Proceedings of the ASME 2010 Dynamic Systems and Control Conference
DSCC2010
September 12-15, 2010, Cambridge, Massachusetts, USA

DSCC2010-4233

A RULE-BASED STRATEGY FOR A SERIES/PARALLEL HYBRID ELECTRIC VEHICLE: AN APPROACH BASED ON DYNAMIC PROGRAMMING

Domenico Bianchi*
Department of
Electrical and Information Engineering
Center of Excellence DEWS,
University of L'Aquila
L'Aquila, 67100 Italy
domenico.bianchi@univaq.it

Luciano Rolando*
Dipartimento di Energetica
Politecnico di Torino
Torino, 10129 Italy
luciano.rolando@polito.it

Lorenzo Serrao
Center for Automotive Research
The Ohio State University
Columbus, OH 43212 USA
serrao.4@osu.edu

Simona Onori
Center for Automotive Research
The Ohio State University
Columbus, OH 43212 USA
onori.1@osu.edu

Giorgio Rizzoni
Center for Automotive Research
Department of Mechanical Engineering
The Ohio State University
Columbus, OH 43212 USA
rizzoni.1@osu.edu

Nazar Al-Khayat
Tung-Ming Hsieh
Pengju Kang
Cummins Inc.
Columbus, IN 47201 USA
nazar.al-khayat@cummins.com
pengju.kang@ge.com
tung-ming.hsieh@cummins.com

ABSTRACT

Dynamic programming (DP) provides the optimal global solution to the energy management problem for hybrid electric vehicles (HEVs), but needs complete a-priori knowledge of the driving cycle and has high computational requirements. This article presents a possible methodology to extract rules from the dynamic programming solution to design an implementable rule-based strategy. The case study considered is a series/parallel HEV, in which a clutch allows to switch from one configuration to another. The strategy works according to a two layer policy: the supervisory controller, which decides the powertrain configuration (either series or parallel), and the energy management, which decides the power split. The process of deriving the rules from the optimal solution is described. Then, the performance of the resulting rule-based strategy is studied and compared with the solution given by the dynamic programming, which functions as a benchmark.

*This work has been developed during a period of 7 months as Visiting Scholar at Center for Automotive Research of the Ohio State University.

1 INTRODUCTION

Hybrid Electric Vehicles (HEVs) represent a powerful means to save fuel and reduce CO_2 emissions. Their performance strongly depends on the energy management strategy onboard of the vehicle. The HEV control problem involves the determination of the optimal power flow and, namely, the power split between the internal combustion engine and the electric motors. Finding the sequence of optimal power split at each time step to minimize the fuel consumption over a driving cycle is the aim of the energy management control for HEVs. It is known [1, 2] that the dynamic programming (DP) seeks for the global optimal solution once there is complete knowledge of driving cycle. However, the strategy is not implementable online, for the need of a-priori knowledge of the driving cycle and the elevated computational requirements. Therefore, other strategies are implemented, mostly based on empirical rules [3,4] or instantaneous minimization [5–7]. The strategy presented in this paper belongs to the first family: it is a rule-based (RB) strategy, thus easily implementable, and is derived from observation of the optimal solution obtained with dynamic programming, as initially proposed in [8]. The motivation for a rule-based strategy for this is to derive a quasi-optimal solution that requires the lowest possible computational requirements, even lower than methods based on instantaneous minimization. The RB approach, being based on a relatively simple set of rules, does not involve minimization or table look-up and therefore is very fast computationally. The paper focuses on a series-parallel hybrid architecture, in which a clutch allows to switch from one mode to the other, as described in Section 2. The presence of a discrete mode change makes the control problem more complex and introduces the need to distinguish between supervisory controller and energy management controller. These concepts are illustrated in Section 3, together with the formulation of the control problem. An optimal solution to the problem is found using Dynamic Programming, as illustrated in Section 5. However, since this algorithm requires complete knowledge of the driving cycle in advance and is extremely requiring in terms of computational loads, we illustrate in Section 6 a method to derive an implementable controller based on its results. Section 7 provides simulation results comparing the rule-based strategy to the optimal solution obtained by DP.

2 POWERTRAIN ARCHITECTURE

Hybrid electric vehicles (HEVs) are commonly classified on the basis of their powertrain architecture, i.e. the way in which the various powertrain components are arranged and the ratio of maximum electrical to mechanical (engine) power on board the vehicle. Traditionally, two main categories have been distinguished: *series* and *parallel* configurations. In series HEVs, at least two electric machines are present: a motor and a generator. The motor is the only mean of providing power to the wheels; it receives electric power from either the battery pack or from the generator, run by an internal combustion engine. Thus, the battery and engine power are summed electrically. In parallel HEVs, on the other hand, mechanical gearings allow the engine and one or more electric machines to drive the wheels; in other words, the power of the engine and the electric machine(s) is summed mechanically at the transmission level.

More sophisticated architectures includes higher degree of freedom and greater opportunity for fuel economy. The architecture considered in this article belongs to this category of powertrain architectures. The application of this powertrain is towards commercial trucks and composed of two electric machines (called in the following *motor* and a *generator*), a Diesel engine and a clutch.

The internal combustion engine is directly connected to the generator GEN while the motor is connected to the powertrain. When the clutch is open, the vehicle behaves as a series HEV, since the engine velocity is independent from the vehicle speed and only the electric motor is able to provide torque to the wheel. However, the presence of the clutch gives some additional alternatives. When the clutch is locked, the engine and both electric machines are connected to the powertrain, the vehicle behaves as in a parallel HEV, summing the torque of the three machines. In this configuration, the engine can drive the vehicle on its own or be assisted by the electric motor.

This particular kind of hybrid powertrain is usually defined *combined series-parallel*. The vehicle can operate in all-electric mode, series-hybrid mode or parallel-hybrid mode, depending upon which is most advantageous given the current operating requirements. The increased flexibility allows to merge the advantages of each architecture: the all-electric mode achieves zero-emission driving for a limited range; the series mode tends to be more efficient at low speed and in stop-and-go cycles, since the engine speed is independent from vehicle speed (and thus low-speed, low-efficiency operation can be avoided); the parallel mode is more efficient at higher load and speed, where it eliminates the double power conversion of the series mode.

With the combined series-parallel architecture, the HEV controller must decide on how to share the total power demand between the available machines and command the clutch to switch between various modes of operation.

3 CONTROL PROBLEM FORMULATION

The optimal control problem in a HEV consists in finding the minimum fuel consumption during vehicle operation, while respecting the design limitations of each component and the drivability/performance specifications. The aim is to minimize a cost function (integral cost) defined as an integral over a finite horizon. The finite horizon typically corresponds to a complete regulatory driving cycle or a short real-world trip. The optimization objective, considered in this work, is the fuel consumption during a trip and the constraints are:

- charge-sustainability: the battery SOC at the beginning and the end of the trip should be equal
- drivability constraints: at each instant, the total torque output of the powertrain should be equal to the driver's demand;
- actuator limitations: at each instant, the output of each machine in the powertrain (engine, motor, and generator) cannot exceed its maximum torque/power rating; similarly, the total battery power must remain within the acceptable limits in both charge and discharge operation.

In this vehicle architecture there are several control variables: the status of the clutch (open or locked) and the status of the engine (on or off) are discrete control variables that determine the *operating mode* of the powertrain; the torques of the individual machines are continuous variables and determine how the power request is shared between components. Formally, the clutch status is represented as:

$$C = \begin{cases} 0 & \text{clutch open} \\ 1 & \text{clutch locked} \end{cases} \tag{1}$$

and, similarly, the engine status is given by

$$E = \begin{cases} 0 & \text{engine off} \\ 1 & \text{engine on} \end{cases}. \tag{2}$$

The power split is defined by the values of the torques delivered by the two electric machines, T_{EM} (motor torque) and T_{GEN} (generator torque). The four input variables are gathered in the following control vector, defined for each time step k:

$$u_k = \left\{T_{mc1,k},\ T_{GEN,k},\ C_k,\ E_k\right\} \tag{3}$$

The hybrid controller is divided in two layers: the value of the variables C and E is determined at the *supervisory controller* level, while T_{EM} and T_{GEN} are determined at the *energy management* level, respecting the constraints on powertrain operation. The remaining degree of freedom of the powertrain, i.e. the transmission gear index g_{tr}, is chosen by the transmission controller, which is assumed to be external to the energy management and supervisory controller, and embedded in the transmission; therefore, the gear index is treated as an external input in this context. The vehicle velocity, the rotational speed of the three machines, and the driver's torque demand are also external inputs.

The problem is formally defined as finding the control law $u_k,\ k = 1,...,N$ that minimizes the cost

$$J = \sum_{k=0}^{N-1} m_f\left(u_k, k\right), \tag{4}$$

where $m_{f,k}$ is the fuel mass flow rate at time k, subject to the constraints:

$$0 \le P_{ICE,k} \le P_{ICE,max}\ \forall k = 0,1,...,N-1 \tag{5}$$

$$P_{EM,min} \le P_{EM,k} \le P_{EM,max}\ \forall k = 0,1,...,N-1 \tag{6}$$

$$P_{GEN,min} \le P_{GEN,k} \le P_{GEN,max}\ \forall k = 0,1,...,N-1 \tag{7}$$

$$P_{batt,min} \le P_{batt,k} \le P_{batt,max}\ \forall k = 0,1,...,N-1 \tag{8}$$

$$SOC_{min} \le SOC_k \le SOC_{max}\ \forall k = 0,1,...,N-1 \tag{9}$$

where P_{ICE} is the engine mechanical power, P_{EM} is the motor electrical power, P_{GEN} is the generator electrical power, P_{batt} is the battery power. The subscripts *max* and *min* refer to the maximum and minimum limits of each variable. An additional constraint is, of course, the dynamic equation of the state of charge, described in Section 4.

4 MODEL OF THE HYBRID ELECTRIC VEHICLE

Two different approaches to the HEV modeling can be adopted: *backward* or *forward* (with respect to the physical causality principles) [9]. In the forward approach the vehicle speed is a consequence of a torque delivered by the powertrain, in response to the demand generated by the driver model (usually a PID controller that compares the actual velocity with the desired value). In the backward approach, instead, no driver is necessary, since the vehicle speed is supposed known and the torque necessary to obtain it is computed by the model.

In this paper a backward, quasi static simulator is used to implement the dynamic programming algorithm, because it allows to treat the vehicle speed as an external input rather than a dynamic state. The vehicle speed, defined by the driving cycle, is used to calculate the vehicle loads; then, through the powertrain model, both fuel consumption and battery SOC are computed. The *vehicle loads* block contains all the parameters needed to compute the power demand. Starting from the driving cycle inputs, it is possible to calculate the tractive force at the wheels as:

$$F_{trac} = F_{inertia} + F_{roll} + F_{aero} + F_{grade}. \tag{10}$$

From F_{trac} and the wheel speed ω_{wheel}, with transmission efficiencies and ratios, it is possible to compute both the power and torque request upstream of the gearbox, as well as the speed of the gearbox input shaft.

Since the transmission ratio is assumed to be independent from the HEV powertrain controller and is considered as a parameter, the torque, speed and power at the gearbox input represent the request that the hybrid propulsion system must satisfy.

If the system is in the series configuration, both the engine and the generator are isolated from the gearbox since the clutch is open; therefore the engine torque can be written as:

$$T_{ICE} = -T_{GEN} + T_{aux,m} \tag{11}$$

where $T_{aux,m}$ is the torque requested by mechanically-driven auxiliary loads. The engine speed is an additional degree of freedom that can be chosen by the energy management strategy. In particular, it was decided to keep the engine working on its best efficiency line, in order to minimize the fuel consumption:

$$\omega_{ICE} = \omega_{ICE,opt}\left(T_{ICE}\right). \tag{12}$$

When the hybrid powertrain works in parallel, all the machines are connected to the gearbox and balance the resisting torque. In this case there are two degrees of freedom because the speed is fixed by the external conditions, while the controller must set two of the three torques (the third is then defined by difference with the total).

Using the electric machines torques as the control variable of the energy management strategy, the engine torque can be written as:

$$T_{ICE} = T_{gb} + T_{aux,m} - T_{EM} - T_{GEN}, \tag{13}$$

while the speed of all the machines is identical:

$$\omega_{gb} = \omega_{ice} = \omega_{EM} = \omega_{GEN}. \tag{14}$$

At this point, regardless of the powertrain configuration, the operating points of all the machines are available and can be used as inputs of the efficiency maps of each component. Quasi-static maps are used to compute the engine fuel consumption and the power demand of the electric machine, given torque and speed.

Another significant variable is the battery power P_{batt}, which can be easily computed as:

$$P_{batt} = P_{EM,e} + P_{GEN,e} + P_{aux,e} \tag{15}$$

where $P_{EM,e}$ and $P_{GEN,e}$ are the electric power of the motor and the generator and $P_{aux,e}$ the power of the electrical auxiliary loads.

The battery is normally very complex to represent. In this case, in order to obtain a reasonably simple simulator, no temperature dependency is considered, hysteresis and dynamics are neglected, and a simple circuit model is used to compute the state of charge variation as a function of the power at the terminals and of the circuit parameters.

The state of charge variation represents the state equation of the energy management problem, and can be written as

$$SOC(k+1) = SOC(k) - \alpha \cdot \frac{I(k)}{Q_{max}} \tag{16}$$

where $I(t)$ is the current flowing through the battery (positive during discharge), Q_{max} is the battery charge capacity, and α is a correction factor that accounts for the charge losses (coulombic efficiency). In order to make this relation implementable in the framework of the model described here, it is necessary to express the current in terms of the battery power; using a simple circuit model including the open circuit voltage and the internal resistance, the following relation is found:

$$I = \frac{(1-b) \cdot V_{oc} + \sqrt{[(1-b) \cdot V_{oc}]^2 - 4 \cdot P_{batt} \cdot R_{eq}}}{2 \cdot R_{eq}} \tag{17}$$

where $V_{oc}(SOC)$ is the open circuit voltage, R_{eq} the equivalent internal resistance, and $b(V_{oc})$ is a coefficient depending on the open circuit voltage [10]. The current I is positive when discharging and negative for charging.

The battery power in (17) can be expressed in terms of the torque/speed of the electric machines using (15).

These are the main physical relations used in the vehicle model and they were implemented using a Matlab function.

5 DYNAMIC PROGRAMMING

Dynamic programming (DP) generates a numerical solution to the optimal control problem defined in Section 3. It gives sufficient conditions for the global optimality (see [11] or [2]).

To implement the DP algorithm on described hybrid architecture, an open-source Matlab code developed at ETH-Zurich [12] was exploited. This function solves discrete-time optimal control problems using Bellman's dynamic programming algorithm. The model equation can include several state variables and input variables. Furthermore, the equations can be time variant (like in this case) and include time-variant state and input constraints. The user has to provide the number of controls that he needs to optimize, the states of the system that have to be monitored and the limits for each of them. Then the code preprocesses this information in order to arrange all the possible input values into multi-dimensional matrices. Then, the controls are applied, step by step, to a vehicle model that generates a grid of possible SOC values, each corresponding to a certain pattern of control inputs, as shown in Figure 1. Thus, given driving cycle, the DP provides the optimal combination of control inputs.

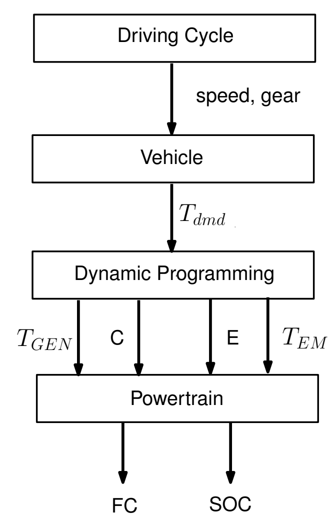

Figure 1. Flowchart of the Dynamic Programming Controller.

The DP solution is computed for several driving cycles, representative of the range of operating conditions for the vehicle considered.

6 RULE-BASED STRATEGY

The control based on a set of rules is computationally efficient for an embedded CPU, but it is based on empirical laws that, usually, have results quite far from the optimality. Its calibration, in addition, could be quite difficult. The dynamic programming, on the contrary, provides the optimal solution on each driving cycle. Therefore, analyzing its control actions, some rules can extracted that try to reproduce the optimal behavior, and, unlike DP control signals, are implementable. This approach is known (see, e.g., [8]), but it is now applied to a complex architecture in which the dynamic programming is used to determine not only the hybrid power split, but also the vehicle operating mode.

The starting point for deriving a rule-based strategy (RB) from DP is an extensive set of simulation in which the optimal driving strategy is found for several driving cycles, covering an ample range of urban and suburban driving conditions. The results are then studied and analyzed in order to find common patterns in the algorithm decisions, that are then replicated by appropriate rules.

The control results are represented in order to emphasize any dependency from significant input variables, such as gearbox power P_{gb}, gearbox speed ω_{gb}, and battery state of charge *SOC*.

As mentioned in Section 3, the powertrain controller can be divided in two parts: the *supervisory control* which decides the best operating mode and the *energy management* which shares the torque among the machines in order to satisfy the overall demand. Therefore the analysis of the DP results has to be performed at two levels: mode selection (engine and clutch status), and torque split. The extraction of rules for each of these two levels is described the following sections.

6.1 Supervisory Control

To understand the behavior of the supervisory control, the operating mode chosen by DP over all the analyzed driving cycles was plotted as function of the gearbox input power and speed, as shown in Fig. 2. Three main areas can be identified:

A. at low speed and torque, the powertrain works either in series or in pure electric (EV) mode - i.e., the clutch is open ($C = 0$) and the engine is either on or off ($E = 1$ or $E = 0$);
B. this area is limited by engine idle speed and positive gearbox torque: here only the parallel configuration is present. Actually there are also a few points of series operation, but they are only used to limit engine speed to its maximum.
C. the third area includes all the points with a negative torque: here the supervisory always switches off the engine in order to save fuel since the vehicle is decelerating. Actually in this situation the engine can also be on, but fuel injection is cut off, which in the simplified model is treated as engine-off.

Fig. 2, nevertheless, is not able to show any dependence on the state of charge. In particular, it was not possible to find any clear correlation between the state of charge and the mode selection in region A. Therefore, a simple threshold was set to distinguish between EV and series mode at lower speed.

The supervisory control rules are therefore implemented as follows:

A. When the gearbox input torque is positive ($T_{gb} \geq 0$) and the gearbox input speed is below engine idle speed ($\omega_{gb} \leq \omega_{idle}$), the clutch is open ($C = 0$); the status of the engine is determined by the SOC values. The vehicle is either in EV or series mode. The speed condition $\omega_{gb} \leq \omega_{idle}$ is not a result of the optimization, but a physical constraint of the powertrain: the engine cannot be connected to the driveline unless the speed is above idle.
B. When $T_{gb} \geq 0$ and $\omega_{gb} > \omega_{idle}$, the clutch is locked ($C = 1$) and the engine is on ($E = 1$), i.e. the vehicle is in parallel mode.
C. When $T_{gb} < 0$, i.e. during regeneration, the clutch remains engaged ($C = 1$), and the engine maintains its previous state, but fuel is cut off.

6.2 Energy Management

Depending on the powertrain mode decided by the supervisory controller, the power split among the three machines is determined in different ways.

6.2.1 Parallel Mode All the machines can directly act on the gearbox input shaft to overcome the resistance torque given by the vehicle. The energy management decides what fraction of the torque is generated by the electric machines and by the engine. The data analysis shows a linear relation between the gearbox input torque T_{gb} and the sum of the electric machine torque T_{elec}. Then, from a simple torque balance, it is possible to compute the fraction given by the engine. In order to split T_{elec} between the two machines, each of the torques computed by DP is related to T_{elec}. As a results, two linear correlations can be observed again. An example of these laws defining the parallel torque split is represented in Fig. 3. In the controller implementation only one of these two will be used, while the third torque will be obtained by difference from the others.

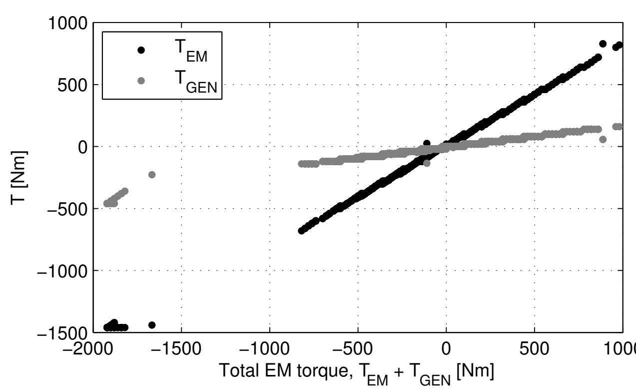

Figure 3. Dependence of the electric machine torque from the total electric request.

This fact allows to match the request in all cases (which would not be possible if all three torques were computed independently). The generator (*GEN*) is chosen as the computed variable because preliminary tests showed better performance in

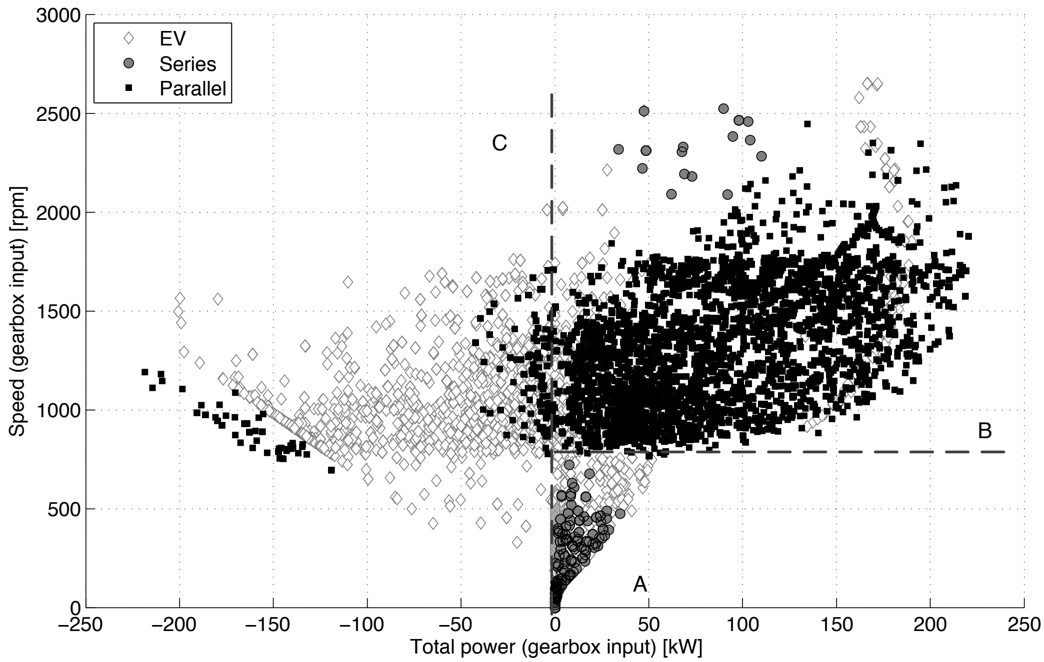

Figure 2. Hybrid Mode as a function of gearbox input torque and speed.

comparison with the dual case in which T_{EM} was the one computed by regression.

6.2.2 Series Mode

In this configuration, an approach based on the torque balance it is not suitable because the engine and the generator are disconnected from the gearbox, thus the entire torque request must be satidsfied by the motor. On the other hand, the electric power demand should be split between the generator and the battery. The total electric power is the sum of the motor electric power and the power needed by the electrical accessories. Following an approach similar to the torque split, the total power request can be correlated with the battery power computed by DP, as shown in Fig. 4. The power fraction that is not supplied by the battery is provided by the motor-generator group. The engine torque T_{ICE} is computed imposing that the engine operates at the speed of maximum efficiency given the power output.

6.3 State of charge control

The state of charge of the battery has to be considered not only in the supervisory decisions but also in the energy management. The torque split should change depending on the energy stored in the battery, but the effect of SOC is not present in all the empirical rules derived from DP and presented in the previous sections. Therefore, these laws needs to be modified to achieve charge-sustainability. One simple way to proceed is to shift up or down the linear laws that compute the electrical loads both in parallel (as described in section 6.2.1) and in series (as described in section 6.2.2). To reach this target, a correction function is introduced in the linear correlations, using an additional coefficient $p(SOC)$ that multiplies the intercept of the regression lines.

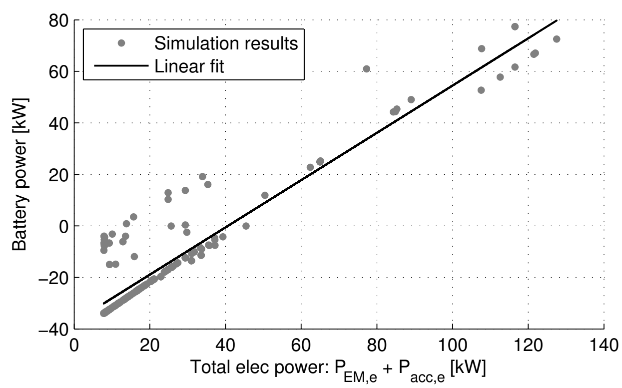

Figure 4. Dependence of the battery power from the total electric power.

It is now necessary to choose the shape of this correction function $p(SOC)$. The correction has to be minor for small deviation from the reference state of charge SOC_{ref}, and increase smoothly when the correction needs to be stronger. A cubic polynomial is a suitable function for this purpose; the correction function is thus defined as:

$$p(SOC) = -\mu \cdot x_{SOC}^3 + 1 \tag{18}$$

where x_{SOC} measures the distance of the state of charge from the reference value:

$$x_{SOC} = \frac{SOC - SOC_{ref}}{(SOC_{max} + SOC_{min})/2}. \quad (19)$$

The parameter μ in (18) defines the amount of correction for the achievement of the charge sustaining condition; a higher value of μ makes the solution more robust, penalizing more the variation of state of charge, but also introduces some deviation from the optimal solution.

It was observed that μ depends on the driving conditions. The lowest value of μ for which the strategy achieves the charge balance is different for each cycle, and guaranteeing robustness under all conditions requires a compromise in terms of performance. This is one of the main drawbacks of a rule based approach: this type of energy management requires a careful calibration of all its parameters, which have to be defined by a compromise among different driving cycles.

7 SIMULATION RESULTS

The rules described in Section 6 were implemented on the simplified model described in Section 4. In this section, the resulting rule-based strategy is compared to the dynamic programming.

The Manhattan bus cycle, representative of real-world driving condition for an urban bus, is here considered to illustrate the performance of RB vs DP. The parameter μ is tuned to the optimum value as described in section 6.3. Figure 5 shows both the overall SOC profile obtained by both strategies and the choices of the supervisory controllers. The SOC profiles are close to each other in terms of shape, which means the power split is similar (the offset is due mainly to punctual differences), and the choices of the supervisory controller (i.e. the vehicle mode in the bottom plot) are also very similar, as expected.

Figure 6 shows the comparison of the torque split. The differences in the electric motor torque (T_{EM}) are quite small, while the generator behavior shows that the rule-based tends to generate more energy to be stored in the battery. Figure 6 also shows that the torque of the electric motor obtained with the RB strategy is smoother in comparison with the DP, which is more "nervous". These discrepancies in the torque may also determine differences in state of charge profile and in fuel consumption. As Fig. 5 demonstrates, the SOC is quite similar but the rule-based is characterized by an higher mean value which leads to a higher final SOC.

The results obtained on several driving cycles are in Table 1, which also includes validation cycles, i.e. cycles not used for the rule extraction. The fuel consumption results listed in the table include the correction for the SOC imbalance present in the RB results.

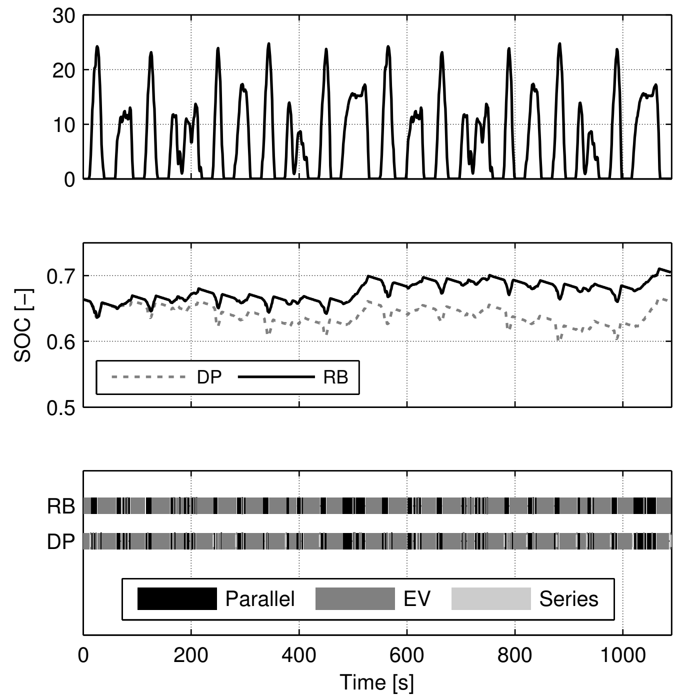

Figure 5. Comparison RB-DP: state of charge profile and mode selection on the Manhattan cycle

Table 1. Comparison of RB and DP over various driving cycles

	Fuel consumption		ΔSOC [%]	
	DP	RB	DP	RB
Manhattan*	100	108	0	6
WVU-suburban*	100	102	0	1
WVU-inter*	100	102	0	3
APTA*	100	100	0	-8.3
UDDS	100	106	0	-3
HTUF	100	110	0	-6

*: cycles used to extract the rules

8 CONCLUSION

The dynamic programming algorithm provides the optimal solution to the HEV energy management problem, and serves as a benchmark to assess the minimum fuel economy achievable along a driving mission. Both the need for a-priori knowledge of the mission profile and the high computational requirements make this strategy unrealistic to implement, since an on-board real time controller has to operate with limited computational and memory resources.

A rule-based strategy, on the other hand, is suitable for online implementation, due to the simple set of *if-then-else* rules. A demanding calibration phase is required though, for making the strategy charge-sustaining with respect to a wide variety of driving cycles.

In fact, rule-based parameters can be strongly affected by the driving conditions. The approach proposed in this paper is to study the results given by the dynamic programming in or-

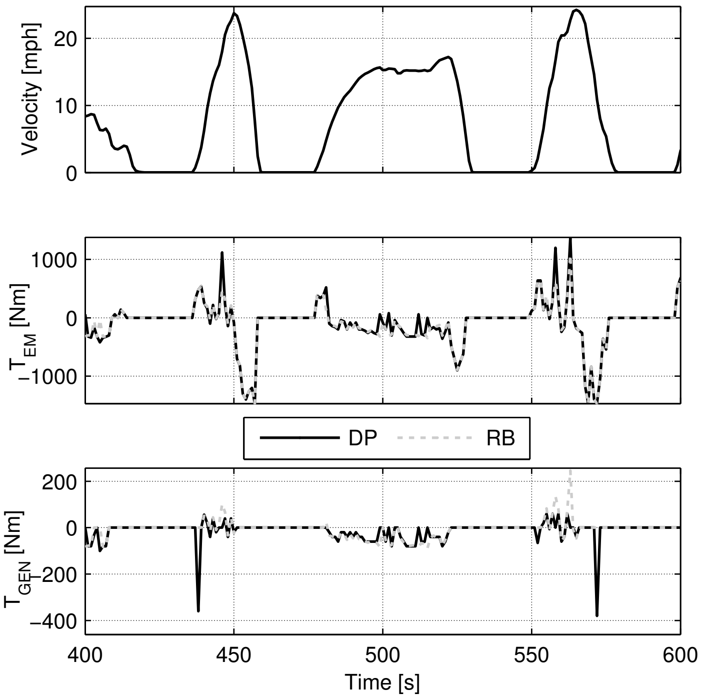

Figure 6. Comparison RB-DP: torque split on the Manhattan cycle (detail). Top: vehicle speed; middle: T_{EM}; bottom: T_{GEN}

der to find some common pattern in its decisions, and extract rules that can be implemented in an "sub-optimal" rule-based controller. This article presents a possible methodology to handle this problem. The DP analysis provided some rules which are able to quickly minimize the fuel consumption. A great number of simulations has been performed to test the robustness of the RB algorithm. This phase is very time consuming while the rule extraction is relatively straightforward. The obtained controller has only one calibration parameter that is tuned in order to satisfy the charge balance. Both fuel consumption and state of charge profile are very close to the optimal, as demonstrated in Section 7.

The study perfomed has shown that the RB controller is dependent on both the powertrain components and vehicle architecture. If these change, the controller needs to be redesigned. A future improvement to the work presented could be looking at the different decisions taken by DP as powertrain components change so as to make the rule-based strategy less dependent on those parameters.

REFERENCES

[1] Brahma, A., Guezennec, Y., and Rizzoni, G., 2000. "Optimal energy management in series hybrid electric vehicles". *Proceedings of the 2000 American Control Conference,* **1**(6), pp. 60–64.

[2] Sundström, O., Ambühl, D., and Guzzella, L., 2009. "On implementation of dynamic programming for optimal control problems with final state constraints". *Oil and Gas Science and Technology - Rev. IFP*, September.

[3] He, X., Parten, M., and Maxwell, T., 2005. "Energy management strategies for a hybrid electric vehicle". *Proceedings of the 2005 IEEE Vehicle Power and Propulsion Conference*, pp. 536–540.

[4] Hofman, T., Steinbuch, M., van Druten, R., and Serrarens, A., 2007. "Rule-based energy management strategies for hybrid vehicles". *Int. J. Electric and Hybrid Vehicles,* **1**(1), pp. 71–94.

[5] Paganelli, G., Ercole, G., Brahma, A., Guezennec, Y., and Rizzoni, G., 2001. "General supervisory control policy for the energy optimization of charge-sustaining hybrid electric vehicles". *JSAE Review,* **22**(4), pp. 511–518.

[6] Musardo, C., Rizzoni, G., Guezennec, Y., and Staccia, B., 2005. "A-ECMS: An adaptive algorithm for hybrid electric vehicle energy management". *European Journal of Control,* **11**(4-5), pp. 509–524.

[7] Serrao, L., Onori, S., and Rizzoni, G., 2009. "ECMS as a Realization of Pontryagin's Minimum Principle for HEV Control". *Proceedings of the 2009 American Control Conference*.

[8] Lin, C., Peng, H., Grizzle, J., and Kang, J., 2003. "Power management strategy for a parallel hybrid electric truck". *IEEE Transactions on Control Systems Technology,* **11**(6), pp. 839–849.

[9] Pisu, P., Hubert, C. J., Dembski, N., Rizzoni, G., Josephson, J. R., Russell, J., and Carroll, M., 2005. "Modeling and design of heavy duty hybrid electric vehicles". *Proceedings of the 2005 ASME International Mechanical Engineering Congress and Exposition*.

[10] Szumanowski, A., 2000. *Fundamentals of Hybrid Vehicle Drives*. Radom, Warsaw.

[11] Serrao, L., 2009. "A comparative analysis of energy management strategies for hybrid electric vehicles". PhD thesis, The Ohio State University.

[12] Sundström, O., and Guzzella, L., 2009. "A generic dynamic programming matlab function". *Proceedings of the 18th IEEE International Conference on Control Applications*, pp. 1625–1630.