Available online at www.sciencedirect.com

ScienceDirect

Energy Procedia 101 (2016) 685 – 692

Energy Procedia

71st Conference of the Italian Thermal Machines Engineering Association, ATI2016, 14-16 September 2016, Turin, Italy

Analysis of the performance of different machine learning techniques for the definition of rule-based control strategies in a parallel HEV

Mattia Venditti*

IC Engines Advanced Laboratory, Dipartimento Energia, Politecnico di Torino
c.so Duca degli Abruzzi 24, 10129 - Torino, Italy

Abstract

Two different machine-learning techniques have been assessed and applied to define rule-based control strategies for a parallel hybrid midsize sport utility vehicle equipped with a diesel engine. Both methods include two phases: a clustering algorithm and a rule definition. In the first method, a homemade clustering algorithm is preliminarily run to generate the set of clusters, while the rules are identified by minimizing an objective function. In the second method, a genetic algorithm provides the optimal size of the clusters, while the associated rules are extracted from the results obtained with a benchmark optimizer. The controllers were tested over NEDC, 1015, AMDC and WLTP.

Peer-review under responsibility of the Scientific Committee of ATI 2016.

Keywords: Machine learning; rule-based controller; genetic algorithm; dynamic programming

1. Introduction

The present paper is focused on the development of machine-learning methods aimed at the identification of an optimal rule-based control strategy for Hybrid Electric Vehicles (HEV). A rule-based control strategy has in fact the advantage of being easily implementable in the vehicle control unit for the on-board application; however, traditional rule-based controllers are generally developed and tuned on the basis of engineering experience or

* Corresponding author. Tel.: +39-011-090-4493; fax: +39-011-090-4599.
E-mail address: mattia.venditti@polito.it

Peer-review under responsibility of the Scientific Committee of ATI 2016.
doi:10.1016/j.egypro.2016.11.087

heuristic approaches, which might be lacking in performance over a broad range of applications and could require additional human effort if they are applied to different vehicles, hybrid architectures or driving conditions [1-4]. Machine-learning techniques have been adopted in order to extract the rules that need to be implemented in the controller. The reduction in human effort to tuning the controller for different applications represents one of the main strengths of this technique, with respect to rule-based and fuzzy control approaches presented in the literature. The online computational time of the rule-based controller algorithm developed in this study has resulted to be much lower than that required by a conventional Equivalent Consumption Minimization Strategy (ECMS). It is also simple to overwrite some rules if other constraints may apply (for instance, safety issues). One of the main weaknesses of the method concerns the training computational time, since more parameters are in general required to be tuned. Dynamic programming [5, 6] has been used to identify the optimal policy and to assess the outcomes of the proposed rule-based controllers. In principle, since the performances of the new proposed technique are very close to that of the DP-based benchmark, no further comparisons have been carried out.

A rule-based controller is here intended to employ an optimal rule that associates the values of the control variables to the clusters obtained from some input variables. The clustering procedure and the rule definition are the two key stages of this approach. The method developed and presented in [7], i.e., Cluster Optimization and Rule Extraction (CORE), has been improved in terms of training computational time and performance. It has also been compared to a second method, i.e. Cluster Extraction and Rule Optimization (CERO), which employs a different approach for generating the clusters and the rule set.

Nomenclature

1015	Japanese Cycle 10-15 mode
AMDC	Artemis Motorway Driving Cycle
CERO	Cluster Extraction and Rule Optimization
CORE	Cluster Optimization and Rule Extraction
CO_2	Carbon dioxide
DP	Dynamic Programming
EM	Electric Machine
FC	Fuel Consumption
GA	Genetic Algorithm
HEV	Hybrid Electric Vehicle
NEDC	New European Driving Cycle
SOC	State of Charge
WLTP	Worldwide harmonized Light Vehicles Test Procedures

2. Description of the vehicle model

This study focuses on the comparison of a hybrid sport utility vehicle to a reference conventional vehicle, using a kinematic vehicle model developed in the Matlab environment [8]. The hybrid vehicle is equipped with a 1.7L diesel engine (rated power of 98 kW), while the conventional vehicle with a 2.9L diesel engine (rated power of 130kW). The hybrid layout also presents of an electric machine (EM), which is connected to the engine via a single-speed gearbox. The driveline consists of a 6-speed transmission and a final drive. Finally, the battery provides power to the electric machine through the inverter. The vehicle chassis mass is 1000 kg, the frontal area 2.57 m^2, the drag resistance coefficient 0.36 and the tire radius is 0.359 m.

The performance of the engine was modelled using experimentally-derived look-up tables. The mass flow rate of both fuel and NO_x emissions were evaluated by interpolating a 2D map. The corresponding CO_2 emissions have instead been linearly determined from the fuel consumption [7, 9]. The mass of each engine and the related after-treatment system has been estimated as a function of the maximum power. The model of the electric machine (a brushless permanent magnet machine that provides a peak power of 20 kW) simulates the power conversion from the mechanical to the electric form, using an efficiency map [10]. The mass of each electric machine has been estimated to be 25 kg. The model of the battery (lithium-ion) is represented by an equivalent resistance circuit, in which the resistance and the open-circuit voltage of the battery are SOC-dependent. The mass of the battery is determined as a function of the power-to-energy ratio (30 W/Wh) and the maximum battery power (20 kW), and is equal to 20 kg. The total energy content is 650 Wh.

The transmission power losses have been estimated by means of an efficiency map, which is a function of the output shaft speed and torque, as well as of the selected gear number. The final drive and the single-speed gearbox have instead been modelled as a torque multiplier, with no power loss. The speed ratio values of the final drive and the gearbox have been set to 1 and 2.75, respectively. Further details of the model can be found in [9-10] and data have been extracted from several works in literature, such as [11-13].

3. Optimal policy

The optimal policy or control strategy is determined by minimizing an objective function, J, while the constraint on the final SOC has to be guaranteed, since the architecture is not of the plug-in type. The mathematical formulation of the problem is:

$$u^*(t) = \min_{u \in S^*(t)} J(u(t)) = \min_{u \in S^*(t)} \gamma \cdot \frac{M_{fc}}{M_{fc,rv}} + (1-\gamma) \cdot \frac{M_{nox}}{M_{nox,rv}} \qquad |SOC(0) - SOC(t_{end})| \le \varepsilon \tag{1}$$

where *u(t)* is the generic policy and $S^*(t)$ is the set of feasible combinations of control variables for each time interval *t*. The policy *u(t)* defines both the transmission speed ratio, τ, and the power flow, α. The α variable defines the power share that has to be managed by the battery, with respect to the total required power demand. Its domain has been discretized in order to implement the DP technique ($N_\alpha = 7$). One pure-electric ($\alpha = 1$) and one pure-thermal ($\alpha = 0$) modes are possible for this hybrid architecture. The electric machine can work either to assist the engine (power-split, i.e. $\alpha = 1/3, 2/3$) or to charge the battery ($\alpha = -1/3, -2/3, -1$). In the J function, M_{fc} and $M_{fc,rv}$ are the cumulated mass of fuel consumed over the mission by the hybrid and reference vehicle, respectively (kg); M_{nox} and $M_{nox,rv}$ are the cumulated NO_x emission mass of the hybrid and the reference vehicle equipped with the same downsized engine, respectively (kg). The weighting factor, γ, has been introduced in order to adjust the relative importance of the related terms and set to 0.95.

Dynamic programming (DP) [5-6] has been used to solve the problem stated in Eq. (1). Previously developed techniques have been used to speed up the computational time required by the optimization process. In particular, the “configuration matrix” approach has been adopted [9]. In this way, DP can rely on pre-calculated data to extract the optimal policy, instead of continuously executing the vehicle model, which is the main cause of high computational time. The DP technique is not suitable for on-board implementation, since the vehicle velocity is in general not a-priori known and since the computational effort is very demanding for the vehicle control unit hardware. However, it is adopted to determine the optimal policy, which represents the benchmark for the assessment of each rule-based controller.

4. Rule-based controllers

4.1. Deterministic automaton and genetic algorithm

Each rule-based controller employs an optimal rule, R, that associates the values of three input variables (the vehicle velocity and acceleration and the battery SOC) to the values of the control variables (τ and α). Genetic algorithms (GA) [14-15] define the optimal set of rules for a given vehicle mission.

The vehicle velocity, vehicle acceleration and the battery SOC generated a 3D input domain, which is discretized in order to obtain a mesh of input clusters. The rule defines the action to take, in terms of τ and α, for each cluster of the 3D space. The clustering procedure and the rule definition are the two key stages of this method.

Machine learning techniques have been employed to offline train specific parameters of each tool, in order to be consistent over different driving missions. The training stage may require some offline computational time;

however, the real-time trained controller algorithm is fast and computationally light. It is also simple to overwrite some rules if other constraints may apply.

4.2. Cluster Extraction and Rule Optimization (CERO)

The outcomes of the CERO tool are the domain clustering and the set of rules. The offline training stage for the CERO tool is here presented. The tool receives the vehicle velocity, acceleration from the input mission data. The two values at a given time instant can be visualized as a point in the 2D space. A set of clusters is determined using a home-made clustering algorithm. This method partitions M points (dots in Fig. 1b) that represent the acceleration and speed of the vehicle, into k clusters (the figure shows 6 regions with different grey shades). The clustered 2D space represents a layer that has to be repeated as many times as the discrete values of the battery SOC (3 layers are shown in Fig. 1b). The number of k clusters times the number of battery SOC layers represents the dimension of the GA individual (Fig. 1a). The GA individual is the rule that univocally associates the clusters of the input domain to the action to take by the controller. In other words, the action specifies the values of the two control variables, τ and α .

The training process aims at identifying the individual, i.e., the list of actions per each cluster, that minimizes the J function (see Eq. (1)), where the energy variation in the battery is converted into an equivalent variation of FC and NO_x emissions. The battery SOC variation is not constrained to be nearly zero over the mission, but equivalent FC and NO_x-emission terms must be introduced in the J definition to account for the final energy content of the battery (see J' in in Fig. 1d). The training process has to be carried out over several driving cycles in order to test the cycle-independence of the results.

After the training process is ended, i.e., the tool has determined the optimal rule associated to the mesh of clusters, for any given driving mission, the mesh of clusters and the set of rules are embedded into the real-time software of the CERO. It then receives the current vehicle velocity and acceleration, as well as the battery SOC. It determines the cluster using a simple Euclid distance algorithm and extracts the associated action from the rule, so that the powertrain can be actuated.

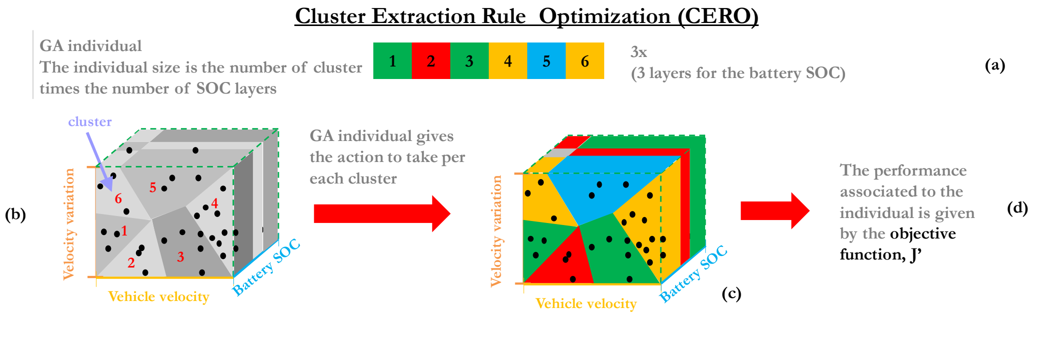

Fig. 1. CERO: illustration of the GA individual (a), the CE (b) and RO (c) algorithms and the performance estimation (d).

4.3. Cluster Optimization and Rule Extraction (CORE)

The CORE tool has to be trained using mission data and the DP results. The outcomes are the clustering of the domain and the set of rules. The offline training stage for the CORE tool is here presented. The tool receives the vehicle velocity, acceleration and the battery SOC. The two first signals come from the input mission data while the battery SOC from the optimal policy obtained with DP in a previous stage. The three values at a given time instant can be visualized as a point in the 3D space. The procedure is repeated for all of the time instants of several driving cycles. The idea is to group close points into a cluster and to select a unique action for that cluster, in the set of admissible actions of the control variables.

The CORE tool consists of two algorithms: one for the smart-clustering of optimal driving conditions (CO), the other for the rule extraction (RE). Let's present the RE algorithm first. Given the 3D space of the operating points and the cluster, the DP actions selected in the points that belong to that cluster are analyzed and the most frequent action becomes the unique action for that cluster, from a frequency distribution. If no points occur in a cluster, a backup rule is used: for instance, pure electric for high levels of SOC, battery charging for low SOC, pure thermal otherwise.

Now let's define the smart clustering of the 3D space, i.e., the CO algorithm. The space is generated by three segments. Each segment is divided into non-equispaced intervals and a mesh of clusters is obtained. The number of intervals per segment is given to the CORE tool as input. The RE algorithm is applied to obtain a rule for each cluster. The rule-based controller is then run over the given mission and the J function represents its performance.

The performance of the rule-based method is expected to be correlated to a great extent to the discretization of the 3D input variable domain. For example, if a different discretization is introduced, some points of the vehicle missions might no longer belong to the same cluster as that of the original discretization. For this reason, GA has been used to define the best size of the grid for each input variable (i.e., the optimal mesh size of the parallelepiped in Fig. 2), in order to minimize the J function value. GA aims at finding the best individual by controlling the evolution of a population of individuals over several generations. In this study the individual stores the grid size information to generate the mesh. As an example, the segments of the vehicle velocity and acceleration are divided into 4 intervals, while the SOC segment into 3. Each individual element represents the interval length. The performance of the rule-based controller is evaluated in terms of the J function. Since it very likely not to obtain a zero net balance of the battery SOC (see Eq. (1)), the battery energy variation over the mission is converted into an equivalent J' value that has to be added for a correct performance analysis.

After the training process is ended, i.e., the tool determines the optimal mesh for the given driving mission, the mesh of clusters and the set of rules are embedded into the real-time software of the CORE. It then receives the current vehicle velocity and acceleration, as well as the battery SOC. It determines the cluster by means of simple table lookups and extracts the associated action from the rule, so that the powertrain can be actuated.

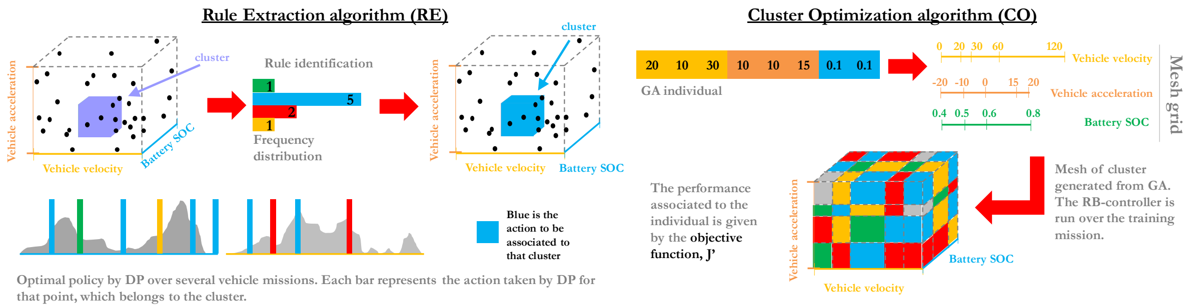

Fig. 2. CORE: illustration of the RE and CO algorithms.

5. Results and Discussion

5.1. Vehicle model validation

The models of the different components, i.e., the engine, the electric machine and the battery, were assessed by means of experimental tests carried out on a diesel mild hybrid powertrain equipped with a belt-alternator starter installed at the dynamic test bed of the ICEAL-PT (Internal Combustion Engine Advanced Laboratory, Dipartimento Energia, Politecnico di Torino). In particular, the validation of the model was carried out by comparing the predicted and experimental values of fuel consumption and NO_x emissions over NEDC under warm and cold start operations. It was found that the inaccuracy resulted to be within ±1 % for the fuel consumption and ±6 % for the cumulated NO_x emissions. For further details on the test rig specifications, the reader can refer to [16], and for further details on the assessment of the model, the reader can refer to [17]. All the data concerning the efficiency of the components have been normalized, for intellectual property reasons.

5.2. Rule-based controller training assessment

Figure 3 compares the performances of the two developed rule-based tools, namely CERO and CORE, in terms of CO_2 reduction (left), and the cost premium (right) over the set of driving cycles. The set consists of the Japanese 10-15 mode driving cycle (referred to as 1015), the Artemis Motorway Driving Cycle (AMDC), the New European Driving Cycle (NEDC) and the Worldwide Harmonized Light vehicles Test Procedures (WLTP). The numbers in the figures indicate the average CO_2 reduction (%) and cumulated CO_2 emissions (g/km) over the different missions (left), and the average cost premium (k$) (right).

The rule-based tools have been specifically trained over each mission. Each optimizer has been given a score according to the achieved J function value: the lower the J value, the higher the score. The average score over the set of driving missions is reported in the rightmost region in Fig. 3 (i.e., 61.6 for CERO, 61.9 for CORE, 62.1 for DP). Both of the developed rule-based tools perform as well as DP, in terms of fuel consumption saving and cost premium.

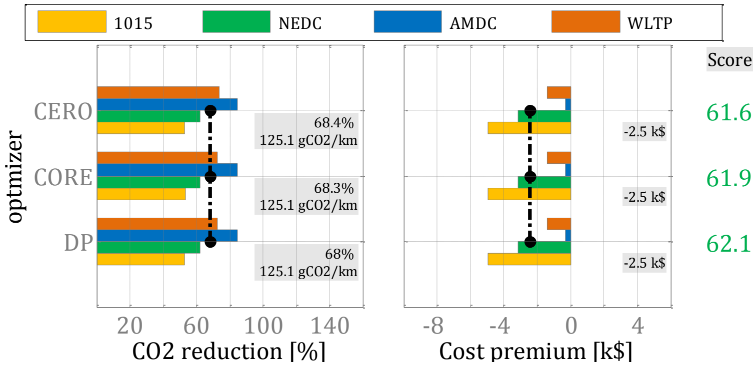

Fig. 3. CO_2 reduction (left) and cost premium (right) over different driving missions.

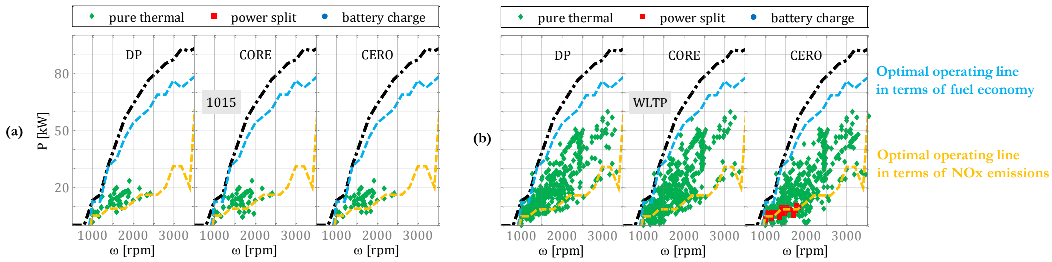

Fig. 4. Power flow distribution on the engine map over the 1015 (a) and WLTP (b) missions, obtained with DP (left), CORE (middle) and CERO (right). The optimal operating line of FC (dashed cyan) and of NO_x (dashed yellow) are also reported.

Figure 4 reports the distribution of the power flow of the hybrid vehicle on the engine map, for two different driving missions. The results obtained with DP are shown in the left column of each chart, while those obtained with CORE and CERO are reported in the middle and right columns, respectively. The rule-based tools have been trained independently over each mission. In general, the power split mode is used to move the engine operating points downward, in order to approach the optimal operating line, in terms of brake specific NO_x emissions. Since each case has also been optimized in terms of NO_x emissions, the operating points that result from the control policy optimization are predominantly distributed in the area that is not critical for NO_x emissions, and which is still efficient in terms of fuel economy. The point distributions reported in Fig. 4 indicate that the results of the proposed rule-based tools are very similar to those of the DP tool.

5.3. Rule-based controller testing

It is very likely that a rule-based controller trained over a given mission, i.e. m_2, and used over a different mission m_1, would lead to partially charge/discharge the battery with respect to the initial content. However, the battery SOC should be balanced over the mission to compare the controller performance to a benchmark control policy. The vehicle has to be run over a fraction of the 1015 mission at the end of each simulated mission, so that the controller charges or discharges the battery to obtain the initial battery SOC value (60%). Several simulations have been carried out over the 1015 mission with the DP tool, for different initial SOC values and 60% as final value.

The fuel consumption and NO_x emission trends have been obtained as a function of the initial SOC. The final SOC obtained by the rule-based controller over mission m_1 is used to interpolate these trends and the values of fuel consumption and NO_x emissions have to be added to the J function value.

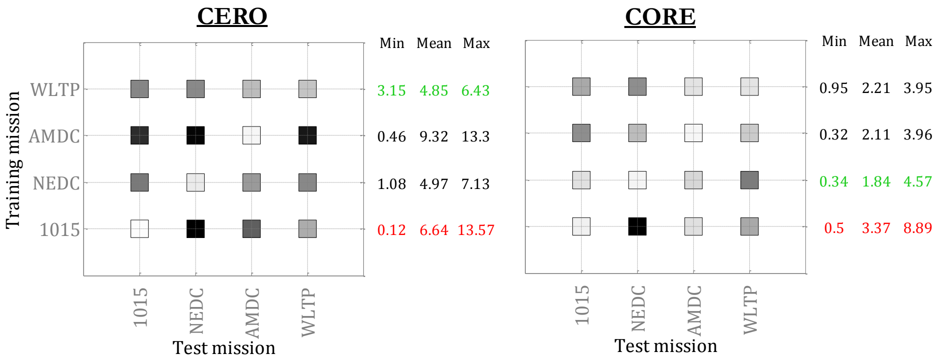

Fig. 5. CO_2 reduction (left) and cost premium (right) over different driving missions.

Figure 5 reports the results of this analysis. The performance of the two tools has been assessed with respect to that of DP for each mission, and a score discrepancy has been calculated.

Each tool was tested over a test mission m_1 (i.e., the driving cycles reported on the x-axis in Fig. 5) adopting the rule that was extracted from the optimization process that had been carried out over a training mission m_2 (i.e., the cycles reported on the y-axis). A square represents each combination of training-testing missions, and the discrepancy of the tool, with respect to DP, is reported with different shades of grey: the darker the shades, the higher the discrepancy. This graphical representation is useful to analyze whether the tool training might lead to bad performance over a specific test mission, analyzing the grey squares horizontally.

As an example, if the training of the CERO tool is carried out over 1015 (fourth row in Fig. 5), and the controller is employed over NEDC (second column), the discrepancy with respect to DP reaches the highest value (13.3%).

The training of the CERO tool over WLTP leads to the least average discrepancy (4.85%), while the CORE training over NEDC leads to the best average performance (1.84%).

The CORE tool can be selected as the best method, as it leads to the lowest discrepancy and the on-board controller has a low-computational demand. However, it comes with a cost, since its training is in general a bit longer than the CERO tool.

6. Conclusions

This present study has been focused on the development of a new machine-learning technique in order to develop rule-based controllers for the energy management of non-plug-in parallel hybrid electric vehicles. Two different methods have been developed (CORE and CERO). The performance of the proposed tools has been assessed over a set of different driving missions and compared with that of dynamic programming (DP). Moreover, the capability of the considered hybrid architectures to reduce fuel consumption and total costs, in comparison with a conventional vehicle, has also been investigated. It has been found that the performance of the CERO and CORE tools is extremely promising, especially for CORE. The average discrepancy of the two methods with respect to DP is in fact 4.85% and 1.84% if the training procedure is carried out over the WLTP and NEDC missions, respectively.

The main reason for the good performance of the proposed tools is that the rule extraction is carried out using machine-learning techniques, and is not based on empirical approaches driven by experience. The tool can easily be implemented in a vehicle control unit for a powerful fast real-time control strategy, as its application requires a very low computational effort.

References

[1] Baumann, B.M., Washington, G., Glenn, B.C., Rizzoni, G., "Mechatronic design and control of hybrid electric vehicles", Mechatronics, IEEE/ASME Transactions 5(1):58-72, 2000, doi:10.1109/3516.828590.

[2] Lee, H.D., Sul, S.K., Kamiya, M., Ikeda, H. et al., "Torque control strategy for a parallel-hybrid vehicle using fuzzy logic", Industry Applications Magazine IEEE 6(6): 33-38, 2000, doi:10.1109/2943.877838.

[3] Kheir, N.A., Salman, M.A. and Schouten, N. J., "Emissions and fuel economy trade-off for hybrid vehicles using fuzzy logic", Mathematics and computers in simulation 66(2):155-172, 2004, doi:10.1016/j.matcom.2003.11.007.

[4] Hajizadeh, A., Golkar, M.A., "Intelligent power management strategy of hybrid distributed generation system", International Journal of Electrical Power & Energy Systems 29(10):783-795, 2007, doi:10.1016/j.ijepes.2007.06.025.

[5] Bellman, R., Kalaba, R. E., "Dynamic programming and modern control theory", Academic Press New York, January 1966.

[6] Bellman, R.E., Lee, E., "History and development of dynamic programming", Control Systems Magazine, IEEE 4(4):24-28, 1984.

[7] Finesso, R., Spessa, E., & Venditti, M. (2016). An Unsupervised Machine-Learning Technique for the Definition of a Rule-Based Control Strategy in a Complex HEV. SAE International Journal of Alternative Powertrains, 5(2016-01-1243).

[8] MathWorks, MatLab, "The Language Of Technical Computing", http://www.mathworks.com/products/matlab/, [Online; accessed 30-May-2016], 2016.

[9] Finesso, R., Spessa, E., Venditti, M., "Layout design and energetic analysis of a complex diesel parallel hybrid electric vehicle", Applied Energy 134:573-588, 2014, doi:10.1016/j.apenergy.2014.08.007.

[10] Finesso, R., Spessa, E., & Venditti, M. (2016). Cost-optimized design of a dual-mode diesel parallel hybrid electric vehicle for several driving missions and market scenarios. Applied Energy, 177, 366-383.

[11] Graham, R. (2001). Comparing the benefits and impacts of hybrid electric vehicle options. Electric Power Research Institute (EPRI), Palo Alto, CA, Report, 1000349.

[12] Delucchi, M., Burke, A., Lipman, T., & Miller, M. (2000). Electric and gasoline vehicle lifecycle cost and energy-use model. Institute of Transportation Studies

[13] Moawad, A., Sharer, P., & Rousseau, A. (2013). Light-duty vehicle fuel consumption displacement potential up to 2045 (No. ANL/ESD/11-4). Argonne National Laboratory (ANL)

[14] Man, K.F., Tang, K.S., Kwong, S., "Genetic algorithms: concepts and applications", IEEE Transactions on Industrial Electronics 43(5):519-534, 1996, doi:10.1109/41.538609.

[15] Joines, J., Houck, C.R. et al., "On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GA's", Evolutionary Computation, IEEE World Congress on Computational Intelligence, 579-584, 1994, doi:10.1109/ICEC.1994.349995.

[16] Catania, A.E., Finesso, R., Spessa, E., "Predictive Zero-Dimensional Combustion Model for DI Diesel Engine Feed-Forward Control", Energy Conversion and Management. 52(10):3159-3175, 2011, doi:10.1016/j.enconman.2011.05.003

[17] Morra, E.P., Spessa, E. and Venditti, M., "Optimization of the operating strategy of a bas hybrid diesel powertrain on type-approval and real-world representative driving cycles", in: ASME 2012 internal combustion engine division spring technical conference, 557-568, 2012, doi: 10.1115/ICES2012-81093.