energies

MDPI

Article

Novel Approaches for Energy Management Strategies of Hybrid Electric Vehicles and Comparison with Conventional Solutions

Fabrizio Donatantonio [1,†], Alessandro Ferrara [1,‡], Pierpaolo Polverino [1,*], Ivan Arsie [2] and Cesare Pianese [1]

[1] Department of Industrial Engineering, University of Salerno, via Giovanni Paolo II 132, 84084 Fisciano, Italy; fabrizio.donatantonio@gmail.com (F.D.); alessandro.ferrara@tuwien.ac.at (A.F.); pianese@unisa.it (C.P.)
[2] Department of Engineering, University of Naples "Parthenope", Centro Direzionale, 80143 Napoli, Italy; ivan.arsie@uniparthenope.it
* Correspondence: ppolverino@unisa.it; Tel.: +39-089-96-4178
† Currently at Kineton s.r.l., via Gianturco 23, 84142 Napoli, Italy.
‡ Currently at Institute of Mechanics and Mechatronics, Division of Process Control and Automation, TU Wien, Getreidemarkt 9/E325, 1060 Vienna, Austria.

Abstract: Well-designed energy management strategies are essential for the good operation of Hybrid Electric Vehicles (HEVs) in terms of fuel economy and pollutant emissions reduction, regardless of the specific powertrain architecture. The goal of this paper is to propose two innovative supervisory control strategies for HEVs derived from different optimization algorithms and to assess HEVs' fuel consumption reduction (compared to conventional vehicles). These approaches are derived from the literature and modified by the authors to present novel algorithms for the optimization problem. One is based on Dynamic Programming (DP), here referred to as the Forward Approach to Dynamic Programming (FADP) and introduces a different implementation of the DP to achieve computational and accuracy benefits. The other is based on the Equivalent Consumption Minimization Strategy (ECMS) approach, and it adapts to the latest driving conditions using information gathered in a finite-length backward-looking horizon. These techniques are used to achieve the optimal power share between the thermal engine and the battery of a parallel HEV. Their performances are compared and analysed in terms of achieved fuel economy and computational time with respect to conventional DP and Pontryagin's Minimum Principle (PMP) approaches.

Keywords: energy management; hybrid electric vehicles; optimization; Dynamic Programming; Pontryagin's Minimum Principle; equivalent consumption minimization strategy

Citation: Donatantonio, F.; Ferrara, A.; Polverino, P.; Arsie, I.; Pianese, C. Novel Approaches for Energy Management Strategies of Hybrid Electric Vehicles and Comparison with Conventional Solutions. *Energies* **2022**, *15*, 1972. https://doi.org/10.3390/en15061972

Academic Editor: Fabrice Locment

Received: 14 December 2021
Accepted: 1 March 2022
Published: 8 March 2022

1. Introduction

In the current world energy scenario, the transport sector accounts for the highest share (one third) of final energy consumption. Indeed, it has been reported that, in 2014, passenger cars alone used more energy than the entire residential sector; moreover, together with freight road vehicles, they accounted for almost a third of global energy-related CO_2 emissions [1]. Therefore, the transition to road vehicles that are not powered by fossil fuels, such as battery electric vehicles (BEVs) and fuel cell electric vehicles (FCEVs), will have a huge positive impact on global CO_2 emissions, as long as electricity and hydrogen, respectively, are produced with renewable energy sources. Nevertheless, several decades will be required to complete this transition because there is still the need for cost reduction, reliability increase and infrastructure development.

In this context, it is authors' opinion that hybrid electric vehicles (HEVs) could represent one of the most feasible and fastest ways in the short term to reduce the impact of the transportation sector on global energy-related CO_2 emissions, combining the advantages of the conventional (i.e., based on internal combustion engines) vehicles with those of BEVs. According to the HEV architecture, it is possible to achieve some of the following improvements [2]:

- downsize the engine to achieve higher average efficiency;
- recover energy during braking instead of dissipating it as friction losses;
- optimize the power split between prime movers;
- eliminate idle fuel consumption by turning off the engine (stop-and-start);
- reduce clutching losses decoupling the engine from the road.

HEVs were firstly classified in two basic types: series or parallel. In series HEVs, the electric motor alone drives the vehicle with the electricity supplied by the battery or by the engine-driven generator. In parallel HEVs, on the contrary, both prime movers can power the vehicle individually or simultaneously.

When the battery is recharged by the grid, like in BEVs, the HEV is referred to as plug-in (PHEV). This vehicle is not limited to charge sustaining operation as non-rechargeable hybrids. Therefore, the battery can be discharged to its lower limit to cover the whole driving mission length. However, the range that PHEVs can cover in pure electric mode is limited.

Choosing the best hybrid powertrain architecture and size of the components for a given vehicle is a trivial task. Indeed, every architecture has strengths and weaknesses that might be relevant depending on the vehicle size, usage, and final cost. Once the architecture is established, the control structure should be defined to obtain the greatest benefits in term of fuel economy increase as well as emissions and life-cycle cost reduction.

In the design of a hybrid powertrain, optimization has a key role in the entire development process. Three main optimization levels can be identified:

- *structural optimization* to find the best possible powertrain layout architecture;
- *parametric optimization* to find the best possible parameters for the components of a fixed structure;
- *control system optimization* to find the best possible supervisory control algorithms.

The structural optimization for HEVs essentially consists in the choice of the best powertrain architecture (i.e., series, parallel, series-parallel or combined hybrid). Whereas, the parametric optimization consists in the optimal sizing of the engine, of the battery pack, of the electric motor (or motors) and of the gearbox gear ratios. Regardless of the configuration, the potential fuel economy improvement and emission reduction rely entirely on the power distribution within the hybrid powertrain. Hence, there is the need for a supervisory controller to determine how the power demand distributes among the powertrain components, providing the set points for all the low-level controllers.

Choosing the proper control strategy, that can be classified in this context as Energy Management Strategy (EMS), is complex and requires the knowledge of both vehicle architecture and driving mission profile, with great burden on computational effort for the online implementation of such strategies [3].

The contribution of this work is to propose and evaluate the performance of two novel supervisory control algorithms: a Forward Approach to Dynamic Programming (FADP) and an adaptation of a causal Equivalent Consumption Minimization Strategy (ECMS-based). The main advantages and drawbacks of such algorithms are compared to conventional optimization techniques: Dynamic Programming (DP) and Pontryagin's Minimum Principle (PMP) are used as reference since they are well established and are commonly used offline to find the optimal control law of HEVs over an assigned driving mission.

The paper is structured as follows: Section 2 briefly addresses the description of the model used to simulate the HEV operation, based on information already available in the literature. Section 3 presents an initial overview on conventional HEVs supervisory control strategies (i.e., DP and PMP) followed by a more detailed description of the novel strategies proposed in this work (i.e., FADP and ECMS-based). In Section 4, a critical analysis of the performances of each optimization technique is performed accounting for required computational time and achieved fuel economy over several driving missions.

2. Model of the Hybrid Electric Vehicle

The hybrid vehicle considered in this study presents a parallel P3 hybrid powertrain configuration, as already presented in [4]. The base vehicle mass is 1400 kg, and the frontal area is 2 m^2, whereas the other key parameters required for the simulation are taken from [5]. The electrical machine is located on the gearbox output shaft so that the mechanical power provided by the electric motor is not affected by engine efficiency. With this architecture, pure electric drive is possible. Such configuration has been chosen instead of a P1 configuration (with the electric motor located nearby the thermal engine) since it shows a lower fuel consumption during urban driving due to higher mechanical efficiency and allows regenerative braking in any driving condition (also when the clutch is not engaged) [6]. Nevertheless, it can be stated that the algorithms proposed in this work can be applied to any architecture without loss of generality.

The model proposed here is a backward-facing one, meaning that the driving force required to move the vehicle is directly computed from the desired velocity trajectory [7]. The driving force propagates backward through the driveline towards the internal combustion engine and the electrical machine. The battery pack consists of 30 branches in parallel, each made of 24 cells connected in series. The battery charging and discharging internal resistances are taken from [8]. The single battery is a $LiFePO_4$ cell manufactured by A123 systems [9], with a nominal voltage of 3.3 V, a nominal capacity of 1.1 Ah and a weight of 39 g. The transmission is a manual 5-speed manufactured by TREMEC model TR-2450 [10]. The mechanical and electrical transmission components are modelled with a constant behaviour, whereas the efficiencies of the internal combustion engine (SI, 1.2 L, 4 in-line cylinders, 105 Nm at 3500 rpm) and the electrical machine (130 Nm max torque, 9000 rpm max speed, 3200 rpm transition speed) are modelled through maps depending on shaft speed and torque [4]. The main drivetrain transmission parameters are taken from [5,10]. Further details on the hybrid vehicle model used in this work can be found in [4].

The main disadvantage of the considered backward approach is that the speed reference trajectory is always assumed perfectly tracked by the vehicle, regardless of the powertrain capabilities (i.e., maximum available torque). Moreover, approximating a transient condition as a sequence of steady states (quasi-static approach) reduces the model accuracy. Nevertheless, it is worth remarking that the simplicity and computational lightness of the proposed model along with the absence of the necessity of a powertrain control structure (including the driver) represent a strong advantage in terms of model usage for control strategies' design and application. Therefore, considering the great computational benefits, these approximations are considered acceptable for the purpose of this work, which aims at performing a preliminary estimation of HEVs' fuel consumption [11]. Moreover, powertrain dynamics can be neglected since it is much faster than the battery state of charge dynamics [12].

3. Supervisory Control Algorithms

A conventional powertrain based on an internal combustion engine has zero degrees of freedom (DOF) on the power demand distribution: all the requests are satisfied by the engine. On the other hand, a hybrid powertrain usually has one DOF, so that the power demand can be optimally distributed (optimal power split) among its components to achieve the maximum improvement in terms of fuel economy and emissions reduction.

The control task of optimal power split is referred to as supervisory control or EMS [11]. The strategies that can be adopted by the supervisory controller can be divided into two main classes: optimal and heuristic. The former uses optimal control theory to formulate a problem that can be solved afterwards, whereas the latter follows rule-based indications calibrated beforehand mainly through optimal control techniques.

Another classification can be performed based on the information used: non-causal and causal. Non-causal control strategies require the perfect knowledge of the inputs (e.g., the road load) and thus assume that they are completely known. On the other hand,

causal strategies use information up to the current instant only. All heuristic strategies are inherently causal, whereas optimal strategies can either be causal or non-causal.

Generally, causal controllers are neither able to find the optimal solution nor to satisfy the constraint on the final state-of-charge (SoC) of the battery. This latter aspect is particularly relevant when it comes to the comparison between different strategies: if the SoCs at the end of the mission are different, the comparison is meaningless.

The main novelty brought by this paper to what is currently available in the literature consists in two novel supervisory control algorithms, being a Forward Approach to Dynamic Programming (FADP) and an adaptation of a causal Equivalent Consumption Minimization Strategy (ECMS-based). The key features and pros and cons of the proposed novel algorithms are described in details and compared to two conventional optimization techniques, which are Dynamic Programming (DP) and Pontryagin's Minimum Principle (PMP).

All considered techniques are non-causal, except the ECMS-based one, which uses current and past information only. Both DP [13–17] and PMP [18,19] techniques are well established and have been extensively presented in the literature for hybrid powertrain optimization, and are here therefore used to evaluate the performance of the newly proposed FADP and ECMS-based approaches. In the following, a brief overview on optimal control problem formulation is given, followed by the introduction of the bases of all the considered methods (both conventional and novel) and the analysis of their performance for hybrid powertrain optimization.

3.1. Optimal Control Problem Statement

The optimal solution to the energy management problem is the one that minimizes (or maximizes) a certain performance index J, suitably defined for each specific problem. The simplest performance index formulation is the one that considers only the fuel mass consumed to realize the driving mission:

$$J = \int_0^{t_f} \dot{m}_f(t, x(t), u(t))dt \tag{1}$$

where x and u are the system state and control variables respectively. Although further factors might be considered in the performance index (e.g., pollutants emissions, battery aging and degradation), this work aims only at minimizing the vehicle fuel consumption.

Several state variables are required to describe the dynamics of HEV systems. Nevertheless, for the purpose of energy management, only the battery SoC is considered as a relevant state variable to be accounted for (i.e., $x = \xi$), whose evolution presents the initial condition $\xi(0) = \xi_0$.

For the case of an HEV under exam, the control variable adopted throughout this paper is the battery power (i.e., $u = P_b$), whose minimum and maximum values depend on both the cell maximum charge and discharge currents and battery pack configuration, along with the battery SoC:

$$P_{b,min}(\xi) \leq P_b(t) \leq P_{b,max}(\xi) \tag{2}$$

The gear engaged is always chosen as the one giving the lowest fuel consumption among those of the feasible set, keeping engine speed within its minimum and maximum values.

The SoC should stay within admissible boundaries during the whole mission $\xi_{min} \leq \xi(t) \leq \xi_{max}$ and meet the constraint on its target value ξ_t at the end of the driving mission (i.e., $\xi(t_f) = \xi_t$).

To consider the constraint on the final SoC, a terminal cost (or penalty) ϕ is usually added to the performance index J:

$$J = \int_0^{t_f} \dot{m}_f(t, \xi(t), P_b(t))dt + \phi(\xi_t) \tag{3}$$

3.2. Dynamic Programming

Dynamic programming (DP) was firstly introduced in 1957 as a solution for multi-stage decision problems in which time plays a significant role and the order of operations may be crucial [20]. DP has widely been used in literature to solve optimal control problems for its capability to handle complex multiple constraints and to guarantee global optimality of the provided solution.

The main drawbacks of the DP are its heavy computational burden, in terms of both memory usage and CPU operations, which increase exponentially with the number of state and control variables of the system and linearly with the mission time span.

The practical implementation of DP algorithms requires the discretization of time and of the feasible sets of state and control variables. Consequently, the solution found with the algorithm will be optimal only up to the error introduced by this discretization.

Once all the variables have been discretized, the overall cost of the mission, given by Equation (3), can be written as the sum of the costs of the N time steps and the final cost:

$$J = \sum_{k=0}^{N-1} \dot{m}_f\left(t^k, P_b^k\right) + \phi\left(\xi^N\right) \tag{4}$$

The constraint on the final SoC is handled as a hard one, yielding to:

$$\phi\left(\xi^N\right) = \begin{cases} 0 & \text{for } \xi^N = \xi_t \\ \infty & \text{otherwise} \end{cases} \tag{5}$$

Starting from the end of the mission, the DP algorithm proceeds backward in time to find and store the optimal control value $P_{b,opt}^k(\xi_i)$ on any state grid point ξ_i at any time t^k, as in Equation (6), and the cost-to-go $J^k(\xi_i)$, that is the minimum cost to go from state ξ_i at time t^k to state ξ_t at time t^N:

$$\begin{aligned} J^k(\xi_i) &= \min_{P_b^k \in V}\left(J^{k+1}\left(f_i^{k+1}\right) + \dot{m}_f\left(t^k, P_b^k\right)\right) \\ P_{b,opt}^k(\xi_i) &= \arg\min_{P_b^k \in V}\left(J^{k+1}\left(f_i^{k+1}\right) + \dot{m}_f\left(t^k, P_b^k\right)\right) \end{aligned} \tag{6}$$

where V is the set of all control variable grid values and f_i represents the value that the SoC would assume according to the control variable P_b^k at the next time step starting from ξ_i:

$$f_i^{k+1} = \xi_i + \dot{\xi}\left(\xi_i, P_b^k\right)\Delta t \tag{7}$$

At time t^N, where the algorithm starts, the *costs-to-go* are known and equal to zero when $\xi_i = \xi_t$ and infinite otherwise, according to Equation (5).

The cost-to-go $J^{k+1}(f_i^{k+1})$ must be interpolated across grid points or approximated with its nearest neighbour because the arrival states f_i^{k+1}, defined by Equation (7), generally do not correspond to the state grid points ξ_i. Once $J^{k+1}(f_i^{k+1})$ is available, it is possible to solve Equation (6) and find the optimal control value $P_{b,opt}^k(\xi_i)$ among the values in the feasible set V.

Once the optimal control is found for all i and k, starting from $\xi_{opt}^0 = \xi_0$, the optimal state trajectory is reconstructed proceeding forward in time:

$$\xi_{opt}^{k+1} = \xi_{opt}^k + \dot{\xi}\left(\xi_{opt}^k, P_{b,opt}^k\left(\xi_{opt}^k\right)\right)\Delta t \tag{8}$$

Since the optimal state ξ_{opt}^k generally do not correspond to the state grid points ξ_i, the optimal control value $P_{b,opt}^k\left(\xi_{opt}^k\right)$ must be evaluated by interpolation or by approximation to its nearest neighbour.

Further detail on Dynamic Programming can be found in [21,22], whereas its application to the hybrid vehicle problem can be found, e.g., in [2,13,15–17,23–26].

3.3. Pontryagin's Minimum Principle

Pontryagin's Minimum Principle (PMP) is a result of optimal control theory that helps identifying necessary optimality conditions for control (and, thus, state) variables of a given dynamic system. Since the optimality conditions are not sufficient, the control candidate identified might not lead to the global optimum. Nevertheless, the results obtained through the application of the PMP could be relatively close to global optimum computed through DP [12]. Further theoretical details on the principle could be found in [21], while examples of its application to the HEV powertrain control can be found, e.g., in [5,19,27–30].

Given the optimal control problem discussed in the previous section with respect to DP, it is possible to define the Hamiltonian function as:

$$H(\xi(t), P_b(t), \lambda(t)) = \dot{m}_f(P_b(t), t) + \lambda(t)\dot{\xi}(\xi(t), P_b(t)) \tag{9}$$

where $\lambda(t)$ is the co-state. The PMP states that a necessary condition for optimality is that the Hamiltonian is minimum in the control variable for all the time window:

$$H\left(\xi_{opt}(t), P_{b,opt}(t), \lambda_{opt}(t)\right) \leq H(\xi(t), P_b(t), \lambda(t)) \; \forall t \in \left[0, t_f\right] \; \& \; \forall P_b(t) \in V \tag{10}$$

and the optimal state and co-state evolve according to the following equations:

$$\begin{cases} \dot{\xi}(t) = \frac{\partial H}{\partial \lambda} = -\frac{I_b(\xi, P_b(t), t)}{Q_b} \\ \xi(0) = \xi_0 \end{cases} \tag{11}$$

$$\dot{\lambda}(t) = -\frac{\partial H}{\partial \lambda} \tag{12}$$

The compliance of the constraint on the final SoC depends on the initial co-state value, i.e., the initial condition of Equation (12). This should be found iteratively (e.g., through the shooting method) until the final state of charge converges to its target value. Subsequently, the Hamiltonian minimization and the integration of Equations (11) and (12) are performed several times until convergence is reached.

To speed up the computation by avoiding the resolution of a minimization problem every time step, the Hamiltonian is evaluated on a finite set of values of the control variable P_b of dimension Q, as presented in [31].

3.4. Forward Approach to Dynamic Programming

From a theoretical point of view, DP algorithms can run forward or backward in time without any difference: the optimal solution will be the same regardless of the approach. The practical implementation of DP to solve optimal control problems requires the discretization of the control variable, leading to the backward procedure described in Section 3.2. DP algorithms involve the use of either the rounding or the interpolation method, with further reduction in the accuracy of the solution and increase in the computational time. Moreover, there is no guarantee that, during the forward-calculation phase of DP, a feasible control trajectory is generated or that such control will lead to the target state at the end of the mission. This issue is particularly problematic in problems where the optimal state trajectory lies along or close to the boundary of its feasible set.

In this paper, a novel procedure is proposed to avoid the discretization of the control variable. This procedure is only possible when a bijective correspondence between state transitions and control variables is present. The proposed algorithm can run forward or backward in time without distinction.

The transition from state ξ_i to ξ_j is expressed as:

$$\Delta\xi_{ij} = \xi_i - \xi_j \tag{13}$$

where the indexes i and j are both related to the same state grid points. A certain cost is required to realize the transition $\Delta\xi_{ij}$ and, at the same time, satisfy the power request at the wheels. This cost is referred to as transition cost and is indicated as L_{ij}. The bijective correspondence between state transitions and control variables is necessary to guarantee that to each $\Delta\xi_{ij}$ corresponds only one possible L_{ij}.

The procedure to evaluate the transition costs requires particular care and it will be detailed in the following. Nevertheless, once these costs are known, the optimal control problem can be basically approached following the same reasoning detailed for the simplified problem discussed in the previous sections.

The algorithm here proposed starts from the initial time step setting the initial cost to infinity for values of ξ different from the initial condition:

$$J_i^0 = \begin{cases} 0 & \text{for } \xi_i = \xi_s \\ \infty & \text{otherwise} \end{cases} \tag{14}$$

For the sake of notation compactness, a slight change in the notation is adopted: J_i^k represents the cost-to-arrive to ξ_i at time t^k starting from the initial state at the beginning of the mission.

With reference to Figure 1, the computation proceeds forward in time solving the minimization:

$$J_j^{k+1} = \min_i \left(J_i^k + L_{ij}^k \right) \tag{15}$$

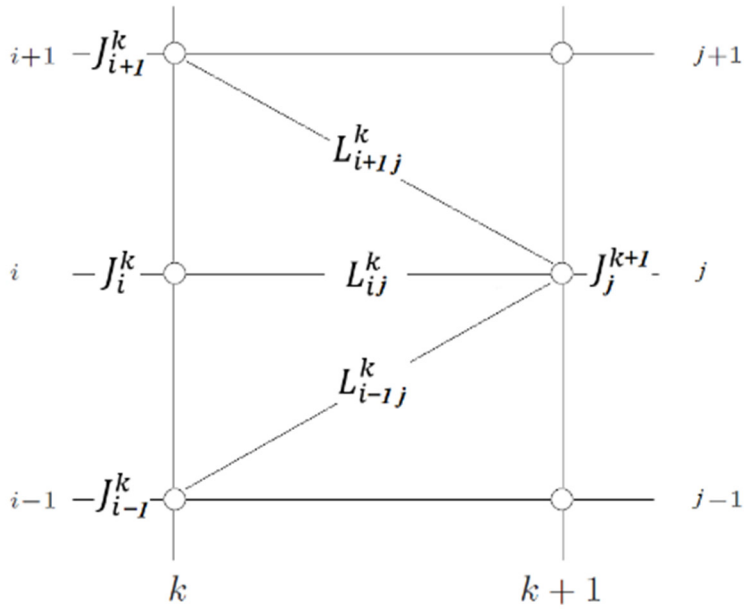

Figure 1. Schematic representation of the Forward Approach to Dynamic Programming (FADP).

This algorithm basically finds the optimal state value ξ_i from which it is convenient to start in order to arrive at ξ_j, satisfying the power request at the wheels. The arguments of the minimization in Equation (15) are stored in a feedback matrix X, whose elements are defined as:

$$X_j^{k+1} = \min_i \left(J_i^k + L_{ij}^k \right) \tag{16}$$

Each entry X_j^{k+1} represents the index i to which corresponds the optimal j state grid value to start from in order to arrive at ξ_j at t^{k+1} with the minimum cost-to-arrive. These values are used to reconstruct the optimal state trajectory $\xi_{opt}(t)$ with a backward procedure,

starting from $j = t$ and $\xi_{opt}^k = \xi_t$, and proceeding backward with k that goes from $N-1$ to 1 according to the following sequence:

$$\begin{array}{c} i = X_j^{k+1} \\ \xi_{opt}^k = \xi_i \\ j = i \end{array} \tag{17}$$

The general implementation of the FADP once each transition cost is known and stored in memory is expressed in the Appendix A (see Algorithm A1). The transition cost calculation depends on the specific problem, and it will be detailed in the next section for an HEV optimal control application.

It is worth noting that the output of this optimization is the optimal state trajectory over time. Nevertheless, since there is a bijective correspondence between each state transition and the control value needed to realize it and satisfy the power request, the optimal control law can be easily calculated.

3.4.1. Transition Cost Calculation

The algorithm described in the previous section is general and could be used for any optimal control problem as an alternative to DP. In this section, the transition cost calculation for the specific case of the HEV energy management problem previously described is presented.

Once the state transitions $\Delta\xi_{ij}$ and the time step Δt have been defined, the rate of charge/discharge is evaluated as:

$$\dot{\xi}_{ij} = \frac{\Delta\xi_{ij}}{\Delta t} \tag{18}$$

and the battery power P_b can be then calculated. At this stage, the compliance with the constraints like those in Equation (2) can be checked.

Once the battery power has been computed, the electric machine mechanical power can be evaluated and the power required to the thermal engine can be finally calculated as the difference between the required wheel power and the electric machine mechanical power.

From engine speed and torque, the fuel consumption $\dot{m}_f$ can be evaluated along with the cost-to-arrive L_{ij}^k. In the Appendix A (Algorithm A2), the whole process of the cost-to-arrive calculation is reported.

3.5. Iterative Approach in FADP

To reduce the computational burden of FADP while preserving the accuracy of the solution, an iterative process is here proposed (IFADP). The main idea is to progressively narrow the state grid in the neighbourhood of the optimal state trajectory obtained in a previous iteration performed on a coarser grid.

Similarly, to what shown in Section 3.4, the state grid is discretized into M grid points:

$$\xi_i = \xi_{down} + \Delta\xi \cdot i \quad \text{for } i = 0, 1, \ldots, M \tag{19}$$

while the upper ξ_{up} and lower ξ_{down} boundaries of the state grid are defined as:

$$\begin{cases} \xi_{up}(t) = \xi_{opt}(t) + \Delta\xi^* \\ \xi_{down}(t) = \xi_{opt}(t) - \Delta\xi^* \end{cases} \tag{20}$$

where $\Delta\xi^*$ is the chosen amplitude of the neighbourhood region related the optimal state trajectory found previously. Hence, the state grid step $\Delta\xi$ can be calculated as:

$$\Delta\xi = \frac{\xi_{up} - \xi_{down}}{M} = \frac{2\Delta\xi^*}{M} \tag{21}$$

At each iteration, the amplitude of the neighbourhood $\Delta\xi^*$ is progressively reduced and, as a consequence, a better fuel economy is achieved. On the other hand, the number of state grid elements M is kept constant to not increase the computational burden.

3.6. Causal ECMS-Based Strategy

In many practical applications dealing with optimal supervisory control of HEVs, the co-state λ introduced with the PMP method can be treated as a constant. The assumption is motivated by the weak dependency of the Hamiltonian on the SoC ξ. In fact, while the fuel flow rate $\dot{m}_f$ does not depend on ξ at all, the battery current depends on ξ through the battery internal resistance and open circuit voltage [4]. Nonetheless, especially for charge sustaining control policies, since ξ fluctuates around an intermediate value, the battery parameters are approximately constant if the range is narrow enough. Subsequently, the dependency of the Hamiltonian H on ξ is weak and can thus be neglected.

According to [19], the Hamiltonian could be redefined as the sum of the fuel chemical power plus the battery electrical power, weighted by an equivalence factor s_0, represented by the constant co-state:

$$H(\xi, P_b, s) = P_f + s_0 \cdot P_b \tag{22}$$

Similarly, the Hamiltonian could be redefined as the sum of the fuel mass consumed by the engine and an equivalent fuel mass consumed by the electric motor:

$$H(\xi, P_b, s) = \dot{m}_f + s_0 \cdot \dot{m}_{EQ} \tag{23}$$

It is worth noting that if $\dot{m}_{EQ} = P_b/H_i$, Equation (23) corresponds to Equation (22).

Nevertheless, in this work, the equivalent fuel consumption is considered equal to the state dynamics:

$$\dot{m}_{EQ} = \dot{\xi}(P_b) \tag{24}$$

thus, the Hamiltonian is closer to the one defined in Equation (9).

The equivalence factor drives the solution alternatively towards battery charging (i.e., high s_0 values) or battery depleting (i.e., low s_0 values). The ECMS-based technique usually minimizes the Hamiltonian while adjusting the equivalent factor dynamically over time using, for instance, a feedback signal on the battery SoC:

$$s = s_0 + \Delta s(\xi, \ldots) \tag{25}$$

This allows keeping ξ within a given range of values and obtaining a reduction of fuel consumption comparable with that obtained through a non-causal method, such as in [31–33]. The constant baseline value s_0 is normally obtained beforehand through an offline optimization using a non-causal method.

In this paper, s_0 is estimated using past driving information and a feedback signal from the battery SoC:

$$s(t) = s_c\left(UD_{t_h}(t)\right) + s_p(\xi(t) - \xi_t) \tag{26}$$

The first term s_c estimates the equivalence factor according to the driving conditions (e.g., urban, sub-urban, highway, etc.) through the Urban Degree UD:

$$s_c\left(UD_{t_h}\right) = \begin{cases} s_{c1} + s_{c2}\frac{UD_{t_h}}{UD_0} & \text{for } UD_{t_h} < UD_0 \\ s_{c1} + s_{c2} & \text{or } UD_{t_h} \geq UD_0 \end{cases} \tag{27}$$

This latter term expresses if a specific driving cycle shows more "urban" conditions or not. It is calculated on the backward horizon (t_h,t) as:

$$UD_{t_h} = \frac{\int_{t_h}^{t} \left|\frac{dv}{dt}\right| dt}{\int_{t_h}^{t} v dt} \tag{28}$$

and its value is greater for urban driving cycles and smaller for highway ones. The second term s_p takes into account the difference between the current SoC and its target value ξ_t.

The parameters presented for this control algorithm, that are s_{c1}, s_{c2}, s_p, UD_0 and t_h, have to be identified for each powertrain configuration through heuristic approaches or iteratively solving the optimal control problem with non-causal techniques.

Analogously to what has been done for PMP in Section 3.3, the control variable P_b is discretized into a finite set of dimension Q on which the Hamiltonian is evaluated to find its minimum value.

The parameters of the causal ECMS-based control strategy were heuristically determined and listed in Table 1, according to the hybrid powertrain configuration defined in the previous sections.

Table 1. Parameters of the causal ECMS-based control strategy.

Parameter	Unit	Value
s_{c1}	-	-2
s_{c2}	-	0.4
s_p	-	20
UD_0	Hz	0.13
t_h	s	120

4. Simulation Results

The performances of each control strategy will now be assessed over several driving missions in terms of achieved fuel economy and required computational time. All the results showed in the following section will refer to charge sustaining strategies: the battery SoC target at the end of the mission is set equal to the initial one ($\xi_t = \xi_0 = 0.6$). The driving cycles accounted in this work are listed in Table 2, with the corresponding *UD* value evaluated on the whole cycle through Equation (28) and ordered in descending *UD*. More information on the accounted driving cycles can be found in [34].

Table 2. Driving cycles considered for the assessment and comparison of the different control algorithms performance.

Driving Cycle	Urban Degree	Duration	Distance
New York City Cycle (NYCC)	0.121	598 s	1.89 km
Common Artemis Driving Cycles (CADC urban)	0.099	993 s	4.974 km
Federal Test Procedure (FTP-72)	0.045	1372 s	12.07 km
Worldwide Harmonized Light Vehicles Test Cycle (WLTC class 3)	0.027	1800 s	23.266 km
Common Artemis Driving Cycles (CADC rural)	0.024	1082 s	17.275 km
Highway Fuel Economy Test (HWFET)	0.008	756 s	16.45 km

The Relative Fuel Economy (RFE) is here assumed as the reference parameter for strategies' assessment and comparison. Such parameter expresses the percentage reduction of the total mass of fuel required by the hybrid vehicle to realize the driving mission with respect to that required by the conventional vehicle (i.e., considering the same powertrain without the electrical components weight):

$$RFE = 1 - \frac{\int_0^{t_f} \dot{m}_{f,HEV} dt}{\int_{t_h}^{t} \dot{m}_{f,conv} dt} \tag{29}$$

A further analysis is performed on the computational effort required by the proposed strategies. In Table 3, the complexity of each algorithm is reported as the number of evaluations of the cost function. Specifically, the terms N, M and Q are the number of

time, state, and control discrete elements, respectively, and i is the number of iterations (where applicable).

Table 3. Complexity of each considered optimal supervisory control strategy.

Optimal Control Strategy	Complexity
DP	$N \cdot M \cdot Q$
PMP	$N \cdot Q \cdot i$
FADP	$N \cdot M^2$
IFADP	$N \cdot M^2 \cdot i$
Causal ECMS-based	$N \cdot Q$

4.1. Analysis of DP Performance

Solving the optimal control problems through DP requires the calculation of the *cost-to-go* from every battery SoC at every time instant. The solution obtained with this method reaches the absolute optimum, compatibly with the state and control discretization. The main drawback consists in the high amount of resources required in terms of both memory and computational time. If the number of state elements M is set equal to the number of control elements Q, the computational time increases quadratically with the number of elements (as shown for the FADP in Table 3).

The optimal SoC trajectory obtained through DP on a WLTP driving cycle is shown in Figure 2. In this figure, on the left y-axis it is reported the SoC value is related to the red SoC trajectory line, whereas the right y-axis relates to the cost-to-go values. It can be observed that the control algorithm tends to recharge the battery during the Low, Medium and High speed of the cycle (i.e., from 0 to almost 1600 s, in urban and extra-urban conditions) and then depletes the battery in the last part of the cycle, when an extra high speed is encountered (i.e., on the highway). For what concerns the cost-to-go values (proposed as a contour plot right beneath the SoC trajectory), they decrease over time and with higher SoC values. This latter result is easily explained by considering that a greater SoC with respect to the one to be achieved as target value implies a reduction in the overall fuel consumption.

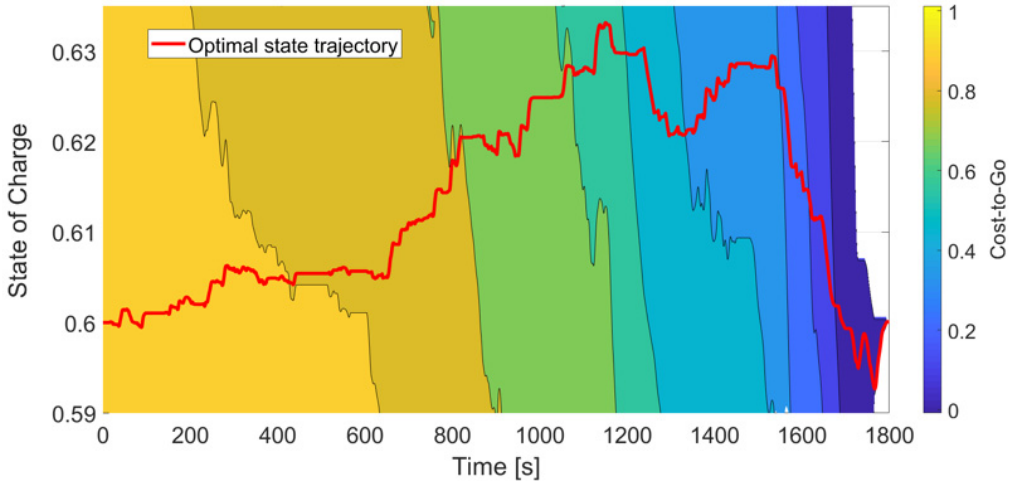

Figure 2. Cost-to-go map and optimal state trajectory obtained with DP optimization on the WLTP driving cycle ($Q = M = 2000$).

Table 4 lists the RFE achieved through the DP optimization for the driving cycles reported in Table 2. As expected, higher RFE are obtained on urban driving cycles (e.g., NYCC and CADC urban) with higher values of *UD* (calculated over the whole driving mission). This is explained with the more frequent braking phases (where energy is recovered) and the lower average efficiency of the thermal engine in low load conditions.

Table 4. Results of DP ($Q = M = 2000$), PMP ($Q = 2000$) and FADP ($M = 2000$) applied to the driving cycles listed in Table 2 in terms of Relative Fuel Economy (RFE).

Driving Cycle	RFE_{DP}	RFE_{PMP}	RFE_{FADP}
NYCC	31.6%	31.6%	31.6%
CADC urban	29.5%	29.5%	29.5%
FTP-72	18.1%	18.1%	18.1%
WLTP class 3	11.5%	11.5%	11.5%
CADC rural	11.8%	11.8%	11.8%
HWFET	1.2%	1.2%	1.2%

On the other hand, on highway missions, the RFE is very limited due to the almost complete absence of braking phases and the high efficiency achieved by the thermal engine alone on the conventional vehicle while operating at medium–high loads. The limited efficiency benefits, along with the additional weight due to the presence of the battery pack, nearly nullify the benefits brought by the hybrid powertrain configuration on highway scenarios.

4.2. Analysis of PMP Performance

As already described in Section 3.3, using PMP entails the replacement of the original boundary value problem (with fixed initial and final conditions on the SoC) with an initial value problem, where the initial condition on the co-state λ is iteratively updated through a shooting method until the final condition on the SoC ξ is met.

Clearly, higher absolute values of the co-state lead to higher values of the final SoC, favouring the recharge mode and penalizing the discharge mode. To find the instantaneous minimum value of the Hamiltonian, the control variable P_b is discretized into a finite set of equally distant points Q within its feasible range (i.e., from its current minimum to its maximum). The Hamiltonian is then evaluated for each possible operating point. Thus, the optimal control is the one that meets the constraints and corresponds to the minimum value of the Hamiltonian.

The computational time required to solve the optimal control problem is proportional to the number of iterations to reach the desired battery SoC at the end of the mission. It increases linearly with the number of elements and iterations in accordance with the estimation given in Table 3. However, if the co-state starts from a proper initial guess and it is suitably updated, just few iterations are required to meet the constraint on the SoC final value within an acceptable tolerance. The above comments are graphically represented in Figure 3 (example related to the results obtained on the FTP-72 cycle), where the linear increase in the computational time with the number of discrete control elements can be clearly visible. However, it can also be observed that the relative fuel economy improvement becomes negligible above a certain number of control elements (2000 in the example), proving that the increase in the computational efforts does not result in further improvement in the optimal solution.

Figure 4 presents the optimal SoC trajectory obtained with PMP on the WLTP driving cycle, to allow a visual comparison with the results obtained with the DP. From this figure, it can be observed that the optimal SoC trajectory is almost the same of that presented in Figure 2, with only hardly detectable changes. Additionally, in this case, the control algorithm tends to recharge the battery in almost the entire cycle and depletes it during highway driving.

Table 4 reports the RFE results obtained through the PMP optimization for several driving missions. Since PMP obtains the same RFE results as DP (Table 4), it is possible to state that this technique allows to find the absolute optimal control on the problem under analysis. Therefore, PMP will be adopted as a comparative reference for the analysis presented in the following with respect to the novel proposed approaches.

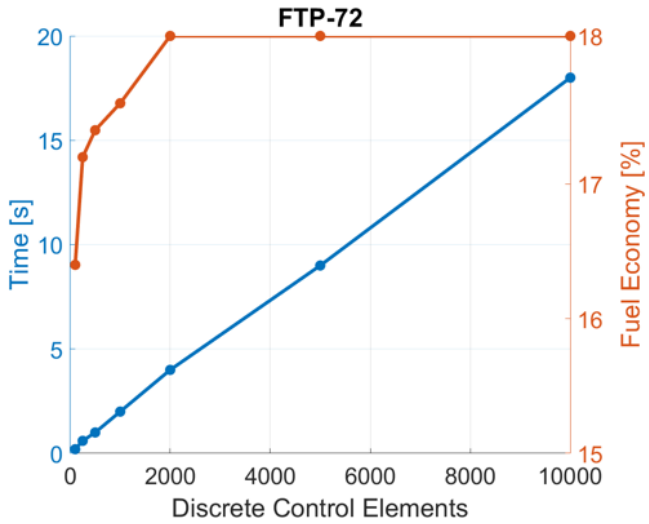

Figure 3. Influence of the number of the discrete control elements Q on computational time per iteration and fuel economy achieved through PMP on the FTP-72 driving cycle.

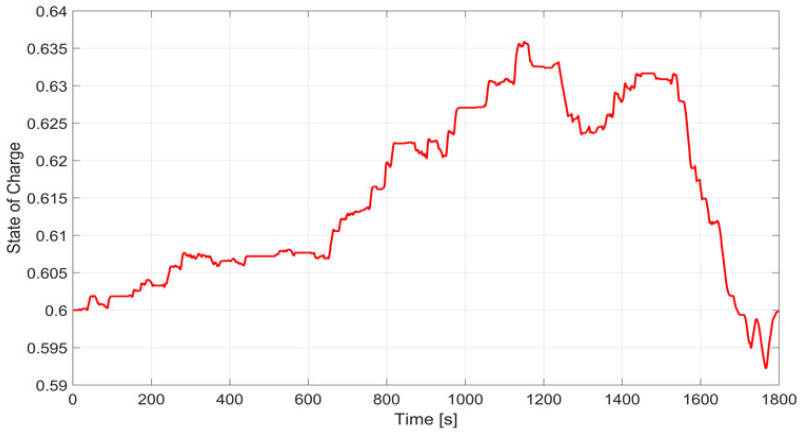

Figure 4. Optimal state of charge trajectory obtained through PMP on the WLTP driving cycle with Q = 2000.

4.3. Analysis of FADP Performance

Approaching the solution of HEV optimal control problem through FADP algorithms allows to evaluate the cost-to-arrive to each battery SoC at every time instants, as reported in Figure 5. The evaluation of the cost-to-arrive over time can be particularly useful in the benchmarking process of causal EMS.

Figure 5 presents the optimal trajectories of the battery SoC obtained with FADP approach. Compared to that of PMP, it can be observed that only minor deviations in the achieved optimal state trajectories are present, mainly due to the discretization of the state and/or control variables.

Figure 6 shows the influence of the number of discrete state elements M on the required computational time and the fuel economy achieved through the resulting optimal control law (the results are presented with respect to the NYCC driving cycle). It is evident that the computational time increases quadratically with the number of elements, in accordance with the estimation given in Table 3, but the maximum fuel economy reaches also in this case a limit value that does not increase with the number of state elements above 1000. The choice of the NYCC driving cycle in this case is performed to give a wider analysis of the results, demonstrating that the achievement of the optimal solution does not require a high computational effort independently from the chosen driving cycle.

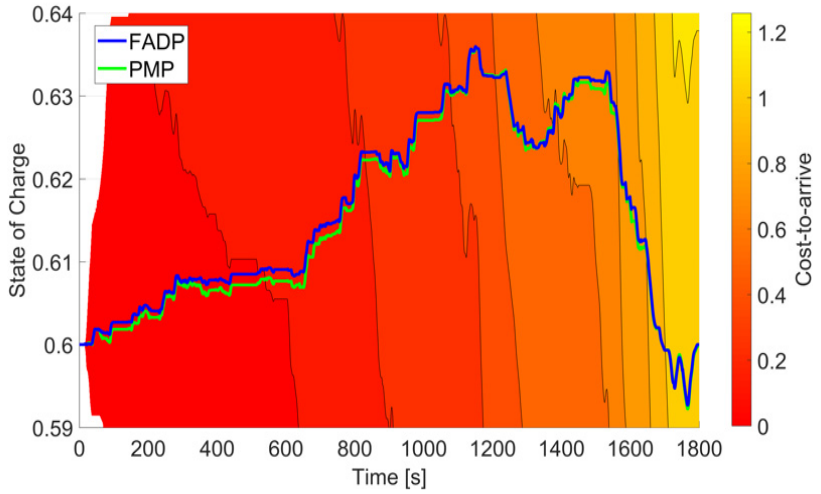

Figure 5. Cost-to-arrive map and optimal state trajectories obtained with FADP optimizations on the WLTP driving cycle (M = 2000). The SoC trajectory obtained with FADP (blue solid line) is compared with that achieved with PMP (cyan solid line).

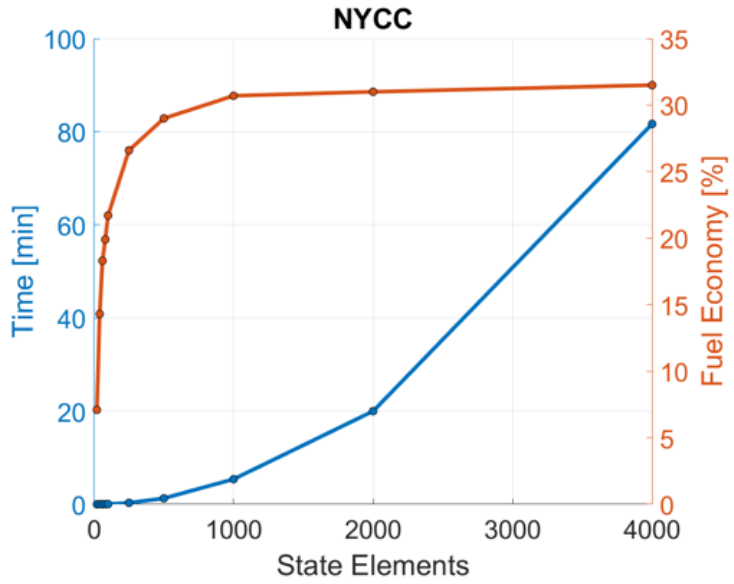

Figure 6. Influence of the number of the discrete state elements on the computational and fuel economy achieved through FADP on the NYCC driving cycle.

Addressing the use of the iterative approach, the SoC optimal trajectory obtained with IFADP is shown in Figure 7 with respect to three successive iterations. The procedure is performed by restraining the search space of the current iteration in a progressively narrower band (of amplitude $\Delta\xi^*$) around the optimal trajectory obtained with the previous iteration (as described in Section 3.5). The values of $\Delta\xi^*$ adopted to obtain the results in Figure 7 are 0.1, 0.01 and 0.005, for the three consecutive iterations. The result achieved with the third iteration coincides with the trajectory already presented in Figure 5. Therefore, after few iterations, the same fuel economy of an FADP with a greater state element number is attained, even though with a reduced computational time. This advancement can be observed from Figure 8, where the influence of the number of discrete state elements M on the required computational time and the fuel economy for the IFADP approach is reported with respect to the NYCC driving cycle. Comparing Figure 8 with Figure 6, it can be seen that, with a given number of state elements, e.g., M = 2000, the IFADP approach achieves the same fuel economy of the FADP in half the required time.

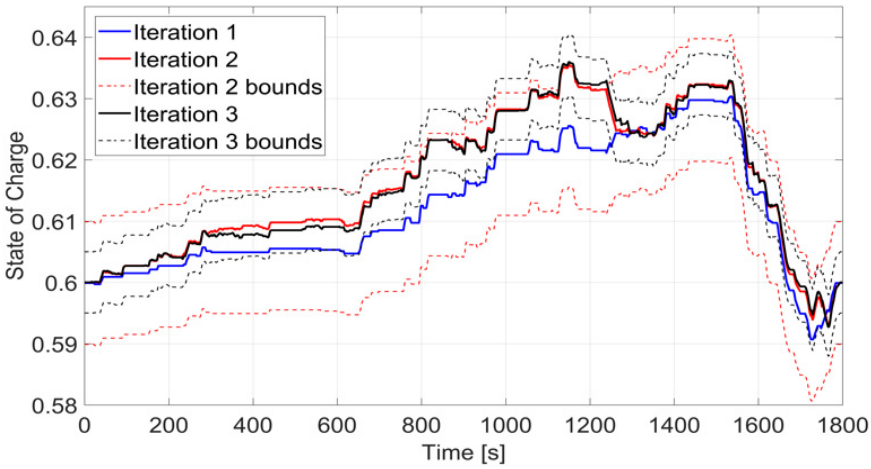

Figure 7. Optimal State of Charge for IFADP, the progressive restriction of the search domain in successive iterations can be observed. The search space of Iteration 1 is the whole feasible range of the battery state of charge $\xi \in (0.5, 0.7)$ ($M = 2000$; WLTP driving cycle).

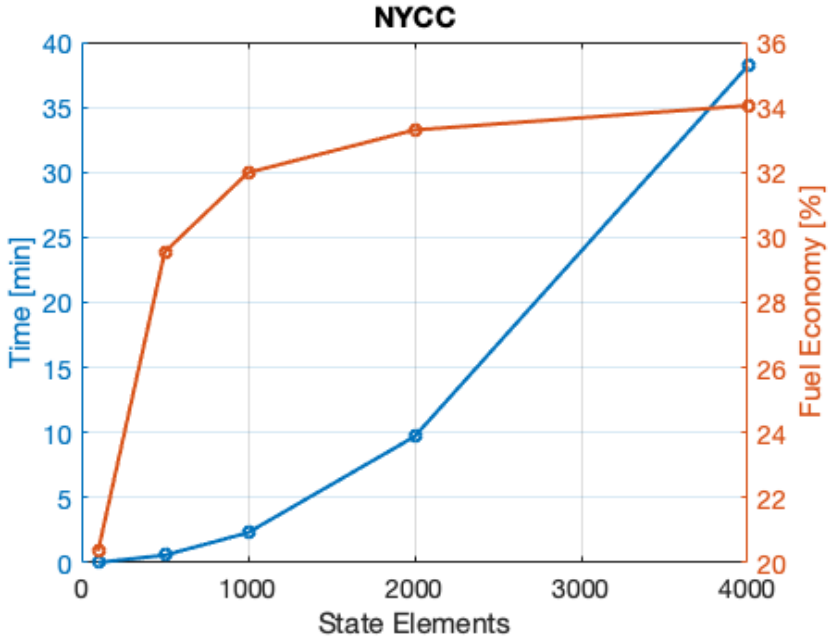

Figure 8. Influence of the number of the discrete state elements on the computational and fuel economy achieved through IFADP on the NYCC driving cycle.

Table 4 reports the RFE results obtained for all investigated driving cycles. It is worth noting that the difference between the optimal state trajectories in Figure 5 does not lead to different results in term of fuel saving (see Table 4).

4.4. Analysis of Causal ECMS-Based Strategy Performance

Similarly to what has been obtained for PMP, to solve the optimization problem through the ECMS-based approach, the control variable P_b is again discretized into Q points, in which the Hamiltonian is evaluated to find its minimum value.

The computational time required to find the sub-optimal SoC trajectory, and thus the battery power control law over the whole driving mission, is comparable with the one required by one iteration of the PMP (see Table 3). The main advantage is that the proposed causal ECMS-based technique can be implemented online since only information up to the current time instant is required.

Figure 9 shows the SoC trajectory obtained with the ECMS-based strategy (green solid line) compared to that of the DP (blue solid line), PMP (red dashed line), FADP (dotted line) and IFADP (dot-dashed line) approaches. It is important to notice that both methods achieve the same final value of SoC $\xi(t_f) = \xi_t$ (although not always guaranteed, as already explained, with the ECMS-based approach), but following two completely different SoC trajectories. However, although this is a substantial difference, the achieved relative fuel economies (reported in Table 5) are very close.

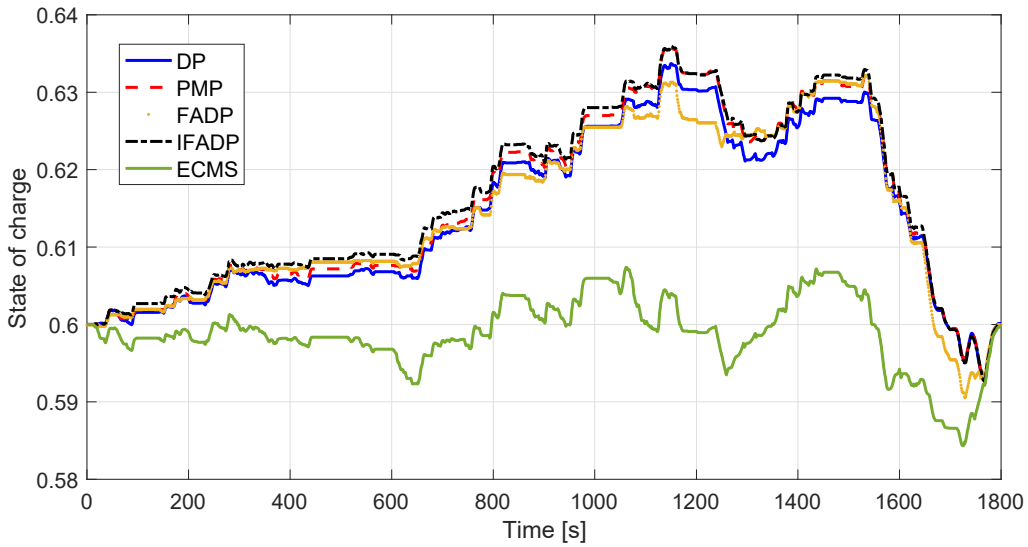

Figure 9. State of charge trajectories through the presented ECMS-based technique (green line) compared to those of DP (blue line), PMP (red dashed line), FADP (dotted line) and IFADP (dot-dashed line) approaches on the WLTP driving cycle.

Table 5. Results of Causal ECMS-based (Q = 1000), applied to the driving cycles listed in Table 2 in terms of Relative Fuel Economy (RFE).

Driving Cycle	Final SoC	RFE_{ECMS}	RFE_{EMCS}/RFE_{PMP}
NYCC	0.5933	41.9%	100%
CADC urban	0.5952	31.8%	99%
FTP-72	0.5989	17.6%	97%
WLTP class 3	0.5997	11.0%	97%
CADC rural	0.6078	9.7%	100%
HWFET	0.6078	−1.4%	100%

To make a fair comparison and to evaluate the distance from the equivalent optimal solution achieved with the PMP approach, the optimal control problem has been solved through PMP by setting the SoC final value ($\xi(t_f)$ reported in Table 5) as the same final value obtained through the ECMS-based technique for each driving cycle. The ratio between the RFE achieved with ECMS, and that of the PMP, has been calculated and reported in the last column of Table 5. The proximity to the optimal fuel consumption is confirmed by the fact that these ratios are all greater than or equal to 97%.

Nevertheless, the RFE values reported in Table 5 highlight some substantial differences with respect to the other techniques presented, due to the differences in the final state of charge. For example, the negative relative fuel economy, with respect to the conventional vehicle, on the HWFET test cycle can be explained by assuming that the battery SoC increases by nearly 1% during the driving mission. This result suggests that a considerable amount of energy from the fuel is used to recharge the battery, similarly to what happens by applying the PMP (as confirmed by the RFE ratio of 100%). Moreover, as shown in Figure 10, it can be observed that the change in Relative Fuel Economy with respect to

the number of control elements is negligible with a value of Q greater than 1000. With a lower elements number, the algorithm is less stable and accurate and the constraint on the final SoC cannot be respected anymore, thus enhancing the RFE change due to battery energy consumption.

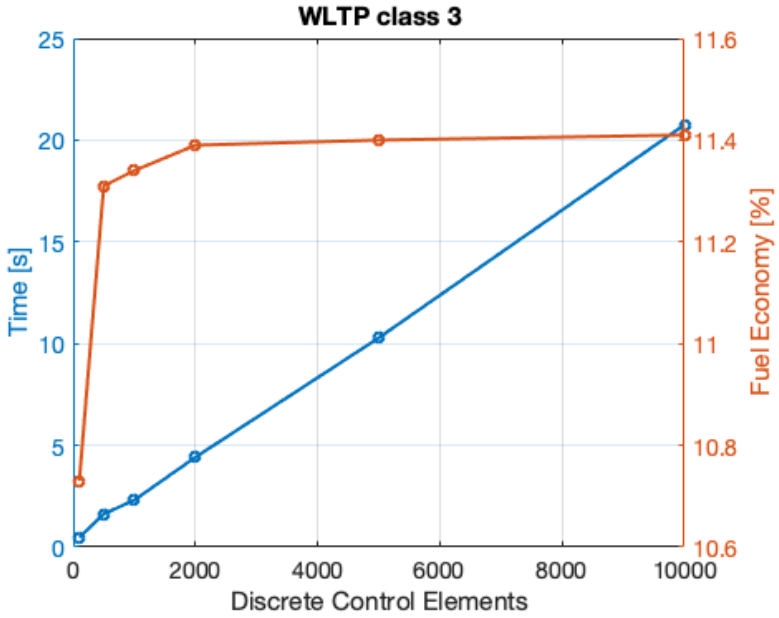

Figure 10. Influence of the number of the discrete control elements on the computational and fuel economy achieved through the ECMS-based technique on the WLTP driving cycle.

The differences between causal and non-causal techniques are particularly evident on short driving cycles. On the other hand, it is possible to practically show the equivalence, and hence the effectiveness, of the ECMS-based optimization in the long run by running multiple repetitions of the same test cycle and evaluating the overall fuel consumption. By doing so, the SoC difference, and thus its fuel equivalent, is compensated by the higher amount of fuel used. These results are shown in Table 6, where the RFE values line up with the ones achieved through the other presented non-causal optimization methods.

Table 6. Results of Causal ECMS-based (Q = 1000) applied to the driving cycles listed in Table 2 in terms of Relative Fuel Economy (RFE) with multiple repetitions of the test cycles.

Driving Cycle	N. of Repetitions	Relative Fuel Economy
NYCC	100	31.5%
CADC urban	100	28.8%
FTP-72	10	17.4%
WLTP class 3	10	11.3%
CADC rural	10	10.9%
HWFET	10	1.1%

5. Conclusions

The paper presented a simulation analysis of four supervisory control strategies for Hybrid Electric Vehicles (HEVs) based on different optimization algorithms, two well-known in the literature, being the Dynamic Programming (DP) and the Pontryagin's Minimum Principle (PMP), and two new algorithms proposed by the authors and modified from the literature, being the Forward Approach to Dynamic Programming (FADP) and the Equivalent Consumption Minimization Strategy-based (ECMS-based).

The techniques' performance comparison has been made focusing on the optimal power distribution between the components of a parallel HEV, and the achieved results have been analysed in terms of fuel economy and computational time. The results obtained with

the FADP, DP and PMP optimizations are in accordance with each other, and all the methods have shown to be able to reach the absolute optimum, although it is worth recalling that absolute optimality cannot be guaranteed with the application of PMP alone. Nevertheless, due to its capability to find the absolute optimum and of its computational speed, PMP stands out as the ideal candidate for the integration of energy management strategies.

For all the test cycles, the results achieved in terms of relative fuel economy through the causal ECMS-based technique are always comparable to the ones of non-causal strategies, which is proof of the goodness of the procedure despite the sub-optimality of the solution and the differences in the state of charge trajectory.

The presented FADP technique shows comparable computational time with respect to the well-established DP and the capability to attain the absolute optimum. Moreover, FADP results are expressed in terms of cost-to-arrive rather than cost-to-go (as in DP), which makes this technique particularly useful in the benchmarking process of causal energy management strategies. Clearly, since FADP and DP are slower than real time, they cannot be used for on-board control applications (along with the fact that they are non-causal). Conversely, PMP and causal ECMS-based optimizations are faster than real time and, hence, are suitable for such applications.

In conclusion, in the current automotive scenario, the ideal candidate for on-board control applications is the causal ECMS-based optimization for the good performance shown without the need for using information in future driving conditions. On the other hand, the technological advances in driving assistance systems and connected vehicles that are leading towards autonomous driving, are the ultimate game changer that can make the driving horizon more deterministic and, hence, render non-causal techniques more interesting from the industrial point of view in the near future.

As future advancement of this research, detailed powertrain components modelling and realistic driving cycles (e.g., Real Driving Emissions cycle) will be considered to provide more accurate results and make the comparison of optimization approaches more robust.

Author Contributions: Conceptualization, F.D., A.F., I.A. and C.P.; methodology, F.D. and A.F.; software, F.D. and A.F.; validation, A.F.; formal analysis, F.D. and A.F.; investigation, A.F. and P.P.; data curation, F.D, A.F and P.P; writing—original draft preparation, F.D., A.F. and P.P.; writing—review and editing, A.F., P.P., I.A. and C.P.; supervision, C.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

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

Nomenclature

Acronyms

BEV	battery electric vehicle
CADC	Common Artemis Driving Cycle
DOF	degrees of freedom
DP	Dynamic Programming
EMS	energy management strategy
ECMS	equivalent consumption minimization strategy
FADP	forward approach to Dynamic Programming
FCEV	fuel cell electric vehicle
FTP	Federal Test Procedure
HEV	hybrid electric vehicle
HWFET	Highway Fuel Economy Test

IFADP	iterative forward approach to Dynamic Programming
NYCC	New York City Cycle
PHEV	plug-in hybrid electric vehicle
PMP	Pontryagin's Minimum Principle
RFE	relative fuel economy
SoC	state of charge
WLTP	Worldwide Harmonized Light Vehicles Test Protocol
Roman symbols	
F	fstate function
H	Hamiltonian function
H_i	fuel lower heating value (J/kg)
I	current (A)
J	performance index
L	transition cost
M	state grid points number
$\dot{\boldsymbol{m}}$	mass flow (kg/s)
P	power (W)
Q	control grid points number
Q_b	battery charge (C)
S	equivalence factor (-)
s_c	equivalence factor parameter (-)
s_{c2}	equivalence factor parameter (-)
s_{c2}	equivalence factor parameter (-)
s_p	equivalence factor parameter (-)
T	time (s)
t_f	final time (s)
U	control variable
UD	urban degree (Hz)
V	control variables set
V	vehicle speed (m/s)
X	feedback matrix
X	state variable
Greek symbols	
φ	penalty/terminal cost (kg)
λ	co-state
ξ	battery state of charge (-)
$\dot{\boldsymbol{\xi}}$	battery state of charge time-derivative (s^{-1})
Subscripts	
0	condition
B	battery
$Conv$	conventional
$Down$	lower boundary
EQ	equivalent fuel consumption
F	fuel
I	state grid point index
J	state grid point index
Min	minimum
Max	maximum
Opt	optimal
Up	upper boundary
Superscripts	
K	time iteration index (-)
N	number of iterations in time (-)

Appendix A

Algorithm A1 General implementation of the Forward Approach to Dynamic Programming (FADP).

if $j == s$ **then**
$J_j^1 = 0$
else
$J_j^1 = +\infty$
end if
$X_j^1 = s$
for $k = 1 : N - 1$ **do**
for $j = 1 : M$ **do**
$J_j^{k+1} = \min_i \left(J_i^k + L_{ij}^k \right)$
$X_j^{k+1} = \arg\min_i \left(J_i^k + L_{ij}^k \right)$
end for
end for
$\xi_{opt}^N = \xi_t$
$j = t$
for $k = N - 1 : -1 : 1$ **do**
$i = X_j^{k+1}$
$\xi_{opt}^k = \xi_i$
$j = i$
end for

Algorithm A2 Calculation of the transition cost for the HEV optimal control problem.

for $i = 1 : M$ **do**
for $j = 1 : M$ **do**
$\Delta\xi_{ij} = \xi_j - \xi_i$
$L_{ij}^k = \infty \cdot \Delta\xi_{ij}$
$\Delta t = t^{k+1} - t^k$
$\dot{\xi}_{ij} = \frac{\Delta\xi_{ij}}{\Delta t}$
$\left(\xi_i, \xi_{ij} \right) \rightarrow P_b$ from battery model
if battery constraints met **then**
$\left(P_b, P_w^k, \omega_w^k \right) \rightarrow \left(P_m, \dot{m}_f \right)$ from electric motor model
if motor constraints met **then**
if $P_w^k < 0$ **then**
if $P_m < 0$ AND $P_m > P_w^k$ **then**
$L_{ij}^k = \frac{P_b}{H_i}$
end if
else
$\left(P_b, P_w^k, \omega_w^k \right) \rightarrow \left(P_e, \dot{m}_f \right)$ from engine model
if engine constraints met **then**
select optimal gear
$L_{ij}^k = \dot{m}_f \Delta t$
end if
end if
end if
end if
end for
end for

References

1. International Energy Agency. *Energy Efficiency Indicators: Highlights*; International Energy Agency: Paris, France, 2017.
2. Sciarretta, A.; Guzzella, L. *Vehicle Propulsion Systems—Introduction to Modeling and Optimization*; Springer: New York, NY, USA, 2005.
3. Saiteja, P.; Ashok, B. Critical review on structural architecture, energy control strategies and development process towards optimal energy management in hybrid vehicles. *Renew. Sustain. Energy Rev.* **2022**, *157*, 112038. [CrossRef]
4. Polverino, P.; Arsie, I.; Pianese, C. Optimal energy management for hybrid electric vehicles based on Dynamic Programming and receding horizon. *Energies* **2021**, *14*, 3502. [CrossRef]
5. Onori, S.; Serrao, L.; Rizzoni, G. *Hybrid Electric Vehicles: Energy Management Strategies*; Springer Briefs in Control, Automation and Robotics: New York, NY, USA, 2016.
6. Cervone, D.; Sessa, B.; Arsie, I.; Pianese, C.; Polverino, P. A comprehensive hybrid vehicle model for energetic analyses on different powertrain architectures. In Proceedings of the 14th International Conference on Engines & Vehicles, Capri Napoli, Italy, 15–19 September 2019. *SAE Technical Paper* **2019** 2019-24-0064.
7. Enang, W.; Bannister, C. Modelling and control of hybrid electric vehicles (a comprehensive review). *Renew. Sustain. Energy Rev.* **2017**, *74*, 1210–1239. [CrossRef]
8. A123 Systems. APR18650M1A Battery Datasheet. Available online: https://www.batteryspace.com/prod-specs/6612.pdf (accessed on 1 December 2021).
9. Lam, L.; Bauer, P.; Kelder, E. A practical circuit-based model for Li-ion battery cells in electric vehicle applications. In Proceedings of the IEEE 33rd International Telecommunications Energy Conference (INTELEC), Amsterdam, The Netherlands, 9–13 October 2011.
10. TREMEC. Tr-2450 Transmission Specifications. Available online: https://www.tremec.com/menu/passenger-vehicle-transmissions/ (accessed on 1 December 2021).
11. Sciarretta, A.; Guzzella, L. Control of hybrid electric vehicles. *IEEE Control. Syst. Mag.* **2007**, *27*, 60–70.
12. Serrao, L.; Onori, S.; Rizzoni, G. A comparative analysis of energy management strategies for hybrid electric vehicles. *J. Dyn. Syst. Meas. Control. Trans. ASME* **2011**, *133*, 031012. [CrossRef]
13. Sundstrom, O.; Ambuhl, D.; Guzzella, L. On implementation of Dynamic Programming for optimal control problems with final state constraints. *Oil Gas Sci. Technol.–Rev. IFP* **2010**, *65*, 91–102. [CrossRef]
14. Heppeler, G.; Sonntag, M.; Sawodny, O. Fuel efficiency analysis for simultaneous optimization of the velocity trajectory and the energy management in hybrid electric vehicles. *IFAC Proc. Vol.* **2014**, *47*, 6612–6617. [CrossRef]
15. D'Amato, A.; Donatantonio, F.; Arsie, I.; Pianese, C. *Enhancing Cruise Controllers through Finite-Horizon Driving Mission Optimization for Passenger Vehicles*; SAE International: Warrendale, PA, USA, 2018; SAE Technical Paper 2018 2018-01-1180.
16. Donatantonio, F.; D'Amato, A.; Arsie, I.; Pianese, C. A multi-layer control hierarchy for heavy duty vehicles with off-line dual stage Dynamic Programming optimization. *Transp. Res. Part C Emerg. Technol.* **2018**, *92*, 486–503. [CrossRef]
17. Eriksson, L.; Thomasson, A.; Ekberg, K.; Reig, A.; Eifert, M.; Donatantonio, F.; D'Amato, A.; Arsie, I.; Pianese, C.; Otta, P.; et al. Look-ahead controls of heavy duty trucks on open roads—Six benchmark solutions. *Control. Eng. Pract.* **2019**, *83*, 45–66. [CrossRef]
18. Stockar, S.; Marano, V.; Rizzoni, G.; Guzzella, L. Optimal control for plug-in hybrid electric vehicle applications. Proceedings IEEE American Control Conference, Baltimore, MD, USA, 30 June–2 July 2010.
19. Serrao, L.; Sciarretta, A.; Grondin, O.; Chasse, A.; Creff, Y.; Di Domenico, D.; Pognant-Gros, P.; Querel, C.; Thibault, L. Open issues in supervisory control of hybrid electric vehicles: A unified approach using optimal control methods. *Oil Gas Sci. Technol.-Rev. IFP Energ. Nouv.* **2013**, *68*, 23–33. [CrossRef]
20. Bellman, R. *Dynamic Programming*; Princeton University Press: Princeton, NJ, USA, 1957.
21. Liberzon, D. *Calculus of Variations and Optimal Control Theory: A Concise Introduction*; Princeton University Press: Princeton, NJ, USA, 2012.
22. Bertsekas, D.P. *Dynamic Programming and Optimal Control*; Athena Scientific: Belmont, MA, USA, 2012; Volume I.
23. Wang, R.; Lukic, S.M. Dynamic programming technique in hybrid electric vehicle optimization. In Proceedings of the IEEE International Electric Vehicle Conference, Greenville, SC, USA, 4–8 March 2012.
24. Sundstrom, O.; Guzzella, L. A generic Dynamic Programming MATLAB function. In Proceedings of the IEEE Control Applications, (CCA) & Intelligent Control (ISIC), St. Petersburg, Russia, 8–10 July 2009.
25. Ozatay, E.; Onori, S.; Wollaeger, J.; Ozguner, U.; Rizzoni, G.; Filev, D.; Michelini, J.; Di Cairano, S. Cloud-based velocity profile optimization for everyday driving: A dynamic-programming-based solution. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 2491–2505. [CrossRef]
26. Maino, C.; Misul, D.; Musa, A.; Spessa, E. Optimal mesh discretization of the Dynamic Programming for hybrid electric vehicles. *Appl. Energy* **2021**, *297*, 116920. [CrossRef]
27. Sciarretta, A.; De Nunzio, G.; Ojeda, L.L. Optimal ecodriving control: Energy-efficient driving of road vehicles as an optimal control problem. *IEEE Control. Syst.* **2015**, *35*, 71–90.
28. Yuan, Z.; Teng, L.; Fengchun, S.; Peng, H. Comparative study of Dynamic Programming and Pontryagin's minimum principle on energy management for a parallel hybrid electric vehicle. *Energies* **2013**, *6*, 2305–2318. [CrossRef]

29. Shi, D.; Liu, S.; Cai, Y.; Wang, S.; Li, H.; Chen, L. Pontryagin's minimum principle based fuzzy adaptive energy management for hybrid electric vehicle using real-time traffic information. *Appl. Energy* **2021**, *286*, 116467. [CrossRef]
30. Song, K.; Wang, X.; Li, F.; Sorrentino, M.; Zheng, B. Pontryagin's minimum principle-based real-time energy management strategy for fuel cell hybrid electric vehicle considering both fuel economy and power source durability. *Energy* **2020**, *205*, 118064. [CrossRef]
31. Serrao, L.; Onori, S.; Rizzoni, G. ECMS as a realization of Pontryagin's minimum principle for HEV control. In Proceedings of the IEEE American Control Conference, St. Louis, MO, USA, 10–12 June 2009.
32. Musardo, C.; Rizzoni, G.; Guezennec, Y.; Staccia, B. A-ECMS: An adaptive algorithm for hybrid electric vehicle energy management. *Eur. J. Control.* **2005**, *11*, 509–524. [CrossRef]
33. Kim, N.; Cha, S.; Peng, H. Optimal control of hybrid electric vehicles based on Pontryagin's minimum principle. *IEEE Trans. Control. Syst. Technol.* **2011**, *19*, 1279–1287.
34. DieselNet. Emissions Test Cycles. Available online: https://dieselnet.com/standards/cycles/index.php (accessed on 1 December 2021).