solution strategy in this paper. Instead, we only use PMP to obtain a real-time implementable solution. This real-time suboptimal solution, denoted by RT-PMP, is based on the observation in [Willems et al., 2013] that λ_1 has little influence on the minimal cost over a drive cycle. It is obtained by solving (20a), with $\lambda_1[k] = 0$ and $\lambda_2[k] = \lambda_2$ for some $\lambda_2 \in \mathbb{R}$ and for all $k \in \{0,1,\ldots,N\}$. This allows the IEM problem to be solved using a simple shooting method to search for λ_2, or by simply taking a fixed value that is tuned over a representative cycle, as was done in [Willems et al., 2013].

5. SIMULATION STUDY

We will now demonstrate and compare the proposed solution strategies to the optimal control problem using a case study of a typical type approval drive cycle. In particular, we compare CDP, LSDP as in [Elbert et al., 2013], the newly proposed BSDP, and the real-time implementable solution from Section 4. This will allow us to both assess the performance of the newly proposed DP method, as well as the amount of performance degradation of the real-time implementable solution.

5.1 Case Description

In this simulation study, a cold-start World Harmonized Transient Cycle (WHTC) will be used. We consider a cold

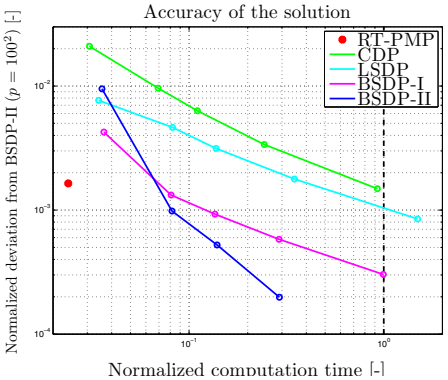


Fig. 3. Computation time vs accuracy.

start, which corresponds to $T_{\text{EAS}}[0] = x_1[0] = 298$ [K] and $T_{\text{amb}} = 298$ [K], as this case is more challenging from a thermal management and emissions point of view. From [Cloudt and Willems, 2011], we find that we can achieve the legislation limit when we choose a NO_x tailpipe emission limit of 0.8 [g/kWh] for the cold-start WHTC. Note that engine heat-up is not incorporated in the modelling and that we use the same models for computing the solutions to the control problem and for doing simulations.

5.2 Implementation Details

Before showing the results of the simulation study, let us discuss some of the implementation details for each of the applied methods. For the number of grid points of $\mathcal{U}^o[k] \subset \mathcal{U}[k]$, we take $q = 21^2$ equidistantly spaced points for all $k \in \{0, 1, \ldots, N-1\}$ for all methods.

For CDP and LSDP of [Elbert et al., 2013], we have to specify grid points $\mathcal{X}^I := [\underline{x}_1, \overline{x}_1] \times [\underline{x}_2, \overline{x}_2] \subset \mathbb{R}^2$. We choose in this example the grid points $x^i \in \mathcal{X}^I$, $i \in \{1, \ldots, p\}$, to be on a regularly-spaced rectangular grid with an equal number of quantization levels for both states x_1 and x_2 and we will vary the number of grid points $p \in \mathbb{N}$ to obtain more accurate solutions. We select $\underline{x}_1 = 298$ [K], as the temperature of the EAS will never drop below the ambient temperature, and $\overline{x}_1 = 900$ [K], as larger temperatures are not forward reachable. Furthermore, we take $\underline{x}_2 = 0$ [kg], as the cumulative tailpipe NO_x cannot decrease, and $\overline{x}_2 = 28.7 \cdot 10^{-3}$ [kg], which corresponds to the emission constraint. The parameter α as in (11), is chosen based on maximizing the operational costs in (2), subject to no emission constraints, which yields $\alpha = 11.67$.

For BSDP, we use the method as outlined in Section 3.4 with $s = \sqrt{p}$ for some $p \in \mathbb{N}$ and take $\mathcal{X}^f[0] := \{298\} \times \{0\}$ and $\mathcal{X}^b[N] = \{x \in \mathbb{R}^2 \mid \underline{x}_2 \leq x_2 \leq \overline{x}_2\}$. We distinguish two cases, denoted by BSDP-I and BSDP-II. For BSDP-I, we take $\underline{x}_2 = 0$ [kg] and $\overline{x}_2 = 28.7 \cdot 10^{-3}$ [kg] and, for BSDP-II, we take $\underline{x}_2 = \overline{x}_2 = 28.7 \cdot 10^{-3}$ [kg]. Hence, we force the solution of the latter case to be exactly on the emission legislation limit. The reason for doing so is that engineering insight tells us that the cost optimal solution has to be on the emission legislation target and it allows us to place the grid points as closely together as possible.

5.3 Optimality of Solutions

Our main objective is to minimize the operational costs over the drive cycle. As mentioned earlier, the accuracy of this cost for each DP method depends on the number of grid points p. Therefore, the algorithms of the different DP methods are simulated for different numbers of grid points, i.e., we take $p \in \{10^2, 20^2, 30^2, 50^2, 100^2\}$.

Table 1. Results of the cold-start WHTC.

Definition	Unit	BSDP-II	RT-PMP
Total fluid costs	[-]	1	1.0016
AdBlue costs	[-]	0.0181	0.0170
Tailpipe emissions $m_{\text{NO}_x,\text{tp}}/W_{\text{tot}}$	[g/kWh]	0.800	0.800
Engine-out emissions $m_{\text{NO}_x,\text{eo}}/W_{\text{tot}}$	[g/kWh]	6.160	5.816

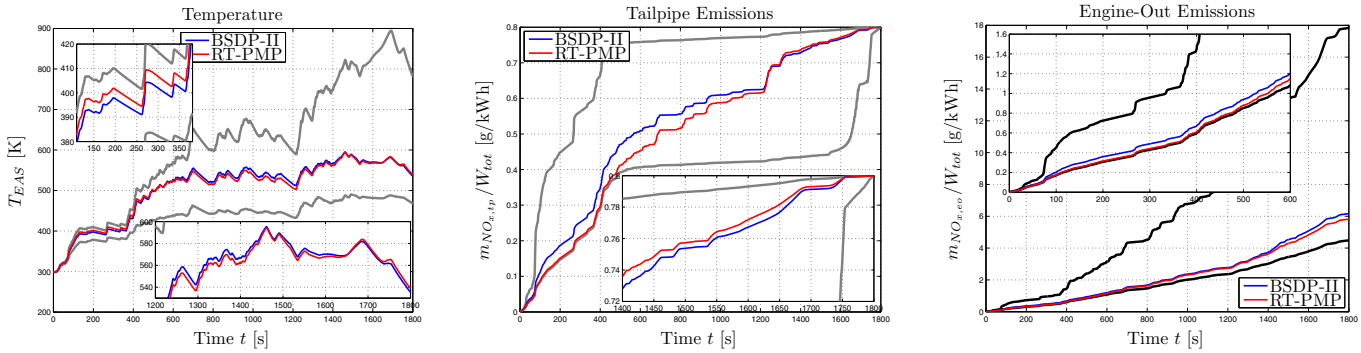
The results of the simulation study are shown in Fig. 3. As BSDP-II with $p = 100^2$ yields to most accurate approximation of the optimal solution, i.e., has the lowest cost, we normalize all results with respect to BSDP-II with $p = 100^2$ for both cost and computation time. All computations are done on a 2.53GHz (quadcore) PC, with 4 GB RAM running Matlab 2010b and the computation time for BSDP-II with $p = 100^2$ equals 19 hours.

As expected, the computation time and the accuracy increases as p increases. To quantify our results, let us look along the dashed line in Fig. 3. We can observe that for this computation time, the normalized costs for CDP, LSDP and BSDP-I are 1.0013, 1.0012, 1.0003, respectively. This shows that BSDP as proposed in this paper is useful in achieving accurate results, while keeping the computational efficiency manageable. Finally, we can conclude that RT-PMP achieves a good approximation of the real optimal solution, as the corresponding cost only deviates 0.16 [%] from BSDP-II with $p = 100^2$.

5.4 Optimal Trajectory

We will now discuss the optimal trajectories of the most accurate DP approach, which is BSDP-II with $p = 100^2$, and the real-time implementable method (RT-PMP). For these two approaches, the states $x_1 = T_{\text{EAS}}$, $x_2 = m_{\text{NO}_x,\text{tp}}$ and the engine-out emissions $m_{\text{NO}_x,\text{eo}}$ are shown as a function of time in Figs. 4a-4c. Note that W_{tot} is the total work delivered over the cycle in Fig. 4b and Fig. 4c. The grey lines in Fig. 4a and Fig. 4b show the intersection of the reachable sets calculated with BSDP-II. The black lines in Fig. 4c show the cumulative minimum and maximum possible $m_{\text{NO}_x,\text{eo}}$, which correspond to the maximum and the no-EGR case, respectively. In Table 1, the values for the trajectories at time step $k = N$ are given, as well as the total fluid cost and the AdBlue costs (both normalized with respect to the total fluid cost of BSDP-II).

From the simulation results, it can be seen that both solutions aim at increasing the EAS temperature almost as fast as possible, see Fig. 4a, while keeping engine-out emissions low (which can be done by using EGR), see Fig. 4c. From 400 to 600 seconds, T_{EAS} increases from 425 [K] to 525 [K], see Fig. 4a. This is the range of temperatures for T_{EAS} where the maximum possible SCR efficiency changes from 58 [%] to 99 [%], which means that less EGR is required and thus a higher $\dot{m}_{\text{NO}_x,\text{eo}}$ is allowed. This results in lower fluid costs, since (a part of the) expensive fuel associated with EGR is replaced by cheaper AdBlue. In other words, the choice for the inputs is a weighted decision between keeping the SCR efficiency sufficiently high by keeping T_{EAS} sufficiently high, while



(a) Temperature EAS. (b) NO_x tailpipe emissions. (c) NO_x engine-out emissions.

Fig. 4. Results of the Cold-Start WHTC.

consuming the least amount of fuel in doing so. Note that we need EGR at the beginning of a cold-start cycle, since the EAS temperature is too low to convert NO_x emissions adequately. Note also that EGR is used over the complete cycle, which implies that a delicate combination of EGR and SCR results in the lowest possible cost, while satisfying the emission constraints.

When we compare the different solutions, we can observe in Fig. 4c that the trajectory for RT-PMP stays near the minimal engine-out emissions during the first 500 seconds and in Fig. 4b that it also stays near the minimal tailpipe emissions. The latter implies that more EGR is used in the RT-PMP strategy with respect to the BSDP-II strategy. We can also see in Fig. 4a that, for RT-PMP, T_{EAS} is about 6 [K] higher in this period than T_{EAS} for BSDP-II. However, at these low temperatures, this has no influence on the SCR's maximum efficiency. At 500 seconds, the difference in cumulative tailpipe emissions is maximal and is equal to 0.060 [g/kWh]. It can be seen that after 500 seconds, T_{EAS} for RT-PMP is about 8 [K] lower than T_{EAS} for BSDP-II. This implies that the SCR efficiency for RT-PMP is slightly lower than the SCR efficiency for BSDP-II, meaning that less AdBlue is used than in the case of BSDP-II and also implies that EGR is mainly used to stay within emission limits. This can also be concluded from Table 1. To summarize, to be able to satisfy the emission constraints, RT-PMP uses, when compared to BSDP-II, more of the relatively expensive fuel associated with EGR instead of the cheaper AdBlue. Another explanation for the difference between RT-PMP and BSDP-II is that $\lambda_1[k] = 0$ in RT-PMP, as $\lambda_1[k]$ corresponds to stimulate heating of the EAS in the optimal control problem. However, the result of this effect is marginal, as can also be seen in Table 1.

6. CONCLUSIONS

In this paper, the optimal solution to the Integrated Emission Management (IEM) problem was determined over a cold-start World Harmonized Transient Cycle (WHTC). This optimal solution was determined using Dynamic Programming (DP) and was compared to the suboptimal real-time implementable strategy that was proposed before and is based on Pontryagin's Minimum Principle. Additionally, we compared different DP methods of which one was a novel extension of an existing approach. The solutions were obtained for a combination of a static engine model and a dynamic engine aftertreatment model.

All proposed methods successfully minimize fuel and AdBlue costs up to a certain numerical accuracy. We found that DP suffers from interpolation errors, but by using the Boundary-Surface Dynamic Programming (BSDP) in the form proposed in this paper, we can increase accuracy of the solution at the cost of only a small amount increase of computational time. The solution obtained through BSDP, where $p = 100^2$ and the terminal constraint is chosen such that the emission legislation is exactly satisfied, delivers the most cost efficient trajectory over the cycle. The real-time implementable strategy based on Pontryagin's minimal principle (RT-PMP) deviates only 0.16 [%] from the most accurate optimal solution.

REFERENCES

ACEA. Commercial vehicles and CO2. Technical report, ACEA, 2011.

D. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 2005.

R. Cloudt and F. Willems. Integrated emission management strategy for cost optimal engine-aftertreatment operation. *SAE Int. J. Engines*, 4(1):1784–1797, 2011.

B. de Jager, T. van Keulen, and J. Kessels. *Optimal Control of Hybrid Vehicles*. Springer, 2013.

P. Elbert, S. Ebbesen, and L. Guzella. Implementation of dynamic programming for n-dimensional optimal control problems with final state constraints. *IEEE Trans. Contr. Syst. Technol.*, 21(3):924–931, 2013.

L. Serrao, A. Sciarretta, O. Grondin, A. Chasse, Y. Creff, D. Di Domenico, P. Pognant-Gros, C. Querel, and L. Thibault. Open issues in supervisory control of hybrid electric vehicles: A unified approach using optimal control methods. *Oil & Gas Science and Techn.*, 68(1): 23–33, 2013.

O. Sundström, D. Ambühl, and L. Guzzella. On implementation of dynamic programming for optimal control problems with final state constraints. *Oil & Gas Science and Techn.*, 65(1):91–102, 2010.

J. Wahlström and L. Eriksson. Modelling diesel engines with a variable-geometry turbocharger and exhaust gas recirculation by optimization of model parameters for capturing non-linear system dynamics. *Proc. IMechE, Part D: J. Automobile Eng.*, 255(7):960–986, 2011.

F. Willems, P. Mentink, F. Kupper, and E. van den Eijnden. Integrated emission management for cost optimal EGR-SCR balancing in diesels. *Proc. IFAC Symp. Adv. Automotive Contr.*, pages 711–716, 2013.