Robust Modeling for Optimal Control of Parallel Hybrids With Dynamic Programming

Robust Modeling for Optimal Control of Parallel Hybrids With Dynamic Programming

Federico Miretti, Daniela Misul
CARS@Polito - Center for Automotive Research and Sustainable Mobility,
Politecnico di Torino, C.so Ferrucci 112, Torino, TO 10138, Italy
Email: federico.miretti@polito.it

***Abstract*—The aim of this work is to provide insight and guidelines for engineers and researchers when developing hybrid powertrain models to be employed in a dynamic programming optimal control algorithm. In particular, we focus on the advantages and disadvantages of the various control sets that can be used to characterize the power flow (e.g. the engine torque or a torque-split coefficient).**

Dynamic programming is the reference optimal control technique for hybrid electric vehicles. However, its practical implementation is not exempt from numerical issues which may hamper its accuracy. Amongst these, some are directly related to the different modeling choices that can be made when defining the system dynamics of the powertrain.

To treat these issues, we first define four relevant evaluation criteria: control bounds definition, numerical efficiency, model complexity and interpretability. Then, we introduce eight different control sets and we discuss and compare them in light of these criteria. This discussion is supported by an extensive set of numerical experiments on a p2 parallel hybrid. Finally, we revisit our analysis and simulation results to draw modeling recommendations.

***Index Terms*—Dynamic programming, optimal control, energy management strategy, Hybrid Electric Vehicles (HEV)**

I. Introduction

Dynamic programming is the most prominent technique used for off-line optimal control of hybrid-electric vehicles [1], [2]. Its applications range from benchmarking against sub-optimal control strategies [3], [4], optimal design and efficiency analysis of HEV powertrains [5], rule extraction for heuristic control strategies [6], [7], and generating datasets for training machine-learning based control strategies [8].

For all these applications, it is of utmost importance to ensure that dynamic programming is correctly applied to ensure accuracy of the obtained solution. Unfortunately, the technique suffers from potential implementation issues which must be accounted for. While some of these issues can be handled by improving the algorithm design [9], other issues require careful consideration in preparing the system dynamical model and defining the optimal control problem of interest.

Furthermore, some of the applications listed at the beginning of this section also require fast computation of a large number of optimal control problems, making numerical efficiency another relevant issue. Once again, while computational efficiency can be improved by algorithm design [10], modeling choices also have a big impact.

In this paper, we focus on the impact of several modeling choices for a parallel hybrid vehicle on the accuracy and numerical efficiency of the optimal EMS (*Energy Management Strategy*) obtained with dynamic programming. The goal of optimal EMS design is to minimize an objective functional; typically, either fuel consumption or a trade-off between this and some other cost such as pollutants emissions [11] or battery degradation [12].

In general, a parallel hybrid modeled with a quasi-static approach has two main quantities to be controlled by the EMS: the gear number engaged by the gearbox (which may possibly be defined by a fixed gear schedule, therefore not taking part in the optimal control problem definition) and the power flow between the thermal engine and the electrical part of the powertrain. While the definition of a control variable for the gear number is straightforward, the power flow can be characterized in many different ways.

Indeed, various different models can be found in the literature for this type of architecture. The most popular choices appear to be the engine torque [11], [13]–[16] or power [17], the e-machine torque [18], the e-machine power [3], [19], the battery power [20], an engine torque-split factor, or an e-machine torque-split factor [21].

In this work, we develop eight different models for the same p2 architecture corresponding to eight different control sets and we analyze their advantages and disadvantages. The main relevant parameters of the analyzed powertrain are reported in Table I. We then test our hypotheses with numerical experiments, and we conclude this paper with a set of recommendations that can be drawn from the evidence.

II. The simulation models

The scope of this comparison is to compare different modeling choices that can be adopted to describe the power flow in a parallel hybrid.

The eight models characterize the power flow with the following control variables:

A) The engine torque T_{eng}.
B) The e-machine torque T_{em}.
C) The battery current i_{b}.
D) The normalized engine torque τ_{eng}.
E) The normalized e-machine torque τ_{em}.
F) The normalized battery current $\imath_{\mathrm{b}}$.
G) The engine torque-split factor α_{eng}.
H) The e-machine torque-split factor α_{em}.

TABLE I
MAIN VEHICLE DATA.

Component	Parameter	Value
Vehicle	Mass	1175 kg
	First coast-down coefficient	150 N
	Second coast-down coefficient	2.24 N/(m · s)
	Third coast-down coefficient	0.44 N/(m · s)^2
	Tyre radius	0.3 m
Engine	Displacement	1.1 l
	Rated power	68 kW
	Maximum torque	130 Nm
E-machine	Rated power	30 kW
	Maximum torque	150 Nm
Battery	Type	Li-ion
	Nominal capacity	5.4 Ah
	Nominal voltage	204 V

Choosing one of these control sets in turn introduces changes in the powertrain model. First, all of these models require different constraints on the powertrain components. Second, models based on the battery current, i.e. models C and F, use different equations for the battery model.

All models, however, share a common sub-model to evaluate a tractive effort.

A. Common path

All models use a longitudinal vehicle model to evaluate a tractive effort F_{veh} as a function of the vehicle speed v_{veh} and a quasi-static powertrain model [22] to consequently evaluate a torque demand. This torque demand is then split between the engine and the e-machine based on the power flow control variable.

Omitting the various driveling efficiencies for ease of notation, the torque demand is evaluated as:

$$T_{\mathrm{d}} = \frac{F_{\mathrm{veh}}(v_{\mathrm{veh}}) r_{\mathrm{wh}}}{\tau_{\mathrm{fd}} \tau_{\mathrm{gb}}(\gamma)}, \tag{1}$$

where r_{wh} is the wheels' radius, τ_{fd} and τ_{gb} are the final drive and gearbox speed ratios, and γ is the gear number.

The battery model is an internal resistance equivalent circuit model, where the relationship between the battery power P_{b} and the battery current i_b depends on the open-circuit voltage and internal resistance characteristics $v_{\mathrm{oc}}(\sigma)$ and $R_0(\sigma)$ as follows:

$$P_{\mathrm{b}} = v_b i_b = \left(v_{\mathrm{oc}}(\sigma) + R_0(\sigma) i_{\mathrm{b}}\right) i_b. \tag{2}$$

The battery SOC (σ) is the only state variable for all models.

III. CONTROL SETS

In this section, we define the control sets and we discuss their individual advantages and disadvantages.

The normalized control sets, i.e. sets D to F, are different from models A to C in that the control variables are normalized by their maximum values, as imposed by the operational limits of the corresponding components. The torque-split factors G and H are defined as the ratio between the engine or e-machine torque and the torque demand given by (1).

A. Engine torque

This model has no particular advantages, if not for the fact that no derived quantity needs to be defined.

This control set obviously has a lower bound at $T_{\mathrm{eng}} = 0$, and an upper bound at the engine maximum torque. However, the maximum torque is strongly speed-dependent; therefore, the upper bound for the control set must be set to the absolute maximum engine torque, which is available at some speed ω^*_{eng}.

Then, a constraint must be set on the engine torque, i.e. $T_{\mathrm{eng}} \leq T_{\mathrm{eng,max}}(\omega_{\mathrm{eng}})$. At all times where the engine speed is different from this ω^*_{eng}, we are wasting computations on unfeasible controls.

After setting the engine torque, the e-machine torque and power and, subsequently, the battery power can be computed. The battery current must then be evaluated by solving (2), which is quadratic in i_{b}:

$$i_{\mathrm{b}} = \frac{v_{\mathrm{oc}} + \sqrt{v_{\mathrm{oc}}^2 - 4 R_0 P_{\mathrm{b}}}}{2 R_0}. \tag{3}$$

This is the most computationally expensive equation in the whole powertrain model.

Let us now consider regenerative braking. With this model, there is no way to directly control the amount of torque demand that gets absorbed by the e-machine to charge the battery. This operating mode is implicitly defined by setting T_{eng} to zero and then letting all of the torque demand be absorbed by the e-machine, up to its generator mode torque limit $T_{\mathrm{em,min}}(\omega_{\mathrm{em}})$.

B. E-machine torque

This model is analogous to the previous one, and it shares a similar issue in that the limit torque is speed-dependent; but since the e-machine has two limit torque curves (for generator and motor mode), the issue affects both the lower and the upper bound for the control set. Regenerative braking is similarly handled by saturating the e-machine torque to its generator mode torque limit.

C. Battery current

This model is different from the previous two in that it essentially inverts the dependent and independent variables in the battery model. In particular, directly controlling the current means that the battery model no longer requires solving a quadratic equation; rather, the battery power is directly computed from (2) and then translated into the e-machine torque which in turn determines the engine torque. Thus, the most expensive computation of the powertrain model is avoided, possibly resulting in a faster-running model.

On the other hand, handling regenerative braking is more troublesome. In order to use a similar logic as the previous models without using (3) inside the optimization algorithm, we pre-calculated a minimum current $\tilde{i}_{\mathrm{b,min}}$ that incorporates the e-machine torque limits:

$$\tilde{i}_{\mathrm{b,min}} = \max \left[i_{\mathrm{b,min}}, \frac{v_{\mathrm{oc}} - \sqrt{v_{\mathrm{oc}}^2 - 4 R_0 \tilde{P}_{\mathrm{b}}}}{2 R_0} \right], \tag{4}$$

where

$$\tilde{P}_{\mathrm{b}} = \eta_{\mathrm{em}}\left(\omega_{\mathrm{em}}, T_{\mathrm{em,min}}(\omega_{\mathrm{em}})\right)\omega_{\mathrm{em}} T_{\mathrm{em,min}}(\omega_{\mathrm{em}}), \tag{5}$$

and used this to saturate the battery current. Note that this $\tilde{i}_{\mathrm{b,min}}$ is a function of both σ and ω_{em}.

D. Normalized engine torque

The normalized engine torque is defined as

$$\tau_{\mathrm{eng}} = \frac{T_{\mathrm{eng}}}{T_{\mathrm{eng,max}}(\omega_{\mathrm{eng}})}. \tag{6}$$

This model prevents discarding controls because the engine torque exceeds its limit torque and the corresponding constraint $T_{\mathrm{eng}} \leq T_{\mathrm{eng,max}}(\omega_{\mathrm{eng}})$ is therefore not needed. The bounds of the control set are naturally set to $\alpha_{\mathrm{eng}} \in [0, 1]$.

E. Normalized e-machine torque

The normalized e-machine torque is defined as

$$\alpha_{\mathrm{em}} = \begin{cases} \frac{T_{\mathrm{em}}}{T_{\mathrm{em,max}}(\omega_{\mathrm{em}})} & \text{if } T_{\mathrm{em}} \geq 0, \\ \frac{T_{\mathrm{em}}}{T_{\mathrm{em,min}}(\omega_{\mathrm{em}})} & \text{if } T_{\mathrm{em}} < 0, \end{cases} \tag{7}$$

where $T_{\mathrm{em,max}}$ and $T_{\mathrm{em,min}}$ are the maximum e-machine torque in motor and generator mode respectively.

This model prevents discarding controls because the em-machine torque exceeds its operating torque range and the corresponding constraint $T_{\mathrm{em,min}}(\omega_{\mathrm{em}}) \leq T_{\mathrm{em}} \leq T_{\mathrm{em,max}}(\omega_{\mathrm{em}})$ is therefore not needed. The bounds of the control set are naturally set to $\alpha_{\mathrm{em}} \in [-1, 1]$.

F. Normalized battery current

The normalized battery current is defined as

$$\iota_{\mathrm{b}} = \begin{cases} \frac{i_{\mathrm{b}}}{i_{\mathrm{b,max}}} & \text{if } i_{\mathrm{b}} \geq 0, \\ \frac{i_{\mathrm{b}}}{i_{\mathrm{b,min}}} & \text{if } i_{\mathrm{b}} < 0, \end{cases} \tag{8}$$

where $i_{\mathrm{b,max}}$ and $i_{\mathrm{b,min}}$ are the maximum charge and discharge battery current respectively.

In some cases, the limit current may be determined by thermal or aging aspects, and can generally be set as a constant. In some other cases, the limiting factor may be the battery voltage limits. Then, if the open-circuit voltage and internal resistance are SOC-dependent, so are the limit currents:

$$i_{\mathrm{b,max}}(\sigma) = \frac{v_{\mathrm{oc}}(\sigma) - v_{\mathrm{b,min}}}{R_0(\sigma)}, \tag{9}$$

$$i_{\mathrm{b,min}}(\sigma) = \frac{v_{\mathrm{oc}}(\sigma) - v_{\mathrm{b,max}}}{R_0(\sigma)}. \tag{10}$$

In these cases, this normalized version of model C prevents discarding controls because the battery current exceeds the allowable current and the corresponding constraint $i_{\mathrm{b,min}} \leq i_{\mathrm{b}} \leq i_{\mathrm{b,max}}$ is therefore not needed. The bounds of the control set are naturally set to $\iota_{\mathrm{b}} \in [-1, 1]$.

The battery model is simplified for the same reason of the battery current model C.

G. Engine torque-split factor

The engine torque-split factor is defined as the ratio between the engine torque and the torque demand:

$$\alpha_{\mathrm{eng}} = \frac{T_{\mathrm{eng}}}{T_{\mathrm{d}}}. \tag{11}$$

The main advantage of the engine torque-split factor is in its interpretability, in that any value for α_{eng} can be attributed to one of the HEV operating modes as shown in Table II.

One issue with this control set is while there is an obvious lower bound for α_{eng} (i.e. $\alpha_{\mathrm{eng}} = 0$), there is no obvious upper bound:

- at times when T_{d} is small compared to $T_{\mathrm{eng,max}}$, a large upper bound would be needed to enable using the engine up to its full power to recharge the battery and a coarse discretization would suffice;
- when T_{d} is close to $T_{\mathrm{eng,max}}$, a small upper bound would be enough but a finer discretization would be needed;
- if T_{d} is larger than $T_{\mathrm{eng,max}}$, an even smaller upper bound (smaller than 1) would suffice.

As a result, this control set generally leads to wasting computations on unfeasible controls whenever the torque demand is relatively high and to an arbitrarily restricted control set for battery charging when the torque demand is relatively low.

TABLE II
ENGINE TORQUE-SPLIT FACTOR AND HEV OPERATING MODE.

value	operating mode
$\alpha_{\mathrm{eng}} = 0$	pure electric
$0 < \alpha_{\mathrm{eng}} < 1$	torque-split
$\alpha_{\mathrm{eng}} = 1$	pure thermal
$\alpha_{\mathrm{eng}} > 1$	battery charging

H. E-machine torque-split factor

The e-machine torque-split factor is defined as the ratio between the e-machine torque and the torque demand:

$$\alpha_{\mathrm{em}} = \frac{T_{\mathrm{em}}}{T_{\mathrm{d}}}. \tag{12}$$

Similarly to the engine-torque split factor, this control set has an unambiguous relation to the HEV operating modes, as shown in Table III; furthermore, for similar reasons, there is no obvious lower bound.

TABLE III
E-MACHINE TORQUE-SPLIT FACTOR AND HEV OPERATING MODE.

value	operating mode
$\alpha_{\mathrm{em}} < 0$	battery charging
$\alpha_{\mathrm{em}} = 0$	pure thermal
$0 < \alpha_{\mathrm{em}} < 1$	torque-split
$\alpha_{\mathrm{em}} = 1$	pure electric

TABLE IV
ACCURACY AND SIMULATION TIME OF THE EXAMINED MODELS.

	Model	$m_{\mathrm{PF}} = 11$			$m_{\mathrm{PF}} = 21$			$m_{\mathrm{PF}} = 41$		
		Δm_{f}	$\psi(\sigma_N)$	t_{sim} (s)	Δm_{f}	$\psi(\sigma_N)$	t_{sim} (s)	Δm_{f}	$\psi(\sigma_N)$	t_{sim} (s)
A	T_{eng}	1.94 %	9.2e-04	2.5	1.26 %	1.3e-03	2.6	0.74 %	1.4e-03	2.9
B	T_{em}	13.15 %	1.0e-02	2.8	2.15 %	4.9e-03	2.8	1.48 %	5.7e-03	3.0
C	i_{b}	3.53 %	3.5e-03	2.8	1.99 %	3.2e-03	3.3	1.17 %	5.2e-03	3.3
D	τ_{eng}	1.95 %	1.3e-03	2.6	1.25 %	1.2e-03	2.6	0.71 %	1.2e-03	2.9
E	τ_{em}	2.28 %	4.3e-03	2.6	1.52 %	5.1e-03	2.6	0.95 %	4.7e-03	2.9
F	ι_{b}	3.53 %	4.4e-03	3.0	2.01 %	3.8e-03	3.1	1.09 %	3.2e-03	3.3
G	α_{eng}	1.07 %	4.3e-03	2.8	0.61 %	4.4e-03	2.6	0.32 %	3.2e-03	2.8
H	α_{em}	0.91 %	1.1e-03	2.5	0.45 %	1.0e-03	2.7	0.33 %	1.0e-03	2.8

IV. EVALUATION CRITERIA

We are now going to evaluate the advantages and disadvantages of the listed models with numerical experiments. In particular, we are looking at four main characteristics.

Control set bounds definition. To apply dynamic programming, we need to discretize the continuous control variables and we need to define a lower and an upper bound. For some control sets, the bounds have an obvious definition; for others, they do not. We can refer to the former as *well-posed* control set bounds.

Numerical efficiency. For each simulation time interval, we exclude all controls that violate the constraints that we set on the powertrain components. These controls do not contribute to the value function update. The more controls we need to exclude because of our formulation of the constraints, the less numerically efficient our model is.

Model complexity. The models which directly control the battery current use simpler equations for the battery model, which may translate into reduced simulation time.

Interpretability. As automotive engineers, we would like to have a direct correspondence between the value taken by our power flow control variable and the HEV operating modes (i.e. pure electric, torque-split, pure thermal and battery charging)[1]. This may also be very relevant if the results are to be used by some rule-extraction algorithm to obtain an heuristic strategy or to train a machine-learning based strategy.

V. SIMULATION RESULTS

To test the ideas discussed in the previous section, we implemented all the listed models and applied dynamic programming with DynaProg[2], a dedicated MATLAB toolbox [10] in order to find the fuel-optimal EMS. We thus compared the control sets in terms of accuracy (which is a product of their numerical efficiency and of the well-posedness of their bounds) and computational time (which is a product of the model complexity).

For all simulations, the SOC grid was defined as ranging from 0.4 to 0.7 with a discretization step of 0.003[3]. The computational grid for power flow control variable was defined by a number m_{PF} of 11, 21 and 41 quantized values in three sets of experiments. Therefore, all simulations had the same number of function evaluation within each set. The gear number was set by a simple gear shift schedule as a function of the vehicle speed, to ensure that all models deal with the same torque demand.

Model accuracy was measured in terms of the model's ability to achieve the true optimal[4] fuel consumption while reaching the terminal SOC of $\sigma = 0.6$. Therefore, it is reported in Table IV and in Fig. 1 for all tested models in terms of two quantities: the difference between the fuel consumption and the true optimal fuel consumption Δm_{f} and the endpoint error $\psi(\sigma_N)$ (i.e. the difference between the terminal SOC and the desired value). A third column reports simulation time (t_{sim}).

The simulation results shown in Table IV raise many points which are worth discussing. Firstly, the torque-split models appear to be the most robust in that they allowed to find a feasible solution with accuracy within 1% even with a coarse discretization grid.

Their good performance is probably explained by the fact that they can accurately reproduce both pure thermal and pure electric modes, thus allowing for stable operation.

Looking at accuracy, as the control set discretization is refined all models tend to the same solution, both in terms of fuel consumption and the state trajectory. Furthermore, inspecting the optimal solution, we also note that it includes some lengthy portions where the optimal operating mode is pure thermal[5] and shorter but frequent portions where the optimal operating mode is pure electric.

However, as we saw earlier, the torque-split models are the only two that can exactly match both these two operating modes. The engine torque-based models cannot reproduce pure thermal exactly. Consider for example the segment shown in Fig. 2, which shows the engine, e-machine and battery power as well as the power demand at the gearbox input for a time

[1]Pure electric includes regenerative braking, which is the only allowed operating mode when the vehicle is braking.

[2]Source code is available at https://github.com/fmiretti/DynaProg, toolbox package available at https://www.mathworks.com/matlabcentral/fileexchange/84260-dynaprog.

[3]I.e. with $n_\sigma = 101$ values.

[4]Obtained by running all models with extremely fine discretization grids ($n_\sigma = 2001$ and $m_{\mathrm{PF}} = 2001$) and selecting the best.

[5]Especially in the extra-high phase of the WLTC.

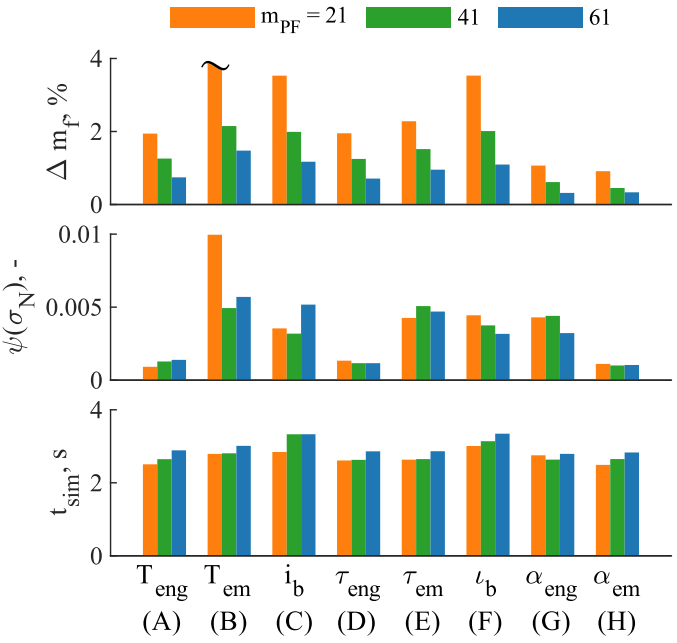

Fig. 1. Accuracy and simulation time of the examined models.

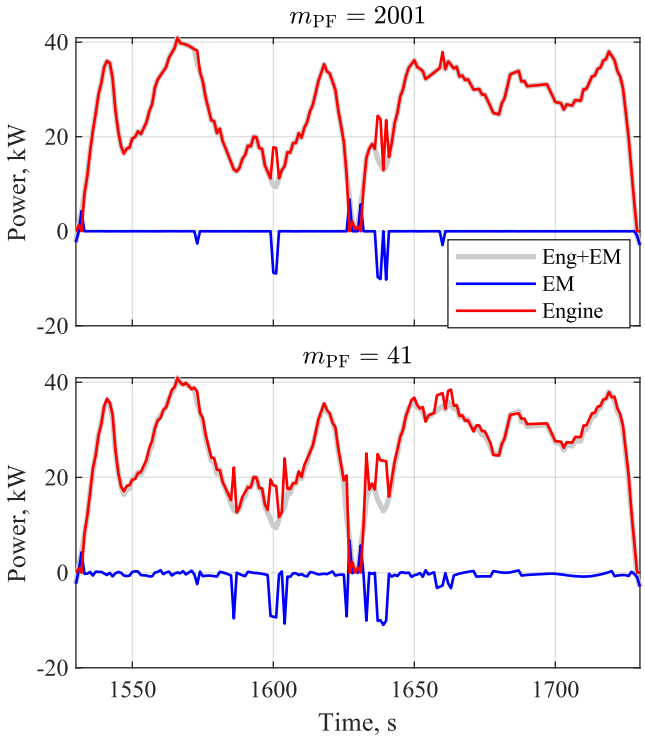

Fig. 2. Pure thermal segment. The top plot shows model A with a fine discretization ($m_{\mathrm{PF}} = 2001$). The bottom plot shows how the same model fails to reproduce the same pure thermal segments accurately with $m_{\mathrm{PF}} = 41$.

segment where the optimal solution involves going in pure thermal[6]. When using an extremely fine discretization, the engine torque model A is able to match this behavior almost exactly; but as we reduce the discretization to $m_{\mathrm{PF}} = 41$, we start seeing how the same model is unable to run in pure thermal, although it attempts to do so by making the e-machine torque as small as possible (given the control set quantization). The same behavior was observed for model D.

A similar issue affects the e-machine- and battery current-based models in that they cannot exactly reproduce pure electric operation. This time, let us consider the segment shown in Fig. 3 for model B, where the optimal solution involves many pure electric portions. While a fine discretization ($m_{\mathrm{PF}} = 2001$) is able to well approximate pure electric, a coarse discretization ($m_{\mathrm{PF}} = 41$) cannot and the algorithm has to use the engine to meet the torque demand. This operation is very inefficient as an engine's efficiency is typically low at low load; to the point were it is sometimes more convenient to just run in pure thermal.

In addition to that, the battery current-based models are less effective at regenerative braking, as it is not possible to saturate the e-machine torque to the torque demand. Hence, when the limiting factor is the torque demand and not one of either the e-machine limit torque or the maximum charge current, the current-based models lose some energy with respect to the other ones. This explains why they generally perform worse than models B and E.

What is less expected is that they are also unable to provide any reduction in computational time. On the contrary, they are the worst performing models in this aspect.

[6]This was confirmed by looking at the behavior of the e-machine torque-based models and the torque-split models.

Inspection of the code performance using a dedicated tool (MATLAB's profiler) revealed that the two most time-consuming operations for these models were the interpolations required to evaluate the e-machine efficiency and engine fuel consumption.

While for all other models these two are only a function of the exogenous inputs (the vehicle speed and acceleration) and control variables, the fact that we saturate the battery current using $\tilde{i}_{\mathrm{b,min}}$ defined by (4), which is also SOC-dependent, means that we have to do a significantly higher number (precisely n_σ times more) of interpolations on the two maps. The corresponding increase in computational cost is enough to overcome the saving induced by avoiding the computation of the battery current with (3).

VI. Conclusion

Based on our simulation results, it appears that the most robust models in terms of simulation accuracy are the torque-split based models, and we attributed this to the fact that, in contrast to the other models, they can exactly match both pure electric and pure thermal operation regardless of the powerflow control variable discretization. In terms of simulation time, all models behave similarly except for the current-based models, which are slightly worse. We attributed this to the need to introduce additional and relatively complex operations to enable fully exploiting regenerative braking.

When considering the results presented in this work, there are many factors that should be taken into account. Firstly, for

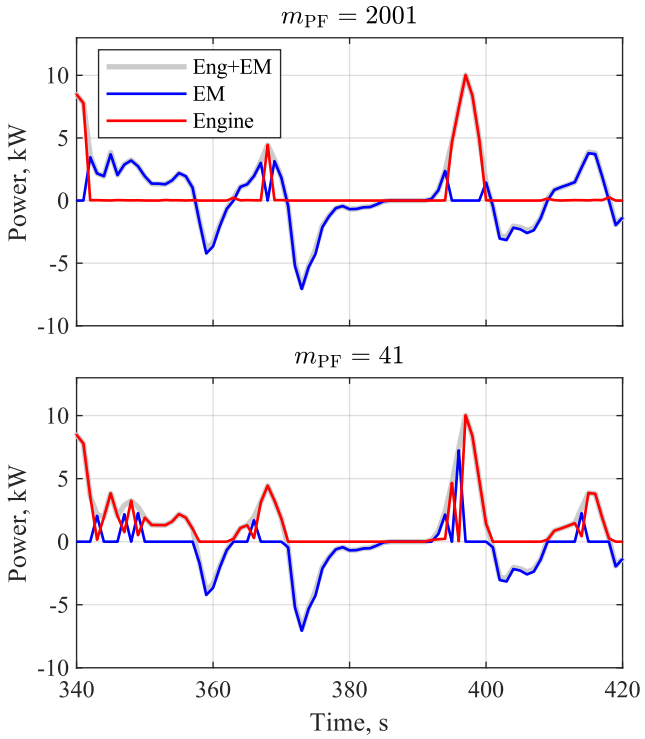

Fig. 3. Pure electric segment. The top plot shows model B with a fine discretization ($m_{\mathrm{PF}} = 2001$). The bottom plot shows how the same model fails to reproduce the same pure electric segments accurately with $m_{\mathrm{PF}} = 41$.

simplicity, we did not consider the engine and e-machine's inertia. Secondly, we did not model any auxiliaries load. This means, for example, that the vehicle can keep the state of charge constant by simply switching off the e-machine, and that the e-machine and the battery do not have to bear any additional load when the engine is turned off. Finally, we only considered one architecture with a hybridization ratio of 0.7[7] and one set of mass and road-load coefficients, corresponding to a small-size passenger car.

All these assumptions and data may somewhat alter the structure of the optimal solution, which may in turn stress differently the strengths and weaknesses of each model. For example, the optimal solution that we inspected in the simulation results makes very scarce use of the battery charging operating mode; hence the most important weakness of the torque-split models does not show.

There are numerous extensions to this work. The most obvious are reintroducing the engine inertia and/or accessory loads as well as to test a wider range of vehicles and hybridization ratios. Then, similar analysis may also be repeated for other powertrain configurations, such as power-split hybrids and series hybrids.

References

[1] S. Onori, L. Serrao, and G. Rizzoni, *Hybrid Electric Vehicles*. Springer-Verlag GmbH, Dec. 2015.

[2] P. G. Anselma and G. Belingardi, "Next generation HEV powertrain design tools: Roadmap and challenges," in *SAE Technical Paper Series*. SAE International, oct 2019.

[3] P. Pisu and G. Rizzoni, "A comparative study of supervisory control strategies for hybrid electric vehicles," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 3, pp. 506–518, may 2007.

[4] W. Enang and C. Bannister, "Modelling and control of hybrid electric vehicles (a comprehensive review)," *Renewable and Sustainable Energy Reviews*, vol. 74, pp. 1210–1239, jul 2017.

[5] E. Silvas, T. Hofman, N. Murgovski, P. Etman, and M. Steinbuch, "Review of optimization strategies for system-level design in hybrid electric vehicles," *IEEE Transactions on Vehicular Technology*, 2016.

[6] D. Bianchi, L. Rolando, L. Serrao, S. Onori, G. Rizzoni, N. Al-Khayat, T.-M. Hsieh, and P. Kang, "A rule-based strategy for a series/parallel hybrid electric vehicle: An approach based on dynamic programming," in *ASME 2010 Dynamic Systems and Control Conference*, jan 2010.

[7] D. Kum, H. Peng, and N. K. Bucknor, "Supervisory control of parallel hybrid electric vehicles for fuel and emission reduction," *Journal of Dynamic Systems, Measurement, and Control*, vol. 133, no. 6, nov 2011.

[8] Y. L. Murphey, J. Park, Z. Chen, M. L. Kuang, M. A. Masrur, and A. M. Phillips, "Intelligent hybrid vehicle power control - part i: Machine learning of optimal vehicle power," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 8, pp. 3519–3530, oct 2012.

[9] P. Elbert, S. Ebbesen, and L. Guzzella, "Implementation of dynamic programming for n-dimensional optimal control problems with final state constraints," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 3, pp. 924–931, may 2013.

[10] F. Miretti, D. Misul, and E. Spessa, "DynaProg: Deterministic dynamic programming solver for finite horizon multi-stage decision problems," *SoftwareX*, vol. 14, p. 100690, jun 2021.

[11] D. Kum, H. Peng, and N. K. Bucknor, "Optimal catalyst temperature management of plug-in hybrid electric vehicles," in *Proceedings of the 2011 American Control Conference*. IEEE, jun 2011.

[12] P. G. Anselma, P. Kollmeyer, J. Lempert, Z. Zhao, G. Belingardi, and A. Emadi, "Battery state-of-health sensitive energy management of hybrid electric vehicles: Lifetime prediction and ageing experimental validation," *Applied Energy*, vol. 285, p. 116440, mar 2021.

[13] P. G. Anselma, A. Biswas, G. Belingardi, and A. Emadi, "Rapid assessment of the fuel economy capability of parallel and series-parallel hybrid electric vehicles," *Applied Energy*, vol. 275, p. 115319, oct 2020.

[14] M. Spano, P. G. Anselma, D. A. Misul, and G. Belingardi, "Exploitation of a particle swarm optimization algorithm for designing a lightweight parallel hybrid electric vehicle," *Applied Sciences*, vol. 11, no. 15, p. 6833, jul 2021.

[15] L. Johannesson and B. Egardt, "Approximate dynamic programming applied to parallel hybrid powertrains," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 3374–3379, 2008.

[16] P. G. Anselma, "Computationally efficient evaluation of fuel and electrical energy economy of plug-in hybrid electric vehicles with smooth driving constraints," *Applied Energy*, vol. 307, p. 118247, feb 2022.

[17] ——, "Optimization-driven powertrain-oriented adaptive cruise control to improve energy saving and passenger comfort," *Energies*, vol. 14, no. 10, p. 2897, may 2021.

[18] V. Larsson, L. Johannesson, and B. Egardt, "Analytic solutions to the dynamic programming subproblem in hybrid vehicle energy management," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 4, pp. 1458–1467, apr 2015.

[19] B. de Jager, M. Steinbuch, and T. van Keulen, "An adaptive sub-optimal energy management strategy for hybrid drive-trains," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 102–107, 2008.

[20] M. Koot, J. Kessels, B. deJager, W. Heemels, P. vandenBosch, and M. Steinbuch, "Energy management strategies for vehicular electric power systems," *IEEE Transactions on Vehicular Technology*, vol. 54, no. 3, pp. 771–782, may 2005.

[21] O. Sundström, L. Guzzella, and P. Soltic, "Torque-assist hybrid electric powertrain sizing: From optimal control towards a sizing law," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 4, pp. 837–849, jul 2010.

[22] G. Rizzoni, L. Guzzella, and B. Baumann, "Unified modeling of hybrid electric vehicle drivetrains," *IEEE/ASME Transactions on Mechatronics*, vol. 4, no. 3, pp. 246–257, 1999.

[7] Defined as: $\frac{P_{\mathrm{eng,max}}}{P_{\mathrm{eng,max}}+P_{\mathrm{em,max}}}$, as in [13].