Multi-Level Energy Management for Hybrid Electric Vehicles—Part I

Vital van Reeven [1,2] and Theo Hofman [1,*]

[1]   Department of Mechanical Engineering, Eindhoven University of Technology,
     5600 MB Eindhoven, The Netherlands; energy@vitalvanreeven.nl
[2]   DAF Trucks N.V., 5600 PT Eindhoven, The Netherlands
[*]   Correspondence: t.hofman@tue.nl; Tel.: +31-40-247-2827

Abstract: The fuel economy of a hybrid electric vehicle (HEV) is improved, by taking the energy relevant system states into account in the energy management system (EMS). With an increasing number of states and decision variables, energy optimizing algorithms in the EMS can be prohibitive for real-time implementation. In part I of this work, a model-based, multi-level approach is taken to subdivide the original (large) optimization problem into computational efficient sub-problems, based on optimal control techniques using a preview. The resulting EMS solves the problem of power-split between engine and motor/generator, mode and gear switching including switching costs, with battery energy constraints. The superior energy efficiency of the multi-level EMS is simulated on a representative heavy duty drive cycle, where it saves 7.0% fuel, compared to a conventional vehicle, where the baseline EMS for the HEV saves 5.8%. In part II, real-world validation of the EMS is performed.

Keywords: energy management; optimal control; multi-level; switching decisions

1. Introduction

Hybrid electric vehicles (HEVs) have emerged as a promising solution to reduce operational cost in commercial road transportation, while complying to increasingly stringent emission legislation. Since HEVs have more than one power converter, they offer additional control freedom, compared to conventional vehicles, which give opportunities for the energy management system (EMS) to decrease fuel consumption and emissions. The EMS has to consider the energy relevant systems of the HEV and already a large amount of solutions have been proposed that take the battery energy dynamics into account [1]. However, the system efficiency can be further improved by taking additional system states into account as suggested by [2,3], referred to as a 'unified', 'integrated', 'total' or 'holistic' energy management. Examples of such additional systems are the battery with its temperature and aging characteristics [4,5], engine after-treatment system [3,6], waste-heat recovery system [7], combustion engine [8], or the cabin heater [9].

Using model information of these systems, the EMS can be posed as an optimization problem, in order to ensure energy efficiency. The computational complexity of this problem, however, increases with the number of states, presence of state constraints, number of decision variables, type of decision variables (continuous/discrete) and nonlinearity of the models. Many methods exist to design an optimizing EMS, using e.g., Dynamic Programming [10,11], Pontryagin's Minimum Principle [12] or Model Predictive Control [13,14] and generate close to optimal results; however they can be computationally too demanding for real-time implementation. In this work, a combination of methods is used, to ensure an efficient computation for each of the sub-problems.

An overview of real-time implementable optimization approaches for a (parallel) hybrid vehicle with the decision variables: power-split, stop-start and gear selection, is given in Table 1. In particular, discrete control variables increase the computational complexity, explaining why gear selection is often omitted from the EMS or solved in a separate step with heuristics [15–17]. In [18], an integrated approach is presented, where gear selection and stop-start are part of a sequential optimization with the power split. In these approaches, the cost of stop-start and gear change events are not considered, which can result in unacceptable switching behaviour, like hunting oscillations. To overcome this problem, costs on switchings are included, solved using Dynamic Programming (DP) and Quadratic Programming (QP) [19] or as one Mixed Integer Linear Program (MILP) [20]. For real-time solving, the allowed model complexity is limiting and adding additional states to these EMSs is computationally prohibitive.

Table 1. Optimizing Energy Management System (EMS) algorithms, for power-split, stop-start and gear. 'Cost' indicates if switching events are penalized in the algorithm. See Table 2 for the used abbreviations.

Reference	Power-Split	Stop-Start	Gear Selection
[15]	PMP	PMP	-
[16]	QP	RB	-
[17]	QP	PMP	RB
[18]	PMP	DP	DP
[19]	QP	DP + cost	DP + cost
[20]	MILP	MILP + cost	MILP + cost
this work	PMP	DP + cost	DP + cost

Table 2. Abbreviations.

Components	
BT	Battery
CE	Combustion Engine
CL	Clutch
EMS	Energy Management System
FT	Fuel Tank
GB	Gearbox
HEV	Hybrid Electric Vehicle
MG	Motor Generator
RD	Road
VH	Vehicle
Algorithms	
DP	Dynamic Program
MILP	Mixed Integer Linear Program
PMP	Pontryagin's Minimum Principle
QP	Quadratic Program
RB	Rule Based (heuristics)

By partitioning a large optimization problem into a set of smaller problems, as in a distributed control system [21], the computational efficiency and robustness are improved, albeit losing the guarantee of global optimality. The process of partitioning is not trivial and many different structures exist. For the power-split problem, two-level structures can be identified in literature [22–24], where planning of the battery energy is separated from the power-split decisions. In this work, a novel EMS is developed as a multi-level system [25,26], in which the higher layers have an increasing level of abstraction of the system, to be able to optimize and coordinate the lower layers [27], while having a large decoupling between the levels. The multi-level EMS solves this control problem with preview, for real-time implementation, using optimal control techniques (PMP and DP), thereby eliminating calibration of parameters. In Figure 1, the proposed control system is illustrated. Each level minimizes

fuel, however, dependent on the function of the level, different model information and corresponding optimal control techniques are used.

- On the first level, the power-split is explicitly solved using the Pontryagin Minimum Principle (PMP), starting from [15]. This method is extended with costs on mode and gear switching, thereby eliminating unacceptable switching behaviour. A Dynamic Programming (DP) routine solves the discrete subproblem. Route and vehicle information determine the switching costs, supporting a full model based approach.
- The second level optimizes the battery state of charge with input- and state-constraints, and provides efficiency information of the hybrid system to the other levels using the battery costate from the PMP solution. Due to the mode switching system on the vehicle, charge sustaining behaviour must be enforced, by using an additional switching algorithm between non-unique solutions.
- The third level provides the necessary route information to the lower layers. By predicting the velocity along the route, using road slope and velocity limitations, the road load on the driveline is determined.
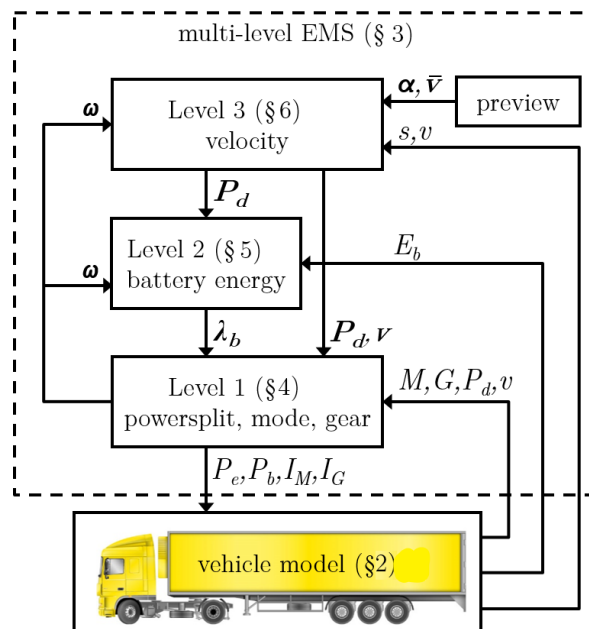

Figure 1. Multi-level optimization of the velocity (v), battery energy (E_b) and power split (P_e, P_b) including mode and gear selection (I_M, I_G). In Section 3 the scheme is described, with the prediction of: road slope (α), velocity limit ($\bar{v}$), driveline rotational speed (ω), driveline power (P_d), velocity (v) and battery costate (λ_b), dependent on: vehicle position (s), velocity (v), battery energy (E_b), driveline power (P_d), mode (M) and gear (G).

In Section 2, the model of the parallel hybrid vehicle is described, after which the multi-level EMS for this vehicle is formulated in Section 3. The algorithms used on each of the three levels are explained in Section 4 (power-split, mode and gear switch optimization), Section 5 (battery energy optimization) and Section 6 (velocity prediction). Offline and online solution schemes of the multi-level optimization are compared in Section 7, where high fidelity simulation results show the fuel benefit of the algorithm. The conclusions are drawn in Section 8. The real-world validation of the multi-level EMS is described in [28].

2. Parallel Hybrid Electric Vehicle Model

For the design of a model-based EMS, the applied models of the hybrid electric vehicle (HEV) are described in this section. The parallel hybrid vehicle under consideration is schematically depicted in Figure 2. The fuel power P_f flows from the tank (FT) to the Combustion Engine (CE), converted to mechanical power P_e, thereby depleting the available fuel energy E_f. Dependent on the position of the clutches CL_1 and CL_2, the power from the Motor/Generator (MG) P_m is added to P_e, resulting in the power at the power-split point P_p. This power is transferred through the gearbox, final drive and wheels (GB), resulting in the driveline power P_d of the vehicle (VH). Dependent on the road load acting on the vehicle (F_0), the travelled distance s and velocity v will change. The MG exchanges electrical power P_b with the battery (BT), thereby (dis-) charging the buffer E_b. Due to the two clutches, both CE and MG can be disconnected and stopped, thereby eliminating their friction losses. In the following sections, the models for this topology are given, with typical model parameters denoted in Table 3.
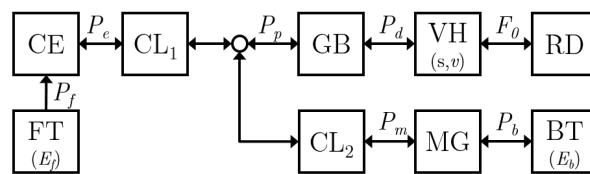

Figure 2. Topology of the parallel Hybrid Electric Vehicle with its relevant power flows, defined positive towards the road and relevant states (between parenthesis).

Table 3. Model parameters for the internal combustion engine (CE), motor generator (MG), battery (BT) and vehicle (VH).

Model	Parameter	Value	Unit	Description
CE	η_e	0.45	-	indicated efficiency
	P_{e0}	$P_{e0}(\omega_e)$	W	friction (cf. Figure 3)
	$\underline{P_e}$	$\underline{P_e}(\omega_e)$	W	minimum power (cf. Figure 3)
	$\overline{P_e}$	$\overline{P_e}(\omega_e)$	W	maximum power (cf. Figure 3)
	I_e	3	kg m^2	moment of inertia
MG	η_m	0.95	-	mechanical efficiency
	P_{m0}	$P_{m0}(\omega_m)$	W	friction (cf. Figure 3)
	$\underline{P_m}$	$\underline{P_m}(\omega_m)$	W	minimum power (cf. Figure 3)
	$\overline{P_m}$	$\overline{P_m}(\omega_m)$	W	maximum power (cf. Figure 3)
	I_m	1.5	kg m^2	moment of inertia
BT	β	1×10^{-6}	1/W	loss coefficient
	$\overline{E_b} - \underline{E_b}$	4×10^6	J	effective battery size
VH	m	20×10^3	kg	vehicle mass
	ρ	1.2	kg/m^3	air density
	c_w	0.65	-	air drag coefficient
	A	10	m^2	frontal area
	c_r	4×10^{-3}	-	rolling resistance coefficient
	r_d	52.2	-	drive ratio
	c_G	1.29	-	gear base constant
	g	9.8	m/s^2	gravitational constant

2.1. Internal Combustion Engine

The internal combustion engine (CE, or 'engine') is modeled as an affine relation between the fuel P_f and the power output P_e, often referred to as a Willans approximation [29]:

$$P_e(t) = P_{e0}(\omega_e(t)) + \eta_e P_f(t), \tag{1}$$

with η_e the indicated efficiency (see Table 3) and $P_{e0}(\omega_e)$ the speed dependent friction losses. It should be noted that $P_f(t) \geq 0$ and $P_{e0}(\omega_e(t)) \leq 0$. The power output is limited by

$$P_e(t) \in [\underline{P_e}(\omega_e), \overline{P_e}(\omega_e)], \tag{2}$$

as shown in Figure 3. The CE has $\underline{P_e}(\omega_e) \leq P_{e0}(\omega_e)$, meaning that additional engine braking can be applied on top of the nominal friction, which is a feature typically available on heavy duty commercial vehicles. For $\underline{P_e} \leq P_e \leq P_{e0}$, no fuel is consumed ($P_f = 0$). The total fuel consumption E_f is the integral of P_f:

$$\dot{E}_f = P_f(t). \tag{3}$$

2.2. Motor Generator

The mechanical output of the motor generator (MG) is modeled as:

$$P_m(t) = \begin{cases} P_{m0}(\omega_m(t)) + \eta_m P_b(t), & \text{if } P_b > 0, \\ P_{m0}(\omega_m(t)) + \frac{1}{\eta_m} P_b(t), & \text{if } P_b \leq 0, \end{cases} \tag{4}$$

with P_m the mechanical output power, P_{m0} the friction, P_b the electrical power from the battery and η_m the constant efficiency of the MG; see Table 3. The power output is limited by

$$P_m(t) \in [\underline{P_m}(\omega_m), \overline{P_m}(\omega_m)] \tag{5}$$

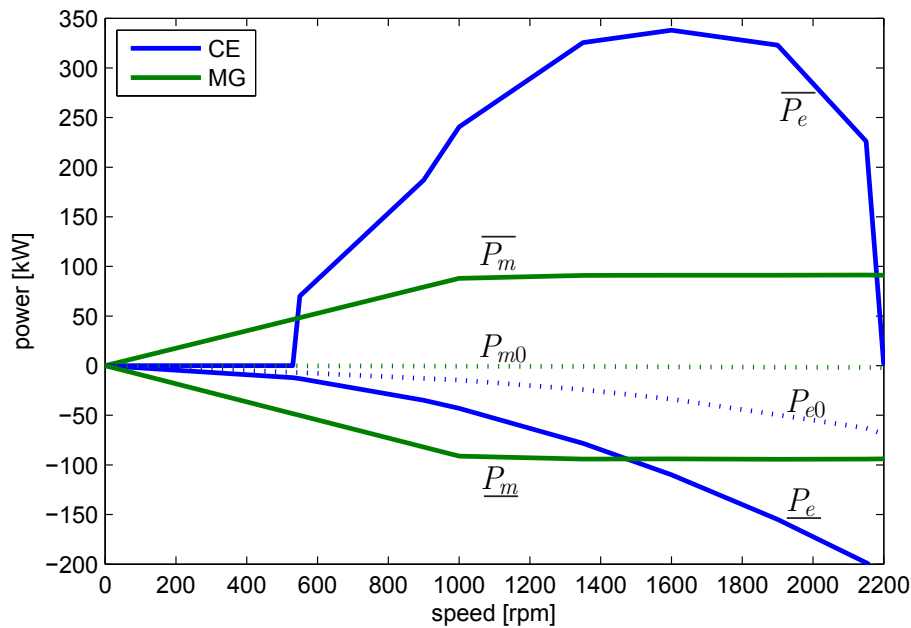and is shown in Figure 3, together with $P_{m0}(\omega_m)$.

Figure 3. Power capabilities of internal combustion engine (CE) and motor generator (MG) as a function of the rotational speed. Dotted lines indicate the nominal friction P_{e0} and P_{m0}.

2.3. Battery

The battery (BT) is modeled as an integrator, with quadratic losses, see [15]:

$$\dot{E}_b = -P_{bi}(t), \tag{6}$$

$$P_{bi}(t) = P_b(t) + \beta \cdot P_b(t)^2, \tag{7}$$

with E_b the energy in the battery, P_{bi} the internal battery power, P_b the power at the terminals and β the loss constant. Note that $P_b > 0$ discharges the battery. The effective size of the battery is limited, such that

$$\underline{E_b} \leq E_b(t) \leq \overline{E_b}. \tag{8}$$

In [8], the validation of the compression-ignition engine (CE), motor generator (MG) and the battery (BT) model is described.

2.4. Mode Selection

The topology has two clutches, CL_1 for connecting the CE to the driveline and CL_2 for connecting the MG to the driveline. When the clutch is open, the respective component is disconnected from the driveline and stopped to eliminate the friction losses in the component. The two clutches create four modes $M \in \{M_c, M_e, M_m, M_o\}$, representing the driveline states, as defined in Table 4. If connected, then the rotational speed of the CE(ω_e), respectively, the MG (ω_m), is equal to the rotational speed ω at the gearbox input shaft, and zero otherwise. The gearbox input power P_p is the sum of the connected components.

Table 4. Driveline modes M.

M	Description	CL_1	CL_2	ω	$P_p(M)$
M_c	combined	closed	closed	$\omega_e{=}\omega_m{=}\omega$	$P_e + P_m$
M_e	engine-only	closed	open	$\omega_e{=}\omega, \omega_m{=}0$	P_e
M_m	motor-only	open	closed	$\omega_m{=}\omega, \omega_e{=}0$	P_m
M_o	open driveline	open	open	$\omega_e{=}\omega_m{=}0$	0

The mode is controlled with $I_M \in \{i_c, i_e, i_m, i_o\}$ for the respective four modes. Mode switch dynamics are defined by the state machine in Figure 4. When a mode switch (ΔM) is performed, the driveline is open (M_o) for a duration of Δt_M, during which no traction is available ($P_p(M_o) = 0$). During the mode switch, a series of events, involving (de-)coupling and synchronization of rotating masses, cause energy losses, represented by the lumped parameter ϵ_M:

$$\epsilon_M = \begin{cases} 0, & \text{if } \Delta M = 0, \\ > 0, & \text{if } \Delta M \sim 0. \end{cases} \tag{9}$$

In Appendix C, the dependency of ϵ_M on the driveline state and the drive cycle is described.
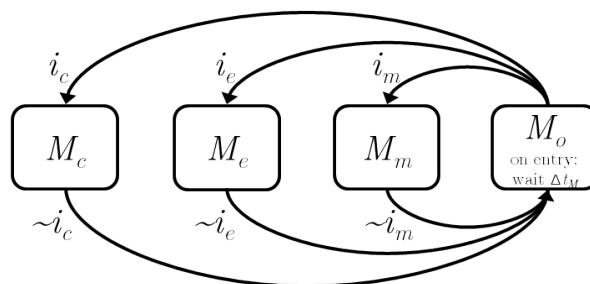

Figure 4. State machine for switching between modes M. All mode switches go through M_o ('open driveline'), which is maintained for Δt_M. The negation of the mode command is denoted with $\sim i$.

2.5. Gear Selection

The gearbox with final drive and wheels (GB) is modeled as a geometrically stepped transmission without losses. The ratio of the gearbox r_G is a function of the gear position $G \in \{1, 2, ..., 12\}$ with the gear base constant c_G

$$r_G(G(t)) = c_G{}^{(12-G(t))}. \tag{10}$$

The gearbox input shaft speed ω [rpm] is related to the vehicle speed v [m/s], through the gearbox ratio, final drive and wheels, with

$$\omega(G(t), v(t)) = r_G(G(t)) \cdot r_d \cdot v(t), \tag{11}$$

where r_d is the drive ratio from speed [m/s] to gearbox out [rpm]. The gear selection is controlled with $I_G \in \{i_1, i_2, ..., i_{12}\}$. Gear switch dynamics are defined by the state machine in Figure 5. When a gear switch (ΔG) is performed, the driveline is open (G_o) for a duration of Δt_G, during which no traction is available:

$$P_d(G) = \begin{cases} 0, & \text{if } G_0, \\ P_p, & \text{if } \sim G_0. \end{cases} \tag{12}$$

During the gear switch, a series of events, involving (de-)coupling and synchronization of rotating masses, cause energy losses, represented by the lumped parameter ϵ_G:

$$\epsilon_G = \begin{cases} 0, & \text{if } \Delta G = 0, \\ > 0, & \text{if } \Delta G \sim 0. \end{cases} \tag{13}$$

In Appendix C, the dependency of ϵ_G on the driveline state and the drive cycle is described.

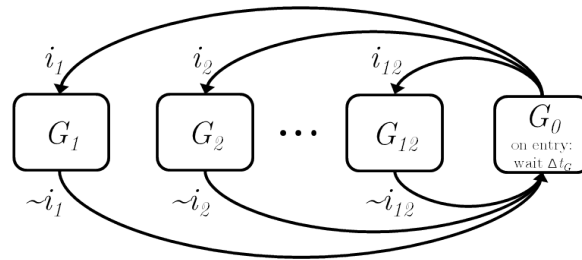**Figure 5.** State machine for switching between gears G. All gear switches go through G_0 ('open driveline'), which is maintained for Δt_G. The negation of the gear command is denoted with $\sim i$.

2.6. Vehicle

The dynamics of the vehicle (VH) are modeled for the longitudinal motion:

$$\frac{d^2 s}{dt^2}(t) = \frac{F_0(s, v(t)) + F_d(t)}{m}, \tag{14}$$

with vehicle mass m, vehicle position s, $v(t) = \frac{ds}{dt}$ the vehicle speed, and the total vehicle road load F_0:

$$F_0(s, v(t)) = F_a(v(t)) + F_r(s) + F_g(s), \tag{15}$$

with:

$$F_a(v(t)) = \frac{1}{2} c_w \cdot A \cdot \rho \cdot v(t)^2, \tag{16}$$

$$F_r(s) = m \cdot g \cdot c_r \cdot \cos \alpha(s), \tag{17}$$

$$F_g(s) = m \cdot g \cdot \sin \alpha(s), \tag{18}$$

respectively, the air drag, rolling resistance and gravitational force, and F_d the driveline force:

$$F_d(t) = \frac{P_d(t)}{v(t)}, \tag{19}$$

where α is the road inclination, and the other parameters as defined in Table 3. The velocity of the vehicle is limited, such that

$$0 \leq v(s) \leq \overline{v}(s), \tag{20}$$

with $\overline{v}$ the maximum speed limit.

3. Multi-Level Energy Management

Optimizing the energy consumption for a system, with multiple states and a mix of continuous and discrete decision variables, is computationally demanding for real-time implementation. Partitioning the optimization into smaller problems reduces the computational burden. Not only are the sub-problems smaller and thereby easier to solve, but also each partition can have its own optimization algorithm, making the selection of more efficient algorithms possible, suited to the specific problem of that partition. e.g., in [19] Dynamic Programming is used for the partition with discrete decision variables, while convex optimization is used for the partition with continuous states. The method of partitioning is, however, not unique nor trivial [27].

For the generic energy management problem, described in Section 3.1, a functional hierarchy is introduced in Section 3.2, inspired by the 'multi-level structure using conjugate variables' in [27]. In this section, the partitioning in levels, and the solution of the multi-level EMS, for two solution schemes, are described:

- the iterative scheme (Section 3.3), used in simulation to show convergence of the EMS in Section 7.1,
- the model predictive scheme (Section 3.4), that is real-time implementable and simulated in Section 7.2.

The control system for each individual level is described in Sections 4–6.

3.1. Generic Energy Management Problem

The general task of the Energy Management System (EMS) is to minimize the fuel energy E_f needed to move the vehicle from distance s_0 to s_f. For the vehicle model in Section 2, this EMS can be posed as a nonlinear, mixed-integer, input- and state-constrained dynamical optimization problem:

$$\min J = \min_u \int_{t_0}^{t_f} \left(P_f(u, x, w, t) + P_\epsilon \right) \mathrm{d}t,$$

$$s.t. \tag{21}$$

$$h_e(u, x) = 0,$$

$$h_n(u, x) \leq 0,$$

with continuous decision variables P_d (driveline power), P_b (battery power) and discrete decision variables $I_M \in \{i_c, i_e, i_m, i_o\}$ (mode request), $I_G \in \{i_1, i_2, ..., i_{12}\}$ (gear request):

$$u = \begin{bmatrix} P_d \\ P_b \\ I_M \\ I_G \end{bmatrix}, \tag{22}$$

continuous states s (distance), v (velocity), E_b (battery energy) and discrete states $M \in \{M_c, M_e, M_m, M_o\}$ (mode), $G \in \{1, 2, ..., 12\}$ (gear):

$$x = \begin{bmatrix} s \\ v \\ E_b \\ M \\ G \end{bmatrix}, \tag{23}$$

disturbances, representing the preview information, α (road slope), $\overline{v}$ (maximum velocity):

$$w = \begin{bmatrix} \alpha(s) \\ \overline{v}(s) \end{bmatrix}, \tag{24}$$

switching cost ϵ_M (for mode switch), ϵ_G (for gear switch):

$$P_\epsilon = \begin{bmatrix} \epsilon_M \\ \epsilon_G \end{bmatrix}, \tag{25}$$

equality constraints on s_0 (initial distance), s_f (final distance), E_b (battery energy) sustenance:

$$h_e = \begin{bmatrix} s(t_0) - s_0 \\ s(t_f) - s_f \\ E_b(t_0) - E_b(t_f) \end{bmatrix} = \mathbf{0}, \tag{26}$$

inequality constraints on v, E_b upper and lower limits, P_e (CE power) upper and lower limits, P_m (MG power) upper and lower limits:

$$h_n = \begin{bmatrix} v(t) - \overline{v}(s) \\ E_b(t) - \overline{E_b} \\ -E_b(t) + \underline{E_b} \\ P_e(t) - \overline{P_e}(\omega) \\ -P_e(t) + \underline{P_e}(\omega) \\ P_m(t) - \overline{P_m}(\omega) \\ -P_m(t) + \underline{P_m}(\omega) \end{bmatrix} \leq \mathbf{0}, \tag{27}$$

together with the model equations in Section 2.

3.2. Multi-Level Optimization

In the multi-level EMS, the global optimization problem is subdivided into three levels based on its *function*: velocity determination, optimizing battery energy and optimizing the power-split including switching of modes and gears. Each level (indicated with subscript $_i$) has the same objective, i.e., minimizing (equivalent) fuel:

$$\begin{aligned} \min J_i = \min_{u_i} & \int_{t_0}^{t_{fi}} (P_f(u_i, x_i, w_i, t) + P_{\epsilon i}) \, \mathrm{d}t, \\ s.t. & \\ & h_{ei} = 0, \\ & h_{ni} \leq 0, \end{aligned} \tag{28}$$

but with a subset of the decision variables, and on each level a different model complexity, belonging to the abstraction and dynamics on that level. In Table 5, the sub-problems of the optimization are defined, including the output y of each level. Numbering of the levels start at 1 for the lowest level, as control of the component itself (e.g., for safe operation) is indicated with Level 0, and is not contained in the EMS.

Information exchange between the three levels is shown in Figure 1 and the function and interfacing of each level is briefly explained next.

Table 5. Generic optimization problem, subdivided into three levels (i).

	'Generic'	'Velocity'	'Battery Energy'	'Power-Split'
i		3	2	1
u_i	$\begin{bmatrix} P_d \\ P_b \\ I_M \\ I_G \end{bmatrix}$	$[P_d]$	$\begin{bmatrix} P_b \\ I_M \end{bmatrix}$	$\begin{bmatrix} P_b \\ I_M \\ I_G \end{bmatrix}$
x_i	$\begin{bmatrix} s \\ v \\ E_b \\ M \\ G \end{bmatrix}$	$\begin{bmatrix} s \\ v \end{bmatrix}$	$[E_b]$	$\begin{bmatrix} M \\ G \end{bmatrix}$
w_i	$\begin{bmatrix} \boldsymbol{\alpha} \\ \overline{v} \end{bmatrix}$	$\begin{bmatrix} \boldsymbol{\alpha} \\ \overline{v} \\ \omega \end{bmatrix}$	$\begin{bmatrix} \boldsymbol{P_d} \\ \omega \end{bmatrix}$	$\begin{bmatrix} \boldsymbol{P_d} \\ \boldsymbol{\lambda_b} \\ v \end{bmatrix}$
$P_{\epsilon i}$	$\begin{bmatrix} \epsilon_M \\ \epsilon_G \end{bmatrix}$	0	0	$\begin{bmatrix} \epsilon_M \\ \epsilon_G \end{bmatrix}$
h_{ei}	$\begin{bmatrix} s(t_0) - s_0 \\ s(t_f) - s_f \\ E_b(t_f) - E_b(t_0) \end{bmatrix}$	$\begin{bmatrix} s(t_0) - s_0 \\ s(t_{f3}) - s_f \end{bmatrix}$	$[E_b(t_{f2}) - E_b(t_0)]$	0
h_{ni}	$\begin{bmatrix} v - \overline{v} \\ E_b - \overline{E_b} \\ \underline{E_b} - E_b \\ P_e - \overline{P_e} \\ \underline{P_e} - P_e \\ P_m - \overline{P_m} \\ \underline{P_m} - P_m \end{bmatrix}$	$\begin{bmatrix} v - \overline{v} \\ P_f - \overline{P_f} \\ \underline{P_f} - P_f \end{bmatrix}$	$\begin{bmatrix} E_b - \overline{E_b} \\ \underline{E_b} - E_b \\ P_e - \overline{P_e} \\ \underline{P_e} - P_e \\ P_m - \overline{P_m} \\ \underline{P_m} - P_m \end{bmatrix}$	$\begin{bmatrix} P_e - \overline{P_e} \\ \underline{P_e} - P_e \\ P_m - \overline{P_m} \\ \underline{P_m} - P_m \end{bmatrix}$
t_{fi}	$t_f = t(s_f)$	$t_{f3} \leq t(s_f)$	$t_{f2} \leq t_{f3}$	$t_{f1} \leq t_{f2}$
y_i	$\begin{bmatrix} P_d \\ P_b \\ I_M \\ I_G \end{bmatrix}$	$\begin{bmatrix} \boldsymbol{P_d} \\ v \end{bmatrix}$	$[\boldsymbol{\lambda_b}]$	$\begin{bmatrix} P_e \\ P_b \\ I_M \\ I_G \\ \omega \end{bmatrix}$

3.2.1. Level 1, 'Power-Split Including Switching' (Section 4)

Level 1 is the lowest control level in the EMS, responsible for the optimal power split (defined by P_b and given P_d), mode (I_M) and gear (I_G) selection. Based on information from the higher levels, it has to act fast, in order to have a responsive vehicle. However, the required fast update rate (typically 10–100 Hz) limits the computational time available for calculations. This is solved by moving computational expensive calculations to higher layers, and having explicit solutions for the remaining optimizations, using optimal control techniques (DP and PMP, see Appendix A). Information from the higher levels is provided by vectors over time (indicated in bold) with estimated quantities:

- $\boldsymbol{\lambda_b}$, estimated equivalent cost of battery energy,
- $\boldsymbol{P_d}$, estimated power demand,
- v, estimated vehicle speed,

and, after optimizing the decision variables, the calculated driveline speed ω is provided to Levels 2 and 3.

3.2.2. Level 2, 'Battery Energy' (Section 5)

Level 2 is responsible for optimizing the battery energy over the cycle, considering the battery constraints. As the battery energy dynamics are slower than the decisions needed in Level 1, its optimizations can run at a lower rate (typically 1 Hz), facilitating the more complex calculations, caused by battery limits and longer horizons. Abstraction of the Level 1 model, by e.g., not considering gear switching and no penalties on switching, combined with PMP techniques, make the optimization computationally efficient. Based on an estimated driveline speed ω, the estimated power demand P_d and the current battery energy E_b, the optimal λ_b is calculated, which is input for Level 1.

3.2.3. Level 3, 'Velocity' (Section 6)

Level 3 determines the velocity v. Of the three levels, it uses the most abstracted model of the hybrid driveline, by e.g., not considering battery dynamics, hybrid modes, power-split or switching. Using-distance based route information for preview, i.e., slope $\alpha(s)$ and maximum speed limits $\bar{v}(s)$, and current vehicle velocity v and position s, an estimate speed v is determined, resulting in an estimated power demand P_d.

3.3. Multi-Level Iteration

The multilevel iteration scheme is used for offline simulation. In this multilevel, nested, optimization, the decisions on one level impact the objective on other levels and thereby influencing optimality [21,30]. This dependency can be seen in Figure 1, where the variables and ω, λ_b are fed back to the higher level controllers. Solving the optimization by starting at the highest level and sequentially transmitting information to the lower levels, we need an a priori estimate of ω and λ_b. When all levels are calculated, the sequence can be repeated, where ω and λ_b are updated from the last sequence, thereby providing the higher levels with the latest decision details of the lower levels. The solution scheme of the optimization results in:

1. Initialization:

    (a)   all levels optimize over the complete drive cycle, from $t(s_0)$ to final time $t_{f3} = t_{f2} = t_{f1} = t(s_f)$,
    (b)   initialize ω with an estimated average speed,

2. Velocity prediction:

    (a)   retrieve $\alpha(s)$, $\bar{v}(s)$ (road slope and speed limit information), $s(t_0)$, $v(t_0)$ (position and speed) and model parameters (Table 3),
    (b)   use the vehicle model and results from optimal control, to predict the velocity profile,
    (c)   store resulting P_d, v.

3. Battery energy optimization:

    (a)   retrieve P_d, ω, $E_b(t_0)$ and model parameters (Table 3),
    (b)   calculate PMP necessary conditions and optimize the remaining boundary value problem(s),
    (c)   store resulting λ_b.

4. Power-split and switch optimization:

    (a)   retrieve λ_b, P_d, v and model parameters (Table 3),
    (b)   calculate PMP necessary conditions and solve the remaining DP problem,
    (c)   store resulting ω and output P_d, P_b, I_M, I_G to the component controllers on the vehicle.

5. Iterate:

    (a)   repeat from step 2 with an improved estimate of ω and λ_b, unless stop conditions are met (number of iterations). Note that the initial conditions at t_0 remain identical.

It is expected that each iteration will increase fuel efficiency, until the solution is converged. The effect of the number of iterations on the fuel efficiency, is analyzed by simulation in Section 7.

3.4. Multi-Level Model Predictive Control

To apply the EMS in real time, the multilevel optimization is implemented as a model predictive (or receding horizon) controller (MPC), which performs the optimization in a regular schedule ('sampling'). Each sampling period the optimization is performed with updated states, over a subset of the cycle ('horizon'), thereby creating feedback for disturbance rejection and model mismatch compensation. In Figure 6, the differences with the multi-level iteration, in horizon and iteration, are illustrated. Compared to the solution scheme in Section 3.3, the following adjustments are made:

- t_0 now refers to the vehicle's current time and t_f is relative to t_0, with a fixed horizon length (which implements the receding horizon),
- each sample time, only one iteration is applied, i.e., step 5 in the iteration scheme is skipped,
- the horizon length decreases on each level: $t(s_f) > t_{f3} > t_{f2} > t_{f1}$,
- each level runs in its own regular schedule (sample time), where the higher levels run slower than the lower levels.

All adjustments are for improving computational efficiency; however, optimality will reduce. In Section 7, we will show that, in simulation, the optimality is marginally decreased and the multi-level approach shows good fuel economy in a high fidelity simulation environment. In Sections 4–6, the algorithms are described in detail for Levels 1, 2 and 3, respectively.
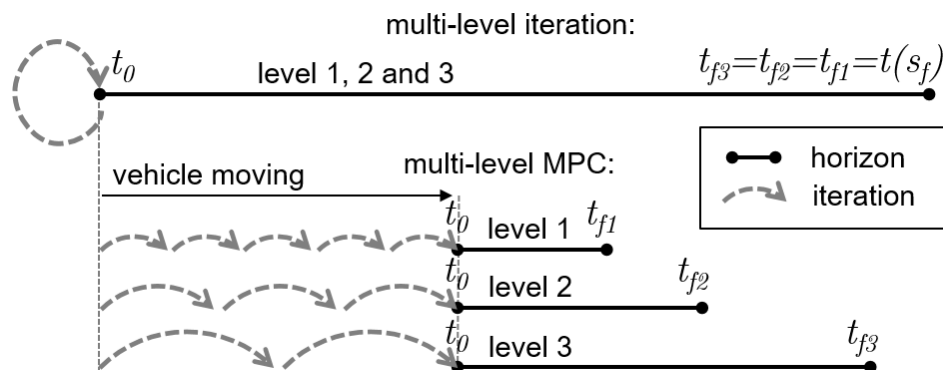

Figure 6. Schematic of the horizon in the 'multi-level iteration' and 'multi-level Model Predictive Controller (MPC)' solution schemes. The multi-level iteration has an identical horizon, for all levels, and for all iterations, comprising the complete drive cycle. The multi-level MPC has a solution scheme, with shorter horizon lengths. With each time sample, the vehicle moves along the drive cycle. At each time sample, one iteration is performed, over a horizon starting at t_0, having a fixed horizon length to t_f. When the level has a faster sample time, iterations are performed more often, thereby improving the feedback performance.

4. Level 1: Power-Split and Switch Optimization

The Level 1 optimization calculates the optimal power-split, mode and gear selection for a simplified plant model, as a subset from optimization (21). In [8,15], it is shown that the power-split problem with battery dynamics can be efficiently solved using PMP, resulting in two solutions steps, see Appendix A:

- minimization of the Hamiltonian as a function of λ_b, resulting in the optimal power-split, mode and gear,
- calculation of λ_b that complies to the battery constraints.

Level 1 performs the first step: minimization of the Hamiltonian. The second step, calculation of λ_b, is performed at Level 2, as described in Section 5.

On Level 1, we assume a predetermined velocity profile $v(t) = v$, which defines P_d using $\alpha(s)$ in Equations (14)–(19). As λ_b is controlling E_b (see Section 5), all continuous states (E_b, s, v) are removed from Equation (23), together with the equality and inequality constraints on the respective states in Equation (26) and Equation (27). As a result, the minimization Equation (28) for Level 1, denoted with subscript $_1$, is

$$\min J_1 = \min_{u_1} \int_{t_0}^{t_{f1}} \left(P_f(u_1, x_1, w_1, t) + P_{\epsilon 1} \right) dt,$$

$$s.t. \tag{29}$$

$$h_{e1} = 0,$$

$$h_{n1} \leq 0,$$

with

$$u_1 = \begin{bmatrix} P_b \\ I_M \\ I_G \end{bmatrix}, \tag{30}$$

$$x_1 = \begin{bmatrix} M \\ G \end{bmatrix}, \tag{31}$$

$$w_1 = \begin{bmatrix} P_d \\ \lambda_b \\ v \end{bmatrix}, \tag{32}$$

$$h_{e1} = 0, \tag{33}$$

$$h_{n1} = \begin{bmatrix} P_e(t) - \overline{P_e}(\omega) \\ -P_e(t) + \underline{P_e}(\omega) \\ P_m(t) - \overline{P_m}(\omega) \\ -P_m(t) + \underline{P_m}(\omega) \end{bmatrix} \leq 0, \tag{34}$$

with the switching cost $P_{\epsilon 1} = 0$ in Sections 4.1 and 4.2 and with $P_{\epsilon 1} \geq 0$ in Section 4.3.

4.1. Explicit Minimization of the Hamiltonian per Mode

On Level 1, the Hamiltonian is minimized. For the hybrid drive train, this Hamiltonian is solved *explicitly* in [8,15], i.e., an analytical expression is found for the minimization. This section extends that solution for the driveline topology with an additional clutch between MG and GB. The Hamiltonian of Level 1 to minimize is

$$H_1 = P_f(P_b, I_M, I_G, P_d, v) - \lambda_b \cdot P_{bi}(P_b, I_M, I_G, P_d, v), \tag{35}$$

using the PMP conditions (Appendix A). When we assume instantaneous switching, i.e., $\Delta t_M = 0, \Delta t_G = 0$, the control signals I_M and I_G bring the system immediately to its corresponding state, hybrid mode M, respectively, gear G, such that $I_M \equiv M, I_G \equiv G$. Then, for each M, G, the minimum of H_1 is found by solving $\frac{dH_1}{dP_b} = 0$. For each M, G, the result is given in Table 6, where only parameter $P_{m0}(\omega(G, v))$ depends on the rotational speed ω, as defined by G and v through Equation (11).

Table 6. Explicit solution of the power-split as a function of the mode.

Mode	Description	P_b^*
M_m	motor-only	$(P_d - P_{m0})/\eta_m$
M_e	engine-only	0
M_c	combined	$\begin{cases} -\dfrac{(\lambda_b + \frac{\eta_m}{\eta})}{2\beta\lambda_b} & \text{if } P_b > 0 \\[3mm] -\dfrac{(\lambda_b + \frac{1}{\eta\eta_m})}{2\beta\lambda_b} & \text{if } P_b \leq 0 \end{cases}$

With the explicit expression for P_b^* (where the superscript * denotes the optimal solution), the reduced Hamiltonian H_{1a} (with subscript $_a$ denoting the variant) is an explicit expression, for each mode and gear combination. Finding the optimal H_1 is thereby reduced to:

$$H_1^* = H_{1a}^* = \min H_{1a}(M, G | \lambda_b, P_d, v). \tag{36}$$

In Figure 7, H_1^* is illustrated. As a function of λ_b, H_1^* changes modes at B_1, B_2 and B_3. Due to the mode changes, the control signal P_b^* for M^* jumps as a function of λ_b at B_1, B_2 and B_3, as shown in Figure 8. That jumping behaviour, caused by the additional clutch between MG and GB, is new to [8,15] and adds complications to the controllability of E_b, as will be shown in Section 5.2.
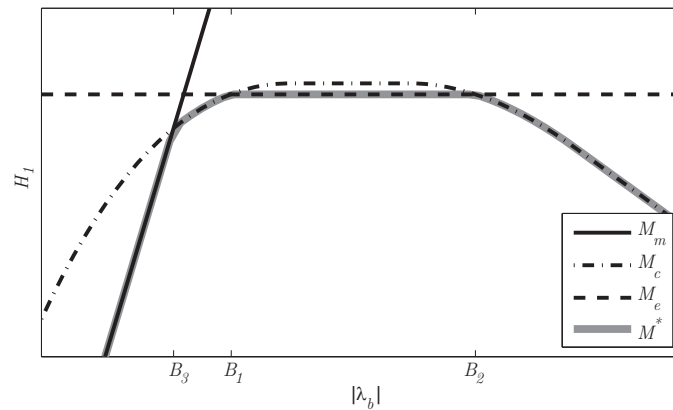
Figure 7. The Hamiltonian H_1^* of the three modes M_m, M_c, M_e as a function of λ_b for a power demand and rotational speed. The optimal mode M^* changes at the boundary points B_1, B_2, B_3.
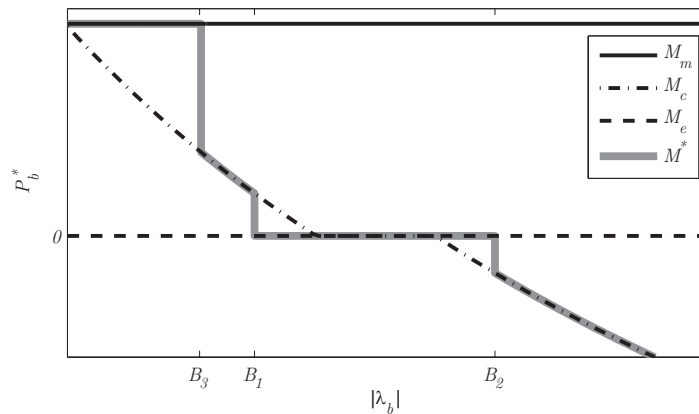
Figure 8. P_b^* as a function of λ_b for the modes M_m, M_c, M_e for a power demand and gear. At B_1 and B_2, the optimal mode M^* switches between M_c and M_e, with a corresponding jump in P_b^*. At B_3 a jump occurs due to switching between M_c and M_m.

4.2. Mode Selection, without Cost on Switching

Instead of calculating all $H_{1a}(M, G)$ and selecting the minimal one in Equation (36), a computationally more efficient approach can be taken where the optimal mode is calculated beforehand. Figure 7 shows that the optimal mode is changed at B_1, B_2 and B_3 where the Hamiltonians of two modes are equal. Equating the Hamiltonians, using Equations (1), (4), (7), (36) and Table 6, results in five expressions, representing all the switching lines (guards B) where the optimal mode M^* is changing. Two guards are a function of $\lambda_b \forall P_d$:

$$B_1 : \lambda_b = -\frac{\eta_m}{\eta} + \frac{2\beta P_{m0}}{\eta} + \frac{2\sqrt{\beta P_{m0}(\beta P_{m0} - \eta_m)}}{\eta}, \tag{37}$$

$$B_2 : \lambda_b = -\frac{1}{\eta \eta_m} + \frac{2\beta P_{m0}}{\eta} - \frac{2\sqrt{\beta P_{m0}(\beta P_{m0}\eta_m - 1)}}{\eta \eta_m}, \tag{38}$$

and three guards are a function of (P_d, λ_b):

$$B_3 : P_d = P_{m0} - \frac{\eta_m + \eta\lambda_b}{2\beta\lambda_b \frac{\eta}{\eta_m}} - \frac{\sqrt{4\beta\lambda_b\eta P_0}}{2\beta\lambda_b \frac{\eta}{\eta_m}}, \tag{39}$$

$$B_4 : P_d = P_{m0} - \frac{\eta_m + \eta\lambda_b}{2\beta\lambda_b \frac{\eta}{\eta_m}} - \frac{\sqrt{4\beta\lambda_b\eta(P_0 - P_{m0}) + (\lambda_1\eta + \eta_m)^2}}{2\beta\lambda_b \frac{\eta}{\eta_m}}, \tag{40}$$

$$B_5 : P_d = P_{m0} - \frac{\eta_m + \eta\lambda_b}{2\beta\lambda_b \frac{\eta}{\eta_m}} - \frac{\sqrt{4\beta\lambda_b\eta P_0 + 2\lambda_b\eta(\eta_m - \frac{1}{\eta_m}) + (\eta_m^2 - \frac{1}{\eta_m^2})}}{2\beta\lambda_b \frac{\eta}{\eta_m}}. \tag{41}$$

All optimal modes M^* are explicitly defined with Equations (37)–(41), as a function of (P_d, λ_b) and the model parameters (Table 3). In Figure 9, the guards and optimal modes are illustrated.
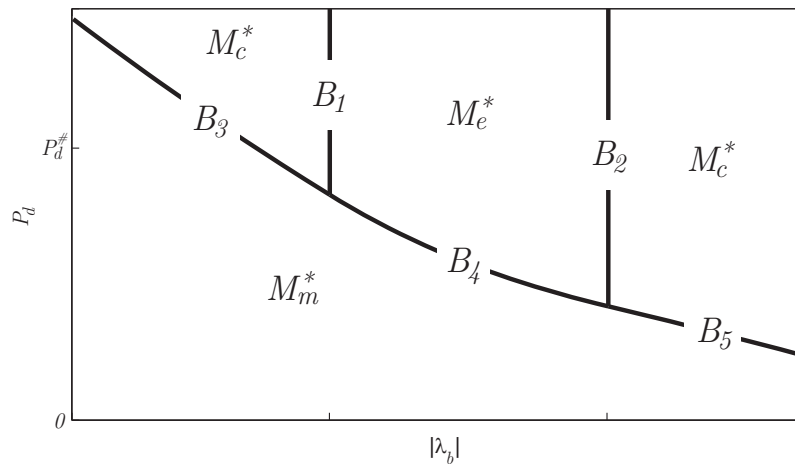

Figure 9. Explicit mode change map as a function of costate λ_b and power demand P_d, indicating the guards B between the optimal modes M_m^* (motor-only), M_e^* (engine-only) and M_c^* (combined). Figures 7 and 8 are cross sections at $P_d^\#$.

A further reduced Hamiltonian $H_{1b}(\lambda_b, P_d, v)$ is now obtained in three steps:

1. for each (feasible) gear, select the optimal mode $M^*(G)$, using Equations (37)–(41),
2. for each (feasible) gear, calculate $H_{1a}(M^*(G), G)$,
3. select G^* that minimizes $H_{1a}(G)$,

thereby reducing Equation (36) to:

$$H_1^* = H_{1b}^* = \min H_{1b}(\lambda_b, P_d, v), \tag{42}$$

which statically defines the decision variables P_b, I_M, I_G as a function of λ_b, P_d, v.

Note that power constraints of the components in Equation (34) are easily added by including the corresponding guards, thereby refining the area of M^*, see [15]. The cost function in Equation (29) is now reduced to

$$\min J_{1b} = \int_{t_0}^{t_{f1}} H_{1b}^*(t|\lambda_b(t), P_d(t), v(t)) \, \mathrm{d}t, \tag{43}$$

which is the integral of all locally minimized Hamiltonians.

4.3. Mode Selection, with Cost on Switching

When the system operates near the guards, frequent switching (so called 'hunting') between modes and gears can occur under influence of disturbances, thus preventing acceptable real-life implementation. Each switch involves connecting and/or disconnecting of components, synchronization of their speeds and additional friction losses, which cause driveability, durability and efficiency issues. As a solution to hunting, the cost function in Equation 21 adds a penalty $P_\epsilon = \epsilon_M + \epsilon_G \geq 0$ on mode switching ΔM and gear switching ΔG.

The parameters ϵ_M, ϵ_G can be tuned to balance hunting behavior, with fuel economy. To prevent tuning of the parameters, Appendix C quantifies the model-based, equivalent fuel losses during a switch, identified by:

- synchronization losses, caused by acceleration and deceleration of rotating masses (Appendix C.1),
- traction interruption, caused by disconnection of the driveline and leading to vehicle speed deviations (Appendix C.2). As the speed deviation during disconnection of the driveline, depends on the road load (e.g., up-hill resulting in a large speed decrease, on flat road resulting in a small speed decrease), the corresponding equivalent fuel costs are a function of the speed change and time loss.

The calculated penalties are then a function of mode change, gear change and vehicle speed. Adding the cost of switching, changes Equation (29) to the new Level 1 cost function

$$\min J_{1c} = \min_{M,G} \int_{t_0}^{t_{f1}} H_{1a}^*(M, G, t|\lambda_b(t), P_d(t), v(t)) + \epsilon_M(\Delta M, t|v(t)) + \epsilon_G(\Delta G, t|v(t)) \, \mathrm{d}t, \tag{44}$$

which is the integral of the explicit Hamiltonian solution in Equation (36), with switching costs ϵ_M and ϵ_G added to the integral, when a switch occurs. By discretizing time,

$$\min J_{1d} = \min_{M,G} \sum_{t_0}^{t_{f1}} H_{1a}^*(M, G, k|\lambda_b(k), P_d(k), v(k)) + \epsilon_M(\Delta M, k|v(k)) + \epsilon_G(\Delta G, k|v(k)), \tag{45}$$

with the discrete time vector $k \in \{0 \cdot T_k, 1 \cdot T_k, 2 \cdot T_k, ..., t_{f1}\}$ and sampling time T_k, the optimization is solved using Dynamic Programming. The outline of the algorithm is as follows, as illustrated in Figure 10:

1. all feasible modes M and gears G over the time horizon t_{f1} are enumerated,
2. for each combination of mode and gear and time, the Hamiltonian is explicitly solved using Equation (36), resulting in $H(M, G, k)$,
3. a Dynamic Program (DP) is formulated with $H(M, G, k)$ as elements for the cost-to-go matrix (M, G, k),
4. to implement the cost for switching, for each mode change and gear change, the cost-to-go matrix is penalized with ϵ_M and ϵ_G,

5. the optimal sequence of modes and gears is calculated using DP,
6. for the MPC implementation, only the first control action is implemented. The next sample time, the algorithm is repeated with updated inputs, disturbances and states.

For the resulting M and G, P_b^* is selected using Equation (36) and with P_d given, P_e and P_b are known with Equations (4), (12) and Table 4. With the selection of T_k such that $\Delta t_M \approx \Delta t_G \approx T_k$, we have $I_M(k) = M(k+1)$ and $I_G(k) = G(k+1)$, which completes the calculation of the output of Level 1.
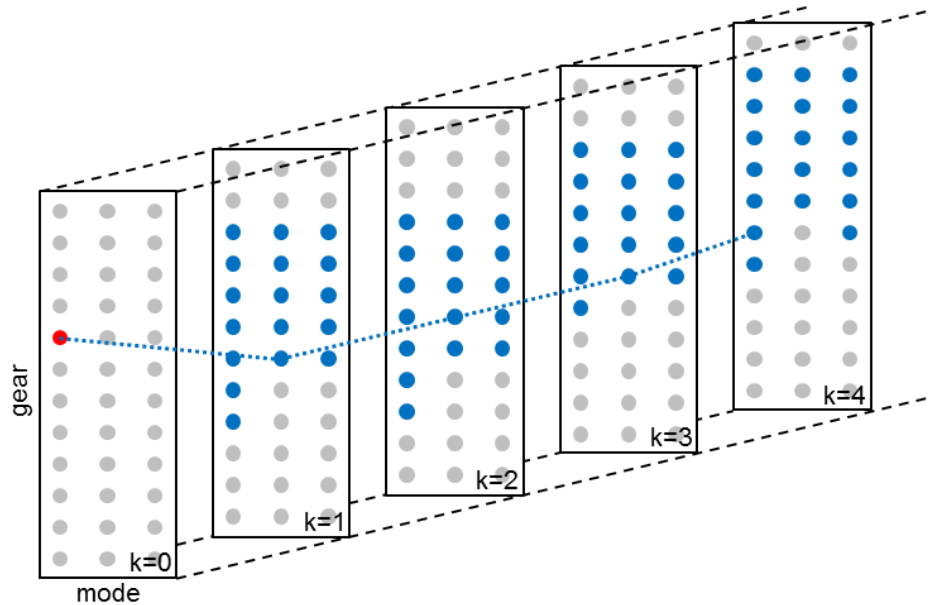
Figure 10. Illustrating the Receding Horizon Integer Program to solve at each time step. Over a horizon of four seconds with a sampling time of one second, all feasible modes and gears are enumerated (blue), starting at the current mode and gear (red). For all feasible modes, H is calculated, increased with an equivalent fuel penalty P_e on each mode or gear change, and the optimal trajectory over the horizon (blue dotted line) is determined, using Dynamic Programming (DP).

5. Level 2: Battery Energy Optimization

The Level 2 optimization determines λ_b, which is an input for Level 1 control. That λ_b is found by solving the power-split problem, taking the battery dynamics E_b in Equation (23), with corresponding limits ($\overline{E_b}$, $\underline{E_b}$, in Equation (27)), into account.

On Level 2, we assume:

- a predetermined velocity profile is available, which defines P_d using Equations (14)–(19).
- a prescribed rotational speed ω, which defines $G \in [1, 12]$ with the predetermined velocity profile. Before iteration over the levels an estimated average ω is selected. After each iteration, ω from Level 1 is used.
- mode and gear switching is instantaneous ($\Delta t_M = 0, \Delta t_G = 0$), without costs associated ($P_e = 0$).

As a result, minimization of Equation (28) for Level 2 is

$$\min J_2 = \min_{u_2} \int_{t_0}^{t_{f2}} \left(P_f(u_2, x_2, w_2, t) \, \mathrm{d}t, \right.$$

$$s.t.$$

$$h_{e2} = 0,$$

$$h_{n2} \leq 0,$$

(46)

with

$$u_2 = \begin{bmatrix} P_b \\ I_M \end{bmatrix}. \tag{47}$$

$$x_2 = \begin{bmatrix} E_b \end{bmatrix}, \tag{48}$$

$$w_2 = \begin{bmatrix} P_d \\ \omega \end{bmatrix}, \tag{49}$$

$$h_{e2} = E_b(t_{f2}) - E_b(t_0) = 0, \tag{50}$$

$$h_{n2} = \begin{bmatrix} E_b - \overline{E_b} \\ -E_b + \underline{E_b} \\ P_e(t) - \overline{P_e}(\omega) \\ -P_e(t) + \underline{P_e}(\omega) \\ P_m(t) - \overline{P_m}(\omega) \\ -P_m(t) + \underline{P_m}(\omega) \end{bmatrix} \leq 0. \tag{51}$$

Following [31], this dynamic state constrained problem is solved using PMP (Appendix A). Using Equations (6) and (7), the system dynamics are:

$$\dot{x} = \dot{E}_b = -P_{bi}(P_b) \tag{52}$$

and with Equation (3) result in the Hamiltonian:

$$H_2 = P_f(P_b, I_M, P_d, \omega) - \lambda_b \cdot P_{bi}(P_b, I_M, P_d, \omega), \tag{53}$$

where λ_b is the costate of E_b. As no switching costs are defined, H_2 is explicitly minimized with Equation (42) as a function of λ_b, among others. The dynamics of λ_b in Equation (A6),

$$\dot{\lambda}_b = -\frac{\partial H}{\partial E_b} = \frac{\partial P_{bi}}{\partial E_b} = 0, \tag{54}$$

show that λ_b is constant for the problem without constraint in Equation (8), or changes stepwise where E_b is constrained [8,32]. Solving the problem is reduced to a boundary value problem [1], with λ_b the decision variable. The boundary value problem, for the topology with clutches, state constraints and a limited horizon, has the following properties, which are handled by the developed algorithms in this section, with the flowchart in Figure 11:

I1 Constraint activation increases the number of boundary value problems to solve, where the number of sub-problems is not known a priori. The iterative solution method from [32] is recapitulated in Section 5.1.

I2 Controllability of $E_b(t_{f2})$ is not a continuous function of λ_b, due to the clutches in the topology, resulting in non-unique solutions. The switching algorithm in Section 5.2 ensures controllability for all λ_b.

I3 When the horizon does not include t_f, the end condition on $E_b(t_{f2})$ is not known, thereby removing charge sustenance $\Delta E_b = 0$ as a control target. In Section 5.3, a solution is proposed, which is guaranteed to be optimal for a subset of scenarios.

In the remainder of this section, $t_{f2} = t_f$ for ease of notation, but without loss of validity of the Level 2 problem where $t_{f2} < t_f$.
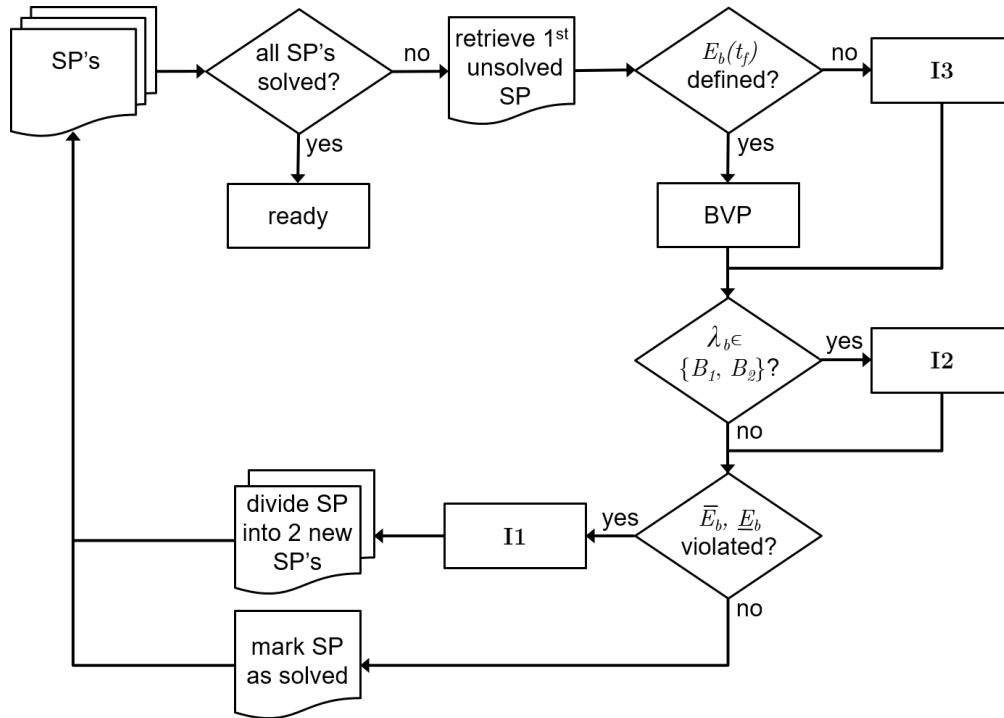
Figure 11. For dealing with constraints, non-unique solutions and a limited horizon, the flowchart shows the procedure for solving the boundary value problems (BVP). Starting with one unsolved (sub)problem (SP), from t_0 to t_f, the process is started. Dependent on the number of constraint violations, the procedure ends with a set of solved subproblems, each providing a segment of λ_b over the complete horizon. Each subproblem contains the solution to a BVP, and dependent on the stated conditions, the additional procedures **I1**, **I2** or **I3** are needed.

5.1. Constraint Handling (I1)

The optimal solution to Equations (21)–(27) results in a constant λ_b when E_b constraints are not present in Equation (54), which can be efficiently found by solving the two-point boundary value problem. When constraints on E_b are violated, the original problem must be subdivided into one or more subproblems dependent on the amount of constraint violations, as described in [32]. As we elaborate on this procedure in the next sections, we repeat an outline of the procedure here:

- The unconstrained subproblem is solved for a constant λ_b.
- If no constraint violation is present, the subproblem is solved. The procedure is repeated for the next unsolved subproblem.
- If constraint violations are present, the time t_c of the largest constraint violation is identified. The subproblem is divided at t_c into two new subproblems, with the first subproblem, having $E_b(t_c) = \overline{E_b}$ as an end point constraint and the second subproblem having $E_b(t_c) = \overline{E_b}$ as a start point constraint, when $\overline{E_b}$ is violated. For $\underline{E_b}$ violations $E_b(t_c) = \underline{E_b}$.

This sequence is repeated, until all subproblems are solved, resulting in a piecewise constant λ_b^* for the whole cycle.

5.2. Controllability of E_b: Non-Uniqueness (I2)

The optimal solution has a non-continuous $E_b(t_f)$ controllability as a function of λ_b, leading to problems in realizing charge sustenance. The cause are the guards in the optimal solution of H (see Section 4.2), where a switch from one mode to another is enforced. When P_d and λ_b coincide with one of the guards B, the optimal solution is non-unique (singular), i.e., several control inputs lead to the same fuel optimum [31,33,34].

The first type of guards are a function of P_d (and λ_b), i.e., B_3, B_4, B_5 (see Section 4). As P_d normally varies in time, the optimal solution will seldom be at one of the guards for a prolonged duration. For that reason, Ref. [15] selects one mode a priori, when P_d is on the guard, thereby being optimal, but without a guarantee to be charge sustaining. In simulations, especially with perfectly constant P_d, switching between the modes could be necessary for a charge sustaining solution. Therefore, Ref. [18] opts to vary λ_b around λ_b^*, which causes switching between the modes, but is inherently sub-optimal because of the deviation from λ_b^*. In [33], this is correctly solved by using a so-called 'sliding mode' control, which minimizes the number of mode changes, which does not change λ_b^*. In our EMS, we also unalter λ_b^* and remain in the last mode, when P_d coincides with the guards B_3, B_4, B_5.

The second type of guards is *not* a function of P_d, only of λ_b, i.e., B_1 and B_2. As λ_b^* is (piecewise) constant, the solution can be on B_1 and B_2 for a prolonged period of time. Figure 8 shows that the optimal solution switches from M_c to M_e when λ_b is at guards B_1 or B_2. Consequently P_b^* jumps at B_1 and B_2, and, for an arbitrary drive cycle, the controllability of $E_b(t_f)$ (or charge sustenance) jumps as a function of λ_b—see Figure 12.
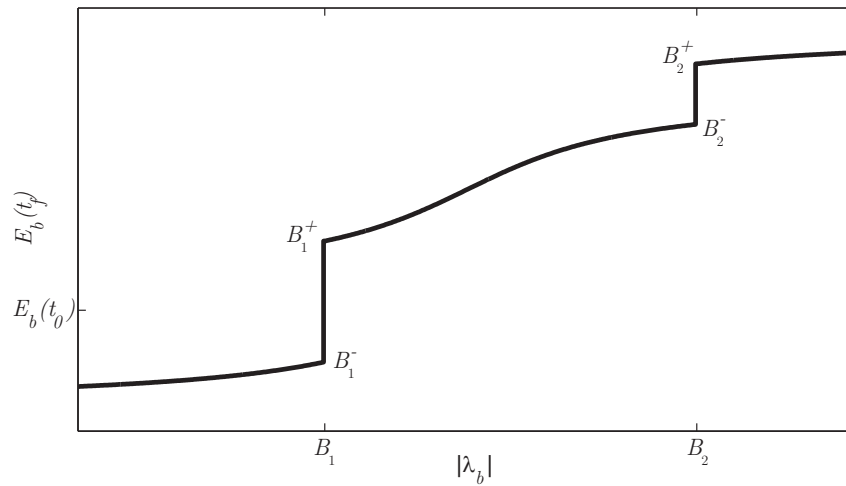

Figure 12. Changing the constant λ_b for a drive cycle, causes jumps in the charge sustenance, due to non-unique (singular) solutions at B_1 and B_2.

To have continuous controllability of $E_b(t_f)$ in B_1 and B_2, a switching sequence must be chosen and, in the following sections, algorithms for the two (extreme) switching scenarios are explained:

I2a with an infinite number of switchings,
I2b with a minimal number of switchings (elaborating on [33]).

The switching scenarios are illustrated for a drive cycle of 2000 s, see Figure 13, for which λ_b^* coincides with B_1. Trajectory B_1^+ (dash-dotted line) starts at $E_b(0)$ and is above charge sustaining at $t_f = 2000$ s. Trajectory B_1^- (dashed line) starts at $E_b(0)$ and is below charge sustaining at $t_f = 2000$ s. In the top plot, P_m is shown for both trajectories, where the difference is in supporting the CE with MG or not (boosting). The charge sustaining solution consists of switching between B_1^+ and B_1^- over time, such that $E_b(0) = E_b(t_f)$. At least one switch must be made, i.e., at t_{s1} or t_{s2} dependent on the initial mode. The next subsections provide algorithms to determine the switching sequence, taking constraints on E_b into account. The algorithms, as illustrated for B_1, also hold for B_2.
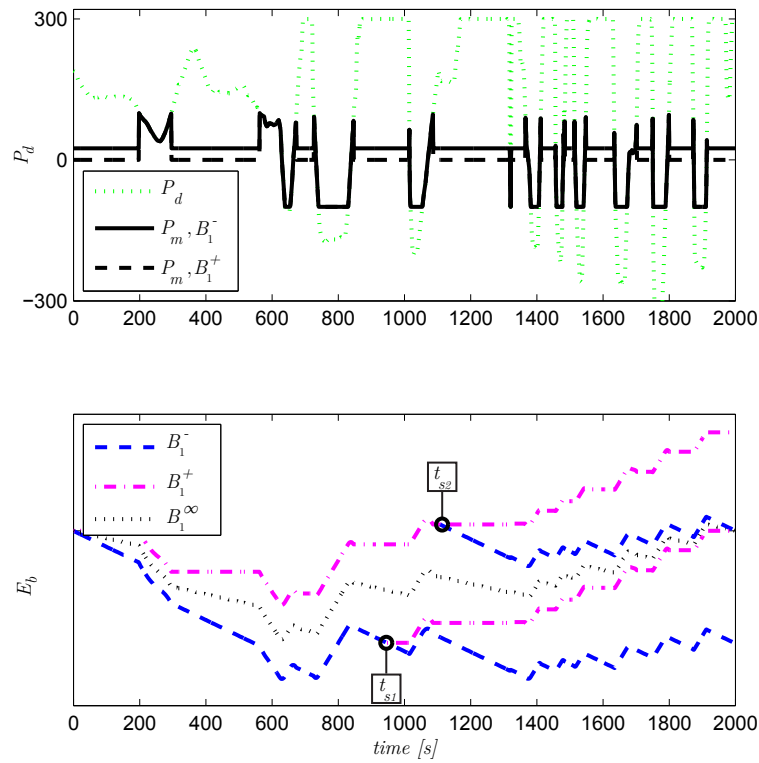
Figure 13. Illustration of algorithm I2a. Power demand P_d of a drive segment with $\lambda_b^* = B_1$ and two optimal modes: B_1^- (P_m is positive during high power demands, resulting in too low $E_b(t_f)$) and B_1^+ (P_m is zero during high power demands, resulting in too high $E_b(t_f)$). A charge sustaining trajectory ($E_b(t_f) = E_b(0)$), is possible with one switch at t_{s1} or t_{s2}, or with infinitely many switchings indicated as B_1^∞.

5.2.1. Infinite Number of Switchings (I2a)

As the dynamics of E_b in Equation (6) are state independent, a charge sustaining solution can be realized by taking a linear combination of the two trajectories B_1^- and B_1^+, such that:

$$r \cdot E_b^{B_1^-}(t_f) + (1-r) \cdot E_b^{B_1^+}(t_f) = E_b(0), \tag{55}$$

where r is a ratio between 0 and 1. In the example in Figure 13, this combined solution is charge sustaining with $r = 0.46$ (dotted line). This combined solution implies an infinite amount of switchings between the two trajectories, and will be referred to as B_1^∞.

5.2.2. Minimal Switching (I2b)

To enforce minimal switching, considering constraints on E_b, a combination of algorithms I1 and I2a is needed. First, it is checked if the subproblem must be further subdivided as in Section 5.1 to guarantee E_b within its bounds:

- if $E_b^{B_1^-} > \underline{E_b}$, then procedure I1 for constraint handling is performed,
- if $E_b^{B_1^+} < \overline{E_b}$, then procedure I1 for constraint handling is performed,
- if $E_b^{B_1^\infty}$ exceeds limits, then procedure I1 for constraint handling is performed,
- else a switching sequence exists in order to maintain E_b within bounds.

The switching sequence with minimal switchings is determined, by choosing when to switch between the two state trajectories B_1^- and B_1^+. The algorithm is illustrated in Figures 14 and 15.

- The initial mode M is maintained, here resulting in B_1^-,
- Check for constraint violations. $\underline{E_b}$ violation is detected at $t = 388$s, so switching to B_1^+ is mandatory. Switching at $t = 388$s results in E_b undershoot of $E_b^\wedge$. With the known relative E_b trajectories of B_1^- and B_1^+, the advancement of the switching time is determined, here at 83 s, Figure 15,
- The previous step is repeated, until t_f is reached,
- $E_b(t_f)$ is now within limits, but not at $E_b(0)$. A switch to the other mode is back propagated from $E_b(t_f) = E_b(0)$, resulting in a last switch at 1191 s.

With this procedure, the discontinuity in the controllability of $E_b(t_f)$ is solved with a switching sequence between modes, where the number of switchings is minimal. The value of λ_b^* has not been altered.

Note that knowing the switching sequence is not necessary on Level 2, as the interface from Level 2 to Level 1 does not use E_b, but λ_b, which is not influenced by the switching sequence. Knowing that a switching sequence exists is thereby sufficient. The Level 1 optimization then decides on switching, taking switching costs into account (Section 4.3).
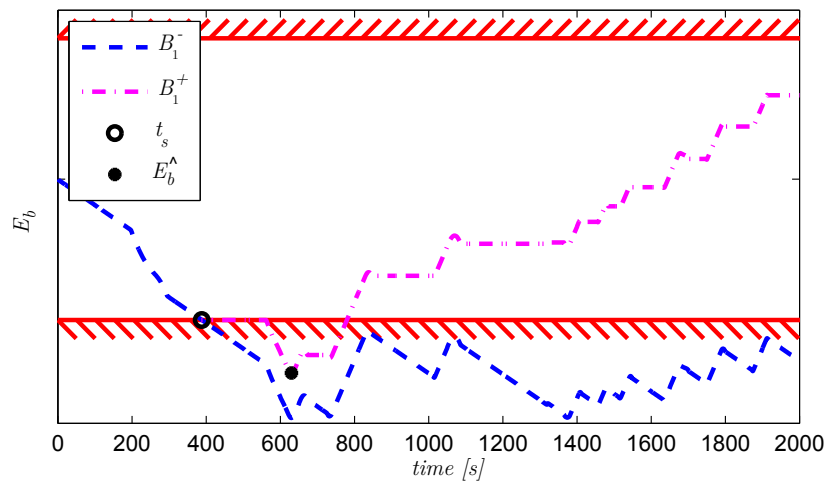

Figure 14. Illustration of algorithm I2b, first part. Starting with B_1^- the lower boundary is violated at t_s and a switch must be made to B_1^+. However, $t_s = 388$ s as switching point is too late, as B_1^+ exceeds $\underline{E_b}$ with $E_b^\wedge$.
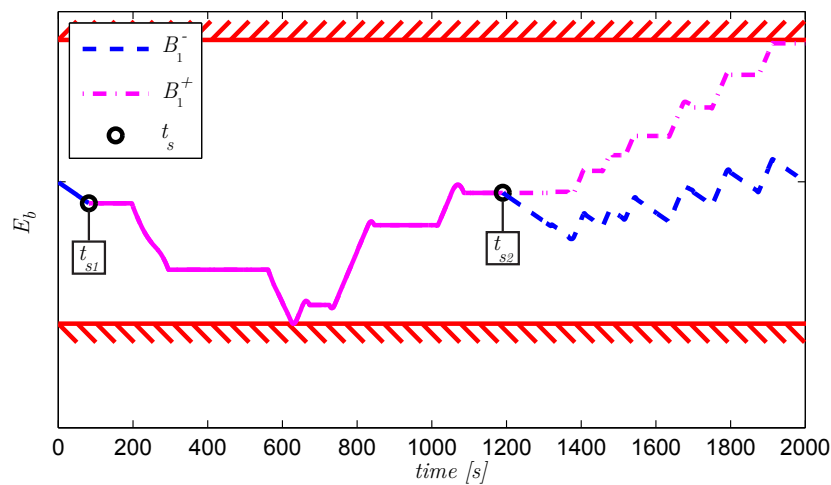

Figure 15. Illustration of algorithm I2b, second part. A new $t_{s1} = 83$ s is found, and B_1^+ is continued, until at $t_{s2} = 1191$ s the switch to B_1^- has to be made to ensure $E_b(t_f) = E_b(0)$.

5.3. Limited Horizon (I3)

When $E_b(t_f)$ is prescribed, the boundary value problem is fully defined and the optimal solution can be found. However, in real life, $E_b(t_f)$ is only prescribed if the complete drive cycle is used (full horizon) and $E_b(t_f)$ has a value, e.g.,

- for a repeating cycle, with $E_b(t_0) = E_b(t_f)$, which is often used in simulation assessments,
- when entering a zero-emission zone at t_f, with $E_b(t_f) = \overline{E_b}$ for maximal electrical driving after t_f,
- when arriving at a charging station at t_f, with $E_b(t_f) = \underline{E_b}$ for cheap charging of the battery after t_f.

If the prediction does not include t_f, e.g., when implemented as a receding horizon algorithm, $E_b(t_f)$ is undefined.

The following, suboptimal, procedure aims at finding a constant λ_b, without constraint violations, resulting in a E_b trajectory that has maximal robustness towards constraint violations. Here, it is chosen to maximize the minimal distance of the E_b trajectory to the constraints, in order to allow for disturbances in the prediction, without violating constraints. The outline of the procedure is:

- choose λ_b based on bi-section and simulate E_b,
- find the time index of $\overline{E_b}$ and $\underline{E_b}$,
- calculate the distance to the boundary for the two indices and select the minimal distance,
- select the next λ_b with bi-section, to increasing the minimal distance, and iterate.

The iteration stops when the increment of the minimal distance is smaller than a tolerance value. When the resulting E_b violates constraints, the problem is subdivided as described in Section 5.2 and each sub-problem is evaluated again.

For two special cases, the above procedure results in optimal solutions:

I3a if within the horizon, a E_b limit is activated, irrespective of the control action e.g., due to a large energy recuperation event, the first subproblem up to this event is fully defined. The solution to this first subproblem is then optimal, and independent from the following subproblems.

I3b if (λ_b^*, P_d) is on guard resulting in non-unique solutions *and* the non-unique solutions are able to span $[\overline{E_b}(t_f), \underline{E_b}(t_f)]$, λ_b^* is optimal for all $E_b(t_f) \in [\overline{E_b}, \underline{E_b}]$.

With these algorithms, the calculation of the output of Level 2 (λ_b) is complete.

6. Level 3: Velocity Prediction

The Level 3 functionality determines the velocity profile of the vehicle, given the slope and velocity limitations of the drive cycle, and provides the Level 2 and Level 1 algorithms with an estimated power demand P_d and velocity v. This section describes a simplified solution using results from the optimal control formulation.

Velocity Prediction Using Three Driving Modes

The Level 3 functionality predicts P_d from information of the road ahead, i.e., road slope α and velocity limitations $\overline{v}$. The velocity of the vehicle determines the resulting road load, so assumptions have to be made how the vehicle will accelerate/decelerate over the route. To determine the fuel optimal velocity profile, an optimization has to be solved. In [35], this optimization is performed for a conventional driveline with an affine fuel map, resulting in a set of three possible driving modes:

1. maximum acceleration with $\overline{F_d}$,
2. minimum acceleration with $\underline{F_d}$,
3. constant speed v_0, with $F_d = F_0$.

The acceleration limits depend on the capabilities of the drive-line force F_d, for a given road load F_0. When we assume a continuously variable gearbox ratio, controlled to a fixed engine speed, and only one hybrid mode M_e, Equations (2), (11), (15)–(19) and Table 4, define the acceleration limits as a function of velocity and road slope: $\overline{F_d}(v, \alpha)$ and $\underline{F_d}(v, \alpha)$. For simulation, the vehicle model is completed with Equation (14). We use the three modes, to predict the velocity profile, given the road information and the vehicle model. The outline of the algorithm is given:

- retrieve $\pmb{\alpha}(s)$ and $\overline{v}(s)$ from the preview data source, for the drive cycle,
- decide on v_0, typically the reference speed of the speed control on the vehicle,
- simulate the vehicle model forwards over the drive cycle, using mode 1 (maximum acceleration). Limit the velocity to v_0 and $\overline{v}(s)$. Store the resulting vector of the forward simulation $\pmb{v_f}(s)$,
- simulate the vehicle model backwards over the drive cycle, using mode 2 (minimum acceleration). Limit the velocity to v_0 and $\overline{v}(s)$. Store the resulting vector of the backward simulation $\pmb{v_b}(s)$,
- calculate the minimum of the two vectors $\pmb{v_c}(s) = \min(\pmb{v_f}(s), \pmb{v_b}(s))$,
- calculate $\pmb{P_d}$ using the vehicle model and $\pmb{v_c}(s)$.

With this procedure, preview information of road slope $\pmb{\alpha}$ and velocity limitations $\overline{\pmb{v}}$ is converted to a prediction of the power demand $\pmb{P_d}$ and velocity $\pmb{v}$ for the lower levels.

A suggestion for future research is to calculate on this level the optimal velocity profile, considering hybrid driving modes, braking, discrete gear shifting and non-affine fuel maps.

7. Simulation Results of the Multi-Level EMS

Fuel optimality of the multi-level EMS is shown in this section, by three simulation scenarios:

- 'Multi-level iteration, short cycle' (Section 7.1), using full horizon optimization on a short, abstract drive cycle. The optimization iterates over the levels, until the fuel consumption is converged to an optimum. With the full horizon, each level has complete information of the drive cycle. The results show how fast the algorithm converges, and, by using an abstract cycle, the decisions on each level are interpreted.
- 'Multi-level MPC, short cycle' (Section 7.2), using receding horizon optimization on a short, abstract drive cycle. By limiting the horizon and implementing the iterations over a receding horizon, the optimality will reduce. This section shows the fuel penalty of this approach, and by using the same abstract cycle, the decisions on each level are compared to the previous scenario.
- 'Multi-level MPC, long cycle' (Section 7.3). To show the real-life fuel saving potential of the multi-level MPC, a representative, long-haul, drive cycle is used, replacing the abstracted cycles of the previous scenarios. Furthermore, a high fidelity model of the vehicle replaces the model from Section 2. The results are compared with a baseline EMS, which uses heuristics and no preview.

7.1. Multi-Level Iteration, Short Cycle

The first two scenarios are demonstrated for a cycle of 1055 m, with a vehicle weight of 25 tons. The elevation profile of this cycle is shown in Figure 16, and contains four slopes of respectively +6%, −12%, +12% and -6%. This elevation profile is also available as a physical test track as described in [28]. The maximum feasible speed on this track is 25 km/h. The plant model uses the models from Section 2, including the effect of open driveline during switching ($\Delta t_M > 0, \Delta t_G > 0$).

The multi-level optimization is iteratively solved for the complete test cycle. Multiple iterations are performed; however, no decrease in fuel consumption is observed by consecutive iterations. The energy consumption of the first two iterations are shown in Table 7, where the energy consumption difference (0.1%) is negligible small. The control decisions of these two iterations are analyzed and compared in Appendix D and show that the calculated decisions after iteration 0, hardly change with iteration 1, with the difference in charge sustenance explaining the marginal change in energy consumption.
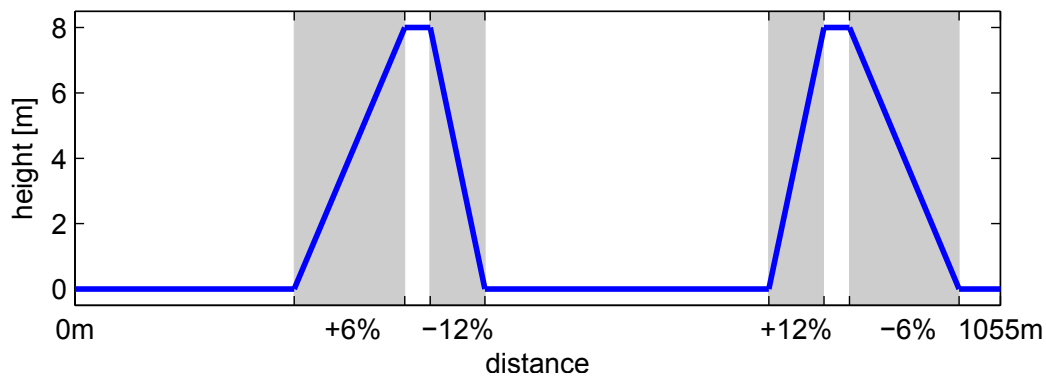
Figure 16. Elevation profile of the short test cycle.

7.2. Multi-Level MPC, Short Cycle

For real-time applications, the Dynamic Program in Level 1 over the complete route is computationally too demanding. In this section, we simulate Level 1 as receding horizon optimization. Here, we choose to optimize Level 1 at a sample time of 1 s, with a horizon of five samples. The other levels are unaltered, as they don't pose computational problems over the cycle's horizon.

The energy consumption is marginally different (0.2%) from the other scenarios. The control decisions of the multi-level MPC are analyzed and compared with the multi-level iteration in Appendix E, showing equal decisions as the multi-level iteration. One essential difference in gear selection can be explained from the shorter horizon, however, without a relevant impact on energy consumption.

Table 7. Energy consumption over the short test cycle. MPC: model predictive controller.

	J [MJ]	Energy Reduction [%]
non-hybrid	16.8	0
iteration 0	7.91	52.9
iteration 1	7.90	53.0
MPC	7.93	52.8

Comparing the three different scenarios, we conclude that the multi-level iteration scheme converges very fast: an update of the estimate of ω is not needed to improve the results of the optimizations on the layers, as iteration 0 is already at a minimal energy consumption. The multi-level MPC has a shorter horizon on Level 1, which leads to slightly different control decisions, but does not significantly alter the energy optimum. For fuel evaluation, the next section presents more realistic long-haul simulations, with the multi-level MPC implemented.

7.3. Multi-Level MPC, Long Cycle

To show the fuel saving potential of the multi-level EMS, a typical long-haul route is simulated, in which a non-hybrid vehicle is compared to a hybrid vehicle with baseline EMS ('baseline') and the multi-level MPC ('preview'). The baseline EMS is a proprietary non-previewing, heuristic algorithm. The proprietary simulation environment is designed for fuel evaluation purposes and contains high-fidelity models of the vehicle. The multi-level MPC is implemented, with the following settings:

- Level 1, 5 s horizon, sample time 0.01 s,
- Level 2, 1000 s horizon, sample time 1 s,
- Level 3, 1000 s horizon, sample time 1 s.

For the 25-ton vehicle, the fuel consumption results are shown in Table 8.

Table 8. Fuel savings on a typical long-haul route. EMS: energy management system.

Vehicle	Fuel [l/100 km]	Fuel [%]
non-hybrid	33.0	0
EMS 'baseline'	31.1	−5.8
EMS 'preview'	30.7	−7.0

The difference between 'baseline' and 'preview' is best illustrated with the fuel saving and battery energy E_b, plotted in Figure 17. There we observe the largest advantage of the previewing strategy, when battery limits are frequently touched, i.e., in the hilly part of the cycle: from 150–350 km. To recuperate a maximum amount of brake energy, 'baseline' discharges the battery as soon as possible, which is not the most efficient use of the recuperated energy. The previewing strategy discharges the battery sufficiently, to recuperate the maximum amount of brake energy, and uses the stored energy more efficiently, e.g., to drive MG-only during low road loads.
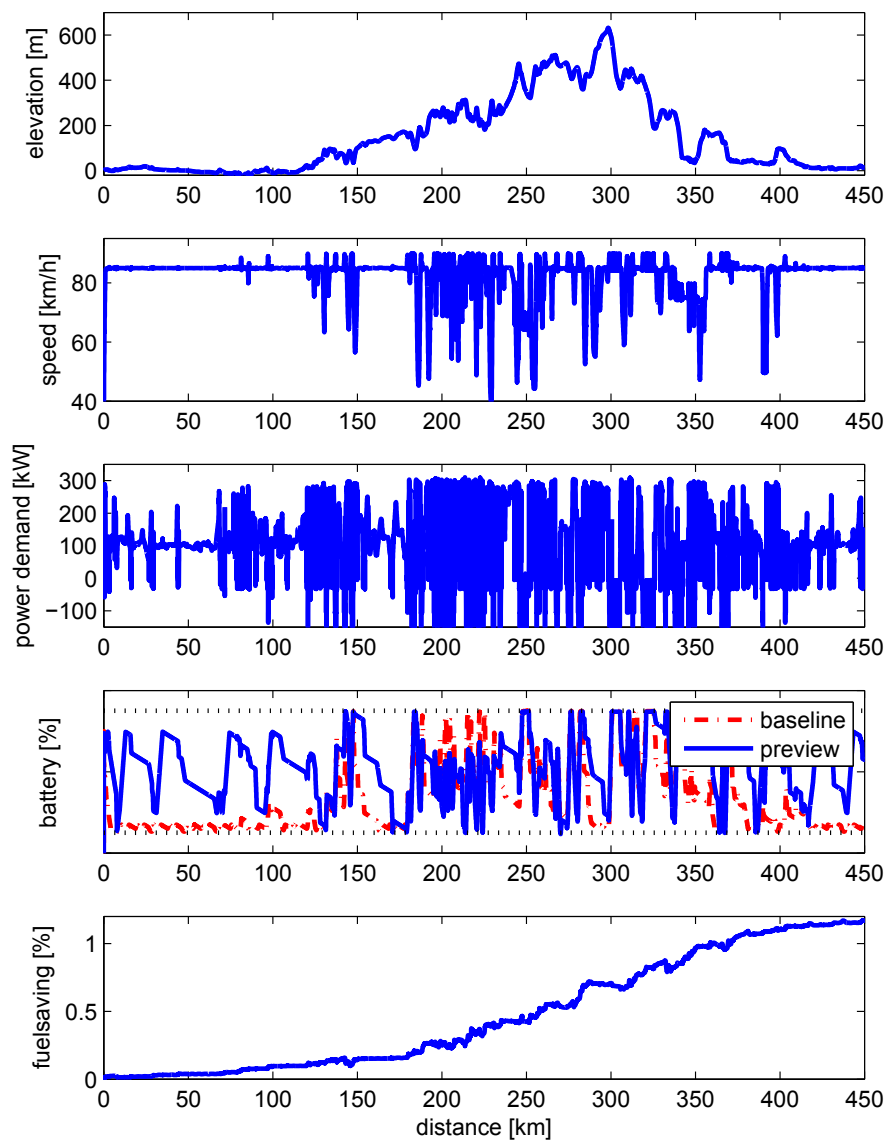
Figure 17. High fidelity simulation results on a representative long-haul cycle. The previewing Energy Management System (EMS) saves an additional 1.2% fuel to the baseline EMS.

Also on segments *without* battery limitations, the previewing strategy outperforms the baseline strategy. On the flat road, e.g., from 0–120 km, λ_b coincides with guard B_2 (see Figure 9 and algorithm I3b in Section 5), resulting in fuel optimal switching control: charging the battery is performed in a relatively short period of time, shown by the steep increase of battery energy, followed by a prolonged period of a disconnected MG, thus eliminating MG friction P_{m0}. The baseline strategy, however, uses lower charging powers, thereby having little periods of MG disconnection, and thus higher MG friction. Furthermore, from a driveability perspective, the larger duty cycles of the previewing strategy, in connecting/disconnecting the MG, are preferred.

8. Conclusions

In this work, a computationally efficient EMS using preview is developed, obtaining excellent fuel economy results (-7.0%) on a typical long-haul route. To manage computational complexity, a multi-level EMS is designed, with on each level a specific algorithm for the subproblem to solve. Each subproblem uses different model information of the vehicle, such that optimal control techniques can be used, to efficiently solve the problem.

On level 1, the power-split, gear, mode decisions are optimized, including costs on switching. For the topology with two clutches, a novel explicit PMP solution is found, which reduces power-split and mode selection to a static calculation, dependent on the current vehicle state and battery costate (or Lagrange multiplier). For taking switching costs into account, a Dynamic Program is designed, using the explicit PMP solution for reducing the dimension of the problem, and thereby limiting the complexity of the calculations.

On level 2, the use of battery energy is optimized, taking the state constraints of the battery into account. A simpler, more abstracted vehicle model is used, e.g., by ignoring switching costs, to reduce computational complexity. Using PMP, the battery costate (or Lagrange multiplier) is calculated, which provides essential cost information to Level 1. The topology with two clutches, causes discrete jumps in the optimal solution, leading to non-unique (singular) solutions. New optimal solutions are found, minimizing the amount of switching events.

On level 3, the velocity is predicted along the route, using road slope and velocity limitations. With the calculation, essential route information is provided to the lower layers. Future work should include velocity optimization on this level, to further reduce the fuel consumption.

In offline simulation, on a short abstract drive cycle, the multi-level approach is shown to converge within one iteration. When the multi-level approach is used as Model Predictive Controller (MPC), comparable fuel energy efficiency is realized, even though the horizon is reduced.

Representative fuel saving results are obtained by running the multi-level MPC in a high fidelity simulation environment, over a typical heavy duty drive cycle: 7.0% fuel reduction compared to a conventional vehicle, where the baseline EMS for the hybrid electric vehicle saves 5.8%.

Author Contributions: V.v.R. developed the presented concepts and methodologies. He performed writing and editing of the original draft. T.H. was responsible for supervision, internal reviewing and editing of the draft and concepts.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Pontryagin's Minimum Principle

The Pontryagin's Minimum Principle provides a set of equations (or conditions) where the optimal solution of a (constrained) dynamic optimization must comply to. Here, the PMP conditions are summarized, following [31,36]. For the fuel-cost functional:

$$\min_u J(u,t) = \min_u \int_{t_0}^{t_f} P_f(x(t),u(t),w(t))dt \tag{A1}$$

with state equations given by:

$$\dot{x} = f(x(t),u(t)), \tag{A2}$$

the Hamiltonian is formulated:

$$H = P_f(x(t),u(t),w(t)) + \lambda(t)^T f(x(t),u(t)), \tag{A3}$$

where $\lambda(t)^T$ are the costates or Lagrange multipliers. For the optimal solution, it holds that

$$H^*(x^*(t),u^*(t),\lambda^*(t),t) \leq H(x^*(t),u(t),\lambda^*(t),t), \tag{A4}$$

with the (Euler–Lagrange) necessary conditions:

$$\dot{x}^*(t) = \frac{\partial H}{\partial \lambda}, \tag{A5}$$

$$\dot{\lambda}^*(t) = -\frac{\partial H}{\partial x}, \tag{A6}$$

where the superscript * denotes the optimal solution. When the state is on a constraint at $t_0 < t_j < t_f$, $\frac{\partial H}{\partial x}$ is not defined and interior boundary conditions apply [37],

$$H^+(t_j) = H^-(t_j) - \zeta(t_j)\frac{\partial h(t_j)}{\partial t}, \tag{A7}$$

$$\lambda^+(t_j) = \lambda^-(t_j) + \zeta(t_j)\frac{\partial h(t_j)}{\partial x}, \tag{A8}$$

where the superscript $^-$ and $^+$ denote the left-hand side and right-hand side limit values, respectively, at t_j. The conditions describe a possible jump condition in λ and H at t_j, when the state makes contact with the boundary h. The magnitude of the jump is determined by the parameter ζ, and is, together with the number of jumps, not known a priori. For the boundary conditions, additional transversality conditions hold, and are given (omitting the penalty on the final state) by:

$$[-\lambda^*(t_f)]^T \delta x_f + [H(x^*(t_f),u^*(t_f),\lambda^*(t_f),t_f)]\delta t_f = 0. \tag{A9}$$

Appendix B. Interpretation of the Costate

For the optimal control problem, the cost function J is minimized, which represents the total amount of fuel energy E_f over a drive cycle:

$$J = \int L dt, \tag{A10}$$

with $L = \dot{E}_f$ and

$$f = \dot{x}, \tag{A11}$$

with x the state(s). Formulating the Hamiltonian leads to:

$$H = L + \lambda^T f, \tag{A12}$$

with λ the costate(s). Using the same variational approach used to derive the PMP conditions, the variation of the optimal J is defined as [31,36]:

$$\delta J = \lambda^T(t_0)\delta x(t_0) + \int_{t_0}^{t_f} \frac{\partial H}{\partial u}\delta u\,dt, \tag{A13}$$

which reveals the interpretation of λ at any time t: when t_0 and t_f approach each other to t, the second term vanishes, and λ defines how much J changes, for a small change in x, i.e., λ is the gradient of the cost function, with respect to the states. This interpretation of λ is known as *equivalent cost* for the well known Equivalent Consumption Minimization Strategy [38], as it provides the equivalence (or price) between fuel energy and battery energy. When the same optimal control techniques are used in other EMS's, e.g., considering more or different energy buffers, the interpretation of the costate is not always straightforward. Looking at the case of vehicle speed optimization as in [35,39], optimal control techniques lead to the optimal solution, but interpretation of λ is lacking. The reason is that, in these examples, the state is chosen to be velocity, which is not energy. If, however, the problem is reformulated with kinetic energy as state, see [40], the corresponding λ dynamics show valuable information. Thus, before λ is easily interpretable, the states of the problem should be reformulated as energies, as, e.g., in [8,9,40], where it is irrelevant if the reformulation is done before or after solving the problem. An example Hamiltonian of an energy minimizing function could be

$$H = \dot{E}_f + \lambda_k\dot{E}_k + \lambda_b\dot{E}_b + \lambda_T\dot{E}_T + \lambda_t, \tag{A14}$$

with fuel energy E_f, kinetic energy E_k, battery energy E_b, thermal energy E_T and λ_t for the cycle duration. Then, by minimizing H, λ determines which energy buffer to charge and which energy buffer to deplete, thus λ being the buffers equivalency, or when the energy is valorized: price.

Appendix C. Cost of Switching

A switch of mode or gear involves a series of events, having some disadvantages, e.g.:

1. synchronization of rotating masses (acceleration/deceleration), possibly with slipping clutches,
2. interruption of traction.

The combination of these events determines the associated (fuel) cost of a mode/gear change event. The first item is defined by the design of the driveline itself and the low-level control of the event, and the energy losses can be calculated irrespective of the drive cycle, [19,41]. The second item is determined by the performance of the low-level control for the given hardware, namely the period of time traction is interrupted. Without traction, the vehicle is freewheeling, and dependent on the road load, will decelerate or accelerate. A cost can be associated with the lack of traction, dependent on the road load F_0—for example driving uphill, interruption of traction is much more expensive than during cruising on a flat road. This second effect is not addressed by [41] or [19]. The novel approach here is to use cycle information from the velocity optimization, i.e., λ_k and λ_t (Appendix B), in order to account for the road load dependent switching cost.

The switch penalty is thereby divided into two parts: cost associated to synchronization, indicated with subscript s, and cost associated with traction interruption, indicated with subscript t, for both mode switches (ΔM) and gear switches (ΔG):

$$\epsilon_M = c_{Ms}(\Delta M) + c_{Mt}(\Delta M, \lambda_t, \lambda_k), \tag{A15}$$

$$\epsilon_G = c_{Gs}(\Delta G) + c_{Gt}(\Delta G, \lambda_t, \lambda_k). \tag{A16}$$

Appendix C.1. Synchronization Cost

During a switch, rotating masses have to be accelerated and decelerated. Here, we assume:

- the largest cost is due to acceleration of rotation masses (inertia),
- during deceleration no energy is recuperated,
- switch events are typical, e.g., based on average behavior,

- each switch involves an inverse switch on a later moment.

The last assumption makes it possible to allocate an average cost to two linked events, e.g., starting the Combustion Engine (CE) has large costs associated, while stopping the CE is free. However, the one can not occur without the other, so the total cost of the two events can be averaged over the two single events. This results in balanced switch decisions in the EMS, when the horizon is small and only one of the linked events can be overseen within the horizon.

With the inertia from Table 3, the equivalent cost of the transitions are quantified with the change of rotational kinetic energy:

$$\Delta U_I = 1/2 I^+ (\omega^+)^2 - 1/2 I^- (\omega^-)^2) \tag{A17}$$

and translated to the equivalent fuel with:

$$\Delta H_m \approx \Delta U_I \lambda_b, \tag{A18}$$

which results in

$$c_{Ms} = \begin{cases} 0, & \text{if } \Delta M = 0, \\ \Delta H_m, & \text{if } \Delta M \neq 0, \end{cases} \tag{A19}$$

with the inertia and speed just before (I^-, ω^-) and after (I^+, ω^+) the mode switch. As λ_b defines the fuel equivalence between electrical and thermal energy, it is not relevant with which energy source the synchronization is performed. For example, an ICE start first involves an electrical acceleration, followed by mechanical acceleration as soon as the rotational speed is above idle speed, but, with λ_b, they are equivalent. The same method is used to calculate c_{Gs}.

For the following rotational speeds,

- disconnected, stopped: 0 rpm,
- connected: 930 rpm,
- downshift: 1200 rpm,

the equivalent costs (here: $\lambda_b = -2.2$) are exemplified in Table A1. The downshift event losses [*] depend on the active mode, respectively M_m, M_e, and M_c, because losses increase with the amount of rotational mass to be accelerated, which is the largest with both CE and MG connected. The transitions involving deceleration of the rotational masses are set to 0. However, some energy recuperation could be implemented with the MG, resulting in lower average equivalent costs.

Table A1. Equivalent cost of transition: synchronisation.

Transition	Direction $\Rightarrow$ [kJ]	Direction $\Leftarrow$ [kJ]	Avg [kJ]
$G_k \Leftrightarrow G_{k+1}$	0	10,20,30[*]	20
$M_e \Leftrightarrow M_m$	16	31	24
$M_e \Leftrightarrow M_c$	16	0	8
$M_m \Leftrightarrow M_c$	31	0	16

It should be noted that the transitions are idealized. The real energy consumption of the transitions will be worse than the posed numbers due to additional losses but can be improved by recuperation on the decelerating masses. Nevertheless they provide a first estimate on the selection of ϵ, which has clear mode dependent values. For a more thorough analysis of the gear switch event, including clutch slip, see [41] (Chapter 6).

Appendix C.2. Traction Interruption Cost

During the transition of a switch, the traction is momentarily interrupted. This interruption causes a deviation from the intended vehicle speed, which has to be compensated during the periods with

traction. Dependent on the road load, the cycle and the component capabilities, this compensation is expensive or cheap. The idea here is to quantify the cost of this compensation to the cycle through the information contained in the costates belonging to the cycle: λ_t, λ_k in Equation (A14). To link this information to the traction interruption cost, the following idealized transition is assumed:

- the transition has a fixed traction interruption time (Δt_o),
- during the transition, the road load is constant,
- the driveline is not saturated, i.e., traction is available to recover the vehicle speed,
- λ_t represents the equivalent cost of time loss for a certain cycle, and is constant.

In Figure A1, the idealized transition is plotted. While cruising at v_0, where $F_d = -F_0$, a switch event is planned at t_0. During the following open driveline period t_0–t_1, constant deceleration causes the vehicle speed to drop. After the transition, we assume immediate recovery to the desired speed, with constant acceleration at $\overline{F_d}$, depending on the mode M and associated component capabilities. During this event, the vehicle loses time, with respect to cruising without transition. If we compare the total transition+recovery time to reach the position s_2 at t_2, with the time needed for constant cruising to reach position s_2, we define the time loss Δt_t of the transition event:

$$\Delta t_t = (s_2 - s_0)/v_0 - (t_2 - t_0), \tag{A20}$$

with

$$
\begin{aligned}
s_1 - s_0 &= v_0(t_1 - t_0) + \frac{1}{2m}F_0(t_1 - t_0)^2, \\
v_1 &= v_0 + (F_0/m)(t_1 - t_0), \\
t_2 &= -(t_1 - t_0)\frac{F_0}{F_v + \overline{F_d}}, \\
s_2 - s_1 &= v_1(t_2 - t_1) + \frac{1}{2m}(F_0 + \overline{F_d})(t_2 - t_1)^2.
\end{aligned}
\tag{A21}
$$

This time loss is converted to equivalent fuel, with λ_t, thus providing a road load dependent penalty on switching,

$$
c_{Mt} = \begin{cases}
0, & \text{if } \Delta M = 0, \\
\Delta t_t \lambda_t, & \text{if } \Delta M \neq 0 \text{ and } |F_0| < \overline{F_d}, \\
1/2m(v_1^2 - v_0^2)\lambda_k, & \text{if } \Delta M \neq 0 \text{ and } |F_0| \geq \overline{F_d}.
\end{cases}
\tag{A22}
$$

When the drive-line saturates ($|F_0| \geq \overline{F_d}$), the vehicle is not able to recover the cruise speed after a short open driveline event. In that situation, the speed deviation $v_1 - v_0$ is used with λ_k to calculate the speed recovery cost. The same method is used to calculate c_{Gt}.

As an example the equivalent cost of a transition due to time loss is given in Table A2, for the model parameters from Table 3, the vehicle driving 25 km/h, and the cycle having a λ_t of -25 kJ/s. The event is tabulated for three segments of, respectively, 0%, 6% and 12% slope. The traction interruption period $t_1 - t_0$ is set to 1 s.

Table A2 shows that more than a magnitude order of difference occurs for a switch cost. This means that the influence of time loss on the equivalent fuel consumption is relevant when the open driveline deceleration (acceleration) is high, where on other route segments the influence is negligible. This shows that transitions can be best planned during the segments with smallest deceleration (acceleration).

The switch cost is now determined by two elements: synchronization cost as a function of the intended transition, and time loss cost as a function of position along a route and the available traction for the transition, in Equations (A15) and (A16).
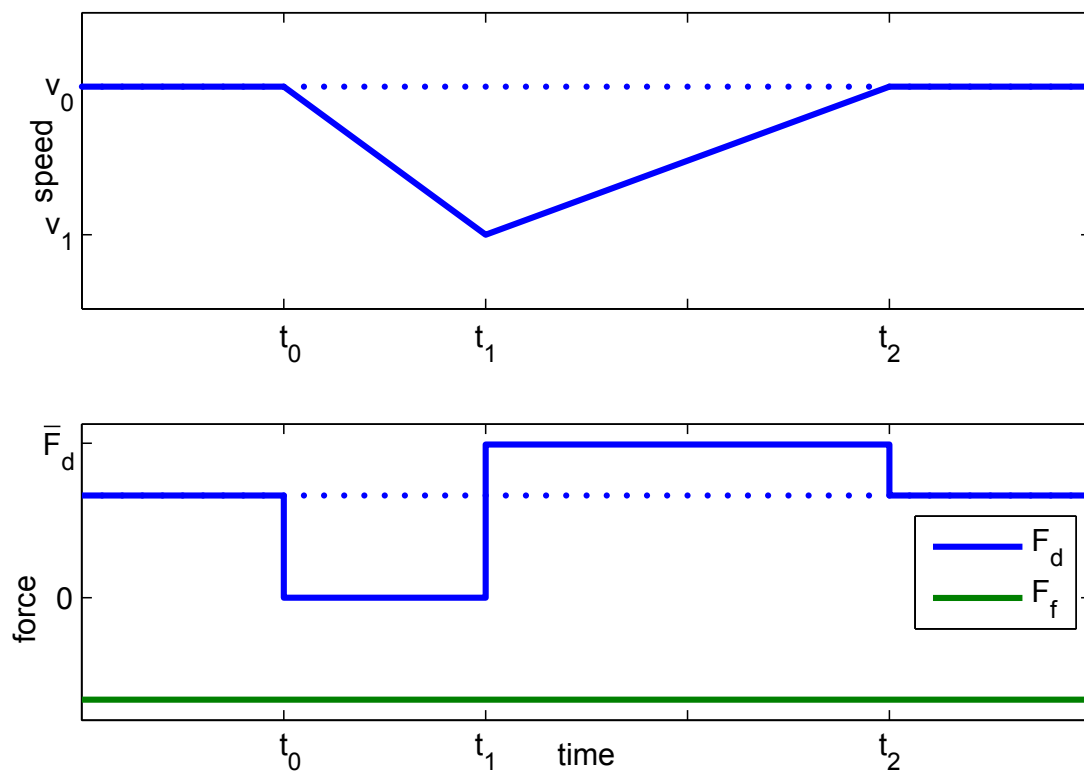
Figure A1. Open driveline behavior for timeloss (Δt_t) calculation. From t_0 to t_1, no traction F_d is available, causing the vehicle to decelerate. From t_1 to t_2, speed recovery takes place at maximum traction $\overline{F_d}$. Arrival at position $s_2(t_2)$ is later than without switch event (dotted line), which is the defined timeloss.

Table A2. Equivalent cost of time loss due to a switch event of 1 s.

Segment	Timeloss [s]	Speedloss [m/s]	Eq. Cost [kJ]
0%	−0.01	−0.075	0.17
6%	−0.07	−0.65	1.8
12%	−0.22	−1.25	5.6

Appendix D. Simulation Results of Multi-Level Iteration

For the simulation on the short drive cycle, multiple iterations are performed in the multi-level EMS of this work. In this section, the decisions of the first two iterations are analyzed, illustrating the fast convergence of the algorithm.

Appendix D.1. Iteration 0

In the zeroth iteration, each level is optimized consecutively, starting at Level 3. The assumptions in each Level are described:

- Level 3: on this level, we assume that the vehicle is cruising constantly at the maximum speed v_0, i.e., 25 km/h. The elevation is given by Figure 16. Assuming an average, constant engine speed (here: 1100 rpm), P_d is calculated.
- Level 2: on this level, charge sustenance is enforced ($E_b(t_0) = E_b(t_f)$) by searching for λ_b^*. The capabilities of the CE and MG at 1100 rpm influence the resulting power-split and mode switching.
- Level 1: on this level, the cost of switching is added to the power-split optimization, using the just found λ_b, P_d and v. The resulting dynamic program is solved for the full horizon, here 10.6 km.

In Figure A2, the result is shown for respectively MG power (P_m), mode (M), gear (G), battery energy (E_b) and vehicle speed (v). The top plot also includes P_d (dotted black) as a reference. On the flat segments, $P_d = 11$ kW and on the slopes $P_d = [113, -192, 215, -90]$ kW for, respectively, the +6%, -12%, +12% and -6% segment. The dashed lines indicate the decisions of Level 2, the solid lines of Level 1.
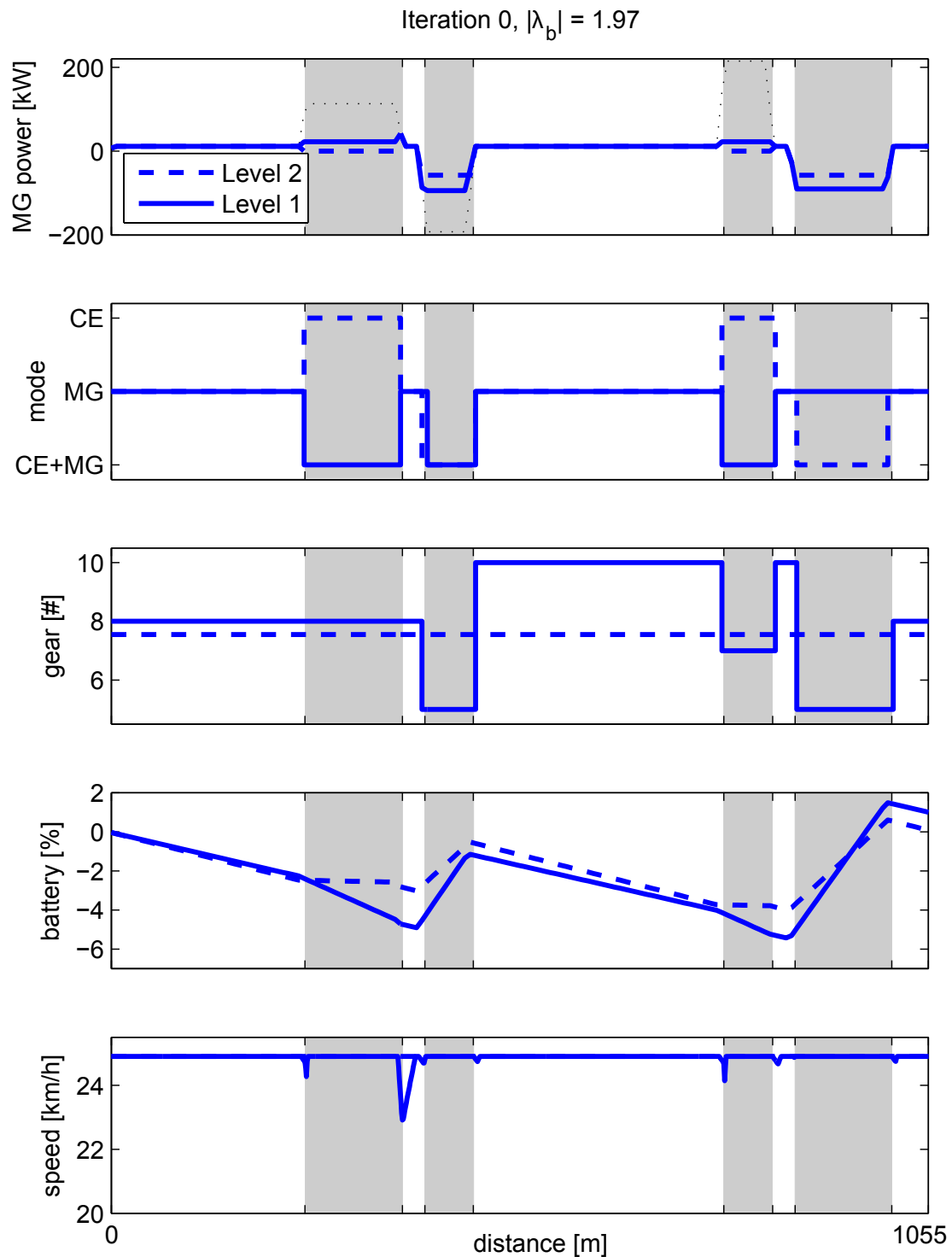

Figure A2. Decisions and results from Level 2 and 1, after the zeroth iteration. Due to the different model abstraction, the solution on Level 1 is not exactly charge sustaining anymore.

Level 2 (dashed line) calculates a charge sustaining $\lambda_b = -1.97$, shown in subplot 4, where E_b starts and ends at 0. It uses a simplified model which keeps the engine speed at 1100 rpm, showing in subplot 3 as a selected gear between 7 and 8. At this engine speed, it is optimal to drive the flat segments in MG mode (M_m), with the CE switched off. The uphill segments have higher power demands P_d, and the CE is used. On the downhill segments, brake energy is recuperated with the MG, but the CE must assist in braking, as $|P_d|$ is larger than the MG can recuperate. As all switches are assumed to be infinitely fast, the bottom plot shows no speed deviation for Level 2.

Level 1 (solid line) uses $\lambda_b = -1.97$ and the switching costs ϵ to determine the optimal mode and gear switching sequence. Most importantly, the gear on the downhill segments is shifted down to 5, in order to maximize the recuperation power of the MG (see Figure 3), which is notable by an increased rise in energy of the battery for these segments. Due to the downshift, the -6% can be driven with the MG alone. The -12% still needs both MG and CE for additional braking power. The uphill segments show a different mode selection than for Level 2. From the control map, we know that $\lambda_b = -1.97$ is close to the guard G_1, where the Hamiltonian of M_e is only slightly lower than of M_c, causing Level 2 to decide for M_e. Level 1, however, takes the switching costs into account, and where both levels decide to start and stop the CE, Level 1 decides that stopping and starting the MG is more expensive than the minor fuel advantage of the lower Hamiltonian, resulting in mode M_c with 22 kW boosting of the MG. Gear selection on the flat segments is towards the highest possible gear where the MG is able to deliver the demanded power, thereby minimizing the friction of the MG.

Due to the switching of mode and gear, several open driveline events occur. The cost of timeloss, as calculated in Section 4.3, determines that, if a switch is needed, it should be done where the timeloss is minimal. For this cycle, this is clearly on the flat segments, not on any of the slopes. This results in only minor vehicle speed deviations in the vehicle speed.

Due to the different control actions between Level 1 and Level 2, charge sustenance is not enforced anymore. This is to be corrected in the next iteration.

Appendix D.2. Iteration 1

In the first iteration, each control level is optimized consecutively, using the results from the zeroth iteration. The differences with the zeroth iteration are:

- Level 3: the constant engine speed from iteration 0 is replaced by the speed resulting from the gear switching sequence. As speed is not explicitly optimized and can always be maintained in this example, P_d is not altered.
- Level 2: using the gear switching sequence, a new charge sustaining λ_b is calculated.
- Level 1: with the adjusted λ_b, the power-split and switching sequence are optimized

In Figure A3, the results are shown for Level 2 and Level 1 for the first iteration. It is evident that no significant differences are present between the two levels. An additional gear shift occurs on the top of the first hill, but, as the friction of the MG at different speeds is very small, the fuel advantage is negligible. Comparing to iteration 0, the difference with Level 2 of iteration 0 is also very small. The main effect of the iteration is that the cycle is charge sustaining again, with $\lambda_b = -1.95$. To realize that, the boosting power of the MG is slightly increased from 22 kW to 29 kW on the uphill slopes.
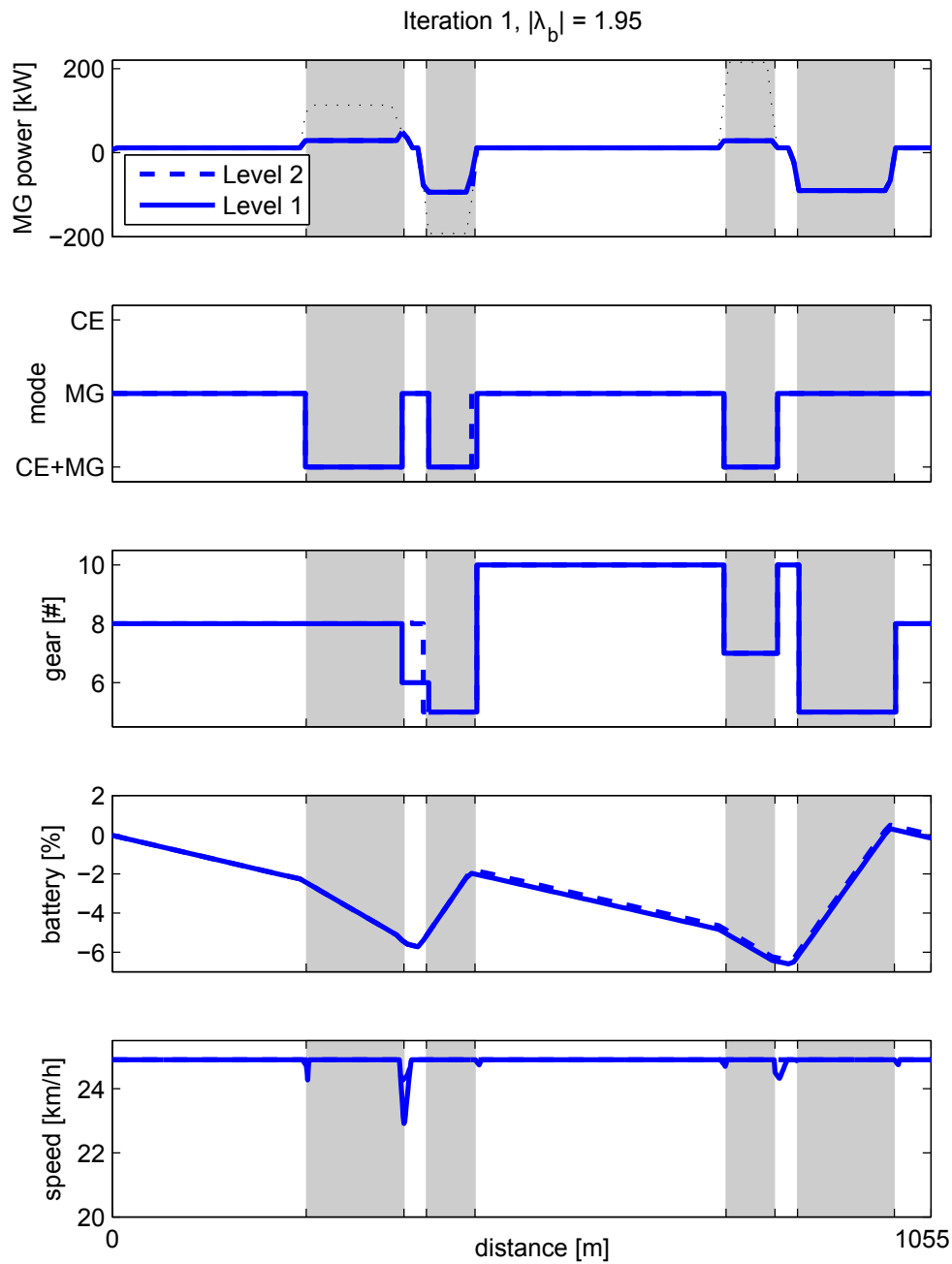
Figure A3. Decisions and results from Level 2 and 1, after the first iteration. After updating the information on λ_b and ω, the solution on both levels converges, and are charge sustaining.

Appendix E. Simulation Results of Multi-Level MPC

In Figure A4, the results are shown for Level 2 and Level 1 for the first iteration. As Level 2 is not changed, the Level 2 traces are the same as in Figure A2. Albeit Level 1 now has a horizon of 5 s, instead of 10 km, the mode decisions are unaltered. However, the gear decisions differ in some segments:

- the trace starts at G_5 and is maintained for a long time. That gear is selected at the end of the -6% slope to maximize brake energy recuperation. After the slope, P_d is low, and a little friction reduction of the MG could be accomplished with an upshift. However, the fuel improvement of switching in MG is an order of magnitude lower than in other modes. Now, for a gear switch to occur, the accumulated fuel improvement over the horizon (5 s) must be higher than the cost of the switch event itself, which it is not, for this short horizon, and the switch will not occur.

- on the +6%, a higher gear G_9 is needed, in order to minimize CE friction losses. Before the +6% slope, this decision enters the horizon, and, as the switch will be made anyhow, the algorithm decides to do the switch immediately, as driving MG in G_9 is slightly more profitable.
- on the top of the first hill, a short shift to G_{10} is performed, simultaneously with a mode switch, which makes the gear switch event cheap. The decision has, however, a marginal impact on the fuel consumption.
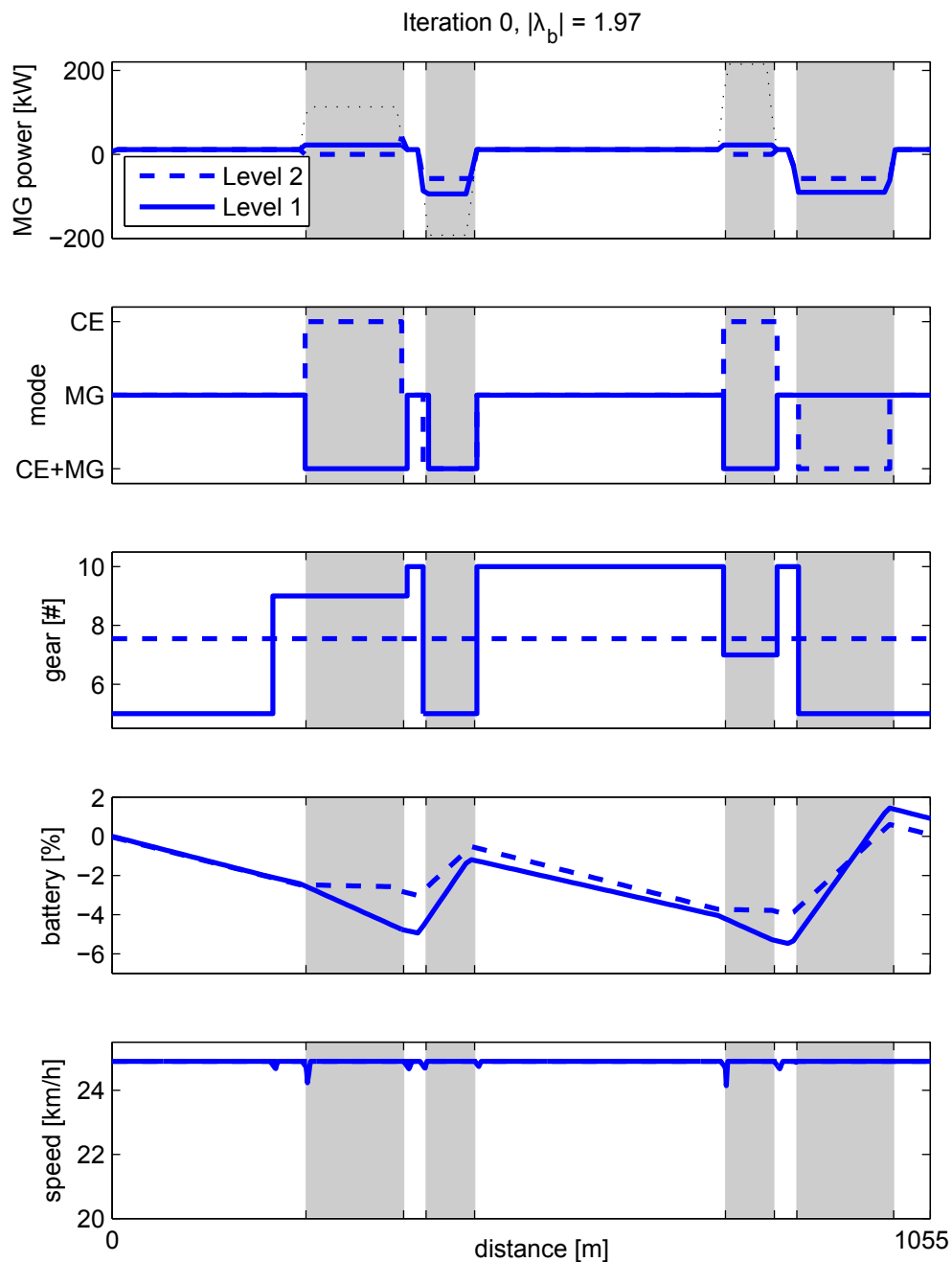

Figure A4. Results for MPC, with a 5-sample horizon on Level 1. Differences with Figure A2 are mainly in the gear decisions, with a marginal impact on fuel.

References

1. Sciarretta, A.; Guzzella, L. Control of hybrid electric vehicles. *IEEE Control Syst. Mag.* **2007**, *27*, 60–70.
2. Serrao, L. Open Issues in Supervisory Control of Hybrid Electric Vehicles: A unified approach using optimal control methods. In Proceedings of the RHEVE Hybrid and Electric Vehicles, Paris, France, 6–7 December 2011.
3. Willems, F.; Spronkmans, S.; Kessels, J. Integrated Powertrain Control to meet low CO_2 emissions for a hybrid distribution truck with SCR-deNO_x system. In Proceedings of the Dynamic Systems and Control Conference, Arlington, VA, USA, 31 October–2 November 2011.
4. Pham, T.; Kessels, J.; van den Bosch, P.; Huisman, R. Integrated Energy and Thermal Management for Hybrid Electric Heavy Duty Trucks. In Proceedings of the IEEE Vehicle Power and Propulsion Conference, Seoul, Korea, 9–12 October 2012.
5. van Reeven, V.; Huisman, R.; Kessels, J.; Pham, T.; Hofman, T. Integrating energy and thermal management of hybrid trucks. In Proceedings of the VDI Commercial Vehicles, Celle, Germany, 5–6 June 2013.
6. Donkers, M.; Schijndel, J.V.; Heemels, W.; Willems, F. Optimal control for integrated emission management in diesel engines. *Control Eng. Pract.* **2017**, *61*, 206–216. [CrossRef]
7. Willems, F.; Kupper, F.; Rascanu, G.; Feru, E. Integrated Energy and Emission Management for Diesel Engines with Waste Heat Recovery using Dynamic Models. *Oil Gas Sci. Technol.* **2015**, *70*, 143–158. [CrossRef]
8. van Reeven, V.; Hofman, T.; Willems, F.; Huisman, R.; Steinbuch, M. Optimal control of engine warmup in hybrid vehicles. *Oil Gas Sci. Technol.* **2014**, *71*, 14. [CrossRef]
9. Merz, F.; Sciarretta, A.; Dabadie, J.C.; Serrao, L. On the optimal thermal management of hybrid-electric vehicles with heat recovery sytems. *Oil Gas Sci. Technol.* **2012**, *67*, 601–612. [CrossRef]
10. Back, M. Pradiktive Antriebsregelung zum Energieoptimalen Betrieb von Hybridfahrzeugen. Ph.D. Thesis, Universitat Fridericiana Karlsruhe, Karlsruhe, Germany, 2005.
11. Johannesson, L.; Asbogard, M.; Egardt, B. Assessing the Potential of Predictive Control for Hybrid Vehicle Powertrains Using Stochastic Dynamic Programming. *IEEE Trans. Intell. Transp. Syst.* **2007**, *8*, 71–83. [CrossRef]
12. Delprat, S.; Guerra, T.; Paganelli, G.; Lauber, J.; Delhom, M. Control strategy optimization for a hybrid parallel powertrain. In Proceedings of the 2001 American Control Conference, Arlington, VA, USA, 25–27 June 2001.
13. Finkeldei, E.; Back, M. Implementing an MPC algorithm in a vehicle with a hybrid powertrain using telematics as a sensor for powertrain control. In Proceedings of the IFAC Symposium on Advances in Automotive Control, Arlington, VA, USA, 25–27 June 2004.
14. Borhan, H.; Vahidi, A.; Phillips, A.; Kuang, M.; Kolmanovsky, I.; Cairano, S.D. MPC-Based energy Management of a power-split hybrid electric vehicle. *IEEE Trans. Control Syst. Technol.* **2012**, *20*, 593–603. [CrossRef]
15. Ambühl, D.; Sundström, O.; Sciarretta, A.; Guzzella, L. Explicit optimal control policy and its practical application for hybrid electric powertrains. *Control Eng. Pract.* **2010**, *18*, 1429–1439. [CrossRef]
16. Murgovski, N.; Johannesson, L.M.; Sjöberg, J. Engine on/off control for dimensioning hybrid electric powertrains via convex optimization. *IEEE Trans. Veh. Technol.* **2013**, *62*, 2949–2962. [CrossRef]
17. Elbert, P. Noncausal and Causal Optimization Strategies for Hybrid Electric Vehicles. Ph.D. Thesis, ETH Zurich, Zurich, Switzerland, 2013.
18. Ngo, V.; Hofman, T.; Steinbuch, M.; Serrarens, A. Optimal Control of the Gear Shift Command for Hybrid Electric Vehicles. *IEEE Trans. Veh. Technol.* **2012**, *61*, 3531–3543. [CrossRef]
19. Nüesch, T.; Elbert, P.; Flankl, M.; Onder, C.; Guzzella, L. Convex optimization for the energy management of hybrid electric vehicles considering engine start and gearshift costs. *Energies* **2014**, *7*, 834–856. [CrossRef]
20. van Reeven, V.; Hofman, T.; Huisman, R.; Steinbuch, M. Extending Energy Management in Hybrid Electric Vehicles with Explicit Control of Gear Shifting and Start-Stop. In Proceedings of the American Control Conference, Montreal, QC, Canada, 27–29 June 2012.
21. Scattolini, R. Architectures for distributed and hierarchical model predictive control. *J. Process Control* **2009**, *19*, 723–731. [CrossRef]
22. Ambühl, D.; Guzzella, L. Predictive Reference Signal Generator for Hybrid Electric Vehicles. *IEEE Trans. Veh. Technol.* **2009**, *58*, 4730–4740. [CrossRef]
23. van Keulen, T.; Jager, B.d.; Serrarens, A.; Steinbuch, M. Optimal Energy Management in Hybrid Electric Trucks Using Route Information. *Oil Gas Sci. Technol. Rev. IFP* **2010**, *65*, 103–113. [CrossRef]

24. Sun, C.; Moura, S.J.; Hu, X.; Hedrick, J.K.; Sun, F. Dynamic Traffic Feedback Data Enabled Energy Management in Plug-in Hybrid Electric Vehicles. *IEEE Trans. Control Syst. Technol.* **2015**, *23*, 1075–1086.

25. Mesarovic, M.D.; Macko, D.; Takahara, Y. *Theory of Hierarchical Multilevel Systems*; Academic Press: Cambridge, MA, USA, 1970.

26. Maciejowski, J. *Predictive Control with Constraints*; Prentice Hall: Upper Saddle River, NJ, USA, 2002.

27. Findeisen, W. *Control and Coordination in Hierarchical Systems*; John Wiley & Sons: Hoboken, NJ, USA, 1980.

28. Van Reeven, V.; Hofman, T. Multi-level energy management: Implementation and validation—Part II. *Vehicles* **2018**, under review.

29. Guzzella, L.; Onder, C. *Introduction to Modeling and Control of Internal Combustion Engine Systems*; Springer: Berlin, Germany, 2010.

30. Bialas, W.F.; Karwan, M.H. *Multilevel Linear Programming*; State University of New York: Buffalo, NY, USA, 1978; p. 78-1.

31. Kirk, D.E. *Optimal Control Theory, an Introduction*; Dover Publications: Chicago, IL, USA, 1970.

32. van Keulen, T.; Gillot, J.; de Jager, B.; Steinbuch, M. Solution for state constrained optimal control problems applied to power split control for hybrid vehicles. *Automatica* **2014**, *50*, 187–192. [CrossRef]

33. Wei, X.; Utkin, V.; Rizzoni, G.; Guzzella, L. Model-based Fuel Optimal Control of Hybrid Electric Vehicle using Variable Structure Control Systems. *J. Dyn. Syst. Meas. Control* **2007**, *129*, 13–20. [CrossRef]

34. Delprat, S.; Hofman, T. Hybrid Vehicle Optimal Control: Linear Interpolation and Singular Control. In Proceedings of the IEEE Vehicle Power and Propulsion Conference, Coimbra, Portugal, 27–30 October 2014.

35. Fröberg, A.; Nielsen, L. *Optimal Control Utilizing Analytical Solution for Heavy Truck Cruise Control*; Internal Report; Linkopings Universitet: Linkoping, Sweden, 2008.

36. Bryson, A.E.; Ho, Y. *Applied Optimal Control*; Taylor and Francis: Abingdon, UK, 1975.

37. Hartl, R.F.; Sethi, S.P.; Vickson, R.G. A survey of the maximum principle for optimal control problems with state constraints. *SIAM* **1995**, *37*, 181–218. [CrossRef]

38. Sciarretta, A.; Back, M.; Guzzella, L. Optimal control of parallel hybrid electric vehicles. *IEEE Trans. Control Syst. Technol.* **2004**, *12*, 352–363. [CrossRef]

39. van Keulen, T.; de Jager, B.; Foster, D.; Steinbuch, M. Velocity Trajectory Optimization in Hybrid Electric Trucks. In Proceedings of the American Control Conference, Baltimore, MD, USA, 30 June–2 July 2010.

40. Hellström, E.; Åslund, J.; Nielsen, L. Horizon length and fuel equivalents for fuel-optimal look-ahead control. In Proceedings of the Advances in Automotive Control, Munich, Germany, 12–14 July 2010.

41. Ngo, V. Gear Shift Strategies for Automotive Transmissions. Ph.D. Thesis, Technical University Eindhoven, Eindhoven, The Netherlands, 2012.