HAL
archives-ouvertes.fr

Constrained Optimization of Energy Management for a Mild-Hybrid Vehicle

G Rousseau, D. Sinoquet, P. Rouchon

Constrained Optimization of Energy Management for a Mild-Hybrid Vehicle

G. Rousseau[12*], D. Sinoquet[1] and P. Rouchon[2]

1 Institut français du pétrole, 1 et 4, avenue de Bois-Préau, 92852 Rueil-Malmaison Cedex - France
2 Ecole des Mines de Paris
e-mail: gregory.rousseau@ifp.fr
* Corresponding author

Abstract — *Hybrid-electric vehicles appear to be one of the most promising technologies for reducing fuel consumption and pollutant emissions. The presented work focuses on a power train architecture for mild hybrid vehicles where the kinematic energy in the breaking phases is stored in a battery to be re-used later via the electric motor. This additional traction power allows to downsize the engine and still fulfill the power requirements. Moreover, the engine can be turned off in idle phases. The complete mild-hybrid vehicle is modelled in AMESim environment and the fuel consumption for given driving cycles is estimated. The control strategies for the energy management between the two power sources are optimized with respect to fuel consumption with a classical dynamic programming (DP) method. We propose an other method based on Pontryagin Minimum Principle which furnishes results very close to the DP results for a significantly reduced calculation time. These optimization results furnish the optimal control laws from which could be derived the control laws to be implemented on the vehicle. To illustrate the potential of optimization for component design, mild hybrid vehicles with varying battery and electric motor sizes, with different types of engine (gasoline / natural gas), are evaluated in terms of consumption gain with the presented methodology.*
Keywords: Hybrid vehicle, Modelling, Optimal control, Optimization

INTRODUCTION

Growing environmental concerns coupled with concerns about global crude oil supplies stimulate research on new vehicle technologies. Hybrid-electric vehicles appear to be one of the most promising technologies for reducing fuel consumption and pollutant emissions [5] :

- Fuel economy. It is well known that the kinetic energy gained by the classical vehicle after an acceleration can not be recuperated when braking. One of the advantage of the hybrid vehicle is the ability to recuperate a part (depending of efficiencies) of this energy, to store it in a battery, and to re-use it latter. This electric energy can therefore be used by an electric motor to power the vehicle and to save fuel.
- Reduction of pollutant emissions. As using an electric motor to supply the traction power, it becomes possible to distribute the power in order to reduce the pollutant emissions. The engine can also be downsized, and still fulfill the maximum power requirements of the vehicle, which can decrease again the pollutant emissions. Last but not least, hybrid vehicles allow to eliminate the idle fuel consumption by turning off the engine when no power is required (system called "stop'n go").

The control of hybrid power trains is more complex than control of classic engine. Indeed, the control laws have to deal with the state of charge of the battery, which provides

the level of energy remaining, and with the variable efficiency of each element of the power train. Optimization of energy management strategies on given driving cycles is often used to derive sub-optimal control laws to be implemented on the vehicle (see among others [1], [11], [13], [3]).

IFP, in partnership with Gaz de France and the Ademe, has combined its downsizing technology with a natural gas engine in a small urban demonstrator vehicle (VEHGAN vehicle), equipped with a starter alternator and super capacity manufactured by Valeo ([12]). In this paper, we focus on this hybrid architecture which is modelled in AMESim environment. Two different optimization algorithms are presented and applied on this model:

– a Dynamic Programming algorithm, classically used to handle this kind of optimal control problem ([13], [11], [1]),

– an algorithm based on Pontryagin Minimum Principle.

In a third section, we will show some results on the architecture of the VEHGAN vehicle with different kinds of engine, different sizes of battery and of electric motor.

1 SYSTEM MODELLING AND OPTIMAL CONTROL PROBLEM

1.1 AMESim Model

AMESim environment combined with IFP libraries Drive, Engine and Exhaust allows to model anything from a whole vehicle to the fine behavior of the engine. Figure 1 presents the scheme of our AMESim model of the VEHGAN : a mild hybrid vehicle with a natural gas engine and a starter alternator ([12]).
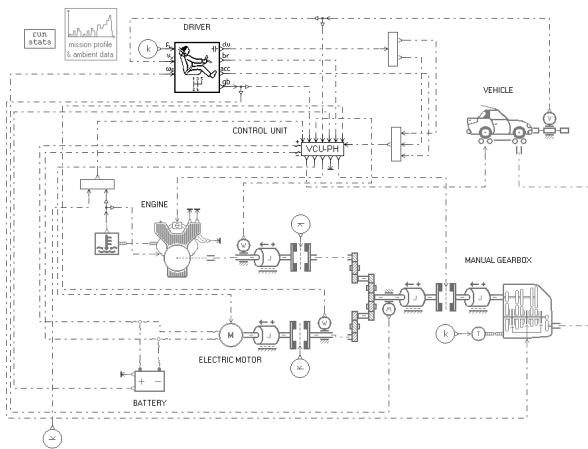

Figure 1

AMESim Model of the VEHGAN vehicle

This AMESim model consists in a model of :

- the vehicle (mass, wheel inertia, resistance force)
- a driver model (which can anticipate the driving cycle)
- a driving cycle to follow (imposing vehicle speed and gear shifts)
- a manual gearbox coupled with a clutch
- a $660cm^3$ Natural gas engine, with fuel consumption map displaying in Figure 2
- a $3kW$ starter alternator
- a battery (voltage source and an internal resistance) of $0.4Ah$
- and the control unit which manages the energy repartition between the engine and the electric motor. Control laws obtained by solving the optimal control problem presented in section (1.2) can be tested thanks to this control unit.
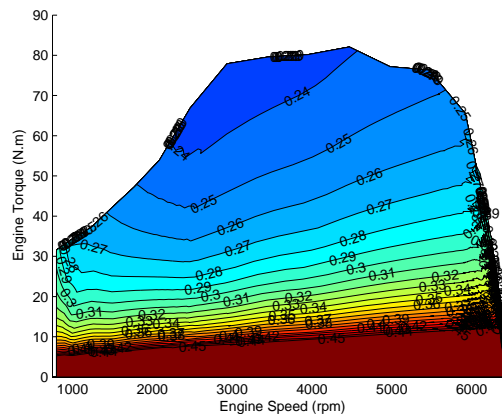

Figure 2

Fuel consumption map of natural gas engine of VEHGAN vehicle

Figure 3 and 4 display results of a simulation of the natural gas hybrid vehicle on the NEDC cycle. In this example, electric motor is used only for small torque requests; we can also see the regenerative breaking, when the battery is recharged.

1.2 Optimal Control Problem

The optimal control problem under study consists in minimizing the fuel consumption of the vehicle along a given vehicle cycle, taking into account physical constraints from battery, engine and electric motor. The general optimization problem is the following :
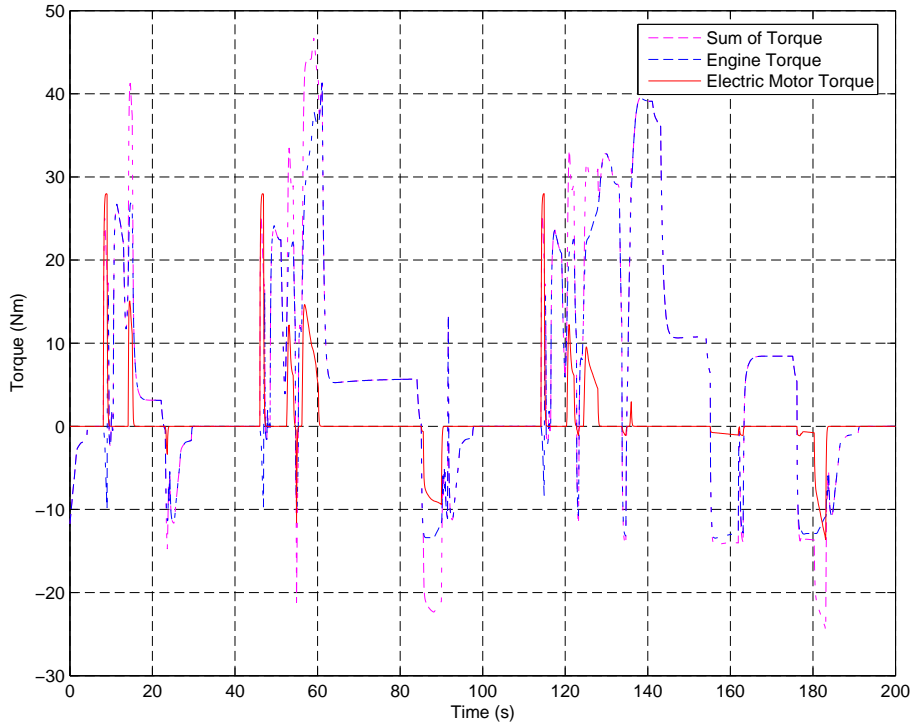
Figure 3

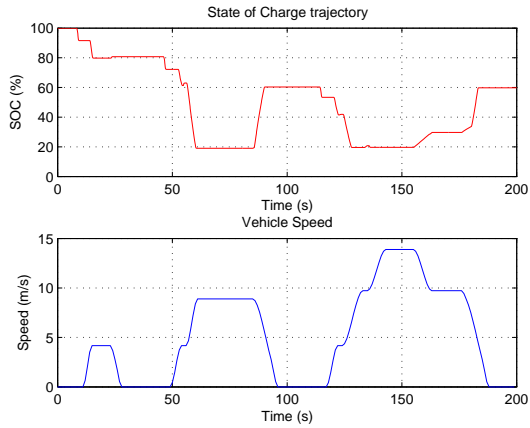Engine and electric motor torque simulated with AMESim model of VEHGAN vehicle of Figure 1

Figure 4

State of charge of the battery simulated with AMESim model of VEHGAN (*Top*); First 200s of the NEDC cycle (*Bottom*)

$$\begin{cases} \min\left\{ J(u) = \int_0^T L(x(t), u(t), t)dt + g(x(T), T) \right\} \\ \text{subject to :} \\ \dot{x} = f(x(t), u(t), t), \qquad x(0) = x_0 \\ x_{min} \leqslant x(t) \leqslant x_{max} \\ u_{min}(t) \leqslant u(t) \leqslant u_{max}(t) \end{cases} \tag{1}$$

with

- 0 and T : respectively the initial and the final time steps of the given driving cycle,
- $u(t)$: Control, i.e. repartition of the energy between the engine and the motor,
- $x(t)$: State of Charge (SOC) of the battery,
- $L(x(t), u(t), t)$: Instantaneous fuel consumption,
- $f(t)$: Function controlling the variations of state of charge of the battery,
- $g(x(T), T)$: penalization term handling a constraint on the final state of charge.

Let introduce other useful notations :

- the requested torque (associated with the chosen cycle) is written as

$$T_{rq}(t) = T_e(t) + T_m(t) + T_{bk}(t), \tag{2}$$

- the engine torque is

$$T_e(t) = u(t)T_{rq}(t), \tag{3}$$

- the electric motor torque is

$$T_m(t) = (1 - u(t))T_{rq}(t) \tag{4}$$

During braking, $T_{rq}(t) < 0$, the braking torque completes the electric motor torque to slow down the vehicle.

$$T_{bk}(t) = T_{rq}(t) - T_m^{min}(t)^{(1)} \tag{5}$$

_____

(1) We neglect any friction torque of the engine

with $T_m^{min}(t)$ is the minimal electric motor torque depending on the engine speed, thus depending on time.

A simple model is implemented for the battery, the state of charge of the battery evolves according to

$$\dot{x}(t) = -\frac{\omega(t)T_m(t)K'}{U_{batt}(t)n_{capa}} \qquad (6)$$

with
- $\omega(t)$: Electric motor and engine speed,
- $U_{batt}(t)$: Battery voltage,
- K' : Scaling constants,
- n_{capa} : Nominal capacity of the battery.

In the following, we assume that the battery voltage is constant along the cycle. We set $K = \frac{K'}{U_{batt}.ncapa}$ and by replacing $T_m(t)$ by its expression, we obtain

$$\dot{x}(t) = -K\omega(t)(1 - u(t))T_{rq}(t). \qquad (7)$$

In this optimal control problem, we make several hypothesis
- the pollutant emissions are not taken into account in the optimization process,
- the engine speed ω_e and the electric motor speed ω_m are equal, $\omega_e = \omega_m = \omega$,
- as the optimization problem is applied on a mild hybrid vehicle, the engine can not be stopped when the torque is provided only by the electric motor, except for the *stop'n go* mode at the idle speed. So, for a command $u(t)$ that cancels the engine torque and for positive torque request, the fuel consumption does not vanish (Figure 2),
- recharging the battery is only possible for negative torques (breaking request), we did not consider recharging by an additional engine torque beyond the driver request torque. Thus the command $u(t)$ remains between 0 and 1,
- we neglect any friction torque of the electric motor.

As in the general optimal control problem (1), the command variable $u(t)$ but also the state variable $x(t)$ are submitted to several constraints :
- the engine can only produce a positive torque, and is limited to a maximum torque which depends on engine speed $\omega(t)$:

$$0 \leqslant T_e(t) \leqslant T_e^{max}(\omega(t)), \qquad (8)$$

- the electric motor has a maximum torque, and also a minimum torque during regeneration breaking :

$$T_m^{min}(\omega(t)) \leqslant T_m(t) \leqslant T_m^{max}(\omega(t)), \qquad (9)$$

- the storage capacity implies a minimum and a maximum state of charge of the battery (which are fixed to 0% and 100% in our example)

$$x_{min} \leqslant x(t) \leqslant x_{max}, \qquad (10)$$

- the final state of charge is constrained to be equal to the initial state of charge to maintain a null electrical energy balance (to avoid to discharge totally the battery for minimizing the consumption)

$$x(0) = x(T). \qquad (11)$$

Constraints on command $u(t)$ are directly derived from (8), (9) and (10)

$$u(t)T_{rq}(t) \leqslant T_e^{max}(t), \qquad (12)$$

$$T_m^{min}(t) \leqslant (1 - u(t))T_{rq}(t) \leqslant T_m^{max}(t). \qquad (13)$$

In the following, we will write $U(t)$ in continuous time (respectively U_k in discrete time) for the feasible domain for $u(t)$ respecting the constraints (12) and (13).

2   DYNAMIC PROGRAMMING OPTIMIZATION

Classically, the Dynamic Programing method (DP) is used to solve the problem (1) ([13], [11]) : it relies on a very simple idea, the principle of optimality, stated by Richard Bellman as follows :

An optimal policy has the property that, whatever the initial state and initial decision are, the remaining decision must constitute an optimal policy with regard to the outcome resulting from the first decision.

The principle of optimality (or Bellman principle) suggests that an optimal policy can be built step by step sequentially, first by building an optimal policy for the "tail subproblem" involving the last step, then by extending the optimal policy to the "tail subproblem" involving the last two stages, and by iterating until an optimal policy is built for the complete time interval. DP algorithm applied to our problem furnished an optimal trajectory of the state of charge of the supercapacitor. First, the optimal control problem (1) is discretized in time :

$$\min_{u \in U} J(u) := \sum_{k=0}^{N-1} L_k(x_k, u_k), \qquad (14)$$

where $L_k(x_k, u_k)$ is the cumulated fuel consumption over the time interval $[k, k+1]$, x_k is the state of charge of the super capacity at time k, and N is the final time of the driving cycle.

The constraint on the final SOC (11) is introduced by a penalizing term $g(x_N) = \beta.(x_N - x_0)^2$ (β, a constant to be chosen [2]) in (14)

$$\min_{u \in U} J(u) := \sum_{k=0}^{N-1} L_k(x_k, u_k) + g(x_N). \qquad (15)$$

----

[2] In the following results, the value of 10 has been implemented

From Bellman principle, the minimum cost $V_k(x_k)$ at the time step k, $0 \leqslant k \leqslant N - 1$, is then expressed as follows :

$$V_k(x_k) = \min_{u_k \in U}(L_k(x_k, u_k) + V_{k+1}(f_k(x_k, u_k))) \qquad (16)$$

f_k being the function that modelizes the battery state of charge dynamic in the discrete form of (7). At time N, the cost function is

$$V_N(x_N) = g(x_N). \qquad (17)$$

This optimization problem is solved backward from final time step to initial time step using a discretization of function V in the command space and in the state space. It allows a fast optimization : $(N-1)$ constrained optimization problems (16) of one parameter, u_k, to be solved with a fine time discretization of the controller. It furnishes the optimal power repartition u_k at each time step regarding fuel consumption with constraints on the battery state of charge. (we refer to [6] for some theoretical results on the convergence of the method and error estimations).

A direct constrained optimization of (15) is an alternative method but leads to a large non linear optimization problem with a large number of inequality constraints. This is the reason why the DP algorithm is often preferred when the number of state variables is small (one or two state variables).

2.1 DP Optimization algorithm

As to get $V_k(x_k)$ for different values of x_k, a 2-D grid (time and SOC nodes) is set. A standard time step used in our examples is $1s$, and the SOC step is 0.5%. A classical DP algorithm consists in computing each feasible command (to go from a point x_k^i to an other point x_{k+1}^j), finally taking the best trajectory [3]. In such a method, the state of charge trajectory remains on the points of the defined grid in the state space. We propose, in our algorithm, to interpolate the function $V(x_k, k)$ in the state space, for each time step k. Therefore, it is possible to use a (state) continuous constrained optimization algorithm to solve each problem (16). The optimization algorithm is only used when $T_{rq} > 0$: when the request torque is negative or null, the optimal command u_k is completely known, as the battery is recharged as much as possible, the command u_k being constrained by the minimal electric motor torque from (13) and by maximum SOC from (10).

2.2 Results obtained with DP method

DP algorithm described in previous section is applied on the VEHGAN vehicle problem : the optimal SOC trajectory, depending of the fuel consumption map and of the vehicle cycle is displayed for NEDC cycle in Figure 5.

_____

(3) the trajectory which minimizes at each step k the sum $(L_k(x_k, u_k) + V_{k+1}(f_k(x_k, u_k)))$
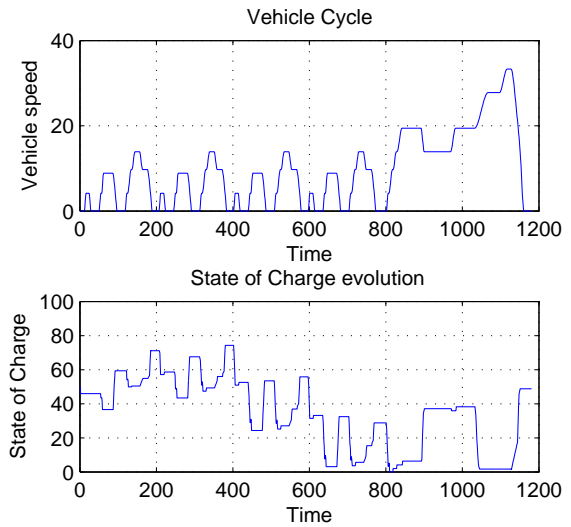

Figure 5

NEDC cycle(*Top*); Optimal state of charge trajectory of VEHGAN vehicle computed for NEDC cycle with DP algorithm (*Bottom*)

Optimization results obtained with DP method are presented and commented in section 4.

3 PONTRYAGIN MINIMUM PRINCIPLE OPTIMIZATION

In this section, we propose an alternative method to solve the optimal control problem (1). It relies on the Pontryagin Minimum Principle (PMP) and unlike the DP method does not require any discretization scheme.

3.1 Pontryagin Minimum Principle

The optimization problem (1) with the constraints defined in section (1.2) is rewritten as :

$$\begin{cases} \min_{u \in U}\left\{J(u) = \int_0^T L(x(t), u(t), t)dt + g(x(T), T)\right\} \\ \text{subject to} \\ \dot{x} = f(x(t), u(t), t), \qquad x(0) = x_0 \\ x_{min} \leqslant x(t) \leqslant x_{max} \end{cases}$$

$$(18)$$

We introduce the Hamiltonian function

$$\mathcal{H}(u(t), x(t), p(t)) = L(x(t), u(t), t) + p(t)\dot{x}(t) + \eta(t)h(x(t), t), \qquad (19)$$

$p(t)$ is called the co-state or adjoint variable of our system. $\eta(t)$ is the Lagrange multiplier associated with the state inequality constraints $x_{min} \leqslant x(t) \leqslant x_{max}$. In this section, we simplify the problem by cancelling this constraint. We

assume here that L is a smooth convex function of u and f is affine linear versus u.

The Pontryagin Minimum Principle states that a necessary condition of the problem (18) is the existence of $p(t)$ such that

$$\frac{\partial \mathcal{H}}{\partial x} = -\dot{p}. \tag{20}$$

If there is no constraint on command variable, we have also

$$\frac{\partial \mathcal{H}}{\partial u} = 0. \tag{21}$$

We refer to [10] and [2] for further details about Pontryagin Principle.

3.2 Application

The fuel consumption $L(t)$ to be minimized in (18), is defined by a discrete map $L(\omega, T_e)$ that can be modelled by a 2-order polynomial [7], to be used for continuous optimization. Choosing a 2-order polynomial ensures an unique solution (under some assumptions on polynomial coefficients) and allows to model a large variety of engine maps as shown in section 4. The fuel map is therefore written as :

$$L(\omega, T_e) = \sum_{i,j=0}^{2} K_{ij}.\omega^i.T_e^j. \tag{22}$$

The problem (18) is written as

$$\begin{cases} \min_{u \in U} \left\{ J(u) = \int_0^T L(\omega(t), T_e(t))dt + g(x(T), T) \right\} \\ \text{subject to} \\ \dot{x} = -K\omega(t)(1 - u(t))T_{rq}(t), \quad x(0) = x_0 \\ x_{min} \leqslant x(t) \leqslant x_{max} \end{cases} \tag{23}$$

From (20) and (7) we obtain

$$\dot{p} = 0 \Rightarrow p = constant = p_0. \tag{24}$$

Without any constraint on the state and on the command, the minimization problem can be easily solved. The minimum fuel consumption is then reached for u^* so as

$$\frac{\partial \mathcal{H}}{\partial u} = \frac{\partial L}{\partial u} + p\frac{\partial f}{\partial u} = 0. \tag{25}$$

The optimal command u^* can be calculated easily by solving the equation (25), which depends linearly on u. u^* finally depends on $p(t)$, $T_{rq}(t)$ and $\omega(t)$

$$u^*(t) = -\frac{\sum_{i=0}^{2} K_{i1}\omega(t)^i + p_0.K.\omega(t)}{2\sum_{i=0}^{2} K_{i2}\omega(t)^i.T_{rq}(t)}. \tag{26}$$

The expression of p_0 (27) is obtained by replacing $u^*(t)$ by its expression in the state equation (7), and by integrating this equation in time

$$p_0(\tau) = \frac{x(0) - x(\tau) - \int_0^\tau (K.\omega(t)(T_{rq}(t) + F(t))dt}{\int_0^\tau \frac{\omega(t)^2 K^2}{2\sum_{i=0}^{2} K_{i2}\omega(t)^i.T_{rq}(t)}dt}, \tag{27}$$

with

$$F(t) = \frac{\sum_{i=0}^{2} K_{i1}\omega(t)^i}{2\sum_{i=0}^{2} K_{i2}\omega(t)^i.T_{rq}(t)}, \tag{28}$$

and τ is the considered final time which can be set to a different value than T as we will see in next section.

3.2.1 Handling constraints on control and state variables

The previous section presents the computation of the optimal control of the continuous problem in a restricted case where no constraint is introduced. While control constraints are generally easily taken into account, handling the state constraints is cumbersome in the continuous optimal control problem. To handle control constraints, several singular cases can be found in [2]. In our application, we are not able to find an analytic solution of the optimal control problem with control constraints : indeed, these constraints depends on time (via their dependency on engine speed) and depends on p_0 which depends on final SOC (cf. previous section). By an iterative method (called *algo1* in the following), we can compute the value of p_0 in order to reach the desired SOC at final time with the control, computed thanks to (26), projected on bound constraints (coming from the feasible domain U).

Concerning the inequality state constraints, [8], [10], [4], [2], [6] have studied the general problem (18). In our application, we can show that $p(t)$ presents discontinuities at the time steps where the state inequality constraints are saturated. The time steps where such discontinuities occur are not a priori known : this prevents us to solve explicitly the continuous optimal control problem with these state constraints.

3.2.2 PMP Optimization algorithm

Considering the difficulties described in the previous section, we propose a heuristic iterative method that allows to find a sub-optimal trajectory from the constrained continous optimal control problem (18). The proposed algorithm consists in an initialization step and 3 steps :
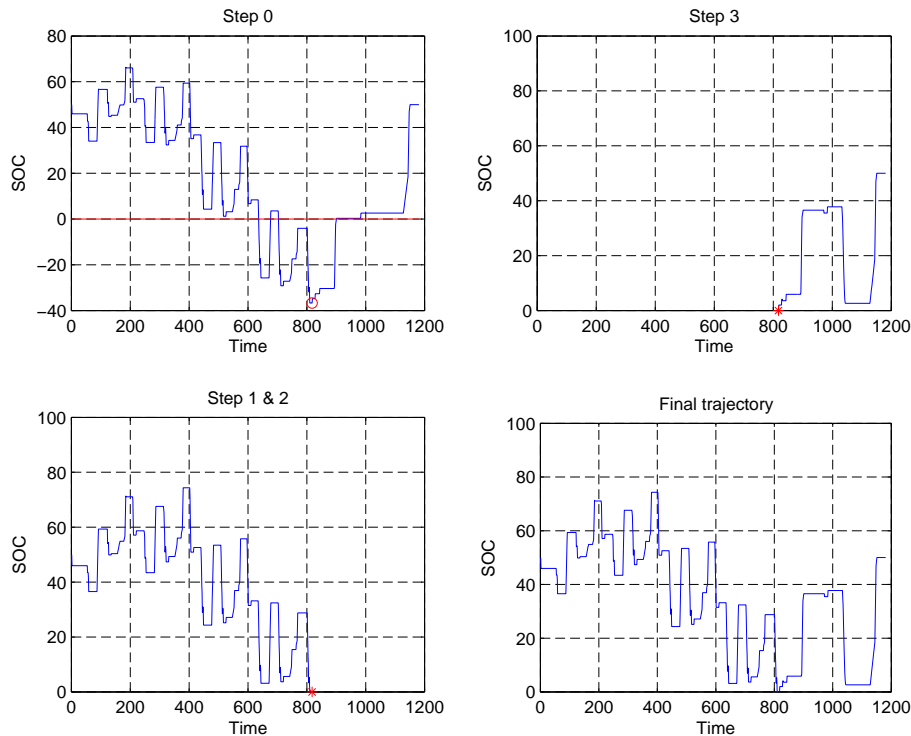
Figure 6

The proposed algorithm based on Pontryagin Minimum Principle.

0. *algo1* is applied on the NEDC driving cycle $[0, T]$ (see Figure 6 Step 0). The obtained optimal trajectory violates the state constraints, the farthest SOC (ie the "most violated point") from the bounds being for instance on the figure at point $(x(t_v) = -37\%, t_v = 818s)$. The initial time is called t_i, here set to 0.

1. The SOC at t_v is projected on the nearest bound of the feasible state domain (for instance, SOC is fixed to $x_{min} = 0$ at point t_v).

2. *algo1* is applied again on $[t_i, t_v]$ (see Figure 6 Step 2). If the obtained trajectory still violates the state constraints on $[t_i, t_v]$, steps 1 and 2 are applied again on the farthest SOC from the bounds (defining a new point t_v). This procedure is repeated until the trajectory remains on the feasible domain. Then the last point t_v becomes the new initial time t_i in step 3.

3. *algo1* is applied on $[t_i, T]$ (see Figure 6 Step 3). If the obtained optimal trajectory still violates the state constraints, steps 1 and 2 are repeated. This sequence is repeated until we reach the final step T at the desired final SOC, without violating the state constraints (Figure 6 bottom right).

3.3 PMP compared to DP Results

Using the same fuel consumption map and vehicle cycle than with Dynamic Programming, we can compare the two optimization algorithms along the NEDC driving cycle on Figure 7.

The two curves are very similar, the commands $u(t)$ being very similar too. We now compare the two algorithms on Urban Artemis driving cycle [9], with the same fuel consumption map, on Figure 8.

We notice again that the 2 curves are almost superimposed. In the following will be presented some results obtained with Pontryagin Principle algorithm, as it is about 1000 times faster than DP (DP algorithm computation time is about 1 hour), and gives quite the same results than DP. The small differences between the 2 algorithms can be minimized by taking smaller time and space discretization steps in DP, but it leads to a too expensive computation time.

4 RESULTS

In this section we present results for different kinds of engine, different sizes of battery, and different sizes of electric motor. As our mild-hybrid vehicle is a urban vehicle, we will present results on the urban Artemis driving cycle. For all simulations, we will assume that initial SOC is 80%, final SOC of 80% being introduced as a constraint to be close to initial SOC.
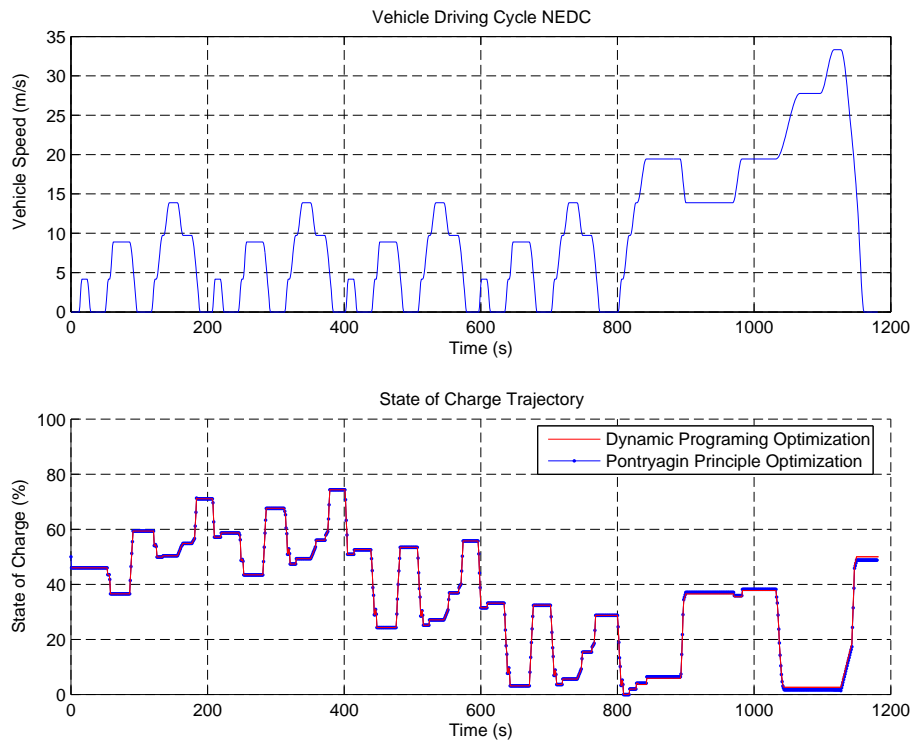
Figure 7

NEDC cycle(*Top*); Optimal state of charge trajectory of VEHGAN vehicle computed for NEDC cycle with PMP & DP algorithm (*Bottom*).
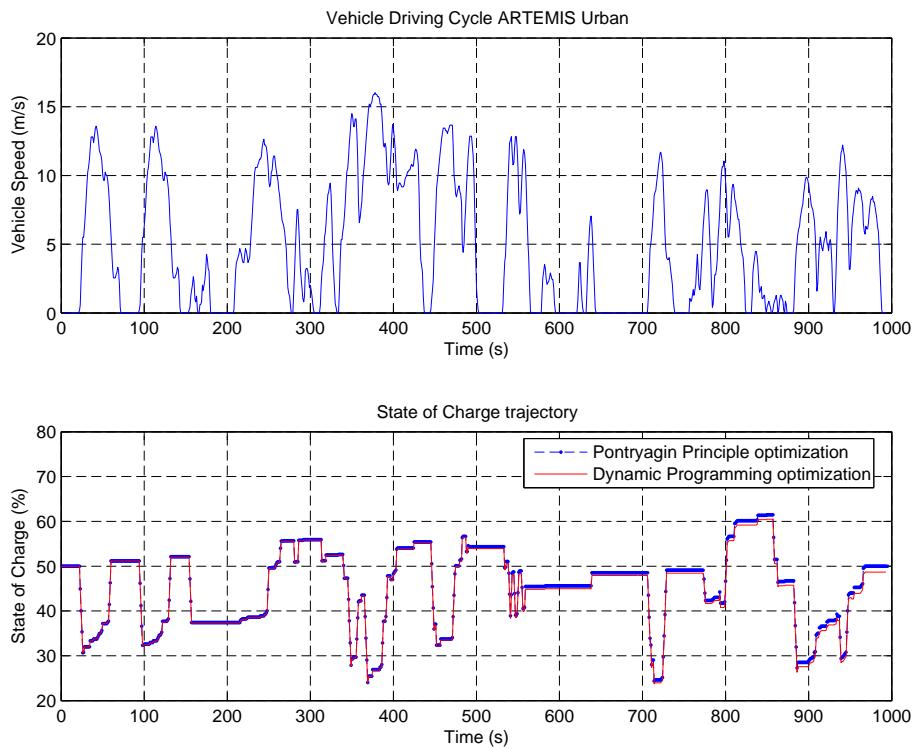

Figure 8

Urban Artemis cycle(*Top*); Optimal state of charge Trajectory of VEHGAN computed with PMP & DP algorithm (*Bottom*).

4.1 Different kind of engine

The both following results, natural gas engine and gasoline engine, were obtained with a battery capacity of 0.4 Ah, and a 3 kW electric motor, as on the VEHGAN vehicle, on Figure 9 and 11.

4.1.1 Natural Gas engine

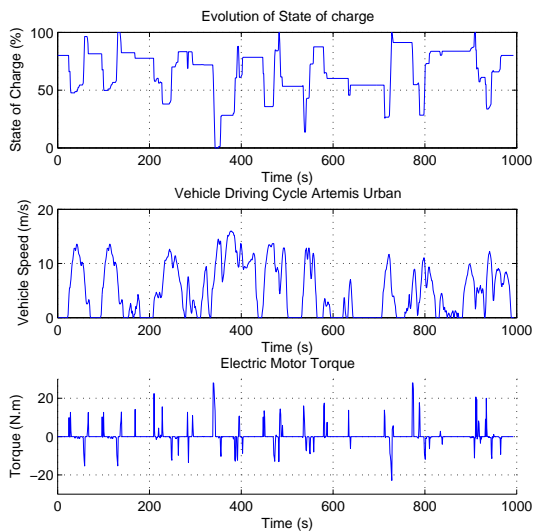The first engine used is a natural gas engine, from the VEHGAN project.

Figure 9

Urban Artemis driving cycle with natural gas engine : optimal state of charge trajectory (*Top*); electric motor torque (*Bottom*)

As the natural gas engine has an efficiency growing with the torque (the best operating points being between 2500 and 4500 rpm and over 60Nm, see Figure 2), the optimization decreases the fuel consumption using the electric motor in the worth efficiency points, i.e. when $\omega < 2500$ rpm. As using electric motor on low torque operating points would lead to worth operating points as illustrated on Figure 10, only high torque operating points are modified by an active hybrid mode.

4.1.2 Small Gasoline Engine

We now use a gasoline fuel map to optimize the fuel consumption for a $600cm^3$ gasoline engine .

As displayed on Figure 12, the optimized operating points are quite the same than ones of natural gas engine. Nevertheless, as the gasoline engine does not present a good efficiency at high torque operating points, the electric motor is also useful in high torque requests.
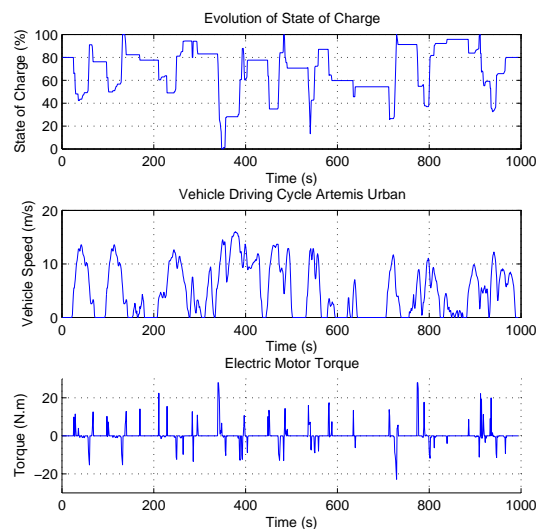
Figure 11

Urban Artemis driving cycle with gasoline engine : optimal state of charge trajectory (*Top*); electric motor torque (*Bottom*)

4.2 Different sizes of battery

In this section we optimize the fuel consumption for the natural gas engine, with different sizes of battery. As we can see on Figure 9, the state of charge crosses the point $x_{min} = 0$ %, meaning that the whole range of the state of charge of the battery is used. Then, if we increase the battery capacity, the optimization algorithm should give better results in term of fuel consumption.

With a battery capacity of 0.4 Ah, we obtain a fuel consumption of 5.90 l/100km, while a pure thermic mode consumption gives 6.08 l/100km on the whole cycle, so we decreased the consumption of 2.9 %. Optimizing the torque repartition with a battery capacity two times bigger, i.e 0.8 Ah, we obtain the same fuel consumption : 5.90 l/100km.

We obtain the same consumptions with the two battery sizes because of the low power of the electric motor, which is often saturated at regenerative breaking.

4.3 Increasing the Electric motor torque power

From the conclusion of these last results, we multiply by two the minimum and maximum electric motor power, to take advantage of a higher energy when regenerative breaking. We also increase the size of the battery, to have coherent battery capacity regarding to regenerative breaking and maximum electric motor power. Figure 14 presents the SOC evolution and the electric motor torque. The fuel consumption with this new electric motor reaches 5.87 l/100km, i.e
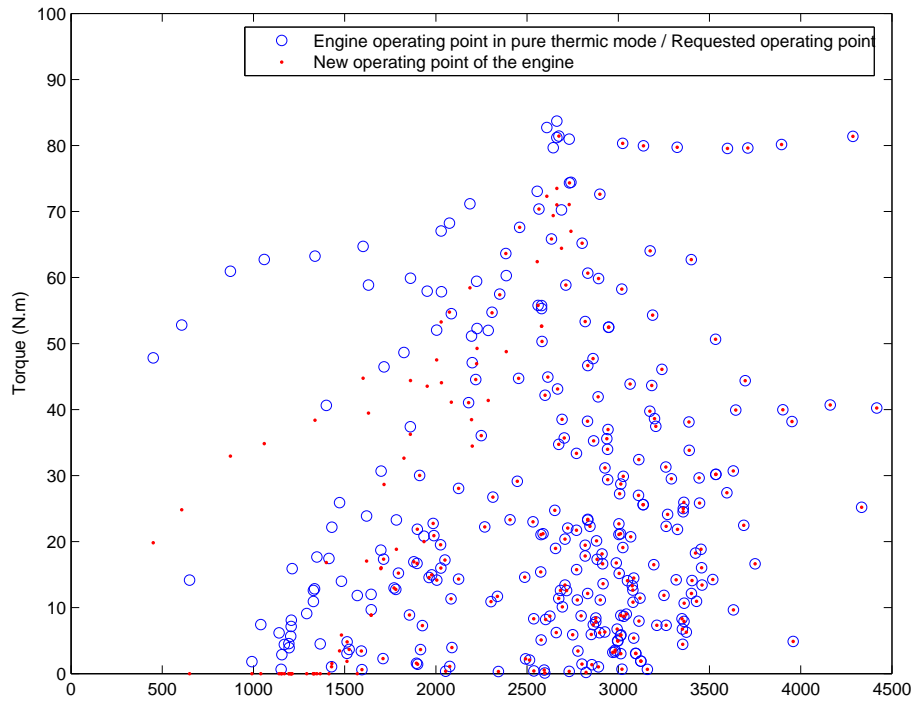
Figure 10

Operating points of engine in hybrid mode obtained by PMP algorithm for the urban Artemis Driving Cycle with a natural gas engine.
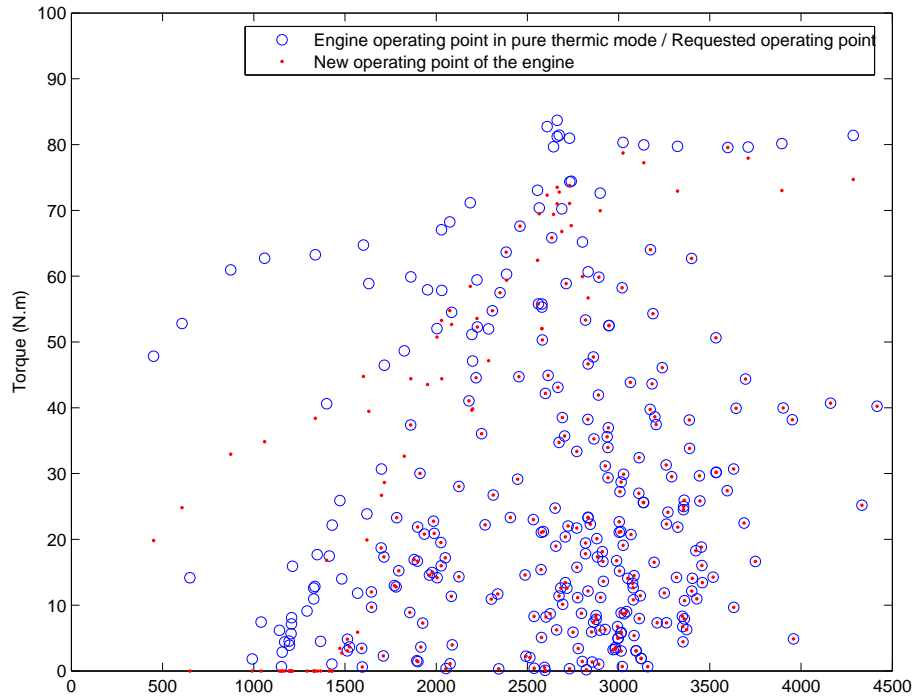

Figure 12

Operating points of engine in hybrid mode obtained by PMP algorithm for the urban Artemis Driving Cycle with a gasoline engine.
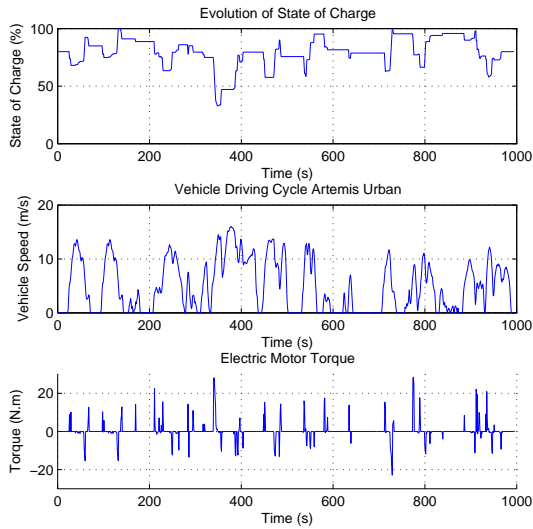
Figure 13

Urban Artemis driving cycle with natural gas engine and a 0.8 Ah battery capacity : optimal state of charge trajectory (*Top*); electric motor torque (*Bottom*)

−3.4 % compared to pure thermic mode, and −0.5 % compared to the smaller electric motor and battery. The Figure 15 resumes all the results obtained.
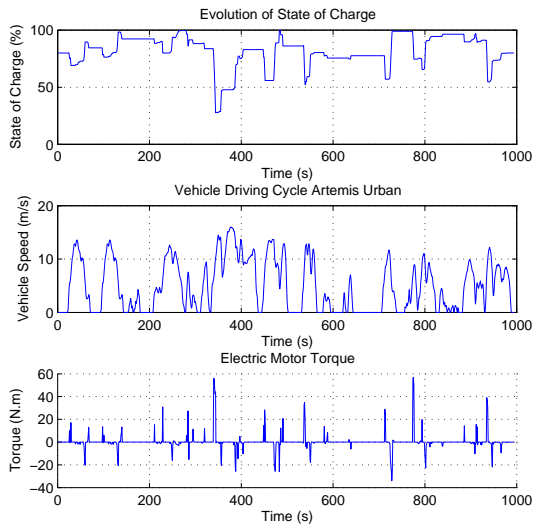

Figure 14

Urban Artemis driving cycle with natural gas engine, a 6 kW electric motor and a 1.2 Ah battery capacity : optimal state of charge trajectory (*Top*); electric motor torque (*Bottom*)

Electric Motor Power (kW)	*3*		*6*	
Battery Capacity (Ah)	*0.4*	*0.8*	*1.2*	*1.6*
Hybrid mode Consumption (l/100km)	*5.90*	*5.90*	*5.87*	*5.87*
Thermic mode Consumption (l/100km)	*6.08*			

Figure 15

Results with different sizes of battery and of electric motor on Artemis Urban Cycle.

4.4 Discussion

The previous results show that it is possible to decrease by 2 to 4% of fuel consumption along a urban driving cycle with this mild hybrid vehicle. Activating the *stop'n go* mode would save around 10 to 15 %, but it would also decrease the available electric energy from battery, as this energy must be used to turn on the engine after a stop.

5 CONCLUSIONS

In this study, we have presented two of the three most used methods in optimal control optimization. The heuristic method based on Pontryagin Minimum Principle, well known in the free state constraint case, has been applied successfully to our state constrained problem, with very similar results to Dynamic Programming methods and a computation time divided by 1000. Nevertheless, there is currently no theoretical proof to confirm the presented validation results. Moreover, there are some limitations to this approach, mainly the assumptions on the fuel consumption map, modelled by a smooth convex function of control u (2-order polynomial) ; this limitation could lead to a bad approximation of the real fuel consumption for some particular engines.

The case study, a mild hybrid vehicle, does not present as good results as could be obtained with a full hybrid, in terms of fuel consumption gain. So in future work will be studied optimization applied to full-hybrid vehicles, as the ability to turn off the engine while the electric motor only power the vehicle is a key advantage. Other degrees of freedom, as the gear-shifting sequence should also be taken into account in the optimization problem. Reduction of pollutant emissions will also be studied by considering a second state.

From optimization results could be derived a suboptimal feedback law [1], [11], [13] based on state of charge and/or speed measurements.

REFERENCES

[1] Sciarretta Antonio, Guzzella Lino, and Back Michael, *A real-time optimal control strategy for parallel hybrid vehicles with on-board estimation of the control parameters*, Proceedings of IFAC Symposium on Advances in Automotive Control AAC04 (2004), 502–507.

[2] E. Bryson and Y.C. Ho, *Applied optimal control*, Hemisphere Pub. Corp., 1975.

[3] S. Delprat, *Evaluation de stratégies de commande pour véhicules hybrides parallèles*, Ph.D. thesis, Université de Valenciennes et du Hainaut-Cambresis, 2002.

[4] Lawrence C. Evans, *An introduction to mathematical optimal control theory*, University of California Berkeley.

[5] J.M. German, *Hybrid powered vehicles*, Society of Automotive Engineers (SAE) (2003).

[6] T. Guilbaud, *Méthodes numériques pour la commande optimale*, Ph.D. thesis, Université de Paris VI, 2002.

[7] L. Guzzella and A. Sciarretta, *Vehicle propulsion systems*, Springer, 2005.

[8] Richard F. Hartl, Suresh P. Sethi, and Raymond G. Vickson, *A survey of the maximum principles for optimal control problems with state constraints*, SIAM Review **37** (1995), no. 2.

[9] André M., *The artemis european driving cycles for measuring car pollutant emissions*, Science of The Total Environment **334-335** (2004), 73–84.

[10] L.S. Pontryagin, V.G. Boltyanskii, R.V. Gamkrelidze, and E.F. Mishchenko, *Théorie mathématique des processus optimaux*, Editions Mir moscou, 1974.

[11] J. Scordia, *Approche systématique de l'optimisation du dimensionnement et de l'élaboration de lois de gestion d'énergie de véhicules hybrides*, Ph.D. thesis, Université Henri Poincare - Nancy 1, November 2004.

[12] R. Tilagone and S. Venturi, *Development of natural gas demonstrator based on an urban vehicle with a down-sized turbocharged engine*, Oil and Gas Science and Technology **59** (2004), no. 6, 581–591.

[13] B. Wu, C-C. Lin, Z. Filipi, H. Peng, and D. Assanis, *Optimization of power management strategies for a hydraulic hybrid medium truck*, Proceeding of the 2002 Advanced Vehicle Control Conference, Hiroshima, Japan (2002).