Online Temperature-aware Equivalent Consumption Minimization Strategy for Mild Hybrid Electric Powertrains

Online Temperature-Aware Equivalent Consumption Minimization Strategy for Mild Hybrid Electric Powertrains

Matteo Acquarone, *Graduate Student Member, IEEE*, Federico Miretti, *Member, IEEE*, Pier Giuseppe Anselma, *Member, IEEE*, and Daniela Anna Misul

***Abstract*—In this work, an online energy management strategy for mild hybrid electric vehicles is developed to minimize the fuel consumption while simultaneously preventing battery overheating. Since mild hybrids are typically equipped with a passively cooled battery pack, the energy management strategy design needs to keep the battery temperature below an upper limit, preventing accelerated aging and thermal runaway. To address this issue, the equivalent consumption minimization strategy (ECMS) approach is extended to develop a real-time capable controller, termed thermal ECMS (Th-ECMS), that is sensitive to the thermal dynamics of the battery and that can enforce constraints on its temperature. The rationale for our formulation is based on Pontryagin's minimum principle from optimal control theory. The online Th-ECMS is developed on the basis of the offline version of Th-ECMS, introduced in a previous work. Exploiting the a priori knowledge of the driving mission, the offline Th-ECMS calibrates the equivalence factors and obtains the optimal solution, which is compared with the globally optimal dynamic programming solution. This offline calibration method is run on a large number of driving missions and the collected data is used to train a feed-forward neural network that estimates optimal equivalence factors as functions of the battery state of charge, battery temperature, and distance yet to travel. The trained network is then used to populate two look-up tables mapping the equivalence factors, and implementable on the vehicle electronic control unit. Finally, the online Th-ECMS obtains the equivalence factors through the look-up tables in real-time. The online strategy was tested in four different driving missions, achieving a fuel economy remarkably similar to the optimal solution and successfully avoiding battery overheating.**

***Index Terms*—Mild hybrid, hybrid electric, energy management, thermal management, passive cooling, battery.**

I. Introduction

MILD hybrid electric vehicles (MHEVs) represent a cost-effective solution that can achieve a reduction in CO_2 emissions in the range of 5% to 30% in real-world driving compared to a conventional vehicle [1], [2], depending on the specific architecture and powertrain design. Consequently, the market penetration of mild HEVs is significant and growing. For example, the European market share increased from 0.2% in 2017 to 14% in 2021 [2]. In monetary terms, a 17.45% annual growth rate is expected for the MHEV market over a 2022–2027 forecast period [3].

Mild hybrids combine an internal combustion engine (ICE) with a small battery pack (approximately 1 kWh) and a relatively small electric machine, such as an integrated starter-generator (ISG) or a belt starter-generator (BSG). Another typical feature of a mild HEV is a 48 V energy storage system which enables power assist during vehicle propulsion and regenerative braking as well as electrification of several accessories (e.g. electrically assisted turbochargers, power steering, air conditioning compressors). The battery pack is a crucial component which requires dedicated management. The performance and lifespan of this component can be seriously compromised by overheating, which can lead to thermal runaway and accelerated aging [4]. For full hybrids, active cooling systems are therefore employed to ensure that battery temperature remains within its optimal range, which is typically between 15 and 40 °C [5]. However, for mild hybrids, the relatively high cost of an active cooling system might reduce the cost-effectiveness of hybridization; for this reason, passive cooling systems are often employed [2], [6], [7]. However, the duty cycle of a MHEV battery under an ordinary energy management strategy (EMS) may be highly cyclic and lead to overheating if passive cooling is used [8].

The energy management strategy is responsible for controlling the mechanical power split between the internal combustion engine (ICE) and the BSG. Based on our previous considerations, it is clear that an EMS for mild HEVs with a passive cooling system must provide compliance with the battery thermal limits by appropriately limiting its charge and discharge currents. On the other hand, limiting battery utilization can jeopardize the CO_2 reduction benefit: identifying the right trade-off between fuel economy and preventing battery overheating is not trivial, and a smart control strategy is needed.

When developing an EMS for a mild HEV, several research works neglect the battery temperature evolution over time in their vehicle modeling approach [9], [10]. Other research works

Manuscript received 15 April 2023; revised 7 September 2023 and 7 November 2023; accepted 12 January 2024. Date of publication 17 January 2024; date of current version 22 April 2024. The review of this article was coordinated by the TVT Administrator. *(Corresponding author: Federico Miretti.)*

Matteo Acquarone, Federico Miretti, and Daniela Anna Misul are with the Department of Energy (DENERG) and also with the Center for Automotive Research and Sustainable Mobility (CARS@Polito), Politecnico di Torino, 10138 Torino, Italy (e-mail: matteo.acquarone@polito.it; federico.miretti@polito.it; daniela.misul@polito.it).

Pier Giuseppe Anselma is with the Department of Mechanical and Aerospace Engineering (DIMEAS) and also with the Center for Automotive Research and Sustainable Mobility (CARS@Polito), Politecnico di Torino, 10138 Torino, Italy (e-mail: pier.anselma@polito.it).

Digital Object Identifier 10.1109/TVT.2024.3355181

model the evolution of battery temperature over time, but they do not use this information as a feedback loop to the EMS of the electrified powertrain [6]. To the best of the authors' knowledge, improving the fuel economy of the real-time EMS for mild HEVs while taking into account the evolution of the battery temperature over time is still an open research question.

In general, EMSs for HEVs can be classified between offline and online approaches depending on the future driving conditions being known a priori beforehand or not, respectively [11]. Offline EMSs are typically adopted to achieve global optimization and set an optimal benchmark. Several online EMSs for mild HEVs have been proposed in the literature, involving, for example, Pontryagin's minimum principle (PMP) [9], equivalent consumption minimization strategy (ECMS) [12], rule-based techniques [13], and reinforcement learning [10], [14]. Obviously, only online EMSs are implementable on-board mild HEVs since knowing the profile of the vehicle speed over time for the entire drive cycle in advance is not possible.

In its traditional formulation, ECMS aims to minimize the equivalent fuel consumption, which is formulated as the sum between the actual fuel consumption and a virtual fuel consumption associated with battery usage, weighted by an equivalence factor (EF) [15]. An application of ECMS for the real-time control of a 48 V mild parallel HEV is found in [16], where the SOC obtained from a dynamic programming algorithm is adopted as the reference SOC for the ECMS. Calibration of the EF is a critical process to minimize fuel economy while satisfying the battery charge-sustaining constraint. Furthermore, the value of the EF has a significant impact on battery temperature and aging entailed by an ECMS HEV controller, as shown in [17].

While the EF calibration process is traditionally carried out offline for a specific driving cycle, several variations of the ECMS have been developed to estimate the equivalence factor online [18]. In general, online control strategies regulate the EF to suppress excessive SOC deviation from a desired reference value. For example, a feedback controller is used to this end in [19], [20]. Indeed, several algorithms based on the proportional-integral-derivative (PID) controller have been used for the EF adaptation. For example, the authors of [21] used linear proportional controllers to adjust the EF, while the addition of an integral gain was shown to improve the ECMS performance in [22].

A different approach involves the use of look-up tables, also referred to as maps, to select the initial EF and/or continuously adapt the EF. The main advantage of look-up tables is the low computational complexity, which makes them particularly appealing for online control [23]. Look-up tables also offer consistent and repeatable behavior, as opposed to other techniques such as directly obtaining the EF from a neural network. In [24], a table that maps the initial EF as a function of initial SOC and driving distance is populated offline using a genetic algorithm. Then, the EF is updated online using a fuzzy control logic based on SOC deviation from a reference trajectory and engine speed. Relying on a similar approach, the control strategy developed in [25] uses a similar lookup table to obtain the EF as a function of SOC and expected driving distance, but it directly uses the table to continuously update the value of the EF. To ensure robustness with respect to varying driving conditions, several maps are created with different driving cycles, and a pattern recognition technique is used online to determine the map to be used.

Another map-based ECMS is developed in [26] to minimize fuel consumption without any knowledge of the future driving mission. An offline optimization is performed to calculate the optimal torque distribution, which is stored in tables and then used in real-time implementation. However, the traditional look-up table-based approach is based on historical data and achieves poor performance when completely different driving conditions are encountered.

Recently, some EF adaption methods based on neural networks (NN) have been developed to improve the NN's ability to generalize to unknown driving conditions. In [27], a neural network is used to this end which takes the current power demand, the ratio of the distance traveled to the total distance, and SOC as inputs, achieving slightly higher fuel consumption than the global optimal solution. The authors of [28] propose an intelligent ECMS based on dual neural networks and a new EF adaptation method for online control of plug-in HEVs which allows to obtain similar fuel economy to the dynamic programming optimal solution.

To the best of the authors' knowledge, no ECMS has been developed to regulate the battery temperature for mild HEV applications.

In this work, an offline and online version of the Thermal ECMS (Th-ECMS) are developed to minimize fuel consumption, guaranteeing that battery overheating is avoided. The offline version of the Th-ECMS, developed in [29], is based on Pontriyagin's Minimum Principle to obtain the optimal solution, exploiting a priori knowledge of the driving cycle. The optimal EFs values, obtained by the offline Th-ECMS algorithm over a set of driving cycles, are stored and used to train an NN which estimates the optimal EFs as a function of SOC, the battery temperature, and the distance yet to travel. Afterward, the trained NN is used to populate a look-up table, which is used in the online Th-ECMS algorithm to obtain the EFs values in real-time. Compared with a baseline ECMS formulation, the proposed battery temperature-aware ECMS is demonstrated to guarantee compliance with battery thermal limits.

The remainder of the article is structured as follows: Section II describes the simulation model developed for the p0 mild hybrid powertrain, including the battery thermal dynamics. Section III introduces the proposed formulation for a battery temperature-aware ECMS and describes its offline solution procedure based on using the Pontryagin minimum principle, to obtain fuel-optimal trajectories for the equivalence factors. Section IV describes the online implementation of the Th-ECMS based on look-up tables as well as the procedure to create the tables themselves. Finally, Section V presents results for the offline calibration procedure using a set of drive cycles, and the obtained look-up tables are tested on another set of drive cycles. For the two sets, both regulatory drive cycles and real driving cycles were used.

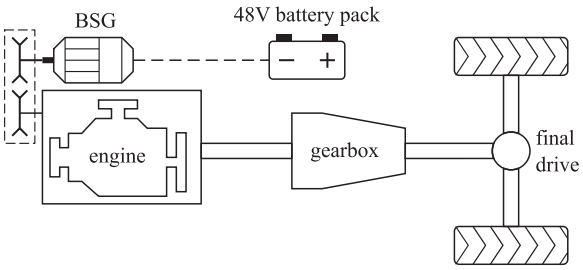

Fig. 1. p0 HEV architecture.

II. Vehicle Model

The p0 HEV powertrain is depicted in Fig. 1. The main goals of the vehicle model are to characterize the evolution of the battery's state (SOC and temperature) and fuel consumption as a function of the torque split between the engine and BSG, which must be set by the EMS. This enables the formulation of an optimal control problem to minimize fuel consumption while enforcing constraints on the battery SOC and temperature.

First, the longitudinal vehicle dynamics model allows to compute the tractive effort F_{veh} as a function of the vehicle's speed v_{veh} and acceleration a_{veh} using a set of road-load coefficients f_0, f_1, f_2 [30], [31]

$$F_{\text{veh}} = f_0 + f_1 v_{\text{veh}} + f_2 v_{\text{veh}}^2 + m_{\text{veh}} a_{\text{veh}}, \tag{1}$$

where m_{veh} is the vehicle mass. The torque demand τ_{d} is then evaluated using a backward-facing quasi-static model [32]:

$$\tau_{\text{d}} = \frac{F_{\text{veh}}(v_{\text{veh}}, a_{\text{veh}}) r_{\text{wh}}}{i_{\text{fd}} i_{\text{gb}}(\gamma)} \tag{2}$$

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

For the scope of this work, the battery pack needs to be characterized from the electrical and thermal standpoint. A zero-order equivalent circuit model (ECM) was used to describe the electrical behavior of the battery pack. Therefore, the battery SOC dynamics, which define the SOC evolution given the power drawn from the battery P_{b} and the battery's current state, were modeled as follows:

$$\dot{\sigma} = -\frac{i_{\text{b}}}{Q_{\text{b}}}, \tag{3}$$

$$i_{\text{b}} = \frac{v_{\text{oc}} - \sqrt{v_{\text{oc}}^2 - 4 R_{\text{eq}} P_{\text{b}}}}{2 R_{\text{eq}}}, \tag{4}$$

where i_{b}, v_{oc}, R_{eq} and Q_b are the battery current, open-circuit voltage, equivalent resistance, and capacity. The battery current i_{b} is assumed positive during discharging and negative during charging. The open-circuit voltage and equivalent resistance are characterized as functions of both the battery state of charge and temperature. Electrical and thermal parameters of the 48 V battery pack are taken from [33]. The battery power was evaluated the sum of the power absorbed by the e-machine and the electric auxiliaries.

For the thermal dynamics, the battery lumped thermal model accounts for the heat generated due to the Joule losses and the convective heat transfer with the surrounding environment [34], [35], [36]:

$$\dot{T}_{\text{b}} = \frac{1}{C_{\text{b}}} \left(R_{\text{eq}} i_{\text{b}}^2 - h_b A_{\text{b}} (T_{\text{b}} - T_{\text{env}}) \right), \tag{5}$$

where C_{b} is the battery's thermal capacity. Heat transfer to the environment is modeled as proportional to the temperature difference between the battery (T_{b}) and the surrounding air (T_{env}) via the heat exchange area A_{b} and heat transfer coefficient h_b.

The electrical power $P_{\text{em,el}}$ absorbed or generated by the e-machine can be computed as a function of its speed ω_{em} and torque τ_{em}:

$$P_{\text{em,el}} = \omega_{\text{em}} \tau_{\text{em}} + P_{\text{em,losses}}(\omega_{\text{em}}, \tau_{\text{em}}) \tag{6}$$

where $P_{\text{em,losses}}$ are the e-machine power losses modeled as a function of ω_{em} and τ_{em}.

The e-machine torque is directly controlled through the e-machine torque-split factor α, which is defined as the ratio between τ_{em} and the torque demand τ_{d}:

$$\alpha = \frac{\tau_{\text{em}}}{\tau_{\text{d}}}; \tag{7}$$

the value of α was limited between 1 and -1. During traction (i.e. the torque demand is positive), if α is positive the e-machine provides traction torque to assist the engine; if α is negative, the engine provides all the torque demand as well as an additional torque which is absorbed by the e-machine to charge the battery. During braking, the engine is turned off and the maximum allowable amount of regenerative braking torque is absorbed by the e-machine. The remaining braking torque demand (if any) is provided by the mechanical brakes.

Controlling the value of α implicitly sets the value of the engine torque as well. Since the torque demand must be met at all times by the e-machine and the engine (during traction), the engine torque τ_{eng} can be expressed as:

$$\tau_{\text{eng}} = \tau_{\text{d}} - \tau_{\text{em}}. \tag{8}$$

The fuel flow rate $\dot{m}_f$ was computed from a steady-state map as a function of the engine speed and torque. The gear number was controlled using a simple gear shift schedule as a function of the vehicle speed. The optimization of the gear shift schedule is not treated in this work.

III. Th-ECMS and Offline Solution

In this subsection, the Th-ECMS and its offline solution method are introduced [29]. The offline solution method is required to obtain fuel-optimal trajectories that satisfy the battery temperature constraint for a given driving cycle. The corresponding co-state trajectories are needed to calibrate the look-up tables for the online implementation, as described in Section IV. The offline solution procedure itself is based on the Pontryagin minimum principle, which leads to the formulation of a *boundary value problem* (BVP). This BVP was solved with a single shooting method that was specifically developed using a particle swarm optimization algorithm, which is outlined in Fig. 2 and described in the following section.

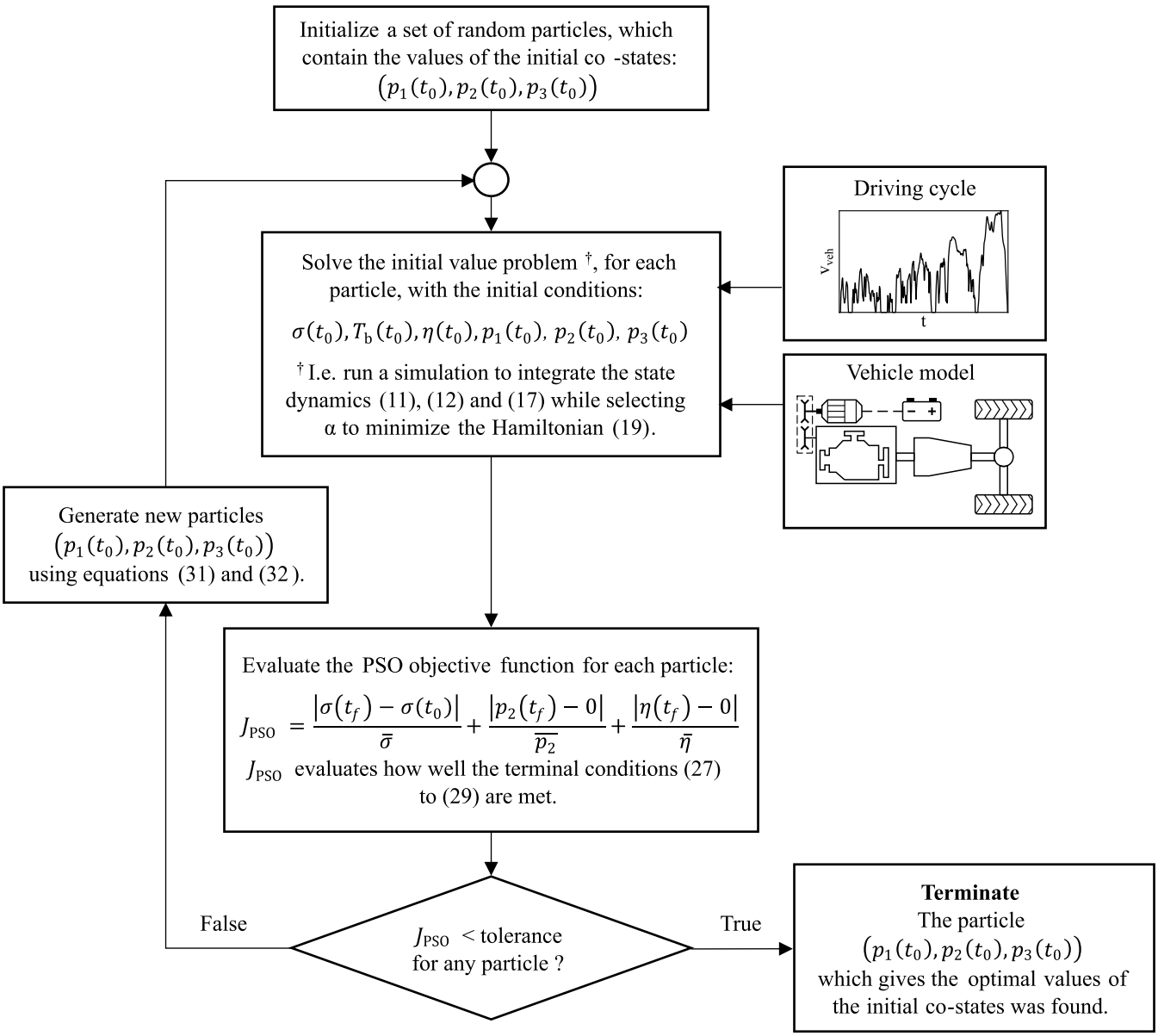

Fig. 2. Indirect single shooting algorithm using PSO.

A. *Optimal Control Problem Formulation*

To justify our formulation of the Th-ECMS, we formulate the EMS design as an optimal control problem. Specifically, we formulate a finite-time constrained optimal control problem to minimize the total fuel consumption [31], [37], [38]:

$$J = \int_{t_0}^{t_f} \dot{m}_f(\alpha, t). \tag{9}$$

The fuel flow rate $\dot{m}_f$ is the running cost, J is the total cost, α is the control variable, and t_0 and t_f are the initial and final time instants of the driving mission, respectively.

Since the vehicle speed and acceleration are defined by the driving mission, and are explicit functions of time, the torque demand τ_d is an exogenous input of the optimal control problem. The state variables are the battery state of charge σ and temperature T_b. Since the battery ECM parameters are functions of the state variables, the current and, therefore, the SOC and temperature dynamics are functions of the states, the torque-split factor α, and time:

$$\dot{\sigma} = \dot{\sigma}(\sigma, T_b, \alpha, t) \tag{10}$$

$$\dot{T}_b = \dot{T}_b(\sigma, T_b, \alpha, t) \tag{11}$$

Several global and optimal constraints need to be satisfied by our EMS. In addition to the various control-dependent constraints that must be formulated to enforce feasibility of the powertrains components' operation (e.g. enforcing the limit torques of the engine and e-machine), state constraints are imposed to maintain the battery SOC and temperature between upper (ub) and lower

(lb) bounds, and to guarantee charge-sustaining operation:

$$\sigma \leq \sigma_{\text{ub}}, \tag{12}$$

$$\sigma \geq \sigma_{\text{lb}}, \tag{13}$$

$$T_{\text{b}} \leq T_{\text{b,ub}}, \tag{14}$$

$$\sigma(t_{\text{f}}) = \sigma(t_0) = \sigma_0, \tag{15}$$

where $\sigma_{\text{ub}} = 0.8$, $\sigma_{\text{lb}} = 0.4$, and the battery temperature upper limit $T_{\text{b,ub}}$ is assumed to be 40 °C [39].

B. Pontryagin's Minimum Principle

The presented optimal control problem was solved by applying the Pontryagin's minimum principle. An indirect method [40] was used to derive necessary conditions of optimality using the PMP; these were then transcribed into a *boundary value problem* whose solution yields the optimal control trajectory as well as the optimal state and co-state trajectories. Compared to other approaches to EMS design such as dynamic programming [41], indirect methods are generally far less computationally expensive while still retaining high accuracy. However, this does not mean that their numerical solution is easy: BVPs of this kind require a good initial guess of the solution in order to achieve convergence [42].

In order to impose the constraints (12), (13), and (14) in the PMP formulation, the approach described in [40, Chapter 5.3] is adopted, and an additional state variable η is introduced, with the following dynamics:

$$\begin{aligned}\dot{\eta} = {} & (\sigma - \sigma_{\text{ub}})^2\, \mathbb{1}(\sigma - \sigma_{\text{ub}}) + (\sigma_{\text{lb}} - \sigma)^2\, \mathbb{1}(\sigma_{\text{lb}} - \sigma) \\ & + (T_{\text{b}} - T_{\text{b,ub}})^2\, \mathbb{1}(T_{\text{b}} - T_{\text{b,ub}})\end{aligned} \tag{16}$$

[1]

Note that, by definition, $\eta(t)$ is monotonically increasing in time and is strictly equal to zero only if the state constraints are never violated. In a way, this variable can be seen as a quantification of the constraints violation.

Since our goal is to minimize this violation, the initial $\eta(t_0)$ and final $\eta(t_{\text{f}})$ states are required to be equal to 0. The reason is that if these conditions are satisfied, the constraints (12), (13), and (14) are never violated since $\eta(t_{\text{f}})$ is calculated as

$$\eta(t_{\text{f}}) = \eta(t_0) + \int_{t_0}^{t_{\text{f}}} \dot{\eta}(t)\, dt. \tag{17}$$

Finally, the Hamiltonian for this control system was written as:

$$\begin{aligned}H(\sigma, T_{\text{b}}, \alpha, p_1, p_2, p_3, t) = {} & \dot{m}_{\text{f}}(\alpha, t) + p_1\, \dot{\sigma}(\sigma, T_{\text{b}}, \alpha, t) \\ & + p_2\, \dot{T}_{\text{b}}(\sigma, T_{\text{b}}, \alpha, t) + p_3\, \dot{\eta}(\sigma, T_{\text{b}}, \alpha, t).\end{aligned} \tag{18}$$

For this problem, the minimum principle states that if $\alpha(t)$ is the optimal control trajectory, then there must exist three co-state functions $p_1(t)$, $p_2(t)$ and $p_3(t)$ satisfying the adjoint equations

$$\dot{p}_1 = -\frac{\partial H}{\partial \sigma}, \tag{19}$$

$$\dot{p}_2 = -\frac{\partial H}{\partial T_{\text{b}}}, \tag{20}$$

$$\dot{p}_3 = -\frac{\partial H}{\partial \eta} = 0, \tag{21}$$

and such that $\alpha \in U(t)$ minimizes the Hamiltonian, where $U(t)$ defines the set of admissible controls and the Hamiltonian is defined by (18).

Furthermore, since the final temperature $T_{\text{b}}(t_{\text{f}})$ is free, the corresponding terminal co-state must satisfy the transversality condition:

$$p_2(t_{\text{f}}) = \frac{\partial F(T_{\text{b}}(t_{\text{f}}))}{\partial T_{\text{b}}} = 0, \tag{22}$$

where $F(T_{\text{b}}(t_{\text{f}}))$ is the terminal cost associated to the battery temperature. Since the formulation of a cost term related to the final battery temperature is not trivial[2] and is beyond the scope of this work, the term $F(T_{\text{b}}(t_{\text{f}}))$ was set to zero.

Summarizing, the application of the minimum principle results in the formulation of a BVP composed by a system of six differential equations (the state dynamics (3), (5), (16) and the adjoint (19)–(21)) and six boundary conditions (the initial states, terminal states and the transversality condition (22)). These boundary conditions are listed below:

$$\sigma(t_0) = \sigma_0, \tag{23}$$

$$T_{\text{b}}(t_0) = T_{\text{env}}, \tag{24}$$

$$\eta(t_0) = 0, \tag{25}$$

$$\sigma(t_{\text{f}}) = \sigma_0, \tag{26}$$

$$p_2(t_{\text{f}}) = 0, \tag{27}$$

$$\eta(t_{\text{f}}) = 0. \tag{28}$$

C. Indirect Shooting Based on Particle Swarm Optimization

In order to solve this BVP, the shooting method is employed. In indirect shooting, an initial guess is generated for all the boundary conditions at the left end point t_0. Then, the initial value problem (IVP) composed of the differential equations coupled to this set of initial conditions is solved by numeric integration. In this work, a forward Euler integration scheme is adopted, and the partial derivatives in (19)–(21) are obtained using finite differences. The difference between the resulting boundary conditions at t_{f} and those imposed in the BVP is used to update the initial guess and the process is repeated.

The same procedure can also be formulated as an optimization problem, where the optimization variables are the initial values of the co-states $p_1(t_0)$, $p_2(t_0)$, and $p_3(t_0)$, and the objective function is the difference between the corresponding $\sigma(t_{\text{f}})$, $p_2(t_{\text{f}})$ and $\eta(t_{\text{f}})$ and those set by the boundary conditions (26)–(28). With this formulation, a particle swarm optimization (PSO) algorithm [43] is adopted to calibrate the initial co-states $(p_1(t_0), p_2(t_0), p_3(t_0))$.

[1] Here, $\mathbb{1}(\cdot)$ denotes the unit step function.

[2] For instance, this term could be formulated as a function of battery aging, which is a particularly complex phenomenon to model

The objective function of the PSO solver is defined as a scalarized objective to minimize the error related to the final boundary conditions $\sigma(t_f)$, $p_2(t_f)$ and $\eta(t_f)$ obtained by solving the BVP with the initial co-states $p_1(t_0)$, $p_2(t_0)$, and $p_3(t_0)$:

$$J_{\text{PSO}}(p_1(t_0), p_2(t_0), p_3(t_0)) = \frac{|\sigma(t_f) - \sigma_0|}{\bar{\sigma}} + \frac{|p_2(t_f)|}{\bar{p_2}} + \frac{\eta(t_f)}{\bar{\eta}} \tag{29}$$

where $\bar{\sigma}$, $\bar{p_2}$ and $\bar{\eta}$ are normalization factors, selected as 0.1%, 0.1, and 1, respectively.

The whole procedure, which is also depicted in Fig. 2, is described in this section. First, a set of particles is initialized. Each particle defines the values of the initial co-states. The values of the particles are uniformly distributed within given bounds. For each particle, the initial value problem is solved; in other terms, a simulation is run using as initial conditions the initial states and the initial co-states defined by the particle. In running this simulation, the torque-split factor α is set by minimizing the Hamiltonian (18), which also requires integrating the adjoint (19)–(21). Then, the PSO objective function (29) is evaluated for each particle, using the values obtained for $\sigma(t_f)$, $p_2(t_f)$ and $\eta(t_f)$ as a result of the simulation.

If there is one particle for which J_{PSO} is within a certain tolerance, i.e., $J_{\text{PSO}} < 1$, that particle defines the values of $p_1(t_0)$, $p_2(t_0)$, and $p_3(t_0)$ which satisfy the necessary conditions of optimality, and the PSO algorithm is terminated. Otherwise, a new set of particles is generated and the process is repeated. To generate a new set, each particle is moved toward its prior personal best[3] position f^* and the global best position g^* in the swarm. More specifically, for the i-th iteration of the PSO algorithm and for particle p, the new values f_p (also called the particle's *position*) are evaluated as [44]:

$$f_{p,i+1} = f_{p,i} + v_{p,i+1}, \tag{30}$$

$$v_{p,i+1} = \omega v_{p,i} + c_1 \mathbf{r_1} (f^* - f_{p,i}) + c_2 \mathbf{r_2} (g^* - f_{p,i}), \tag{31}$$

where v_p is the particle's *velocity*, ω, c_1 and c_2 are parameters of the algorithm and $\mathbf{r_1}$ and $\mathbf{r_2}$ are uniformly distributed random vectors in the range [0, 1].

The main advantage of the PSO solver is its robustness with respect to a poor first guess. This can be attributed to the fact that PSO is a derivative-free algorithm, hence it is less sensible to the strong lack of smoothness in the problem. This lack of smoothness is ultimately caused by the presence of non-continuously differentiable functions such as linear interpolants[4] in the vehicle model. It should be noted that, although the PSO algorithm is able to reduce the objective function (29) to a very small quantity, it is never able to reduce it to exactly zero. Among other aspects, this is due to the many approximations in the optimization process, such as the numerical integration error and the approximation of the adjoint equations by finite differences. Indeed, this reflects the fact that we are obtaining an approximate solution to the exact optimality conditions, as is typical of all indirect methods.

[3]I.e. the one with the lowest objective function value.

[4]For instance, the fuel consumption and the ECM parameters are computed by linear interpolants as is typical in HEV powertrain models.

IV. Online Th-ECMS

The online Th-ECMS strategy is designed to achieve a real-time control strategy, implementable on a vehicle electronic control unit. Therefore, the fundamental goals were to obtain a controller that is computationally lightweight and does not depend on future driving conditions. The only information that is assumed to be available about the trip is the remaining distance to be traveled, which can be easily obtained from any GPS navigation system. In the online Th-ECMS, the two EFs (which we rename $p_{1,on}$ and $p_{2,on}$ for the sake of clarity) are obtained in real-time from two different look-up tables, one for each EF. These look-up tables are calibrated using the results of the offline solution procedure; the whole workflow is represented in Fig. 3.

The independent variables of the look-up tables are chosen to be the state variables, SOC and battery temperature, and the distance yet to travel. The reason for including the two state variables is readily explained by Bellman's principle of optimality, which implies that for a deterministic and autonomous system, optimal decisions depend on the current value of the state only. However, our system is not autonomous since its evolution is strongly time-dependent through the future development of the driving mission. Therefore, the remaining trip length is added to the inputs of the look-up table as the simplest variable that is informative enough to achieve charge-sustaining operation without requiring any additional prediction algorithm. All of this information is available in real-time on a vehicle without requiring a significant additional computational effort, under the assumption that the current location is known and the final destination is set by the user before starting the driving mission.

Another reason for only considering the remaining trip length to encode information about the driving cycle is the concern that adding more variables would make the calibration of the look-up tables significantly harder. Nonetheless, this could be replaced by more informative variables that better encode the future driving conditions, therefore leading to a better performance; one example would be to use an estimate of the remaining traction energy that could be computed from e.g. remaining trip length and information about traffic and geography (urban/extra-urban/highway, flat/hilly terrain) that can be obtained from the navigation system.

A. Calibration of the EF Look-Up Tables

To calibrate the look-up tables for the equivalence factors, a wide set of driving cycles is selected and the offline solution procedure described in Section III is adopted to obtain the optimal EFs trajectories. More specifically, since each simulation is run in discrete time, the EFs values are obtained for every time step of each driving cycle. Collectively, these values build up a training dataset.

Afterward, a feed-forward neural network, which takes the SOC, battery temperature, and distance yet to travel as inputs, is trained on this training dataset to estimate the values of the two EFs. In general, the selection of a sufficiently large and diverse training dataset is needed to enhance the generalization ability of an NN. Therefore, the set of driving cycles of the training dataset needs to be comprehensive of both regulatory and real-world

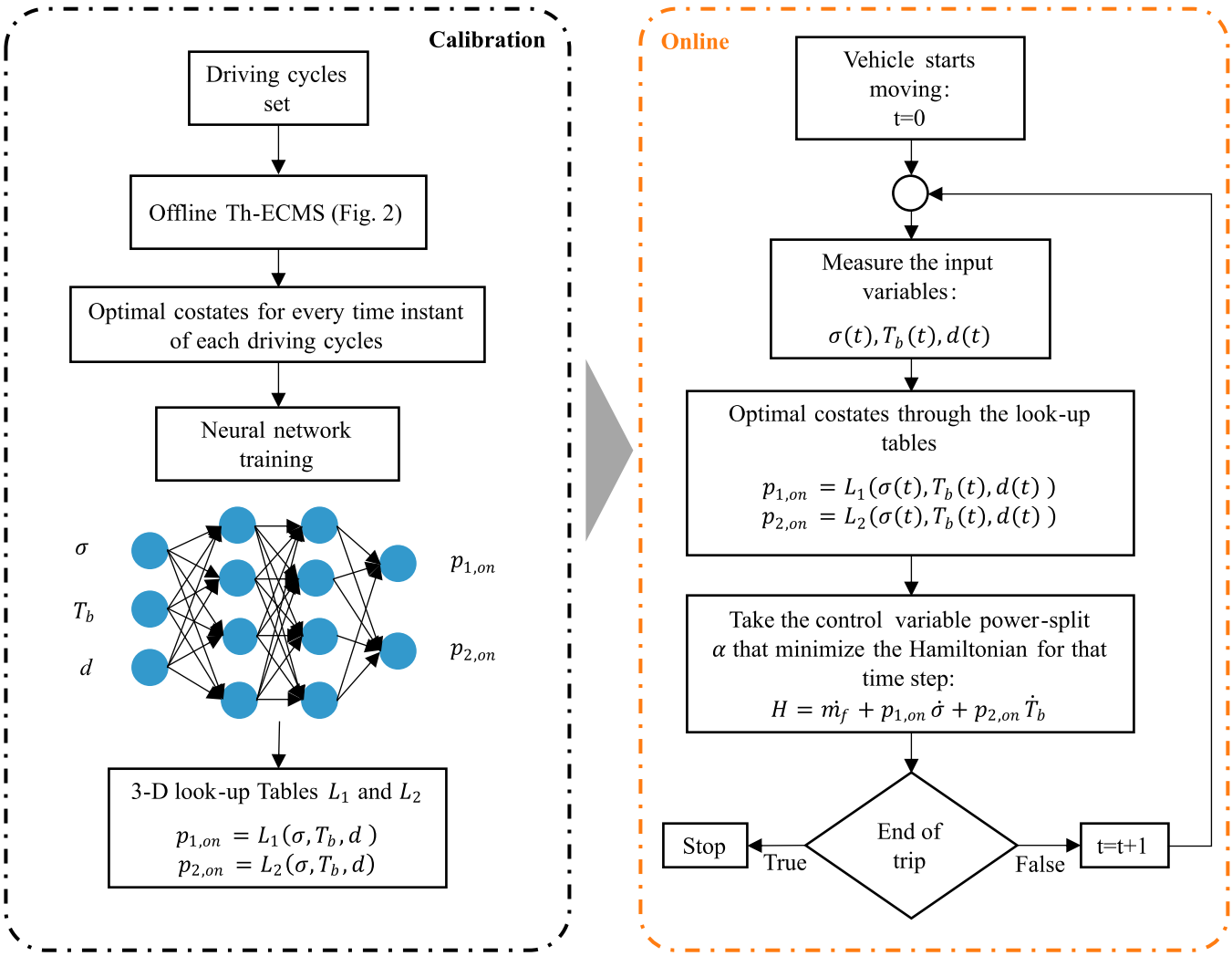

Fig. 3. Workflow of the online Th-ECMS. The calibration phase allows to obtain the EF look-up tables as functions of SOC, battery temperature and distance yet to travel. Afterwards, the online Th-ECMS uses the calibrated look-up table in real-time to select the power split.

driving scenarios, covering a wide variety of driving conditions in order to improve the NN ability to perform well on new driving cycles that are not part of the training dataset. For this reason, as will be discussed in Section V, different real-world driving cycles were included both in our training and our validation sets. When the training procedure is completed, the neural network is used to populate the two look-up tables related to the two EFs. The three parameters that the look-up tables take as input, i.e. SOC, battery temperature and distance yet to travel, are discretized and the optimal EFs are calculated by the NN for each grid point.

B. Online Controller

The online controller starts whenever the driver sets a new destination on the GPS. For each sampling time (which we set to one second), the battery SOC, temperature, and remaining distance are measured and fed into the look-up tables to obtain the two EFs $p_{1,on}$ and $p_{2,on}$. Afterwards, the controller selects the control variable α to minimize the Hamiltonian in real-time:

$$H(\sigma, T_\mathrm{b}, \alpha, p_{1,on}, p_{2,on}, t) = \dot{m}_\mathrm{f}(\alpha, t) + p_{1,on}\,\dot{\sigma}(\sigma, T_\mathrm{b}, \alpha, t) + p_{2,on}\,\dot{T}_\mathrm{b}(\sigma, T_\mathrm{b}, \alpha, t) \quad (32)$$

Note that in the online framework there is no need to track the co-state p_3 that was present in the offline Th-ECMS, since p_3 is only needed to find a solution of the BVP that does not violate the state constraints.

V. Results and Discussion

The results presented in this section were obtained using the vehicle model described in Section II. The main parameters of the vehicle and powertrain components are reported in Table I.

A. Offline Th-ECMS

First and foremost, we assess the validity of the offline solution procedure described in Section III. The procedure is implemented in MATLAB and tested on a number of driving cycles.

TABLE I
MAIN VEHICLE SPECIFICATIONS

Parameter	Value
Vehicle mass	1978 kg
ICE maximum power	243 kW
Electric machine maximum power	30 kW
Battery maximum power	27 kW
Battery energy capacity	1.3 kWh

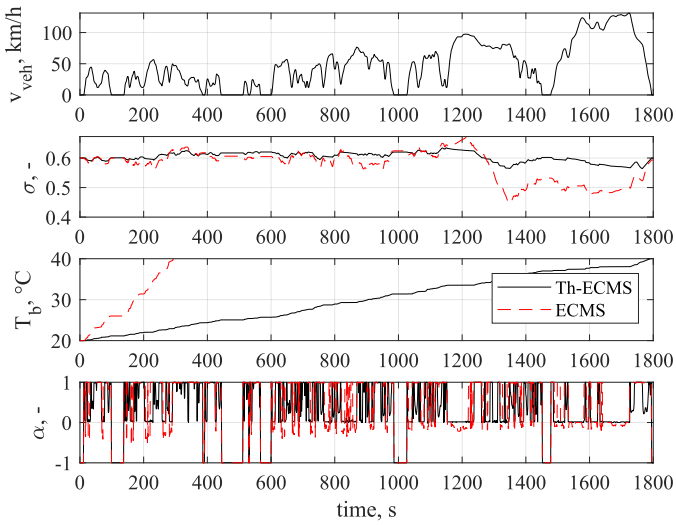

Fig. 4. Simulation profiles with a regular ECMS and the proposed formulation. The ECMS formulation proposed in this digest is labeled as *Th-ECMS*, while the regular implementation is denoted simply as ECMS.

TABLE II
FUEL ECONOMY OF THE PROPOSED TH-ECMS COMPARED TO THE DYNAMIC PROGRAMMING BENCHMARK

Driving cycle	Fuel economy, l/(100 km)		Difference
	Th-ECMS	DP	
UDDS	5.79	5.83	0.6 %
WLTP	6.67	6.70	0.4 %

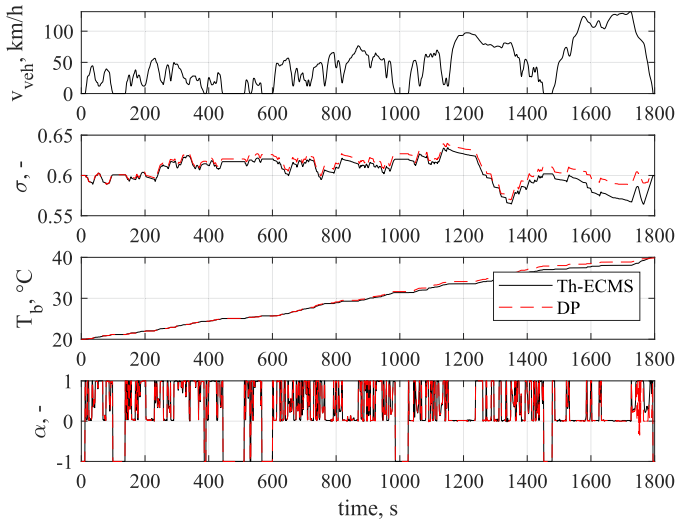

Fig. 5. Comparison of the Th-ECMS and the solution obtained with dynamic programming on the WLTP cycle.

Here, we report results for the WLTP (Worldwide Harmonised Light vehicles Test Procedure) and UDDS (Urban Dynamometer Driving Schedule) cycles. For each simulation, the parameters $p_1(t_0)$, $p_2(t_0)$, and $p_3(t_0)$ were calibrated using the PSO algorithm as described in the previous section in order to minimize fuel consumption while keeping the battery temperature below 40 °C, with an environment temperature of 20 °C. To minimize the Hamiltonian, the torque-split factor α was discretized with a uniformly spaced grid of 121 values ranging from -1 to 1. The PSO algorithm was implemented using the MATLAB function in [45], using default values for the parameters. The number of particles was set to 300.

The effectiveness of the proposed implementation is shown in Fig. 4. The Th-ECMS produces a solution hitting the constraints with great accuracy whereas a regular ECMS[5] implementation violates the battery temperature upper bound, making a larger use of the battery to maximize fuel economy and producing higher currents as a consequence. Furthermore, the regular ECMS produced a fuel economy of 6.46 l/100 km while the Th-ECMS 6.67 l/100 km. Since both are fuel-optimal, with the only difference being in the introduction of the battery temperature constraint, the gap between the two can be seen as the minimum gap that can be attained without introducing an active cooling system while keeping the battery under 40 °C, acting on the EMS design only.

[5]In the standard ECMS, no battery temperature constraint is imposed

As mentioned in the previous section, an approximate solution to the necessary conditions of optimality is obtained by the offline Th-ECMS. Moreover, the fact that these conditions are necessary but not sufficient does not guarantee global optimality but only local optimality. Hence, to assess the optimality of our method, the Th-ECMS results are compared with the solution obtained through dynamic programming (DP). While DP uses its own approximations to obtain a solution, it is based on both necessary and sufficient conditions of optimality and is therefore often used as a benchmark.

By searching through all possible discretized values of state (σ and T_b) and control (α) variables, this algorithm relies on Bellman's priciple of optimality to determine the optimal solution.

For our simulations, we used a dedicated MATLAB toolbox called DynaProg [46]. The control variable α was discretized with a uniformly spaced grid of 121 values ranging from -1 to 1 (as for the ECMS). For the purpose of the value function update and evaluation, the state variables (the battery SOC and temperature) were discretized with uniformly spaced grids of 801 values ranging from 0.4 to 0.8 and 212 values ranging from 20 °C to 41 °C, respectively.

As reported in Table II, the Th-ECMS comes very close to dynamic programming in terms of fuel economy: in both the WLTP and UDDS, the difference is below 0.6 %. The fact that the Th-ECMS produced a slightly lower fuel consumption can be explained by inspecting the simulation results shown in Figs. 5

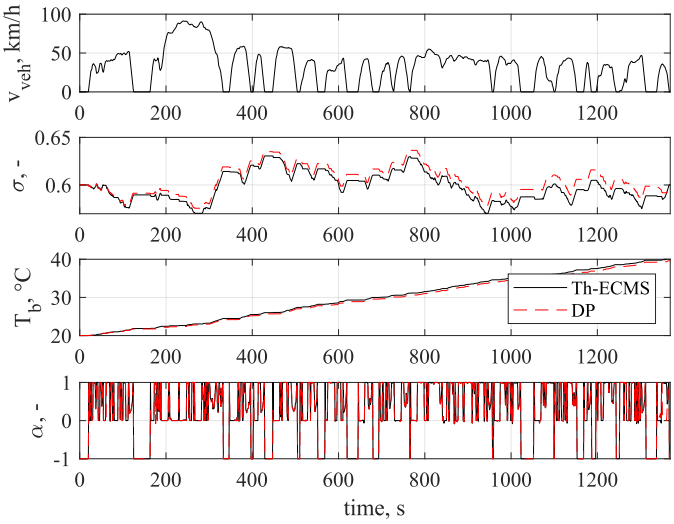

Fig. 6. Comparison of the Th-ECMS and the solution obtained with dynamic programming on the UDDS cycle.

and 6. Although the control and state profiles match quite well overall, there is a small but notable difference in the maximum temperatures reached: while the Th-ECMS reaches maximum temperatures of 40.3 °C and 40.2 °C on the UDDS and WLTP cycles respectively, dynamic programming only reaches 39.6 °C and 39.2 °C.

This is due to the difference in the way the two algorithms handle state constraints. The dynamic programming algorithm enforces the state constraint as hard constraints; moreover, the inherent characteristics of the value function approximation scheme tend to artificially penalize operating close[6] to the constraint boundary. On the other hand, the formulation of the minimum principle employed for the Th-ECMS treats the state constraints as something similar to a soft constraint, where violations are strongly penalized by the requirement that the additional state $\eta(t)$ remains zero for all t.

B. Online Th-ECMS

After assessing the offline solution method, the online implementation of the Th-ECMS described in Section IV was tested. To this end, a set of training and a set of validation driving cycles were defined.

The set of training cycles was composed of five regulatory cycles and five real-world driving cycles. The regulatory driving cycles include the Urban Dynamometer Driving Schedule (UDDS) and the HD-UDDS variant from the US EPA type approval procedure, which are representative of urban driving conditions, the Rural (ARDC) and Motorway (AMDC) cycles from the Common Artemis Driving Cycles (CADC) [47], and the Worldwide Harmonized Light Vehicles Test Cycle (WLTC) Class 3b cycle (used for type approval in Europe, Japan, India and other countries), which is composed of urban, rural and highway segments. The real-world driving cycles included in the training set are depicted in Fig. 7.

[6]The meaning of *close* here depends on the state grids' discretization.

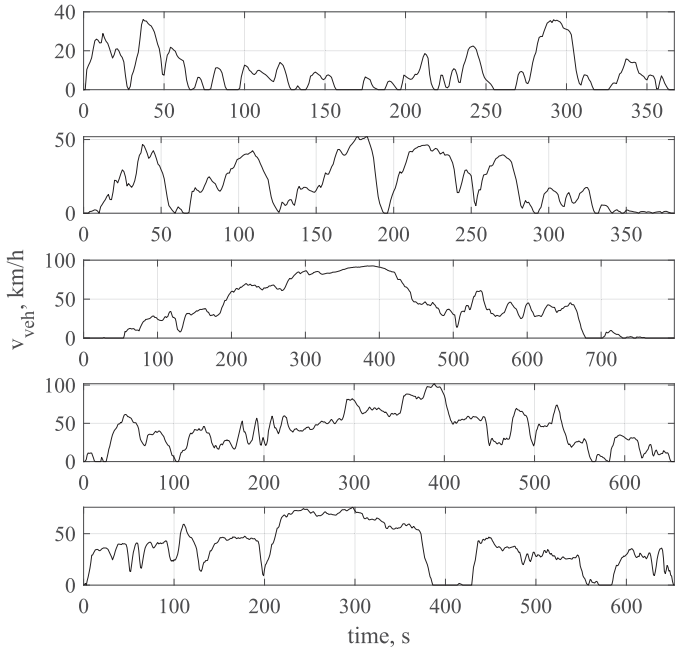

Fig. 7. Speed profiles of the real driving cycles included in the training set.

TABLE III
NEURAL NETWORK HYPERPARAMETERS

Hyperparameter	Value
Batch size	64
Learning rate	0.001
Number of hidden layers	3
Number of neurons per hidden layer	64
Number of neurons at input layer	3
Number of neurons at ouput layer	2
Activation function	ReLU
Optimizer	Adam
Number of epochs	5000

This training set was used to calibrate the look-up tables, running the offline solution method for each of them and using the corresponding co-state trajectories to train the neural network, as depicted in Fig. 3. More specifically, the offline solution method was run several times for each training cycle with different values of initial battery SOC (0.5, 0.55, 0.6, 0.65, and 0.7) and initial battery temperature (20 °C and 30 °C) to cover a broader range of operating conditions. The hyperparameters of the neural network used to calibrate the look-up tables are shown in Table III. The NN was implemented using MATLAB Deep Learning Toolbox [48].

The calibrated look-up tables map the equivalence factors as functions of the SOC, battery temperature, and distance yet to travel, as shown in Fig. 8. Although all three input variables influence the equivalence factor values, some noticeable trends can be highlighted. Since the equivalence factor $p_{1,on}$ affects the SOC dynamics, higher values of $p_{1,on}$ are selected at higher SOC values to encourage battery discharging, especially as the SOC approaches the upper bound. A similar trend can be noticed for the equivalence factor $p_{2,on}$ with respect to the battery temperature: high values of T_b correspond to high values of

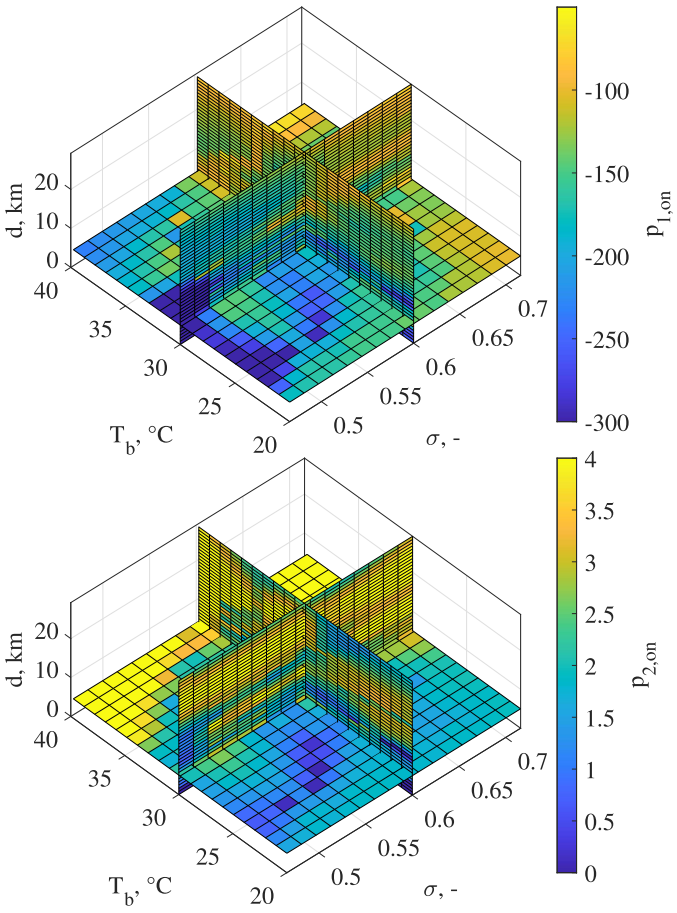

Fig. 8. Calibrated $p_{1,on}$ and $p_{2,on}$ look-up tables: the equivalence factors $p_{1,on}$ and $p_{2,on}$ are mapped as functions of SOC, battery temperature, and distance yet to travel.

TABLE IV
PERFORMANCE OF THE ONLINE TH-ECMS WITH RESPECT TO THE OFFLINE SOLUTION

Driving cycle	Difference wrt offline		
	Fuel economy	Final SOC σ_f	Final temperature $T_{b,f}$
JC08	+1.95 %	+0.018	-0.67 °C
AUDC	+0.03 %	-0.008	-0.10°C
CLUST3	+2.50 %	+0.009	-0.02 °C
CLUST4	+1.61 %	+0.017	-1.40 °C

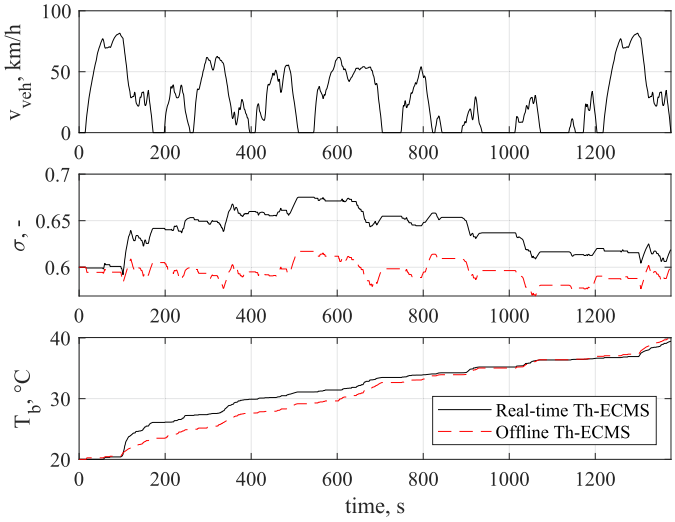

Fig. 9. Comparison of the offline and online Th-ECMS solutions on the JC08 cycle.

$p_{2,on}$, which prevents overheating. It is also worth noting that the higher the distance yet to travel, the higher the equivalence factor $p_{2,on}$ is. This behavior can be explained by the fact that if the vehicle needs to travel higher distances, the battery is more exposed to a potential temperature increase due to the higher energy demand over the remaning driving mission. Therefore, a higher $p_{2,on}$ is selected to keep the battery temperature under control.

Then, in order to assess the validity of the method in different driving conditions, the online controller with the calibrated look-up tables was tested with four cycles that compose the validation set. This set is composed of driving cycles that are not part of the training cycles set: two regulatory cycles and two real-world cycles. The two regulatory cycles are the Urban cycle from the Artemis cycles (AUDC) and the JC08 cycle, formerly part of the Japanese type approval procedure, which represents driving in congested city traffic). The two real-world cycles are referred to as CLUST3 and CLUST4. These real driving cycles have a few peculiarities that help broaden the range of test conditions with respect to the regulatory driving cycles. The CLUST3 cycle is an urban cycle which includes a long segment at low speed followed by a short phase with steep accelerations; the CLUST4 cycle is a urban/rural cycle which again includes steep accelerations and decelerations.

As can be seen in Table IV, the online Th-ECMS performs reasonably well with respect to the offline solution: the degradation in fuel economy remains below 2.5%, and the deviation in the state of charge at the end of the trip is within 0.02. The battery temperature never exceeds 40 °C, though it can get very close.

As shown by the time profiles in Figs. 9 to 12, the online controller tends to overcharge the battery on all cycles, except for the very last portion of the AUDC. The difference is never excessive though, and charge-sustaining behavior is always achieved within a reasonable tolerance. Nonetheless, this noticeable difference between the SOC profiles reflects the sub-optimality of the online controller's behavior and highlights room for further improvement.

The suboptimality of the solution is also evident from the AUDC test, where the online controller achieves a slight increase in fuel consumption despite reaching a lower terminal SOC with respect to the offline solution. Since the final temperature of the battery reaches almost exactly the 40 °C threshold, as is the case for the offline solution, this difference in fuel economy cannot be attributed to reduced usage of the battery and can be attributed instead to a less smart usage. Similar considerations can be made for the CLUST3 cycle, where a relatively large degradation in

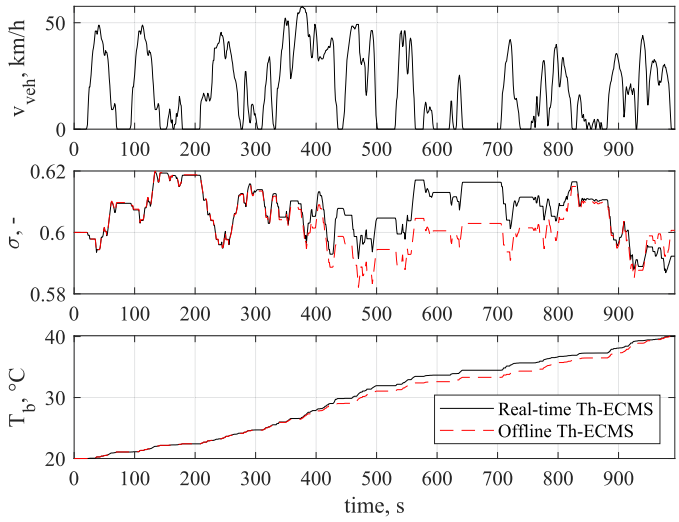

Fig. 10. Comparison of the offline and online Th-ECMS solutions on the AUDC cycle.

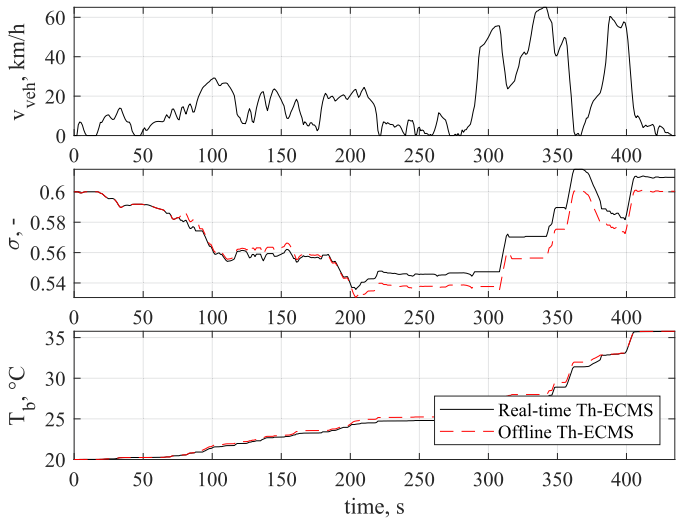

Fig. 11. Comparison of the offline and online Th-ECMS solutions on the CLUST3 cycle.

fuel economy is observed even when taking into account the small positive deviation in the final SOC.

Another interesting observation that can be drawn from these results is that the method is quite robust in meeting the temperature constraint. Indeed, even in the face of the relatively larger deviations in the SOC trajectory that can occur in some intermediate segments of the driving missions, the final temperature always meets the constraint as it tends to converge to the offline results towards the end of the trip. This shows that the look-up tables are quite robust in avoiding the constraints violation, even if they do not enable fully optimal operation in terms of fuel economy.

To illustrate the real-time capability of the Th-ECMS algorithm, the computational times for the test driving cycles are shown in Table V. The experiments were run on a computer with an 11th Gen Intel(R) Core(TM) i7-1165G7 processor, running

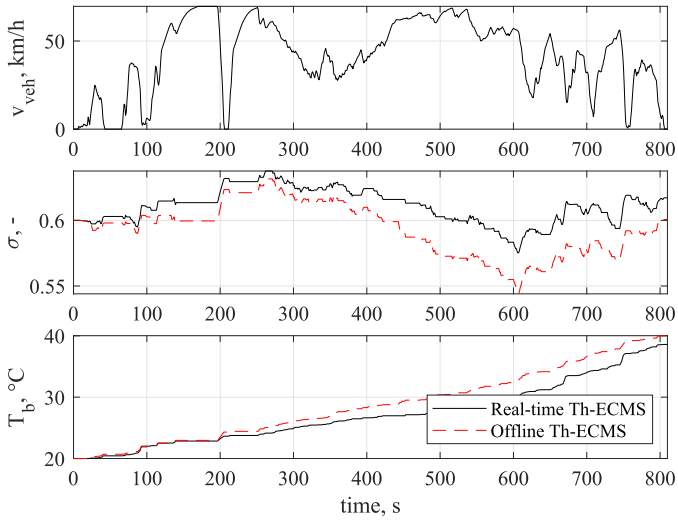

Fig. 12. Comparison of the offline and online Th-ECMS solutions on the CLUST4 cycle.

TABLE V
COMPUTATIONAL TIMES OBTAINED IN THE NUMERICAL EXPERIMENTS

Driving mission	Computational time
JC08	0.24 s
AUDC	0.15 s
CLUST3	0.07 s
CLUST4	0.14 s

at 2 GHz, with 16 GB of RAM. The online Th-ECMS proves to be computationally lightweight, achieving computational times lower than 0.3 s for all the driving cycles. Since all the simulations are carried out using the same discretization of the control variable, the computational time is proportional to the cycle length.

VI. CONCLUSION

In this article, an online energy management strategy is proposed that minimizes fuel consumption while keeping the battery temperature below an upper threshold. The proposed strategy (Th-ECMS) has a particular significance for the mild HEVs since they are typically equipped with a passive cooling system which can only provide a limited cooling power. In this context, a battery temperature-aware EMS becomes particularly useful to avoid thermal runaway and reduce battery aging while maximizing fuel economy.

In the first part of the article, a novel EMS with a constraint on the battery temperature is formulated as an optimal control problem and solved using Pontryagin's minimum principle. The offline solution method for solving the boundary value problem that arises from the application of the minimum principle adopts a derivative-free particle swarm optimization algorithm due to its suitability for multi-objective optimization, robustness with respect to a poor first guess, and low sensibility to the strong lack of smoothness in the problem.

In the second part of the article, the online implementation of Th-ECMS is introduced. The Th-ECMS selects the control variable in real-time thanks to two look-up tables that map the equivalence factors. Following and offline calibration process, the look-up tables can be used online since they only require as inputs the current state of charge, battery temperature, and the remaining trip length, which can be easily obtained from any consumer GPS system. Therefore, the strategy is computationally lightweight and does not rely on complex prediction algorithms to anticipate future driving conditions.

After calibrating the look-up tables on a training set of five regulatory drive cycles and five real-world driving missions, the online controller was tested on a validation set of four missions, which proved the method to be robust in meeting the temperature constraint. Nonetheless, the results also show that there is still room for improvement in terms of fuel economy.

To this end, several approaches could be pursued. The first approach would be to optimize the values of grid points of the look-up tables. A second approach would be to replace the remaining trip length as one of the look-up tables inputs with the estimated energy required to complete the trip that should better encode information about future driving conditions [49].

REFERENCES

[1] D. S. Cardoso, P. O. Fael, and A. Espírito-Santo, "A review of micro and mild hybrid systems," *Energy Rep.*, vol. 6, pp. 385–390, Feb. 2020.

[2] J. Dornoff, J. German, A. Deo, and A. Dimaratos, "Mild-hybrid vehicles: A near term technology trend for CO2 emissions reduction," *Int. Council Clean Transp.*, White paper, Jul. 2022. [Online]. Available: https://theicct.org/wp-content/uploads/2022/07/mild-hybrid-emissions-jul22.pdf

[3] Mordor Intelligence, "Mild hybrid vehicles market - growth, trends, Covid-19 impact, and forecasts (2022–2027)," Feb. 2022. Accessed: Mar. 2023. [Online]. Available: www.mordorintelligence.com/industry-reports/mild-hybrid-vehicles-market

[4] C. Yu et al., "Cost-efficient thermal management for a 48V li-ion battery in a mild hybrid electric vehicle," *Automot. Innov.*, vol. 1, no. 4, pp. 320–330, Nov. 2018.

[5] E. Hosseinzadeh, R. Genieser, D. Worwood, A. Barai, J. Marco, and P. Jennings, "A systematic approach for electrochemical-thermal modelling of a large format lithium-ion battery for electric vehicle application," *J. Power Sources*, vol. 382, pp. 77–94, Apr. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378775318301411

[6] S. Lee, J. Cherry, M. Safoutin, J. McDonald, and M. Olechiw, "Modeling and validation of 48V mild hybrid lithium-ion battery pack," *SAE Int. J. Altern. Powertrains*, vol. 7, no. 3, pp. 273–287, Apr. 2018.

[7] E. Ezemobi, G. Yakhshilikova, S. Ruzimov, L. M. Castellanos, and A. Tonoli, "Adaptive model predictive control including battery thermal limitations for fuel consumption reduction in P2 hybrid electric vehicles," *World Electric Veh. J.*, vol. 13, no. 2, Feb. 2022, Art. no. 33. [Online]. Available: https://www.mdpi.com/2032-6653/13/2/33

[8] J. Hall, S. Borman, B. Hibberd, M. Bassett, S. Reader, and M. Berger, "48V high-power battery pack for mild-hybrid electric powertrains," *SAE Int. J. Adv. Curr. Pract. Mobility*, vol. 2, no. 4, pp. 1893–1904, Apr. 2020.

[9] L. Thibault, A. Sciarretta, and P. Degeilh, "Reduction of pollutant emissions of diesel mild hybrid vehicles with an innovative energy management strategy," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2017, pp. 1274–1279.

[10] B. Xu, F. Malmir, D. Rathod, and Z. Filipi, "Real-time reinforcement learning optimized energy management for a 48V mild hybrid electric vehicle," SAE, Warrendale, PA, USA, Tech. Paper, 2019-01-1208, 2019, doi: 10.4271/2019-01-1208.

[11] E. Silvas, T. Hofman, N. Murgovski, P. Etman, and M. Steinbuch, "Review of optimization strategies for system-level design in hybrid electric vehicles," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 57–70, Jan. 2017.

[12] A. Sciarretta et al., "A control benchmark on the energy management of a plug-in hybrid electric vehicle," *Control Eng. Pract.*, vol. 29, pp. 287–298, Aug. 2014.

[13] R. M. Bagwe, A. Byerly, E. C. d. Santos, and Z. Ben-Miled, "Adaptive rule-based energy management strategy for a parallel HEV," *Energies*, vol. 12, no. 23, Nov. 2019, Art. no. 4472.

[14] M. Acquarone et al., "Influence of the reward function on the selection of reinforcement learning agents for hybrid electric vehicles real-time control," *Energies*, vol. 16, no. 6, 2023, Art. no. 2749.

[15] G. Paganelli, T. M. Guerra, S. Delprat, J.-J. Santin, M. Delhom, and E. Combes, "Simulation and assessment of power control strategies for a parallel hybrid car," *Proc. Inst. Mech. Eng. Part D: J. Automobile Eng.*, vol. 214, no. 7, pp. 705–717, Jul. 2000.

[16] Y. Wang and Z. Huang, "Optimization-based energy management strategy for a 48-V mild parallel hybrid electric power system," *J. Energy Resour. Technol.*, vol. 142, no. 5, Mar. 2020, Art. no. 052002.

[17] B. Zhou, A. Rezaei, and J. Burl, "Effect of battery temperature on fuel economy and battery aging when using the equivalent consumption minimization strategy for hybrid electric vehicles," SAE, Warrendale, PA, USA, Tech. Paper, 2020-01-1188, 2020, doi: 10.4271/2020-01-1188.

[18] Z. Chen, Y. Liu, M. Ye, Y. Zhang, Z. Chen, and G. Li, "A survey on key techniques and development perspectives of equivalent consumption minimisation strategy for hybrid electric vehicles," *Renewable Sustain. Energy Rev.*, vol. 151, Nov. 2021, Art. no. 111607.

[19] A. Chasse, A. Sciarretta, and J. Chauvin, "Online optimal control of a parallel hybrid with costate adaptation rule," *IFAC Proc. Volumes*, vol. 43, no. 7, pp. 99–104, Jul. 2010.

[20] S. Onori, L. Serrao, and G. Rizzoni, "Adaptive equivalent consumption minimization strategy for hybrid electric vehicles," in *Proc. Dyn. Syst. Control Conf.*, 2010, pp. 499–505.

[21] S. Qiu, L. Qiu, L. Qian, and P. Pisu, "Hierarchical energy management control strategies for connected hybrid electric vehicles considering efficiencies feedback," *Simul. Modelling Pract. Theory*, vol. 90, pp. 1–15, Jan. 2019.

[22] S. Onori, L. Serrao, and G. Rizzoni, *Hybrid Electric Vehicles*. Berlin, Germany: Springer, 2016.

[23] K. Koprubasi, "Modeling and control of a hybrid-electric vehicle for drivability and fuel economy improvements," Ph.D. dissertation, The Ohio State University, Columbus, OH, USA, 2008.

[24] Z. Lei, D. Qin, L. Hou, J. Peng, Y. Liu, and Z. Chen, "An adaptive equivalent consumption minimization strategy for plug-in hybrid electric vehicles based on traffic information," *Energy*, vol. 190, Jan. 2020, Art. no. 116409.

[25] X. Lin, Q. Feng, L. Mo, and H. Li, "Optimal adaptation equivalent factor of energy management strategy for plug-in CVT HEV," *Proc. Inst. Mech. Eng. Part D: J. Automobile Eng.*, vol. 233, no. 4, pp. 877–889, Mar. 2019.

[26] M. Sivertsson, C. Sundström, and L. Eriksson, "Adaptive control of a hybrid powertrain with map-based ECMS," *IFAC Proc. Volumes*, vol. 44, no. 1, pp. 2949–2954, Jan. 2011.

[27] S. Xie, X. Hu, S. Qi, and K. Lang, "An artificial neural network-enhanced energy management strategy for plug-in hybrid electric vehicles," *Energy*, vol. 163, pp. 837–848, Nov. 2018.

[28] Z. Chen, Y. Liu, Y. Zhang, Z. Lei, Z. Chen, and G. Li, "A neural network-based ECMS for optimized energy management of plug-in hybrid electric vehicles," *Energy*, vol. 243, Mar. 2022, Art. no. 122727.

[29] M. Acquarone, P. G. Anselma, F. Miretti, and D. Misul, "Battery temperature aware equivalent consumption minimization strategy for mild hybrid electric vehicle powertrains," in *Proc. IEEE Veh. Power Propulsion Conf.*, 2022, pp. 1–6.

[30] C. E. Chapin, "Road load measurement and dynamometer simulation using coastdown techniques," SAE, Warrendale, PA, USA, SAE Tech. Paper 810828, 1981, doi: 10.4271/810828.

[31] M. Ehsani, Y. Gao, S. E. Gay, and A. Emadi, *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles*. Boca Raton, FL, USA: CRC Press, 2004.

[32] L. Guzzella and A. Sciarretta, *Vehicle Propulsion Systems*. Berlin, Germany: Springer, 2013.

[33] S. Luciani, S. Feraco, A. Bonfitto, and A. Tonoli, "Hardware-in-the-loop assessment of a data-driven state of charge estimation method for lithium-ion batteries in hybrid vehicles," *Electronics*, vol. 10, no. 22, Jan. 2021, Art. no. 2828. [Online]. Available: https://www.mdpi.com/2079-9292/10/22/2828

[34] C. Forgez, D. V. Do, G. Friedrich, M. Morcrette, and C. Delacourt, "Thermal modeling of a cylindrical LiFePO4/graphite lithium-ion battery," *J. Power Sources*, vol. 195, no. 9, pp. 2961–2968, May 2010.

[35] T. Huria, M. Ceraolo, J. Gazzarri, and R. Jackey, "High fidelity electrical model with thermal dependence for characterization and simulation of high power lithium battery cells," in *Proc. IEEE Int. Electric Veh. Conf.*, 2012, pp. 1–8.

[36] L. Saw, Y. Ye, and A. Tay, "Electro-thermal characterization of lithium iron phosphate cell with equivalent circuit modeling," *Energy Convers. Manage.*, vol. 87, pp. 367–377, Nov. 2014.
[37] J. D. Bishop, M. E. Stettler, N. Molden, and A. M. Boies, "Engine maps of fuel use and emissions from transient driving cycles," *Appl. Energy*, vol. 183, pp. 202–217, Dec. 2016.
[38] J. Heywood, *Internal Combustion Engine Fundamentals*. New York, NY, USA: McGraw-Hill, 2018.
[39] H. Park, "A design of air flow configuration for cooling lithium ion battery in hybrid electric vehicles," *J. Power Sources*, vol. 239, pp. 30–36, Oct. 2013.
[40] D. E. Kirk, *Optimal Control Theory; an Introduction*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1970.
[41] C. R. Taylor, "Dynamic programming and the curses of dimensionality," in *Applications of Dynamic Programming to Agricultural Decision Problems*. Boca Raton, FL, USA: CRC Press, 2019, pp. 1–10.
[42] T. J. Böhme and B. Frank, *Hybrid Systems, Optimal Control and Hybrid Vehicles*. Berlin, Germany: Springer, 2017.
[43] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: An overview," *Soft Comput.*, vol. 22, pp. 387–408, 2018.
[44] A. G. Gad, "Particle swarm optimization algorithm and its applications: A systematic review," *Arch. Comput. Methods Eng.*, vol. 29, no. 5, pp. 2531–2561, Apr. 2022.
[45] "Particleswarm," 2023. [Online]. Available: https://it.mathworks.com/help/gads/particleswarm.html
[46] F. Miretti, D. Misul, and E. Spessa, "DynaProg: Deterministic dynamic programming solver for finite horizon multi-stage decision problems," *SoftwareX*, vol. 14, Jun. 2021, Art. no. 100690.
[47] M. André, "The ARTEMIS European driving cycles for measuring car pollutant emissions," *Sci. Total Environ.*, vol. 334/335, pp. 73–84, Dec. 2004. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0048969704003584
[48] "Deep learning toolbox," 2023. [Online]. Available: https://it.mathworks.com/help/deeplearning/ref/trainingoptions.html
[49] J. Holden, H. V. Til, E. Wood, L. Zhu, J. Gonder, and M. Shirk, "Trip energy estimation methodology and model based on real-world driving data for green-routing applications," *Transp. Res. Record: J. Transp. Res. Board*, vol. 2672, no. 24, pp. 41–48, Sep. 2018.

Matteo Acquarone (Graduate Student Member, IEEE) received the B.S. and M.S. degrees in mechanical engineering in 2019 and 2021, respectively, from the Politecnico di Torino, Turin, Italy, where he is currently working toward the Ph.D. degree in energetics. From 2021 to 2022, he was be a Visiting Scholar with Stanford University, Stanford, CA, USA. He is the author of five peer-reviewed conference and journal publications. His research interests include development of artificial intelligence control algorithms for energy management strategies of hybrid electric vehicles and energy savings of connected vehicles.

Federico Miretti (Member, IEEE) received the M.S. and Ph.D. degrees in energetics from the Politecnico di Torino, Turin, Italy, in 2018 and 2022, respectively. He is currently a Postdoctoral Research Fellow with the Politecnico di Torino. He is the author of 13 peer-reviewed conference and journal publications. His research interests include the design of optimization-based energy management strategies for hybrid-electric power trains as well as autonomous and connected vehicles.

Pier Giuseppe Anselma (Member, IEEE) received the M.S. and Ph.D. degrees (Hons.) in mechanical engineering from the Politecnico di Torino, Turin, Italy, in 2017 and 2021, respectively. In 2016 and 2019, he was a master's Student Researcher and a Visiting Ph.D. Student Researcher with the McMaster Automotive Resource Centre (MARC), McMaster University, Hamilton, ON, Canada. He is currently a Postdoctoral Research Fellow with the Politecnico di Torino. He is the author or coauthor of more than 60 peer-reviewed international conferences and journal publications. He holds one patent. His research interests include the design of propulsion and brake systems and the development of smart energy management strategies related to electrified and automated road vehicles. He was the recipient of three awards for his research, including the First Prize at the TRA Visions 2020 Young Researcher Competition, the Siebel Scholar, Class of 2021, and the IEEE Italy Section ABB 2022 Ph.D. Thesis Award.

Daniela Anna Misul received the graduation degree (*cum laude)* in mechanical engineering from the Politecnico di Torino, Turin, Italy, in 1997. She is currently an Associate Professor with the Energy Department of Politecnico di Torino. The scientific production of Prof. Misul consists of several publications, covering the following research streams: SI engine technology for natural gas and bi-fuel ICEs, model-based control approaches for Diesel ICEs, diagnostic and predictive 0D/1D tools for evaluating injection, combustion process and emission formation in diesel and SI engines, in-cylinder fluid-dynamics and combustion (CFD) of ICEs, optimization of hybrid architectures, CFD modeling activities for turbomachinery, AI algorithms for xEVs control and ADAS applications. She has also been contributing to the coordination of the research activities within the PT-ERC (Polito - Engine Research Center) Research Group with specific reference to the scientific responsibility and coordination of some of the research contracts as well as to team building and fund-raising activities. She held the scientific responsibility of National and International research projects, ruled through partnership agreements with companies and/or public private bodies, which are leaders in their own sector, and she has also been recently coordinating activities related to the use of AI for smart application in the automotive sector. She is also coordinating activities within the Center for Automotive Research and Sustainable mobility@PoliTO (CARS@PoliTO) Group.

Open Access funding provided by 'Politecnico di Torino' within the CRUI CARE Agreement