Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

A predictive control strategy based on A-ECMS to handle Zero-Emission Zones: Performance assessment and testing using an HiL equipped with vehicular connectivity

01 May 2024

A predictive control strategy based on A-ECMS to handle Zero-Emission Zones: performance assessment and testing using an HiL equipped with vehicular connectivity

Lorenzo Brunelli, Alessandro Capancioni, Stella Canè, Giammarco Cecchini, Alessandro Perazzo, Alessandro Brusa, Nicolò Cavina

Highlights

- Simulation environment with real vehicular connectivity and prototype control units
- Speed Profile Prediction based on real-time traffic and navigation data
- Energy Management Strategies to allow electric drive within an urban area
- Reduction of CO_2 production between 20% and 26%

Keywords

Zero-Emission Zones, hybrid electric vehicles, Adaptive-ECMS, vehicular connectivity, Hardware-in-the-Loop

Abstract

Recently, several metropolitan cities introduced Zero-Emissions Zones where the use of the Internal Combustion Engine is forbidden to reduce localized pollutants emissions. This is particularly problematic for Plug-in Hybrid Electric Vehicles, which usually work in depleting mode. So, the risk of not having enough energy stored to carry out the driving mission and then paying a fee is substantial. This work presents a viable solution by exploiting vehicular connectivity to retrieve navigation data of the urban event along a selected route. The battery energy needed, in the form of a minimum State of Charge (SoC), is calculated by a Speed Profile Prediction algorithm and a Backward Vehicle Model. That value is then fed to both a Rule-Based Strategy, developed specifically for this application, and an Adaptive Equivalent Consumption Minimization Strategy (A-ECMS). The effectiveness of this approach has been tested with a Connected Hardware-in-the-Loop (C-HiL) on a driving cycle measured on-road, stimulating the predictions with multiple re-routings. The tests have been conducted with different initial SoC values for each strategy, showing a maximum error in the SoC prediction of 2.4% and up to 26.1% of CO_2 saving with the A-ECMS.

1. Introduction

The urbanization trend is continuously increasing during the last decades, leading to congestions and localized high concentrations of emissions pollutants [1]. On one hand, governments are forced to introduce more stringent regulations [2]. On the other hand, local administrations start to manage urban mobility to optimize traffic flows to decrease congestion. The most frequently applied measure is the introduction of urban areas with increasing traffic limitations, regarding both the conventional vehicles (Low-Emission Zone – LEZ) and the non-conventional vehicles (Zero-Emission Zone – ZEZ) [3]. Conventional vehicles and mild-hybrid electric vehicles are excluded from this analysis since they can do nothing but pay fees to access the restricted areas. Battery Electric Vehicles (BEV) are also excluded since electric driving is the only possible driving mode. Differently, Plug-in Hybrid Electric Vehicles (PHEV), having two different sources of propulsion (Internal Combustion Engine (ICE) and High Voltage (HV) battery), offer the additional degree of freedom of the power division, in case of a parallel powertrain topology. While the possibility to split the torque demanded by the driver avoids the less efficient operating points of the ICE, it multiplies the possible splitting solutions, increasing the controller complexity.

In the last years, different solutions have been proposed in literature on the topic of energy management for hybrid electric vehicles. The most widespread solution for on-board implementation is represented by heuristic controllers that use fixed rules to manage the energy flows in the powertrain. Thus, they are referred to as Rule-Based Strategies (RBS). This approach is considered simple and robust, but far from the optimal solution [4,5]. Other supervisory controllers perform a local or global minimization of equivalent fuel consumption that considers both the real fuel consumption and the electrical power requested by the battery. Such controllers are usually considered sub-optimal [6] or optimal [7] respectively and several of them have been already deployed on the vehicles. An example is the ECMS, which was originally introduced in [8] and applied as a practical solution in [9]. Then, in [10] and more recently in [6] remarkable improvements have been obtained to make the ECMS a real-time control strategy. In particular, in the latter work, an adaption of the equivalence factor based on feedback from SoC has been proposed.

In parallel, innovative technologies are being broadly implemented both on the vehicle and the infrastructure, such as wireless communication, and cloud computing. These technologies are divided into vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and vehicle-to-network (V2N) communication. Together with the latest Advanced Driver-Assistance Systems (ADAS), these technologies allow calculating an electronic horizon (shortly known as eHorizon) which represents a virtual reconstruction of the trip ahead for a planned route and can provide it to the control strategies, resulting in more efficient energy management. Several studies highlighted the benefits of future driving information for Energy Management Strategies (EMS). An improved A-ECMS based on long-term target driving cycle recognition and short-term vehicle speed prediction is presented in [11]. It can optimize the equivalence factor based on mileage, SoC, long-term driving cycle, and real-time vehicle speed, resulting in a reduction of fuel consumption of 8,7%. Similarly, algorithms can determine the optimal SoC trajectory according to the traffic information, while the equivalence factor is tuned dynamically, thus enabling effective tracking of the reference SoC trajectory, as done in [12]. A different approach is presented in [13] where the A-ECMS uses a historical driving profile for equivalence factor estimation, thus the proposed strategy is able to foresee the change of the driving behaviors and adjust the equivalence factor more reasonably. Another methodology to determine the future operating conditions of the vehicle is analyzed in [14] using different driving pattern recognition and prediction algorithms based on type-approval cycles. With this information, a potential improvement of up to 4% in fuel economy was achieved with a 0-dimensional model on an RDE driving cycle with respect to the baseline ECMS. Furthermore, a neural network algorithm can also be used to recognize and group the speed patterns, and consequently applied to a dynamic ECMS by adjusting the equivalence factor, leading to a potential improvement of 7,8% in fuel economy [15].

Regarding control strategies related to the presence of restrained area, in [16] a LEZ-anticipating control strategy for a PHEV bus with a P2-type parallel powertrain configuration is presented. The control strategy is based on a combined RB/ECMS, and it is superimposed by generating an optimal reference SoC trajectory aimed at enabling pure electric driving through forthcoming LEZs and minimizing the overall fuel consumption. Following this trend, the strategy used in this work is an A-ECMS fed with a minimum target SoC needed to perform a ZEZ event in pure electric mode, whose development and testing have been presented in [17]. In this case, the target SoC is calculated starting from the navigation data provided by the connectivity and the use of an analytical model of the vehicle.

Clearly, the more the control strategies grow in complexity, the more their testing and validation become demanding both in terms of costs and time. In fact, they must be tested in a huge number of scenarios, regarding dangerous and highly unpredictable situations to be

declared reliable. The challenge is therefore to develop a validation and testing framework that can replicate the complexity and unpredictability of road conditions and traffic scenarios. In other words, the aim is to move the tests from the road to the virtual simulation, saving both money and time. In [18], the authors improve the commercial simulation software with optimal energy management, which is tested using the short horizon information coming from the leading vehicle in the collaborative environment. In [19] an advanced simulation framework is presented, with several On-Board Unit (OBU) / Road-Side Unit (RSU) hardware connected to a microscopic traffic simulator (Simulator of Urban Mobility - SUMO) to integrate real vehicular communication devices. A step further was made in [20] which developed an Engine-in-the-Loop system integrated with a real-time traffic simulator (named VISSIM) to evaluate the performance of emerging connected vehicle applications. Then, a real vehicle equipped with an OBU is driving along with other connected vehicles while the data is transmitted to the HiL, which reacts consequently. Finally, [21] focused on the development of a sustainable framework for testing control strategies for connected automated vehicles. They presented an HiL where vehicle dynamics are up to ETAS DESK-LabCar, controlled by onboard control unit i.e., MicroAutoBox and Matrix embedded PC-Adlink. The latter oversees the communication with the OBU and the cloud, respectively through Ethernet and LTE. Such an advanced simulation framework is very interesting, but it has been presented with a short driving routine. In conclusion, the paper presents improvements to the simulation environment presented in [22], which represents a state-of-the-art for development and testing of energy management control strategies based on vehicular connectivity.

1.1. Novel contribution

This paper aims at proving that the predictive functions and the energy management strategy presented in [17] are suitable for real-time applications, which means implementation in a control unit and under real driving conditions (i.e. driving cycles not known a priori). Moreover, this work also demonstrates that the presented EMS leads to a remarkable reduction in fuel consumption when approaching a ZEZ even if it is fed with real-time navigation data. To properly test the proposed approach, the simulation environment presented in [22] has been improved to replicate situations as realistic as possible, such as when the traffic changes during time and/or the driver decides to change direction, leading to several re-routings.

In this regard, the evaluation of an energetically equivalent velocity trace derived from the navigation data is required by the predictive strategy. To do so, a Speed Profile Prediction (SPP) algorithm represented by a space-based driver model derived from literature, firstly developed at Model-in-the-Loop (MiL) level in [23], has been implemented in the simulation environment.

The speed profile is then forwarded to the Backward Vehicle Model (BVM) described in [17], which predicts the necessary amount of electrical energy to perform the ZEZ in full electric mode. This value, expressed in terms of a target SoC, is used as input for the strategies under test. Regarding the latter, since the RBS introduced in [17] is not designed to work differently from the conventional Charge-Depleting / Charge-Sustaining (CD/CS) mode, in this work an Adaptive-RBS (A-RBS) has been defined. In particular, it uses the same reference SoC used by the A-ECMS to dynamically adapt the electric drive thresholds depending on the remaining distance from the ZEZ, thus working in a Charge Blended (CB) mode. To verify the improvements with respect to the RBS in terms of CO_2

reduction, the two strategies are compared on two different RDE cycles at the Software-in-the-Loop (SiL). The results show that the A-RBS outperforms the RBS only when the initial SoC is lower than the target SoC and so the HV battery must be charged. Conversely, the RBS is still more efficient when the initial SoC is higher than the target SoC and the CD/CS mode is required. Consequently, since the aim is to design a strategy that reduces the energy consumption in all the possible scenarios, a Combined-RBS (C-RBS) is finally defined to perform CB mode or in CD/CS depending on the initial value of the SoC. The C-RBS defined in this way may then be used as the reference to be compared with energy-based approaches such as A-ECMS, since it represents the best performance that can be reached with heuristic approaches.

Finally, the A-ECMS and the C-RBS have been tested at the C-HiL on a driving profile measured on the road during a specific time slot of the day. Differently from the one presented in [22], this simulation environment presents a layout reduced in complexity as the sole Telecommunication Control Unit (TeCU) is in charge of querying the Map Service Provider (MSP) and gatewaying the data. Moreover, in this work, a Human Machine Interface (HMI) has been introduced to be as close to the prototype vehicle as possible. More in detail, the vehicle HMI is replicated by a tablet running an Android-based Navigator App, that allows the driver to set the desired destination and to see the suggested route based on actual traffic conditions. Meanwhile, the related navigation data are collected from the MSP and sent to the TeCU. Moreover, if the actual position of the vehicle deviates from the route proposed by the app or if the actual level of traffic changes, the app can trigger the request for a re-routing. In these cases, the navigation data are refreshed, and the prediction is performed again, evaluating the updated target SoC. Besides, the tests have been conducted starting from different initial SoC values, during the same time slot of the recorded driving profile to reduce the natural randomness of the traffic conditions as much as possible. In conclusion, the simulations highlight both the goodness of the predictions even under varying traffic conditions, and the improvements brought by the A-ECMS in terms of CO_2 reduction.

2. Simulation Environment

In this section, the simulation environment is described in detail. Firstly, the C-HiL adopted for the test is analyzed in each of its components, with a focus on the vehicular connectivity and the control units. Then, the vehicle layout and simulation model are presented alongside its specifications. Finally, the developed SPP algorithm and the BVM, responsible for the target SoC evaluation, are introduced.

2.1. Connected Hardware-in-the-Loop

In this work, the focus is put on the long-range connectivity provided by the layout presented in [22]. With respect to the latter, the C-HiL has been improved and its complexity reduced. In particular, as shown by the blue boxes in Fig. 1, the TeCU is now in charge of both the internet service manager and the data processing, while an Android-based tablet is connected via Wi-Fi to the TeCU itself to replicate the vehicle's HMI. Furthermore, the data exchange handled by the TeCU is divided into a direct flow ("NAV Data" label in Fig. 1), for the transmission of the navigation data (listed in Table 1) from the MSP to the HCU, and a backflow ("HMI Data" label in Fig. 1), for querying and transmitting the actual GPS position from the Real-Time PC to the MSP and the HMI.
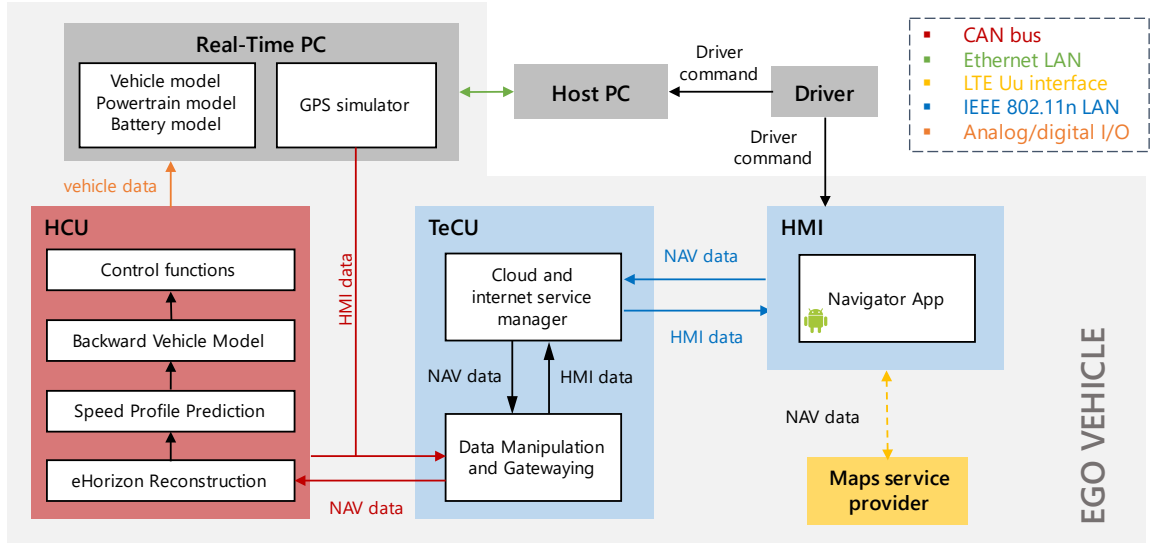
Fig. 1   Connected HiL layout for testing predictive functions exploiting long-range connectivity
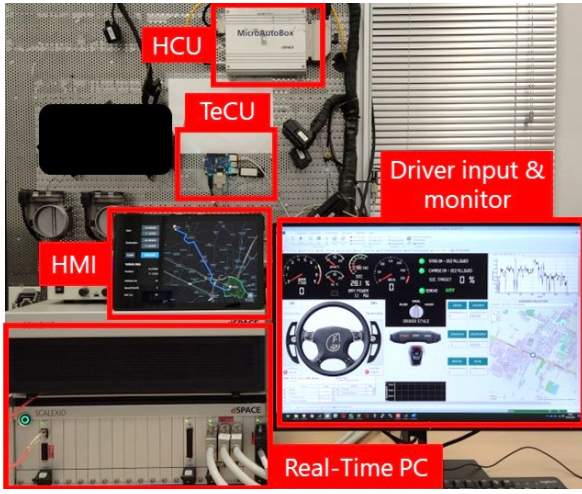

Fig. 2   Photo of the C-HiL during a simulation

To give a proper outlook of the simulation environment, in Fig. 2 a photo taken during a simulation is reported. Hence, the components of the C-HiL are described in detail in the next paragraphs, following the workflow of the simulations. Firstly, the driver selects the desired destination directly on the HMI Navigator App whose screenshot is represented in Fig. 3. The latter, integrating HERE SDK [24], includes the start and destination coordinates boxes, the actual vehicle's position (blue dot), and data (coordinates, altitude, speed, and SoC), and the ZEZ plotted as a green area. The latter has been designed by increasing the limits of the legal Limited Traffic Zone of Bologna to also include the residential areas nearby. In this way, the average urban trip to reach the city center is between 2-4 kilometers and so the maneuver in pure electric is more demanding.
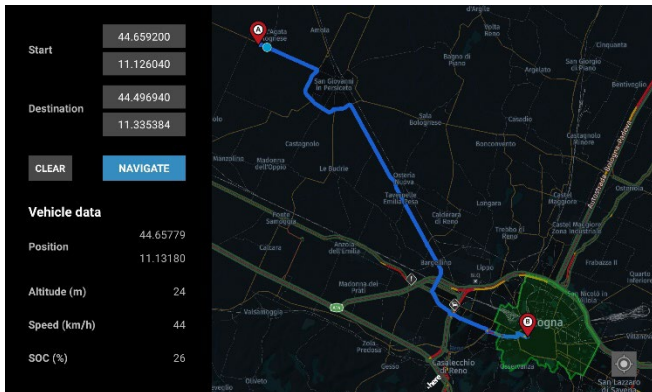

Fig. 3   Screenshot of the Navigator App: suggested route from point A to point B

Then, the MSP evaluates the fastest route according to actual traffic conditions and sends the navigation data via LTE Uu Interface to the tablet (yellow line in Fig. 1), which forwards them to the TeCU via IEEE 802.11n LAN (blue line). The latter performs the necessary data processing to make the navigation data compliant with the HCU computational power capability and sends them via CAN bus (red line in Fig. 1) to the HCU itself.

Table 1   Navigation data sent by the MSP

Navigation Data	Dimensions	Format
Legal speed limits	[1x1500]	uint8
Road segments	[1x1500]	uint16
Slope	[1x1500]	uint8
Stop events typologies	[1x200]	uint8
Stop events positions	[1x200]	uint16
Traffic codes	[1x100]	uint8
Traffic codes segments	[1x100]	uint16
ZEZ entrance and exit	[1x10]	uint16

In particular, the data processing includes the removal of redundant data such as segments where both the speed limit and the slope do not change, to respect the dimensions limits listed in Table 1.

Consequently, the HCU performs the eHorizon reconstruction picking up all the CAN messages and concatenating the values in dedicated vectors. The number of CAN messages needed for a particular route depends on the number of data and the format chosen for them. After that, the SPP algorithm (described in detail in sec. 2.3 and in [23]) calculates a space-based velocity profile that proved to be energetically equivalent to the one effectively driven. This is motivated by the fact that the aim is to evaluate the electrical energy usage of the HV battery and not to predict the exact maneuvers of the driver. Therefore, the speed profile is converted into time-domain and used by the BVM, which is described in detail in sec. 2.4 and in [17], to predict the amount of energy to perform the ZEZ in pure electric mode, thus respecting the local legislation. Finally, the value of the State of Charge (SoC) target becomes the input of the Energy Management Strategies. Concurrently, the Real-Time PC sends back information about actual GPS position, altitude, vehicle speed, and SoC via CAN bus, through the TeCU, to be plotted into the HMI, as shown in the bottom left corner of Fig. 3. Moreover, the Navigator App can also detect if the actual position of the vehicle is moving away from the proposed route and trigger another request to the MSP to retrieve the updated navigation data. These data are then forwarded again to the TeCU as described above.

In case of missing navigation data, due for example to connectivity loss, the conventional A-ECMS would perform its optimization with a target SoC typical of CD/CS mode.

2.2. Vehicle model

The 0-D model of the prototype vehicle, developed in the Simulink® environment, runs in the Real-Time PC. It represents a high-performance gasoline PHEV with a P1-P4 configuration, so with an Electric Motor (EM) directly mounted on the crankshaft (P1) which acts as an integrated started and generator (ISG), and two EMs coupled to the front wheels (P4) through a fixed gear ratio. All three EMs are identical, thus having the same performance.

Since the model has been already discussed in [17,22] and its description is out of the scope of this paper, only a schematic representation of the powertrain layout is reported in Fig. 4, while the specifications of the ICE, the gearbox, the HV battery, and the EMs are summarized in Table 2.
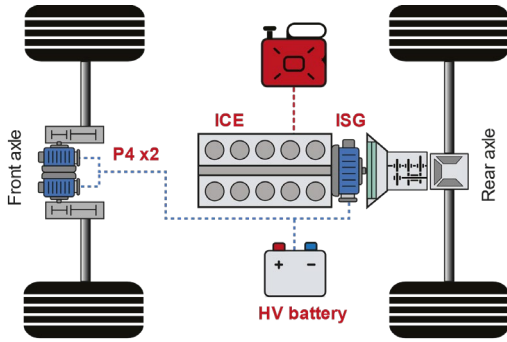

Fig. 4   Prototype vehicle layout [22]

Table 2   Vehicle powertrain specifications

Vehicle parameter	Specification	
Battery nominal capacity (1C @ 25°C)	19.4	Ah
Battery nominal / max. voltage	384 / 391	V
Motor torque (continuous / peak)	145 / 350	Nm
Motor power (continuous / peak)	64 / 140	kW
Engine max. torque	533	Nm
Engine max. power	449	kW
Overall max. power (continuous / peak)	577 / 729	kW
Transmission	6	gears

3. Control functions

In the following chapters, the control functions and the algorithm developed and implemented into the HCU are presented. First, the navigation data received by the map service provider are processed by the predictive functions, which have to evaluate the amount of energy required in the form of a target SoC to perform the ZEZ in pure electric mode. Consequently, the energy management strategies implemented to handle that information are described with a focus on the necessary adaptations done on the conventional RBS to have a fair comparison.

3.1. Predictive functions

Looking at the red block in Fig. 1, the predictive functions start with the eHorizon reconstruction, which processes the CAN messages received from the TeCU converting them into arrays. Then, the processed navigation data are used by the Speed Profile Prediction algorithm to evaluate the energetically equivalent velocity trace related to them. Finally, the Backward Vehicle Model predicts the amount of energy required to be stored in the HV battery, expressed as a target SoC.

3.1.1. Speed Profile Prediction algorithm

The application of predictive functions upon the control unit is still challenging since they are inherently dependent on future

trajectories of velocity, road slope, and external disturbances, which are generally unknown for real-world drive cycles. However, with the help of the navigation data provided by the MSP, an estimation of the trajectories can be made and then used by the energy management strategy.

To do so, different approaches can be identified, which are divided into parametric and non-parametric [25]. In the first category, the driving task is modeled as a stimulus-response system, so, as a control problem where the driver's goal is to keep a safe distance from the vehicle in front or to pursue a target speed according to some imposed constraints. The second one includes algorithms based on probabilistic and artificial intelligence theories such as Artificial Neural Networks, Markov chains, or Monte Carlo methods. As reported in [25], advanced parametric models are commonly used for long-range energy management prediction, representing a valuable trade-off between reliability and ease of implementation. A remarkable algorithm for Speed Profile Prediction has been proposed by [26], which for this reason has been assumed as the reference for the one developed and tested at the MiL level in [23].

The algorithm is implemented in the simulation environment as follows:

Algorithm 1 - SPP algorithm

1:   **begin**
2:       nodes vector creation from navigation data
3:       **for** $j = 1:N_s$
4:           legal speed limit and traffic code assignment
5:       **end for**
6:       $d = L_{trip}/(n_v \cdot c_s) \rightarrow$ discretization step
7:       **for** $j = 1:N_s$
8:           $N = (s_j - s_{j-1})/d \rightarrow$ sub-segmentation
9:           **for** $i = 1:N$
10:              next point i MAS evaluation
11:              next point i speed evaluation
12:          **end for**
13:          speed and space values vector assignment
14:      **end for**
15:      space-to-time conversion
16: **end**

The output of the algorithm are the vectors of the speed and slope trajectories, whose size is limited by the computational power of the HCU as well as the navigation data. As a consequence, their accuracy is dependent on trip segmentation, which is the result of:

1. a first segmentation that depends on the road characteristics (slope, stop events, legal speed limits) and on the traffic, so on the navigation data. This kind of segmentation is represented by N_S "nodes" in Algorithm 1. The nodes have been defined as the coordinates along the trip that correspond to the presence of a stop event or a variation in the speed limit, slope, or traffic code.

2. a second segmentation that is then applied by the algorithm to actually generate the speed trace, as shown in Fig. 8 (vertical black lines). In particular, a discretization step, d, is evaluated as the ratio between the total length of the trip, L_{trip}, and the size of the output vector, n_v. Then, each segment defined by the first segmentation, with a length of $s_j - s_{j-1}$, is subdivided again into N sub-segments having a length equal to d (Algorithm 1, line 8). In this way, the density of the sub-segmentation is kept constant, since the longer is the j-th segment, the higher is the number of required sub-segments. When:

$$(s_j - s_{j-1}) < k_d \cdot d \qquad (1)$$

i.e., the two adjacent nodes are too close to each other, a smaller discretization step is applied to have sufficient values to create the speed trace within the j-th segment. The coefficient $k_d = 4$ represents the minimum number of points required to create an acceleration-deceleration maneuver following Eq. (5) and (6). Moreover, a corrective factor c_s is applied in line 6 to take into account the additional elements required by the condition

expressed by Eq. (1) and avoid the complete saturation of the memory. The latter is evaluated as $c_s = 1 - k_d \cdot n_c/n_v$, where n_c is the number of segments that verify the condition expressed by Eq. (1).

As a final remark on the second segmentation, it must be said that the discretization is dependent on the overall length of the route, so the accuracy of the speed profile could be affected for longer trips. However, this algorithm is applied only to the ZEZ, which is usually a limited event for the test cases under evaluation, and the overall accuracy is more than satisfying as proved by the results of the calibration and of the tests at the C-HiL. Moreover, the problems related to the computational power will be overcome by moving the predictive functions from the HCU to the cloud, as it will be proposed in future work.

After that, the starting values of the speed limit and traffic code are assigned to every sub-segment $i = 1, \dots, N$ of each segment $j = 1, \dots, N_S$ (line 4 of Algorithm 1). Consequently, for each sub-segment, the Maximum Allowed Speed (MAS) is calculated (line 8 of Algorithm 1). If a node coincides with a stop event position, then the MAS is imposed by the kind of the stop event itself. In particular, a stop event is commonly referred to as an event whose presence implies that the speed of the vehicle in that position must be partially or totally decreased. They are divided into static, if the breaking or stop is mandatory (e.g., bumps, stop signal), and dynamic, if they may not affect the current speed even if their position is known (e.g., green lights and right of ways). For the latter, a stop-over probability is introduced, as in [26], depending on traffic codes modeled by means of binomial probability and summarized in Table 3.

Table 3   Stop-over probability with respect to the traffic codes

Traffic code, c	Traffic color	Congestion	Stop-over probability [%]	Code weight[1] CW
1	🔵	absent	15	0.85
2	🟠	light	40	0.60
3	🔴	medium	60	0.40
4	🔴	heavy	70	0.30

[1] the code weight and its value will be discussed in the next paragraphs.

For the other nodes, the MAS value is affected by the traffic density, which influences the maximum speed due to the presence of the other cars and induces oscillations around that limit due to the variation of traffic flow. Thus, the MAS can be expressed as:

$$MAS = MAS_t + MAS_n \tag{2}$$

where:

- MAS_t considers the effect related to the *traffic condition*, expressed in terms of colors (blue, orange, red, and dark red) and can be expressed as:

$$MAS_t = v_{lim} \cdot CW \tag{3}$$

where v_{lim} is the legal speed limit, and CW is the code weight that depends on the traffic code c.

- MAS_n replicates the effect of the *driver's behavior* that depends on the traffic condition. In fact, the velocity often exhibits oscillations around the speed limit due to inharmonic traffic flow. As proposed by [26], to account for these oscillations, the MAS_n can be expressed as a sum of R-cosines:

$$MAS_n = \sum_{r=1}^{R} A_r \cdot cos(2\pi f_r x) \tag{4}$$

where A_r and f_r are the amplitudes and frequencies of the oscillation respectively, and R is the number of the considered cosine terms (in this case $R = 3$).

After the evaluation of MAS for each i-th point, the vehicle speed can be calculated (line 11 of Algorithm 1). Since the objective of the

algorithm is to generate a speed vector, with the assumption of traveling time minimization, the driver will always try to reach the MAS, if possible. Therefore, the driver decides whether to accelerate, decelerate or keep the velocity constant depending on the actual value of the speed and on the boundary conditions of the $i + 1$-th point. Thus, if the MAS has been already reached, the driver can maintain the speed or start braking. Otherwise, it starts to accelerate. For this reason, an exponential acceleration and a linear deceleration law are adopted:

$$a_{acc}(x) = a_{acc,max} \cdot k_{acc}(1 - e^{-\tau x}) \tag{5}$$

$$a_{dec}(x) = -a_{dec,max} \cdot k_{dec} \cdot x \tag{6}$$

where $a_{acc,max}$ and $a_{dec,max}$ are the maximum acceleration and deceleration that depend on vehicle performance, k_{acc} and k_{dec} are the reductive factors that consider the different driver behaviors, and τ is the time constant defining the acceleration transient.

Finally, the speed profile is converted from space to time domain by following a linear interpolation of the values to feed the backward vehicle model with proper input signals (line 13 of Algorithm 1). Moreover, a stop time is assigned to each stop event depending on the traffic code and the typology of the stop event itself.

After the analysis of the algorithm, its calibration has been performed, focusing on:

- *driver-related parameters*, so k_{acc}, k_{dec}, and τ, regarding the acceleration and deceleration maneuvers described by Eq. (5) and (6), respectively. Thus, the conceived scenario is represented by a ramp-up ramp-down cycle with a nominal speed of 50 km/h (i.e., a typical urban speed limit) and performed on a straight road;

- *traffic-related parameters*, that are CW, A_r, and f_r, the algorithm has been calibrated over three urban driving scenarios within the city of Bologna, the more representative of which is shown in Fig. 5.
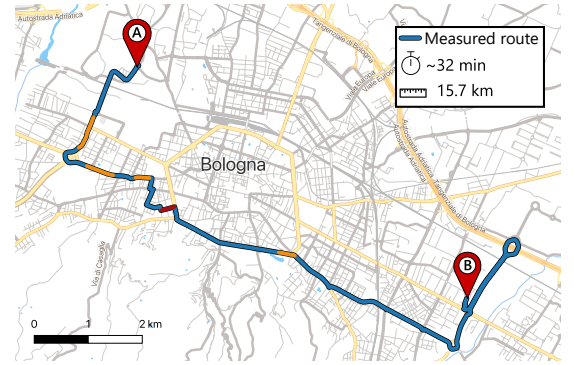

Fig. 5 Driving scenario with different traffic codes and speed limits, within a ZEZ

Furthermore, as suggested by [26], the measurements have been performed to include all the possible common situations. In particular, they have been performed at different times of day to assess the influence of various traffic conditions (from the less congested road of the night to the rush hour) and on different types of urban roads, to consider as many driving conditions as possible (from the urban roads to the high-velocity ones). Since the purpose of the algorithm is to predict the amount of energy related to an urban event, the calibration and validation campaign are focused only on scenarios inside a ZEZ.

The weight of the traffic codes, the amplitudes, and the frequency of Eq. (4) have been determined by dividing the measured space-based velocity traces into segments and clustering them with respect to the traffic code. Then, a Fast Fourier Transform (FFT) is applied in the space domain for each traffic code cluster, as suggested in [26], obtaining a magnitude-frequency diagram. In fact, the influence of traffic congestion leads to different oscillations for each traffic code. The oscillations are intended as the variation of the vehicle speed with respect to the average value every certain number of meters,

whose inverse can be seen as the space-domain frequency [m^{-1}]. By filtering the measurements, it is possible to isolate the first four space-domain frequencies and amplitudes. The first one represents the average values that are used to calculate the code weight (Eq. 8) determining the corrected MAS value. Finally, the oscillations due to the frequencies and amplitudes going from $r = 1,\ldots,3$ are added to the average value. For a matter of brevity, only the one related to the orange traffic color (code $c = 2$) is reported in Fig. 6.
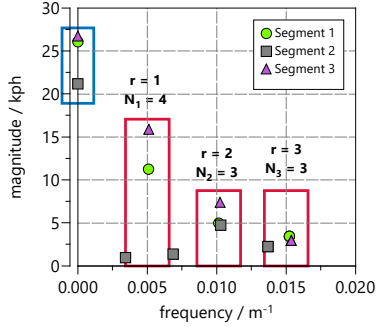

Fig. 6   Frequency analysis with FFT for the traffic code $c = 2$

In this case, $J = 3$ measured speed signals have been investigated, being each of them related to a certain space segmentation of length l_j. Then, the three ranges ($r = 1,\ldots,3$) of frequencies have been identified to describe the oscillatory phenomenon with acceptable accuracy (red boxes in Fig. 6, where N_r is the number of points within each range). At this point, focusing on Eq. (4), both the frequencies and the amplitudes of each range can be expressed as arithmetic means of the relative measured values:

$$A_{c,r} = \frac{\sum_{j=1}^{J}\left(1/N_r \cdot \sum_{n=1}^{N_r} A_{n,r,j}\right) \cdot l_j}{\sum_{j=1}^{J} l_j} \tag{7}$$

$$f_{c,r} = \frac{\sum_{j=1}^{J}\left(1/N_r \cdot \sum_{n=1}^{N_r} f_{n,r,j}\right) \cdot l_j}{\sum_{j=1}^{J} l_j} \tag{8}$$

where N_r is the number of pairs of magnitudes $A_{n,r,j}$ and the frequencies $f_{n,r,j}$ corresponding to the r-th range of the j-th signal. Moreover, since the amplitude A_0 of the oscillation at $f = 0$ m^{-1} (blue box in Fig. 6) represents the average speed on the given segments, it can be used to determine the parameter CW for each traffic code as follows:

$$CW_c = \frac{\sum_{j=1}^{J} \frac{A_{0,j}}{v_{lim,j}} \cdot l_j}{\sum_{j=1}^{J} l_j} \tag{9}$$

being $v_{lim,j}$ the legal speed limit for the j-th segment. In this way, according to Eq. (3), the static contribution to the MAS can be determined.

Afterward, two different kinds of Key Performance Indicators (KPI) have been identified to evaluate the goodness of the prediction: speed-based and energy-based KPIs. Regarding the first ones, reliable parameters for speed traces comparison are provided by [25,27] and they can be represented by the mean absolute error (MAE) and the BIAS, both expressed in [km/h] and defined as:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |v_{p,i} - v_{r,i}| \tag{10}$$

$$BIAS = \frac{1}{n}\sum_{i=1}^{n} v_{p,i} - v_{r,i} \tag{11}$$

where $v_{p,i}$ and $v_{r,i}$ are respectively the predicted and the measured speed at point i, and n is the total amount of points where the differences are calculated. Moreover, since the MAE does not consider the algebraic signs of the errors, it is used to express the mean distance between the prediction and real data. Differently, the BIAS represents a good instrument to identify eventual issues related to a systematic under/overestimation of the speed.

On the other hand, energy-based KPIs suggested by SAE [28] have been assumed as references. At first, three energy components are calculated for both the predicted and the measured cycle, and they are the road load, the positive and the negative inertia, expressed as follows:

$$E_{RL} = \int_0^L F_{RL}\,dl = \int_0^L (F_0 + v \cdot F_1 + v^2 \cdot F_2)\,dl \tag{12}$$

$$E_{I^+} = \int_0^L F_{I^+}\,dl = \int_0^L (m \cdot a^+)\,dl \tag{13}$$

$$E_{I^-} = \int_0^L F_{I^-}\,dl = \int_0^L (m \cdot a^-)\,dl \tag{14}$$

where L is the total length of the route, E_{RL} represents the energy required to win rolling resistance and drag force, E_{I^+} represents the energy required by the vehicle mass m to be accelerated, and E_{I^-} represents the energy required by the vehicle mass to be decelerated. Now, three energy KPIs can be introduced in form of energy rate, where subscripts p and r are respectively referred to as the predicted and real (measured) speed profile:

$$\Delta E_{RL} = \left(E_{RL_p} - E_{RL_r}\right)/E_{RL_r} \cdot 100 \tag{15}$$

$$\Delta E_{I^+} = \left(E_{I_p^+} - E_{I_r^+}\right)/E_{I_r^+} \cdot 100 \tag{16}$$

$$\Delta E_{I^-} = \left(E_{I_p^-} - E_{I_r^-}\right)/E_{I_r^-} \cdot 100 \tag{17}$$

The results of the calibration and validation campaign are described in detail in [23]. For the sake of brevity, in this work, only a representative test case's validation is reported in Fig. 7. In this case, the scenario under test includes different traffic codes, stop events, and speed limits to consider as many conditions as possible. The numeric results of that validation are summarized in Table 4.
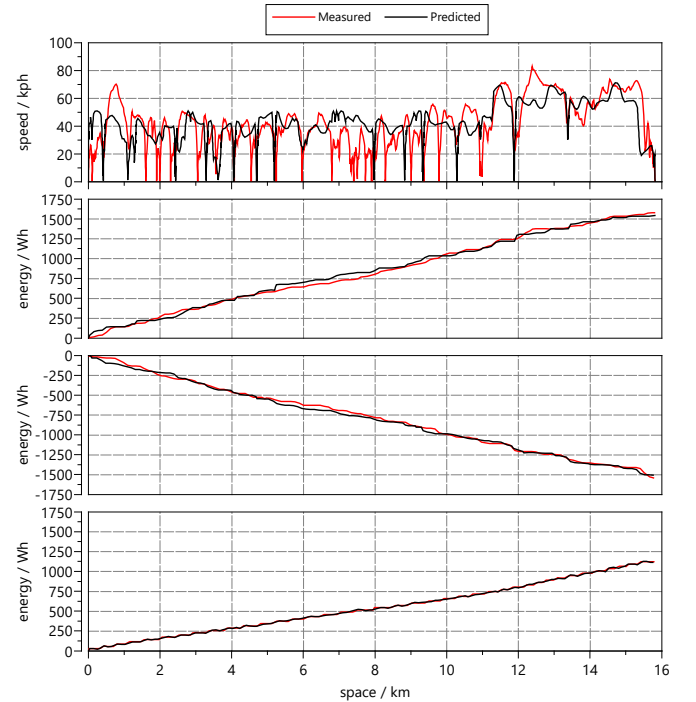

Fig. 7   Top plot: speed profile predicted from the navigation data (black) and the measured speed on the road (red). Three bottom plots: comparison between the three energy components related to the prediction (black) and the measurements (red)

Table 4   Speed and energy KPI for the presented use case

MAE [km/h]	BIAS [km/h]	$|E_{RL}|$ [Wh]	ΔE_{RL} [%]	$|E_{I^+}|$ [Wh]	ΔE_{I^+} [%]	$|E_{I^-}|$ [Wh]	ΔE_{I^-} [%]
12.4	-0.1	5	-0.4	32	-2	32	-2

In conclusion, Fig. 8a shows the calibrated predicted speed profile with respect to the test case presented in Fig. 5, alongside the legal speed limits on that route (magenta line) and the related traffic colors

(filled area below the speed profile) listed in Table 3. As shown, the velocity trace is always below the limits because the CW, as expressed by Eq. (9), affects the MAS_t, the static component of MAS, lowering the v_{lim}, while the oscillations described by Eq. (4) simulate the natural speed deviation from that value.
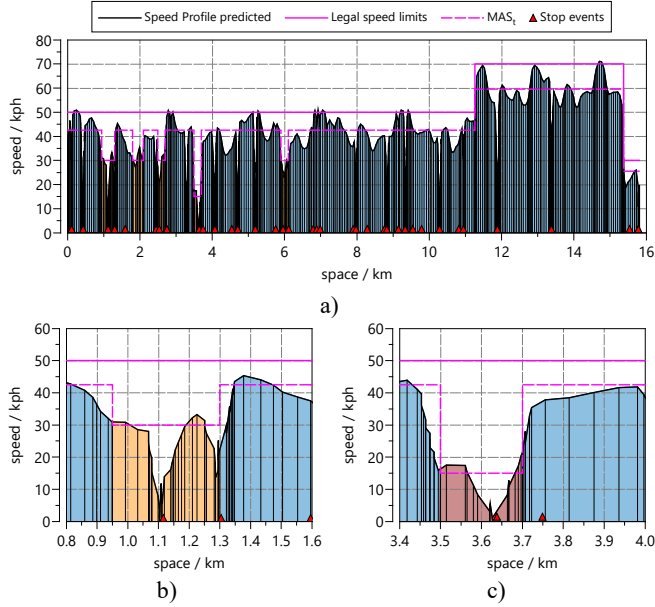

Fig. 8   a) Speed profile predicted by the algorithm after calibration b) detail of MAS limited by the orange traffic code c) detail of MAS limited by the red traffic code

Moreover, as shown in Fig. 8b and Fig. 8c, the speed is also reduced in correspondence to the orange and dark-red traffic codes respectively by the values reported in Table 3, showing the impact that each traffic code has on the predicted profile. Furthermore, on the x-axis, the stop events positions are plotted with the red triangles. As it can be noticed, since the stop-over probability is applied, the vehicle's speed is not always zero.

3.1.2. Backward Vehicle Model

The quasi-static analytical model of the vehicle used for the prediction has been developed through a backward-facing approach based on the inverted path of the energy flow inside the vehicle. Thus, the source is represented by the wheels and the sinks are the energy storage devices. Here, the traction force, and consequently torque and power, are evaluated on the base of the vehicle speed and road slope, which can be considered as external disturbances acting on the dynamic system represented by the vehicle [29–31]. Thus, there is no closed-loop control on the speed, i.e., a driver model is not needed. Moreover, dynamic effects such as torque control are not inherently included. Therefore, this results in a less complex model with benefits regarding computational load [32], which is a relevant aspect to be considered for algorithm implementation inside the hardware.

The main equations of the backward vehicle model, presented in [17], are briefly reported in the following paragraphs. Firstly, since the vehicle dynamics analytical model only considers the longitudinal forces acting on the car, the fundamental equation is:

$$m \frac{d}{dt} v(t) = F_{mot}(t) - F_{res}(t) \qquad (18)$$

where F_{mot} is the propulsion force, while F_{res} is the resistance force acting on the vehicle. The latter can be expressed as follows:

$$F_{res}(t) = F_a(t) + F_r(t) + F_g(t) \qquad (19)$$

that are, respectively, aerodynamic resistance, rolling resistance, and slope-related resistance, whose equations are not reported here for the sake of brevity. Then, as already discussed in literature [26,33], and reported in [17], the global electrical power request can be evaluated from Eq. (18) and expressed as follows:

$$P_b(t) = P_{EM}(t) + P_{aux}/\eta_{DCDC} \qquad (20)$$

in which $P_{EM}(t)$ is the power requested from the electrical motors on the front axle either for traction or regenerative braking, P_{aux} is the constant power to be supplied to the low-voltage battery and the other auxiliaries, and η_{DCDC} is the efficiency of the DCDC converter. At this point, the battery power request is modified according to power limitation maps related to charge-discharge and peak-nominal working conditions of the HV battery. These parameters are calculated by the battery electrical and thermal models, respectively, which are the same used in the vehicle's model. Thus, the electrical behavior of the cell has been represented by a single-polarization equivalent circuit model, also known as the first-order RC equivalent circuit model. Battery voltage V_b and current I_b can be calculated from the following equations system:

$$\begin{cases} V_b(t) = \left(V_{OC} - R_0 \cdot I_b(t)/n_p - V_1 \right) \cdot n_s \\ P_b(t) = V_b(t) \cdot I_b(t) \end{cases} \qquad (21)$$

where V_1 is the voltage drop related to the RC circuit, n_p, n_s are the number of cells in parallel and in series, respectively, and $P_b(t)$ is the battery power request coming from Eq. (20). Finally, the battery state of charge is estimated with an Ampere-hour (Ah) integral method (also known as Coulomb counting) [34]. In formula:

$$\xi(t) = \xi_i - \int_0^{\Delta t} \frac{\eta_c \, I_b(t)}{C_n} dt \qquad (22)$$

being C_n the nominal battery capacity, ξ_i the initial value of the state of charge, and η_c the coulombic efficiency. From Eq. (22), the net amount of SoC needed to drive the ZEZ in pure electric drive can be calculated as the difference between the maximum and minimum value of the predicted SoC profile ξ:

$$\Delta\xi_Z = \max(\xi) - \min(\xi) \qquad (23)$$

Then, the target SoC value is calculated as follows:

$$\xi_t = \xi_{min,b} + \Delta\xi_Z + \xi_s \qquad (24)$$

where $\xi_{min,b}$ is the minimum SoC to drive the vehicle in pure electric mode (in this application, it is set to 20%), ξ_s is a positive offset value that has been set to 5% to compensate for the physiological inaccuracies of the navigation data, as proved in [35].

Finally, ξ_t is the output of the model representing the minimum value of SoC to be stored in the HV battery to perform the ZEZ in pure electric, which is then forwarded to the control strategy.

3.2. Energy Management Strategies

The presence of a ZEZ along the route adds a boundary condition for the EMS. Without the connectivity and the information from the server, the conventional strategy works blindly, thus the driver could run into fees or traffic limitations imposed by the local municipalities. Conversely, the control strategy has to fulfill an additional objective besides the fuel consumption, which is granting the energy to drive the ZEZ in full electric drive. Then, when the ZEZ is reached by the vehicle, the strategies are bypassed, and the electric drive is forcedly switched on.

In this chapter, the RBS presented in [17] is modified into an Adaptive-RBS to increase its efficiency and then compared with the A-ECMS at the SiL on two complete Real-Driving Emissions (RDE) cycles. Consequently, analyzing the results, a Combined-RBS is finally defined to test the predictive functions at the C-HiL.

3.2.1. Adaptive ECMS

According to the works presented in literature and summarized in the sec. 1, an attempt to apply the adaptive formulation of the ECMS to PHEVs is proposed in [36,37]. It can be noticed that there are two main contributions to the adaptive formulation of the equivalence factor: the penalty function and the adaptive function. The first one aims at maintaining the state of charge of the battery within the range $[\xi_{min,b}, \xi_{max,b}]$ that are the battery limits. The second one is the

adaptive function and since the paper focuses on it, only the latter is reported:

$$f_a(\xi(t), t) = k_a\big(\xi_r(t) - \xi(t)\big) + (s_{k-1} + s_{k-2})/2 \qquad (25)$$

The adaptive function represents a proportional correction of the equivalence factor s by considering an adaptive factor, k_a, and the difference between the reference value of the SoC, ξ_r, and the actual one. Since the adaption is performed periodically, the terms s_{k-1} and s_{k-2} are the values of the equivalence factor used in two previous time intervals, namely adaptation steps, working as an integral correction.

To handle the ZEZ, in [17] the k_a has been calibrated as a 2-D map depending on the remaining distance to the ZEZ ($\Delta d_Z(t) = d_{i,Z} - d(t)$) and the difference between the reference SoC and actual SoC $\Delta\xi_{r,f}(t) = \left|\xi(t) - \xi_{r,f}\right|$.

Moreover, for a PHEV the reference SoC to be followed by the control policy can be expressed as a linear function of the total distance of the trip to gradually discharge the battery during the driving mission. Thus, neither a battery discharging nor sustaining behavior is favored, and then the resulting working mode could be referred to as charge blended. So, the general formulation for a generic drive cycle has been modified as:

$$\xi_r\big(d(t)\big) = \xi_i + \frac{\xi_f - \xi_i}{d_f - d_i}(d(t) - d_i) \qquad (26)$$

with the following assumptions related to the ZEZ:

$$\begin{aligned} t_f &= t_{i,Z} \\ d(t_f) = d_f &= d_{i,Z} \\ \xi(t_f) = \xi_f &= \xi_{i,Z} \end{aligned} \qquad (27)$$

Where $t_{i,Z}$ is the instant of time when the vehicle reaches the ZEZ, $d_{i,Z}$ is the distance of the ZEZ from the actual position, and $\xi_{i,Z}$ is the value of the SoC at the beginning of the ZEZ.
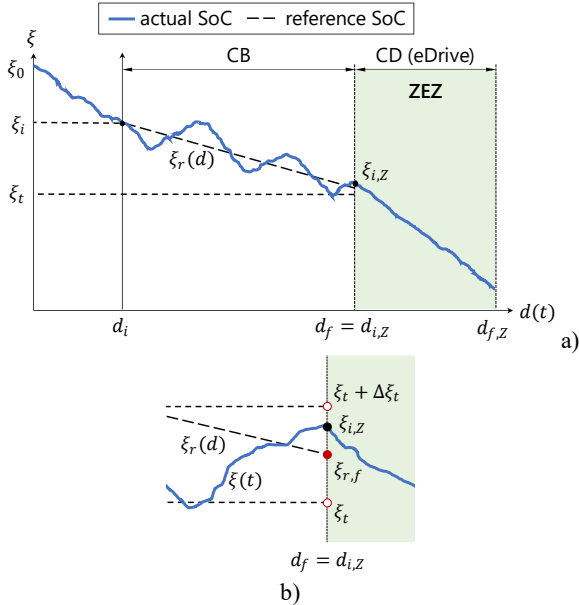

Fig. 9   *Charge Blended (CB) A-ECMS handling the ZEZ: a) actual and reference SoC trends, b) particular of the beginning of the ZEZ. [17]*

A qualitative use case is shown in Fig. 9a, where a generic prediction happens in between the driving cycle (point $d_i \in [d_0, d_{f,Z}]$). In general, the first prediction occurs when the driver selects the destination on the navigator, and it is performed again every time a re-routing or changes in traffic condition occurs. Here, the CB mode is applied by the A-ECMS following the reference SoC expressed by Eq. (26) until the urban area is accessed at $d_f = d_{i,Z}$ (received from the MSP via LTE, as shown in Table 1) reached at the instant $t = t_f$.

Moreover, it is important to mention that in contrast to other formulations of the reference SoC, the global constraint $\xi_r(t_f) = \xi_{r,f} = \xi_t$ has been softened, as shown in Fig. 9b. The unused electrical energy associated with the SoC difference $\xi_{i,Z} - \xi_t$ is justified by the more important aim of pursuing a reliable and robust energy management control strategy for handling all the ZEZ in pure-electric driving mode. Therefore, the final reference SoC value becomes the reference SoC range $[\xi_t, \xi_t + \Delta\xi_t]$, centered in $\xi_{r,f}$, with $\xi_{r,f} > \xi_t$ and $\Delta\xi_t = 5\%$.

3.2.2. Adaptive RBS

The conventional RBS originally implemented in the HCU [17,22] of the prototype vehicle controls the electric drive, the front axle torque vectoring, the 4WD control, the torque filling, and the boosting. Focusing on the electric drive, the RBS works in a CD/CS mode through fixed hysteresis concerning the SoC ξ, the torque at the wheels T_w, and the actual vehicle speed v_{veh}. Formally, the switch from electric to hybrid mode, and vice versa, is controlled by Eq. (28) and Eq. (29) respectively:

$$\begin{cases} \xi < \xi_{min} \\ T_w > T_{max}(n) \qquad (28) \\ v_{veh} > v_{max} \end{cases} \qquad \begin{cases} \xi > \xi_{max} \\ T_w < T_{min}(n) \qquad (29) \\ v_{veh} < v_{min} \end{cases}$$

Following the trend in literature, a blended mode leads to increased efficiency for the PHEV [38], especially if road and traffic data are accessible [39]. In fact, if the information about the ZEZ is available, thus the target SoC ξ_t can be higher than the initial SoC of the battery. In [17], the RBS has been modified to receive the navigation data as input, but it can only recharge the HV battery and then performs a conventional CS around the target, as follows:

$$\begin{cases} \xi_{min} = \xi_t \\ \xi_{max} = \xi_t + 5\% \end{cases} \qquad (30)$$

This could lead to an imbalanced comparison since the RBS has not been conceived to work in this operating condition. Thus, the control policy has been adapted to take as input the same SoC reference, ξ_r, defined by Eq. (26), and use it to define the new values of ξ_{max}, ξ_{min} instant by instant:

$$\begin{cases} \xi_{min}\big(d(t)\big) = \xi_r(d(t)) \\ \xi_{max}\big(d(t)\big) = \xi_r(d(t)) + 5\% \end{cases} \qquad (31)$$

3.2.3. CO₂ Correction

The CO_2 production has been chosen as the assessment parameter for comparing the energy consumption related to the strategies under test with respect to the reference one (namely the RBS). To make a proper comparison, a correction of the CO_2 production has been proposed. In particular, the electrical energy additionally used or saved, $\Delta E = E^{AR} - E^{AE}$ (AR: A-RBS, AE: A-ECMS), at the end of the driving mission should be converted into an equivalent amount of fuel as already described in [17]. Hence, the energy correction can be:

- *negative*, meaning that more electrical energy was used by the strategy under test than the reference strategy. In this case, the assumption is that the ΔE is provided by the ICE and the P1 is working as a generator. Thus, considering the energy balance, the mass fuel correction is expressed by:

$$\Delta m_f = \frac{\Delta E}{Q_{lhv}} \cdot \frac{1}{\bar{\eta}_{ICE} \, \bar{\eta}_{ISG}} \qquad (32)$$

where Q_{lhv} is the lower heating value of gasoline, $\bar{\eta}_{ICE}$ and $\bar{\eta}_{ISG}$ are the average efficiencies of the ICE and the ISG respectively. The instantaneous efficiency of the ICE is calculated as the ratio between the mechanical power provided by the engine and the chemical power related to the injected fuel.

- *positive*, on the contrary, if less electrical energy was used by the strategy under test. Here, the ΔE is assumed to be provided by the EMs, as the vehicle is driving in electric mode. Thus, the energy balance can be expressed as:

$$\Delta m_f = \frac{\Delta E}{Q_{lhv}} \cdot \frac{\bar{\eta}_{EM}}{\bar{\eta}_{ICE}} \qquad (33)$$

where $\bar{\eta}_{EM}$ is the average efficiency of the EMs, calculated as the ratio between the input and output power.

Finally, to evaluate the CO_2 production, the corrected fuel consumption $m_{f,c} = m_f + \Delta m_f$ is multiplied by a conversion factor whose value is $k_{CO_2} = 2370$ g$_{CO2}$/l$_{fuel}$. The latter can be calculated as suggested by [40] from the following equation:

$$FC = \left(\frac{0.1206}{\rho_f}\right)(0.829 \cdot HC + 0.429 \cdot CO + 0.273 \cdot CO_2) \qquad (34)$$

where $\rho_f = 0.75$ kg/l is the fuel density, and HC, CO and CO_2 are the production of the relative chemical agents [g/km]. For the conversion factor evaluation, the conservative assumption $HC = CO = 0$ g/km has been made.

3.2.4. Strategies comparison

The simulations of two RDE cycles have been performed at the SiL to compare the strategies in terms of CO_2 production, and in particular to understand if the A-RBS outperforms the conventional RBS and can be used as the new reference. Differently from the conventional RDE cycle, the two used here have been driven in the opposite direction, thus the urban event (and so the ZEZ) is located at the end of the trip. This allows the energy management strategies to prepare the battery with the proper amount of energy for the urban area. The driving cycles are represented in Fig. 10 along with the altitude profile (the 0 value is the initial vehicle position) and the ZEZ (green area) while the length in km of the cycles and of the urban event are listed in Table 5:
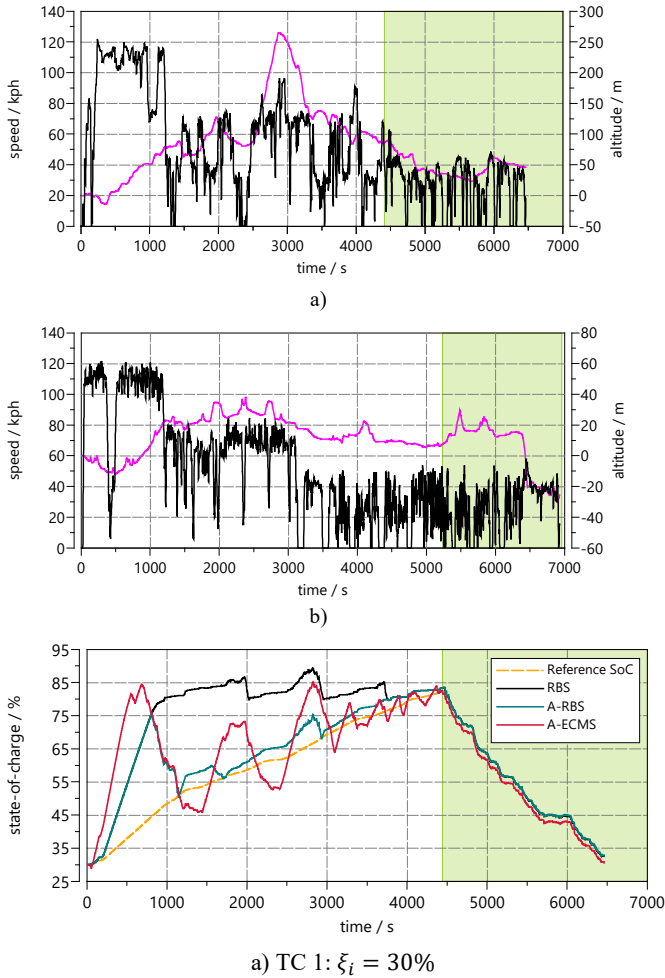

a)

b)

a) TC 1: $\xi_i = 30\%$

Table 5   Length of the route and the ZEZ for each RDE

Cycle	ξ_t [%]	d_Z [km]	$d_{i,Z}$ [km]	$d_{f,Z}$ [km]
RDE 1	80	15.0	75.1	90.1
RDE 2	80	11.8	80.9	92.7

where ξ_t is the target SoC calculated by the BVM, d_Z is the duration of the ZEZ in km, while $d_{i,Z}$ is $d_{f,Z}$ identify the start and the end of the ZEZ.

The comparisons of the different Test Cases (TC) have been made starting from different values of the initial SoC ξ_i (30%, 50%, and 90%), whereas the ZEZ event remains the same and consequently the target SoC. The initial values have been chosen to cover different operation modes of the strategies due to the difference between the initial SoC and the target SoC (in both test cases 80%):

- $\xi_i = 30\%$: implies a significant negative difference and so a demanding recharging phase;
- $\xi_i = 50\%$: in this case, the difference is less pronounced and so the recharging phase;
- $\xi_i = 90\%$: here the difference is positive, as commonly happens with PHEV. This is necessary to understand if the A-RBS proposed is suitable even for a conventional CD/CS mode.

The Fig. 11 a), b), and c) show the simulation results for the RDE 1 at 30%, 50%, and 90% respectively, and likewise the Fig. 11 d), e), and f) for the RDE 2. In particular, the SoC trends for each strategy have been plotted, along with the reference SoC $\xi_r(d(t))$, used by both the A-RBS and the A-ECMS, expressed by Eq. (26). The simulation results for all the test cases are reported in Table 6 in terms of a relative percentage difference of corrected CO_2. The A-RBS and the A-ECMS are compared to the RBS by means of the parameters $\Delta CO_{2,\%}{}^{AR}$ and $\Delta CO_{2,\%}{}^{AE}$.

$$\Delta CO_{2,c,\%}{}^{AR} = \left(CO_2{}^{AR} - CO_2{}^R\right)/CO_2{}^R \cdot 100 \qquad (35)$$

$$\Delta CO_{2,c,\%}{}^{AE} = \left(CO_2{}^{AE} - CO_2{}^R\right)/CO_2{}^R \cdot 100 \qquad (36)$$

where the superscript R stands for RBS, AR for A-RBS, and AE for A-ECMS. Analyzing the results listed in Table 6, for the test cases with $\xi_i = 30\%$ and $\xi_i = 50\%$, where the strategies must work in CI mode (negative values of $\Delta\xi = \xi_i - \xi_t$), the A-RBS performs better than the RBS, as shown by the negative values of $\Delta CO_{2,c,\%}{}^{AR}$. On the other hand, if the strategies must work in the conventional CD/CS mode (positive values of $\Delta\xi$), the RBS still represents the most efficient solution. Differently, the values of $\Delta CO_{2,c,\%}{}^{AE}$ for the A-ECMS highlight the considerable improvements in all the test cases, confirming the results previously obtained in [17].
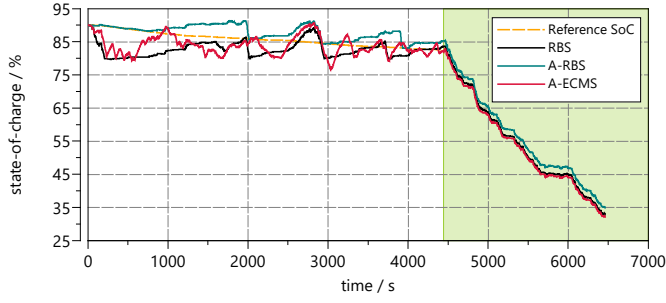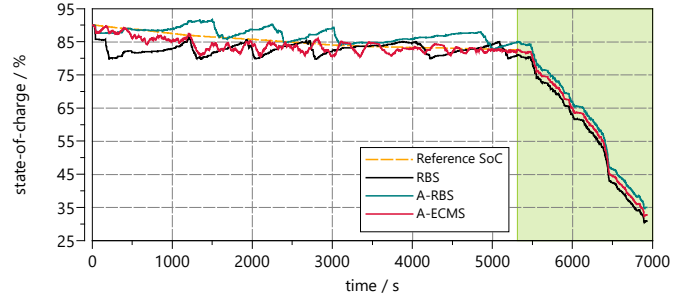

d) TC 2: $\xi_i = 30\%$
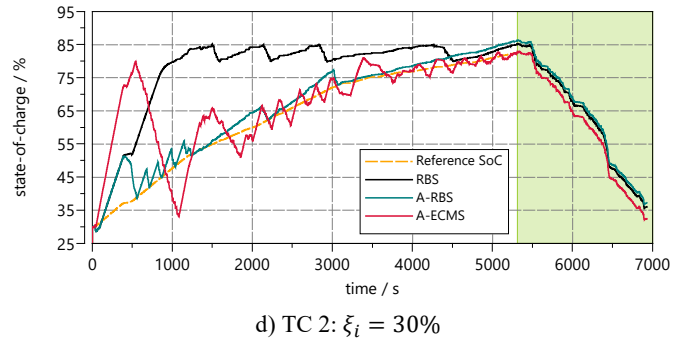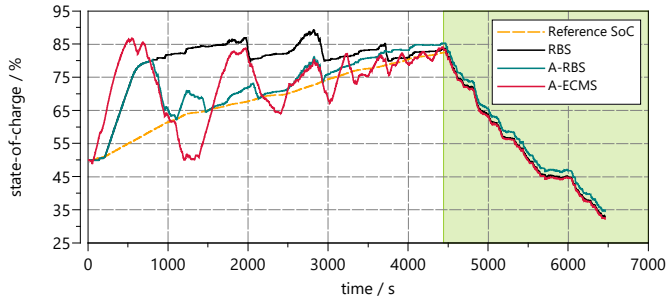
b) TC 3: $\xi_i = 50\%$
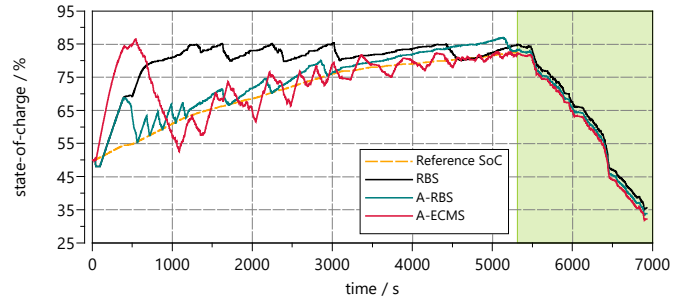
e) TC 4: $\xi_i = 50\%$

c) TC 5: $\xi_i = 90\%$

f) TC 6: $\xi_i = 90\%$

Fig. 11   a, b, c) RDE 1: SoC trends for each strategy approaching the ZEZ (green area) with initial SoC of 30%, 50%, and 90% respectively; d, e, f) RDE 2: SoC trends for each strategy approaching the ZEZ (green area) with initial SoC of 30%, 50%, and 90% respectively

Table 6   Results of the simulations in terms of CO₂ production and relative errors for each test case

TC	RDE	ξ_i [%]	ξ_t [%]	$\Delta\xi$ [%]	$CO_{2,c}^{R}$ [g/km]	$CO_{2,c}^{AR}$ [g/km]	$CO_{2,c}^{AE}$ [g/km]	$\Delta CO_{2,c,\%}^{AR}$ [%]	$\Delta CO_{2,c,\%}^{AE}$ [%]
1	1	30	80	-50	319	305	268	**-4.3**	**-16.1**
2	2	30	80	-50	360	352	296	**-2.2**	**-17.6**
3	1	50	80	-30	307	295	240	**-3.7**	**-21.7**
4	2	50	80	-30	344	334	272	**-3.1**	**-21.1**
5	1	90	80	10	280	288	208	**3.0**	**-25.4**
6	2	90	80	10	312	321	230	**2.8**	**-26.3**

3.2.5. Combined RBS

In conclusion, the tests highlight that for this specific powertrain the A-RBS is more efficient only in CI, while the RBS is still better otherwise. Hence, a Combined-RBS has been identified merging the limits expressed by Eq. (30) and by Eq. (31) in Table 7.

Table 7   C-RBS: state of charge limits for the electric drive

	ξ_{min}	ξ_{max}	
$\xi_i > \xi_t$	ξ_r	$\xi_r + 5\%$	(37)
$\xi_i < \xi_t$	$\xi_r(d(t))$	$\xi_r(d(t)) + 5\%$	(38)

This new strategy, defined as a combination of the RBS and the A-RBS to be the most efficient in terms of CO_2 reduction, is adopted from now on as the reference strategy and it will be tested at the C-HiL and compared to the A-ECMS to confirm the results obtained at SiL level. In Table 8 all the EMS describe so far, and the relative working modes are summarized with respect to the SoC difference $\Delta\xi = \xi_t - \xi_i$.

Table 8   EMS working mode summary

	$\Delta\xi > 0$	$\Delta\xi < 0$
RBS [17]	CD/CS	CI/CS
A-RBS	CB	CB
C-RBS	CD/CS	CB
A-ECMS [17]	CB	CB

It is worth mentioning that the selection of such strategy as the reference one is purely based on nominal efficiency optimization, without taking into consideration effects due to battery C-rating characteristics, aging, or overheating.

4.   C-HiL testing

In this chapter, the C-RBS and the A-ECMS are tested at the C-HiL on a driving profile measured on-road (blue line in Fig. 12). To do so, six different test cases have been defined. More in detail, each test starts querying the MSP to retrieve the online navigation data in real-time for the same destination as the measured trace (namely point B in Fig. 12). Consequently, the HCU calculates the target SoC, as described in sec. 2.4. Since the navigation data depends on the actual traffic conditions, they could differ from the measured data both for the proposed route and the traffic data. When the Navigator App detects that the distance between the GPS position of the vehicle and the planned route is higher than a certain threshold, it triggers a re-routing. Hence, it queries the MSP again for updated navigation data and the HCU calculates a new target SoC. In this way, the tests force the control strategies to work in conditions as near as possible to reality, validating both the predictions of the target SoC and the effectiveness of A-ECMS with respect to the C-RBS in terms of CO_2 production.

4.1.   Test cases

The tests have been performed on a driving profile measured on-board using MATLAB® Online, installed on a smartphone, to retrieve the actual vehicle speed, and GPS position (latitude, longitude, and altitude). To replicate a plausible and realistic scenario, the starting point has been set in the rural area near the city of Bologna, point A in Fig. 12, while the destination has been set on a parking lot in the middle of the ZEZ, point B.
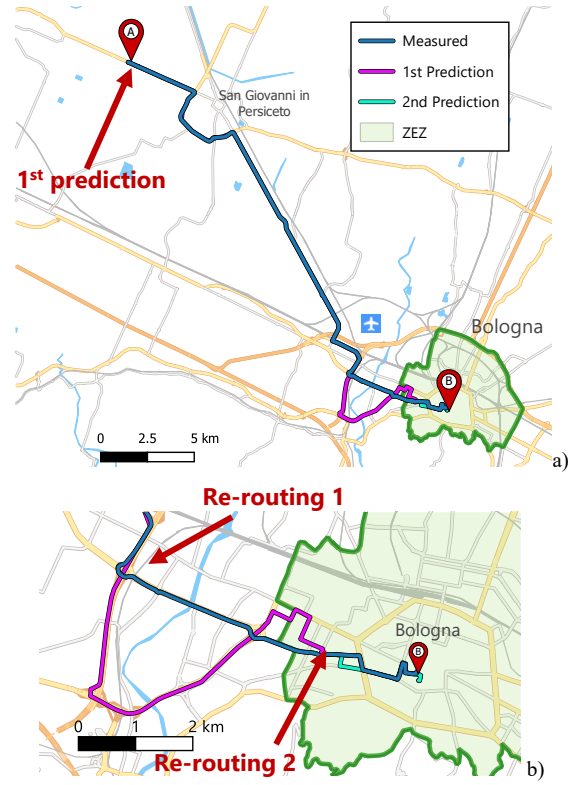

Fig. 12   Representation of the measured route (blue line) from point A to point B and the proposed routes at each query: the first one at the beginning of the test (purple line) and the second related to re-routing 1 (cyan line)

The measurement has been done in the afternoon of a working day to consider medium traffic congestion, while maintaining a driving behavior neither aggressive nor cautious. Then, the same measured data have been set as the reference speed profile to be followed by the vehicle model and then deployed in the Real-Time PC. Finally, the tests have been conducted starting from different values of initial SoC, as for the comparison described in sec. 3.4, at comparable hours of the day, to reduce the physiological variability of the traffic as much as possible. The measured driving cycle is reported in Fig. 13, as well as the altitude and the ZEZ.
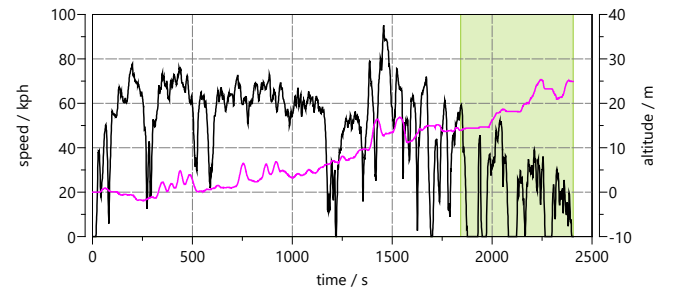

Fig. 13   Real driving profile under test (black), altitude profile (magenta), and the ZEZ area (green)

Table 9   Length of the driving profile and ZEZ for the measured test case

Cycle	d_Z [km]	$d_{i,Z}$ [km]	$d_{f,Z}$ [km]
Measured route	3.0	27.5	30.5

In Table 9, the data regarding the length of the route $d_{f,Z}$, the remaining distance to the ZEZ $d_{i,Z}$, and the urban event are reported. Furthermore, in Table 10 all the test cases are summarized, along with the initial value of SoC ξ_i, and the number of re-routing n_r (excluding the first prediction that is mandatory), which occurred during the simulations.

| TC	EMS	ξ_i [%]	n_r [-]
1	C-RBS	30	0
2	A-ECMS	30	1
3	C-RBS	50	1
4	A-ECMS	50	1
5	C-RBS	90	1
6	A-ECMS	90	0

4.2. Results

In Fig. 14 the results of the testing at the C-HiL have been reported. The subfigures a), b) and c) are referred to the simulations performed with the C-RBS, while d), e) and f) to the ones performed with the A-ECMS at the initial SoC value of 30%, 50%, and 90% respectively. In particular, each subfigure is composed as follows:

- *top plot*: showing the actual HV battery SoC (black line), the reference SoC (dotted line) calculated with Eq. (26), and the target SoC (petrol line) calculated by the BVM with the Eq. (24). The latter is related to the real-time navigation data received during the simulation, so it changes when a re-routing occurs;

- *central plot*: it represents the comparison of the energy associated with each predicted speed profile with respect to the driven one (grey). The comparison is referred to the energies defined by Eq. (12), (13), and (14), namely the positive inertia energy (cyan), the negative inertia energy (green), and the road load (yellow). In addition, the battery consumption predicted by the BVM is also compared (orange) to the actual one. Moreover,

for a matter of graphical representation, the predictions are distributed on the *x*-axis, whereas during the simulations they are performed at the same time once all the navigation data are received. The predictions are highlighted in each plot by means of grey boxes, but since the central plot is an enlargement of the predictions themselves the grey boxes are wider;

- *bottom plot*: the turnaround time (TAT) of the HCU, expressed in milliseconds, is reported to prove the real-time capability of these strategies. It can be noticed that the TAT is well below the maximum allowable time-step of 10ms of the HCU software for the entire duration of the test. However, the TAT presents a peak every time a prediction occurs, reaching higher values closer to the limit. This can be accepted since it occurs very few times during the cycle, and the dimensions of the vector listed in Table 1 have been chosen to avoid software overruns.
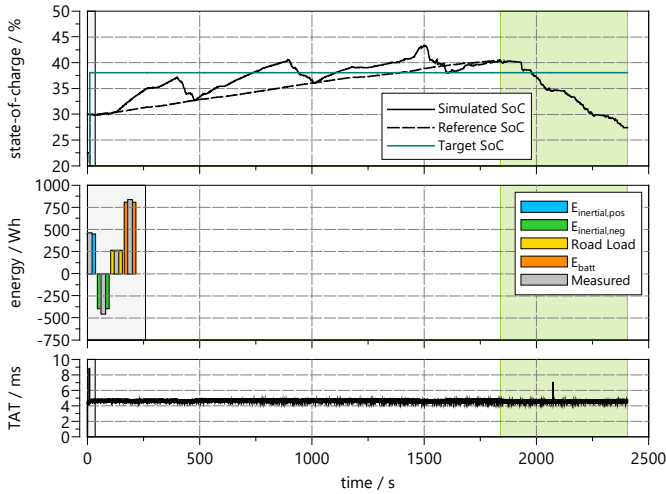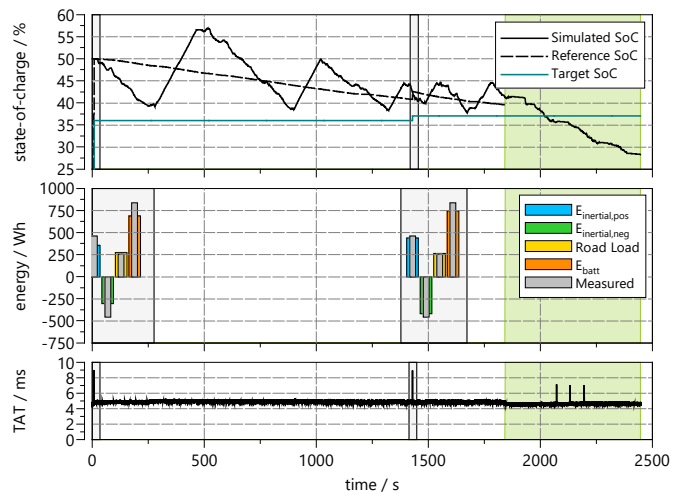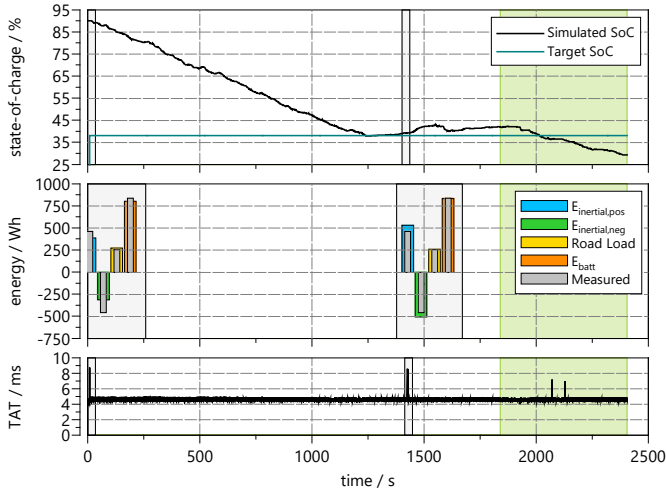

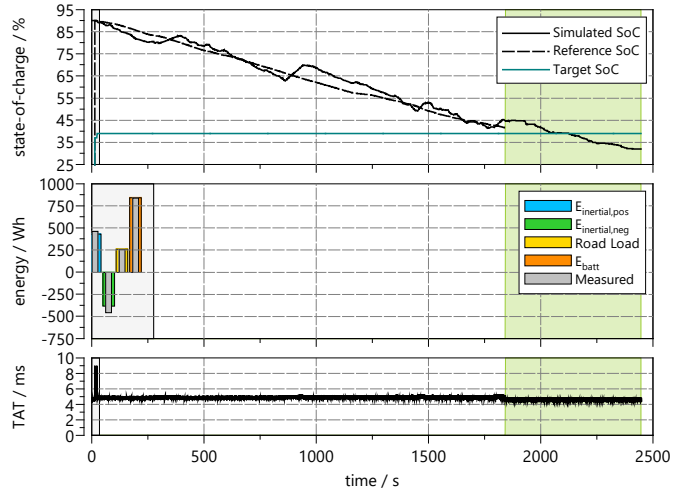a) TC1: $\xi_i = 30\%$
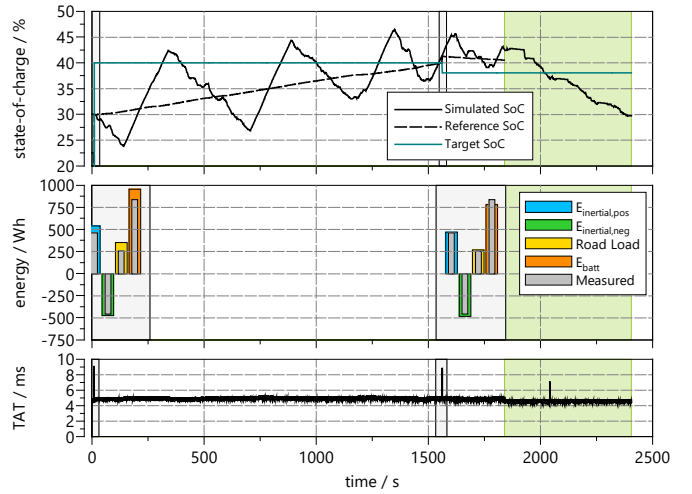
d) TC2: $\xi_i = 30\%$
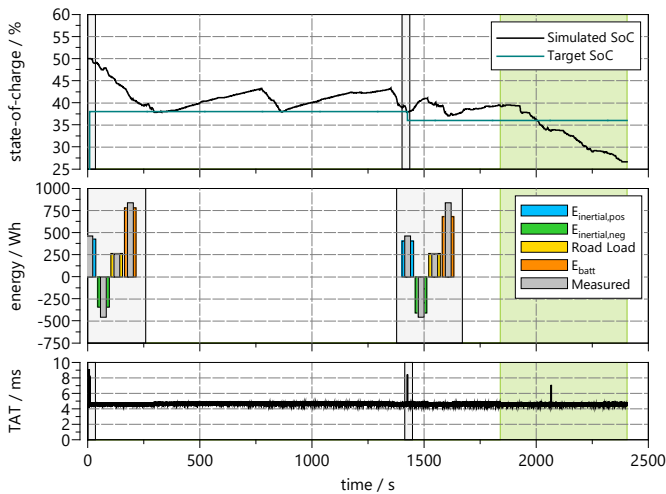
b) TC3: $\xi_i = 50\%$

e) TC4: $\xi_i = 50\%$

c) TC5: $\xi_i = 90\%$

f) TC6: $\xi_i = 90\%$

Fig. 14   Results of the tests at the C-HiL: a) b) c) simulations performed with C-RBS at ξ_i = 30%, 50%, 90% respectively; d) e) f) simulations performed with A-ECMS at ξ_i = 30%, 50%, 90% respectively

Table 11   Results of the tests at the C-HiL: comparison of predictions depending on the initial SoC and the meters remaining to the ZEZ event

| | C-RBS (TC1) | | | | | | A-ECMS (TC2) | | | | | |
| | 1st prediction ($d_{i,Z}$ = 27516 m) | | | 2nd prediction - | | | 1st prediction ($d_{i,Z}$ = 30538 m) | | | 2nd prediction ($d_{i,Z}$ = 3307 m) | | |
	Meas. [Wh]	Pred. [Wh]	$\Delta_\%$ [%]	Meas. [Wh]	Pred. [Wh]	$\Delta_\%$ [%]	Meas. [Wh]	Pred. [Wh]	$\Delta_\%$ [%]	Meas. [Wh]	Pred. [Wh]	$\Delta_\%$ [%]
E_{I^+}	463	449	-3.1	-	-	-	463	543	17.2	463	469	1.2
E_{I^-}	-456	-395	-13.2	-	-	-	-456	-474	4.0	-456	-482	5.7
E_{RL}	257	266	3.2	-	-	-	257	351	36.3	257	270	5.1
E_{batt}	855	808	5.4	-	-	-	836	958	14.6	836	782	6.9
	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]
SoC	12.9	12.4	-0.6	-	-	-	12.9	14.7	1.9	12.9	12.0	-0.8

| | C-RBS (TC3) | | | | | | A-ECMS (TC4) | | | | | |
| | 1st prediction ($d_{i,Z}$ = 26984 m) | | | 2nd prediction ($d_{i,Z}$ = 5910 m) | | | 1st prediction ($d_{i,Z}$ = 26984 m) | | | 2nd prediction ($d_{i,Z}$ = 5917 m) | | |
	Meas. [Wh]	Pred. [Wh]	$\Delta_\%$ [%]	Meas. [Wh]	Pred. [Wh]	$\Delta_\%$ [%]	Meas. [Wh]	Pred. [Wh]	$\Delta_\%$ [%]	Meas. [Wh]	Pred. [Wh]	$\Delta_\%$ [%]
E_{I^+}	463	426	-7.9	463	406	-12.3	463	355	-23.4	463.1	439.9	-8.3
E_{I^-}	-456	-342	-24.9	-456	-409	-10.2	-456	-302	-33.7	-455.5	-417.6	-5.0
E_{RL}	257	264	2.6	257	265	2.9	257	274	6.4	257.4	264.9	2.9
E_{batt}	841	780	7.2	841	682	-18.8	839	690	17.8	839.4	746.5	-11.1
	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]
SoC	12.9	12.0	-0.9	12.9	10.5	-2.4	12.9	10.6	-2.3	12.9	11.5	-1.4

| | C-RBS (TC5) | | | | | | A-ECMS (TC6) | | | | | |
| | 1st prediction $d_{i,Z}$ = 26984 m | | | 2nd prediction $d_{i,Z}$ = 5913 m | | | 1st prediction $d_{i,Z}$ = 27539 m | | | 2nd prediction - | | |
	Meas. [Wh]	Pred. [Wh]	$\Delta_\%$ [%]	Meas. [Wh]	Pred. [Wh]	$\Delta_\%$ [%]	Meas. [Wh]	Pred. [Wh]	$\Delta_\%$ [%]	Meas. [Wh]	Pred. [Wh]	$\Delta_\%$ [%]
E_{I^+}	463	387	-16.5	463.05	529.75	14.4	463	431	-6.9	-	-	-
E_{I^-}	-456	-312	-31.4	-455.53	-505.37	10.9	-456	-380	-16.5	-	-	-
E_{RL}	257	273	6.1	257.44	259.20	0.7	257	261	1.4	-	-	-
E_{batt}	837	802	4.2	837.12	834.15	-0.4	834	844	1.2	-	-	-
	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]
SoC	12.9	12.3	-0.5	12.9	12.8	-0.1	12.8	13.0	0.2	-	-	-

Table 12   Results of the tests at the C-HiL: comparison between the two strategies in terms of raw and corrected CO_2

TC	EMS	$\bar{\eta}_{ICE}$ [%]	$\bar{\eta}_{EM}$ [%]	$\bar{\eta}_{ISG}$ [%]	ξ_i [%]	ξ_f [%]	FC [l/100km]	E [Wh]	CO_2 [g/km]	$\Delta CO_{2,\%}$ [%]	ΔE [Wh]	$CO_{2,c}$ [g/km]	$\Delta CO_{2,c,\%}$ [%]
1	C-RBS	16.0	80.3	75.0	30	27.2	13.3	-139	315				
2	A-ECMS	24.3	80.9	75.6	30	29.7	10.6	-196	250	-20.4	57	250	-20.4
3	C-RBS	14.8	80.0	73.7	50	26.4	10.8	1444	256				
4	A-ECMS	23.6	80.9	75.6	50	28.3	8.4	1184	198	-22.5	260	189	-26.1
5	C-RBS	15.3	81.5	74.4	90	29.1	4.8	3902	113				
6	A-ECMS	21.4	81.2	76.1	90	31.8	4.2	3684	99	-12.5	218	90	-20.1

In Table 11, the predicted energies E_{I+}, E_{I-}, E_{RL}, and E_{batt} are reported and compared to the ones related to the driven route in terms of percentage difference $\Delta_\%$, as in Eq. (15), (16), and (17). Analogously, the predicted battery energy consumption and SoC are also analyzed.

Since the TC2 is the most representative test case, the related scenario is reported on the map in Fig. 12 and it is analyzed in detail in the following paragraph. In fact, it simulates the realistic scenario of a driver that does not follow exactly the suggestion of the navigator, independently if on purpose or not, inducing several re-routing.

Focusing on it, the MSP suggests the fastest route related to the actual real-time traffic data, the purple line in Fig. 12, and the HCU makes the prediction referring to that route. This leads to the first target SoC represented in Fig. 14d and as long as the driver follows the proposed route, the target SoC remains constant. Then, when the driver takes a different road, a re-routing occurs leading to a new query to the MSP, so updated navigation data, cyan line in Fig. 12b, and thus a recalculation of the target SoC, second grey box in Fig. 14b. Looking at the results in terms of energy in Table 11, the first prediction presents relevant errors for most of the KPIs due to the differences of the routes, and consequently a not precise target SoC calculation. After the re-routing, the driver follows the suggested route, so in this case, the energy KPIs present significantly lower errors leading to an accurate prediction. Even if there is such a considerable initial error, and the re-routing occurs close to the ZEZ ($d_{i,Z}$ = 3307 m), the initial target SoC does not change significantly after re-routing, and the EMS can grant the ZEZ in full electric. In Fig. 12b, another re-routing occurs but this time within the ZEZ, identified also by the spike within the green area in the TAT bottom plot. Thus, the TeCU queries the MSP and retrieves the navigation data, but no prediction will be performed. In fact, at this point, the vehicle can only go in electric drive and no countermeasures can be taken by the EMS if the SoC is not enough.

In general, it can be noticed that if the driver follows the proposed route (and so no re-routings occur), the energy predictions well-represent the energy necessary to perform the ZEZ in full electric drive. That is the case of TC1 and TC6, where the SoC errors are -0.6% and 0.2% respectively. For TC2, TC4, and TC5, the driver does not follow the suggested route, inducing re-routings. Even in these cases, the SoC prediction error is below 2.3%, which is covered by the offset ξ_s expressed in Eq. (24), and then improved after the rerouting, as well as the KPIs and the predicted battery's energy consumption. Differently, for what concerns the TC3, the relative errors increase after the re-routing, worsening the energy prediction. This was due to inaccurate values of the real-time traffic data provided by the MSP, which underestimates the congestion along the route. Even if this represents a common situation that can happen when the vehicle is driving on the road, however, the error produced is still compensated by the offset ξ_s.

Then, focusing on Table 12, the C-RBS and the A-ECMS are compared in terms of corrected CO_2 production, calculated with Eq. (32), (33), and (34). In particular, the Table 12 reports the ICE's and electric machines' average efficiency, the fuel consumption FC, the energy consumption E, the raw CO_2, and the corrected CO_2. Similar to the comparison made in sec. 3.4, the relative percentage difference has been chosen as the assessment parameter, both for the raw $\Delta CO_{2,\%}$ and corrected value $\Delta CO_{2,c,\%}$. This comparison can be considered representative since the driven profile is always the same, while the only variable parameter is the target SoC. Although, the latter varies in a limited range of a few percentage points, so it does not significantly affect the EMS. Hence, it can be noticed that for the A-ECMS the average efficiencies of the ICE and the ISG are higher, leading to lower fuel consumption and, apart from the case $\xi_i = 30\%$, a lower energy consumption. This means a reduction of raw and corrected CO_2 production, in a range of 12.5% - 22.5% and 20.1% - 26.1%, respectively. Moreover, the final values of SoC are well above the minimum SoC, $\xi_{min,b}$, proving that the ZEZ has been performed in full electric drive even if the navigation data present some normal inaccuracies, due to different proposed routes and traffic data that do not reflect exactly the current situation.

5.   Conclusions and future works

In this paper, a supervisory controller architecture for PHEVs based on predicted functions and an Adaptive-ECMS has been proposed to handle a Zero-Emission Zone using navigation data retrieved in real-time from the map service provider. With this information, the Speed Profile Prediction is performed to evaluate an energetically equivalent driving profile that is then fed to the BVM. The latter calculates the amount of energy necessary to drive the ZEZ in pure electric mode and forward that target SoC to the control policy. For a proper comparison, also the conventional RBS has to be modified into an A-RBS to handle the ZEZ even in situations where a Charge Increasing mode is required. Thus, the latter has been tested on two inverted RDE cycles while facing a ZEZ event. As a consequence, the test results helped to define a more efficient Combined-RBS that is finally tested on a real driving scenario measured on the road and compared to the A-ECMS at the C-HiL. The latter exploits the real vehicular connectivity, the TeCU, and the HMI to manage the communication with the MSP itself, as it would be in the real vehicle. Since the navigation data from the MSP are real and referred to the current traffic situation, the proposed route can differ from the driven one, leading to re-routing and consequent adaptation of the target SoC.

The results show that:

- The SPP evaluates an energetically equivalent profile with a relative error of the required battery's energy between 0.4% and 11%, which results in a variation of the target SoC between 0.1% to 2.4%. These values are compensated by the SoC offset used to calculate the target SoC, so the ZEZ in pure electric is always granted. However, the SPP prediction is strongly dependent on the accuracy of the navigation and traffic data as shown in TC3, and on the driver behavior;

- The A-ECMS outperforms the C-RBS in all the test cases in terms of fuel consumption and so corrected CO_2 production, with a reduction between 20.1% and 26.1%. Thus, the proposed predictive strategy not only grants the fully electric drive in an urban event, preventing the payment of fees, but also optimizes the fuel consumption while driving outside the ZEZ;

- Both strategies have been deployed into the real HCU and tested with real vehicular connectivity, proving the real-time capability and robustness of the predictive functions under different and unpredictable conditions. This has accelerated the function development laying the ground for future on-road tests.

Even if the C-HiL tests already provide positive results under challenging scenarios, further tests are required. Firstly, a scenario simulator and an improved driver model must be implemented at the C-HiL to create multiple test cases without measuring them on the road. Then, additional tests on the vehicle have to be fulfilled to definitely prove the effectiveness of the control policy. In addition to that, the influence of different charging and discharging strategies on battery health needs to be analyzed to consider the effects due to battery C-rating characteristics, aging, or overheating. Moreover, the testing with real connectivity underlined how the SPP still presents room for improvement. On one hand, despite the good results presented in the paper, the validation of the SPP is limited to the scenarios under test, thus a wider and more generic campaign is needed. On the other hand, it should be enhanced with the addition of Machine Learning algorithms to adapt itself with respect to the driver's behavior and so preventing physiological variability. Finally, the TAT measurements show how the predictions affect the HCU, thus the next step is to transfer all the computational burden firstly to the TeCU and then to a cloud server, since the information exchange does not need very low latencies.

Nomenclature

Abbreviations

A-ECMS	Adaptive ECMS
A-RBS	Adaptive RBS
BVM	Backward Vehicle Model
CB	Charge Blended
CD	Charge-Depleting
C-HiL	Connected Hardware-in-the-Loop
CI	Charge-Increasing
C-RBS	Combined RBS
CS	Charge-Sustaining
ECMS	Equivalent Consumption Minimization Strategy
EM	Electric Machine
EMS	Energy Management Strategy
FC	Fuel Consumption
HCU	Hybrid Control Unit
HMI	Human Machine Interface
ICE	Internal Combustion Engine
MAS	Max Allowed Speed
MSP	Map Service Provider
PHEV	Plug-in Hybrid Electric Vehicle
RBS	Rule-Based Strategy
RDE	Real Driving Emissions
SoC	State of Charge
SPP	Speed Profile Prediction
TAT	Turnaround Time
TC	Test Case
TeCU	Telecommunication Control Unit
V2N	Vehicle-to-Network
ZEZ	Zero-Emission Zone

Roman Symbols

a_{acc}, a_{dec}	Vehicle acceleration and deceleration
$a_{acc,max}, a_{dec,max}$	Maximum vehicle acceleration / deceleration
A_r, f_r	Amplitude and frequency of the r-th range
c	Traffic code

c_s	Corrective factor for close segments
$\Delta CO_{2,\%}$	Relative percentage difference of raw CO_2
$\Delta CO_{2,c,\%}$	Relative percentage difference of corrected CO_2
CW	Code weight
d	Predicted speed profile discretization step
$d_{i,Z}, d_{f,Z}$	Initial/final distance delimiting the ZEZ
d_Z	ZEZ length
E_{I^+}, E_{I^-}	Positive / negative inertia energy
E_{batt}	Battery energy
E_{RL}	Road load
$\Delta E_\%$	Relative percentage difference of energy
f_a	Adaptive function
k_a	ECMS adaptive factor
k_{acc}, k_{dec}	Acceleration / deceleration reductive factors
k_d	Minimum number of sub-segments to create an acceleration-deceleration maneuver
n_c	Number of close segments
n_r	Number of re-routings
n_v	Maximum number of elements of output vector
P_{aux}	Auxiliary components power
P_{batt}	Battery power
P_{EM}	Electric motor power
v_{lim}	Road segment's legal speed limit

Greek symbols

$\bar{\eta}_{ICE}, \bar{\eta}_{EM}, \bar{\eta}_{ISG}$	Engine, EM, and ISG average efficiencies
$\xi_{Z,i}, \xi_{Z,f}$	SoC at ZEZ entrance / exit
ξ_i, ξ_f	Initial / final value of SoC
$\xi_{min,b}, \xi_{max,b}$	Minimum / maximum allowed HV battery SoC
ξ_{min}, ξ_{max}	Minimum / maximum SoC thresholds for RBS
ξ_r	Reference SoC
ξ_s	SoC offset
ξ_t	Target SoC
$\Delta\xi_Z$	Net amount of SoC for ZEZ in full electric drive

References

[1] European Commission, Joint Research Centre. Atlas of the human planet 2020: open geoinformation for research, policy, and action. Publications Office; 2021, url: https://data.europa.eu/doi/10.2760/562514 (accessed March 17, 2022).

[2] Regulation (EU) 2019/631 of the European Parliament and of the Council of 17 April 2019 setting CO2 emission performance standards for new passenger cars and for new light commercial vehicles, and repealing Regulations (EC) No 443/2009 and (EU) No 510/2011 (Text with EEA relevance.). vol. 111. 2019.

[3] Mueller J, Le Petit Y. Low-Emission Zones are a success - but they must now move to zero-emission mobility. Eur Fed Transp Environ AISBL 09/19. url: https://www.transportenvironment.org/sites/te/files/publications/2019_09_Briefing_LEZ-ZEZ_final.pdf.

[4] Banvait H, Anwar S, Chen Y. A rule-based energy management strategy for Plug-in Hybrid Electric Vehicle (PHEV). 2009 Am. Control Conf., 2009, p. 3938–43. doi: 10.1109/ACC.2009.5160242.

[5] A Rule-Based Strategy for a Series/Parallel Hybrid Electric Vehicle: An Approach Based on Dynamic Programming | DSCC | ASME Digital Collection. url: https://asmedigitalcollection.asme.org/DSCC/proceedings-abstract/DSCC2010/44175/507/345430 (accessed March 21, 2022).

[6] Onori S, Serrao L, Rizzoni G. Equivalent Consumption Minimization Strategy. In: Onori S, Serrao L, Rizzoni G, editors. Hybrid Electr. Veh. Energy Manag. Strateg., London: Springer; 2016, p. 65–77. doi: 10.1007/978-1-4471-6781-5_6.

[7] Tian Y, Liu J, Yao Q, Liu K. Optimal Control Strategy for Parallel Plug-in Hybrid Electric Vehicles Based on Dynamic Programming. World Electr Veh J 2021;12:85. doi: 10.3390/wevj12020085.

[8] Paganelli G. Conception et commande d'une chaîne de traction pour véhicule hybride parallèle thermique et électrique. Doctoral Thesis. Université de Valenciennes et du HainautCambrésis, 1999, url: https://www.theses.fr/1999VALE0024.

[9] Paganelli G, Delprat S, Guerra TM, Rimaux J, Santin JJ. Equivalent consumption minimization strategy for parallel hybrid powertrains. Veh. Technol. Conf. IEEE 55th Veh. Technol. Conf. VTC Spring 2002 Cat No02CH37367, vol. 4, 2002, p. 2076–81 vol.4. doi: 10.1109/VTC.2002.1002989.

[10] Musardo C, Rizzoni G, Guezennec Y, Staccia B. A-ECMS: An Adaptive Algorithm for Hybrid Electric Vehicle Energy Management. Eur J Control 2005;11:509–24. doi: 10.3166/ejc.11.509-524.

[11] Yu Y, Jiang J, Wang P, Li J, Yu Y, Jiang J, et al. A-EMCS for PHEV based on real-time driving cycle prediction and personalized travel characteristics. Math Biosci Eng 2020;17:6310–41. doi: 10.3934/mbe.2020333.

[12] Chen D, Kim Y, Stefanopoulou AG. Predictive Equivalent Consumption Minimization Strategy With Segmented Traffic Information. IEEE Trans Veh Technol 2020;69:14377–90. doi: 10.1109/TVT.2020.3034552.

[13] Climent H, Pla B, Bares P, Pandey V. Exploiting driving history for optimising the Energy Management in plug-in Hybrid Electric Vehicles. Energy Convers Manag 2021;234:113919. doi: 10.1016/j.enconman.2021.113919.

[14] Zanelli A, Servetto E, De Araujo P, Vankayala SN, Vondrak A. Numerical Assessment of Auto-Adaptive Energy Management Strategies Based on SOC Feedback, Driving Pattern Recognition and Prediction Techniques. Energies 2022;15. doi: 10.3390/en15113896.

[15] Feng J, Han Z, Wu Z, Li M. A Dynamic ECMS Method Considering Vehicle Speed Pattern and Minimum Engine Operation Time for a Range-Extender Electric Vehicle. IEEE Trans Veh Technol 2022;71:4788–800. doi: 10.1109/TVT.2022.3148268.

[16] Soldo J, Skugor B, Deur J. Optimal Energy Management Control of a Parallel Plug-In Hybrid Electric Vehicle in the Presence of Low Emission Zones. Warrendale, PA: SAE International; 2019. doi: 10.4271/2019-01-1215.

[17] Capancioni A, Brunelli L, Cavina N, Perazzo A. Development of Adaptive-ECMS and predictive functions for Plug-in HEVs to Handle Zero-Emission Zones Using Navigation Data. Warrendale, PA: SAE International; 2021. doi: 10.4271/2021-24-0105.

[18] Xu F, Shen T. Look-Ahead Prediction-Based Real-Time Optimal Energy Management for Connected HEVs. IEEE Trans Veh Technol 2020;69:2537–51. doi: 10.1109/TVT.2020.2965163.

[19] Szendrei Z, Varga N, Bokor L. A SUMO-Based Hardware-in-the-Loop V2X Simulation Framework for Testing and Rapid Prototyping of Cooperative Vehicular Applications. In: Jármai K, Bollò B, editors. Veh. Automot. Eng. 2, VAE 2018: Springer, Cham; 2018, p. 426–40. doi: 10.1007/978-3-319-75677-6_37 .

[20] Shao Y, Zulkefli MAM, Sun Z, Huang P. Evaluating connected and autonomous vehicles using a hardware-in-the-loop testbed and a living lab. Transp Res Part C Emerg Technol 2019;102:121–35. doi: 10.1016/j.trc.2019.03.010.

[21] Kim Y, Tay S, Guanetti J, Borrelli F, Miller R. Hardware-In-the-Loop for Connected Automated Vehicles Testing in Real Traffic. 21st Int. Conf. Intell. Transp. Syst., Maui, HI: IEEE; 2018, p. 2878–83. doi: 10.1109/ITSC.2018.8569753.

[22] Brunelli L, Capancioni A, Gonnella P, Casadio R, Brusa A, Cavina N, et al. A Hybrid Vehicle Hardware-in-the-Loop System With Integrated Connectivity for Ehorizon Functions Validation. IEEE Trans Veh Technol 2021;70:4340–52. doi: 10.1109/TVT.2021.3073807.

[23] Boccolini M. Development of a speed profile prediction algorithm based on navigation data for energy management optimization. Master Thesis. University of Bologna, 2021. url: https://amslaurea.unibo.it/22543/.

[24] HERE SDK: Online, Offline and Global Maps for iOS and Android | HERE n.d. https://developer.here.com/products/here-sdk (accessed March 22, 2022). url: https://developer.here.com/products/here-sdk (accessed March 22, 2022).

[25] Lefèvre S, Sun C, Bajcsy R, Laugier C. Comparison of parametric and non-parametric approaches for vehicle speed prediction. 2014 Am. Control Conf., 2014, p. 3494–9. doi: 10.1109/ACC.2014.6858871.

[26] Böhme TJ, Frank B. Hybrid Systems, Optimal Control and Hybrid Vehicles: Theory, Methods and Applications. Cham: Springer International Publishing; 2017. doi: 10.1007/978-3-319-51317-1.

[27] Liu K, Asher Z, Gong X, Huang M, Kolmanovsky I. Vehicle Velocity Prediction and Energy Management Strategy Part 1: Deterministic and Stochastic Vehicle Velocity Prediction Using Machine Learning. Warrendale, PA: SAE International; 2019. doi: 10.4271/2019-01-1051.

[28] SAE Technical Standard. J2951: Drive Quality Evaluation for Chassis Dynamometer Testing - SAE International. url: https://www.sae.org/standards/content/j2951_201401/.

[29] Lekshmi S., Lal Priya P.s. Mathematical modeling of Electric vehicles - A survey. Control Eng Pract 2019;92:104138. doi: 10.1016/j.conengprac.2019.104138.

[30] Hofman T, van Leeuwen D. Analysis of modeling and simulation methodologies for vehicular propulsion systems. 2009 IEEE Veh. Power Propuls. Conf., 2009, p. 1619–26. doi: 10.1109/VPPC.2009.5289633.

[31] Trigui R, Vinot E, Jeanneret B. Backward Modeling and Energy Management Optimization of a Two-Clutches Series-Parallel HEV for Efficiency Assessment. IFAC Proc Vol 2012;45:422–7. doi: 10.3182/20120902-4-FR-2032.00075.

[32] Mohan G, Assadian F, Longo S. Comparative analysis of forward-facing models vs backwardfacing models in powertrain component sizing. IET Hybrid Electr. Veh. Conf. 2013 HEVC 2013, 2013, p. 1–6. doi: 10.1049/cp.2013.1920.

[33] Guzzella L, Sciarretta A. Vehicle Propulsion Systems. Berlin, Heidelberg: Springer Berlin Heidelberg; 2013. doi: 10.1007/978-3-642-35913-2.

[34] Rui Xiong, Weixiang Shen. Advanced Battery Management Technologies for Electric Vehicles. Wiley; 2019. ISBN: 978-1-119-48164-5

[35] Capancioni A . Development of predictive energy management strategies for hybrid electric vehicles supported by connectivity. Doctoral Thesis. Alma Mater Studiorum - Università di Bologna, 2022. doi:10.48676/unibo/amsdottorato/10044.

[36] Cavina N, Caramia G, Patassa S, Caggiano M. Predictive Energy Management Strategies for Hybrid Electric Vehicles: Fuel Economy Improvement and Battery Capacity Sensitivity Analysis, SAE Technical Paper 2018-01-0998, 2018. doi: 10.4271/2018-01-0998.

[37] Caramia G, Cavina N, Caggiano M, Patassa S, Moro D. Battery state of charge management strategies for a real-time controller of a Plug-in Hybrid Electric Vehicle. Energy Procedia 2018;148:258–65. doi: 10.1016/j.egypro.2018.08.076.

[38] Lei Z, Qin D, Zhao P, Li J, Liu Y, Chen Z. A real-time blended energy management strategy of plug-in hybrid electric vehicles considering driving conditions. J Clean Prod 2020;252:119735. doi: 10.1016/j.jclepro.2019.119735.

[39] Padmarajan BV, McGordon A, Jennings PA. Blended Rule-Based Energy Management for PHEV: System Structure and Strategy. IEEE Trans Veh Technol 2016;65:8757–62. doi: 10.1109/TVT.2015.2504510.

[40] Commission Regulation (EU) 2017/1151 of 1 June 2017 supplementing Regulation (EC) No 715/2007 of the European Parliament and of the Council on type-approval of motor vehicles with respect to emissions from light passenger and commercial vehicles (Euro 5 and Euro 6) and on access to vehicle repair and maintenance information, amending Directive 2007/46/EC of the European Parliament and of the Council, Commission Regulation (EC) No 692/2008 and Commission Regulation (EU) No 1230/2012 and repealing Commission Regulation (EC) No 692/2008 (Text with EEA relevance). vol. 175. 2017.