A branch and bound approach for minimizing the energy consumption of an electrical vehicle

Abdelkader Merakeb, Frédéric Messine

A Branch and Bound Approach for Minimizing the Energy Consumption of an Electrical Vehicle

Abdelkader Merakeb
Département de Mathématiques, Université Mouloud Mammeri de Tizi-Ouzou, 15000 Algérie.
Email: merakeb_kader@yahoo.fr

Frédéric Messine
ENSEEIHT-IRIT, UMR-CNRS 5055,
2 rue Camichel, 31000 Toulouse, France.
Email: frederic.messine@n7.fr

***Abstract*—In this paper we discuss about the way to approximate the solution of an optimal control problem with a switched command. Our Method is based on a discretization technique associated with a Branch and Bound algorithm. The problem that we focuss on is the minimization of the consumption of the energy of an electrical vehicle during some imposed displacements.**

***Keywords*—Discretization Techniques; Optimal Control; Discrete Switch Control; Branch-and-Bound; Electrical Vehicle.**

I. Introduction

In this paper, we discuss about the way to solve efficiently the problem of the minimization of the energy which is consummated by an electrical car during an imposed displacement, see [1] for an overview on this type of problems.

The problem that we are interested with, can be formulated as follows:

$$\left\{\begin{array}{l} \min_{i_m(t),\Omega(t),pos(t),u(t)} E(i_m,u) = \int_0^{t_f} u(t) i_m(t) V_{alim} \\ \qquad\qquad\qquad\qquad ... + R_{bat} u^2(t) i_m^2(t) dt \\ u.c. \\ \dot{i_m}(t) = \frac{u(t)V_{alim} - R_m i_m(t) - K_m \Omega(t)}{L_m} \\ \dot{\Omega}(t) = \frac{1}{J}\left(K_m i_m(t) - \frac{r}{K_r}\left(MgK_f + \frac{1}{2}\rho S C_x \left(\frac{\Omega(t) r}{K_r}\right)^2\right)\right) \\ \dot{pos}(t) = \frac{\Omega(t) r}{K_r} \\ |i_m(t)| \leq 150 \\ u(t) \in \{-1, 1\} \\ (i_m(0), \Omega(0), pos(0)) = (i_m^0, \Omega^0, pos^0) \in I\!R^3 \\ (i_m(t_f), \Omega(t_f), pos(t_f)) \in \mathcal{T} \subseteq I\!R^3 \end{array}\right. \quad (1)$$

where E represents the electrical energy consummated during the displacement. The state variables are: (i) i_m the current inside the motor; (ii) Ω the angular speed, with $V(t) = \frac{3.6 \times r}{K_r} \times \Omega(t)$ which represents the speed of the car in km/h (r is the radius of the wheel); (iii) pos is the position of the car. The control u is in $\{-1, 1\}$ (a switch control); the physical system can switch in $10^{-9}s$. In this problem, we have a constraint on a state variable to limit the current inside the motor in order to discard the possibility to destroy it. The other terms are fixed parameters and represent some physical things: -$K_r = 10$, the coefficient of reduction; -$\rho = 1.293 kg/m^3$, the air density; -$C_x = 0.4$, the aerodynamic coefficient; -$S = 2m^2$, the area in the front of the car; -$r = 0.33m$, the radius of the wheel; -$K_f = 0.03$, the constant representing the friction of the wheels on the road; -$K_m = 0.27$, the coefficient of the motor torque; -$R_m = 0.03\Omega$, the inductor resistance; -$L_m = 0.05$, inductance of the rotor; -$M = 250kg$, the mass; -$g = 9.81$, the gravity constant; -$J = M \times r^2/K_r^2$; -$V_{alim} = 150V$, the battery voltage; -$R_{bat} = 0.05\Omega$, the resistance of the battery. The initial conditions are given but the target set $\mathcal{T}$ is free and depends on the instances of the problem; it could be a point of $I\!R^3$ but one or two variables could not be fixed: for example, $\mathcal{T}$ just could be the final position equal to $100m$ is required (see the numerical section).

This problem is hard to solve directly by using classical optimal control techniques. We tried to solve it by using the Pontryaguin method based on shooting techniques and also by using a direct shooting algorithm, [3]. For the moment, the fact that we have a constraint on the state associated with the fact that it is a switched control involves a lot of difficulties which does not allow us to obtain interesting solutions (even local ones). The dynamic programming of Bellman is also difficult to apply to this problem, [2].

Thus, in this paper we propose another original methodology to solve this problem yielding to some discretized problems which are solved using an exact Branch and Bound algorithm. This new method provide exact results for the discretized formulations which correspond to approximations of the global solutions of Problem (1).

In Section II, we present an approximation of Problem (1) by a discretization of it based on current references. In Section III, the Branch and Bound algorithm is presented. In Section IV, some numerical examples are discussed and validate our approach. We conclude in Section V.

II. Approximation of Problem (1)

First, we remark that if we discretize all the interval of time $[0, t_f]$ by fixing the value of the control u, it is necessary to have very small steps about $10^{-3}s$ else the value of the current will change too roughly. That will generate a very huge mixed integer non-linear global optimization problem which is, for the moment, impossible to solve using direct methods of optimal control.

Another idea, which directly comes from the numerical simulation of the behavior of the car, is to impose during some short laps of time the value of the current inside the electrical motor of the vehicle. This is possible using the control parameter $u(t)$. Thus, if we impose a reference current $iref$, if

$i_m(t) > iref + \frac{\Delta}{2}$ then $u(t) := -1$ and if $i_m(t) < iref - \frac{\Delta}{2}$ then $u(t) := 1$ and else $u(t)$ keep the same value. This technique is just a way to construct a current regulator which is a first step before making a speed regulator for an electrical car. Hence, using this, the following differential system of equations can be solved:

$$VS(iref, t_0, t_f) :=$$

$$\left\{ \begin{array}{l} \dot{E}(t) = u(t) i_m(t) V_{alim} + R_{bat} u^2(t) i_m^2(t) \\ \dot{i_m}(t) = \frac{u(t) V_{alim} - R_m i_m(t) - K_m \Omega(t)}{L_m} \\ \dot{\Omega}(t) = \frac{1}{J} \left(K_m i_m(t) - \frac{r}{K_r} \left(M g K_f + \frac{1}{2} \rho S C_x \left(\frac{\Omega(t) r}{K_r} \right)^2 \right) \right) \\ \dot{pos}(t) = \frac{\Omega(t) r}{K_r} \\ u(t) := \left\{ \begin{array}{l} -1 \text{ if } i_m(t) > iref + \frac{\Delta}{2} \\ +1 \text{ if } i_m(t) < iref - \frac{\Delta}{2} \\ u(t) \text{ else.} \end{array} \right. \\ (E(t_0), i_m(t_0), \Omega^(t_0), pos(t_0)) = (E^{t_0}, i_m^{t_0}, \Omega^{t_0}, pos^{t_0}) \in I\!R^4 \\ u(t_0) := 1; \end{array} \right. \quad (2)$$

where t_0 is the initial time which is not necessary equal to 0. This system of differential equations can be efficiently solved using a classical differential integrator such as for example *Euler, RK2, RK4* with a step of time less than 10^{-3}. The function $VS(iref, t_0, t_f)$ will compute in theory all the values for $E(t), i_m(t), \Omega(t), pos(t)$, for all $t \in [t_0, t_f]$ but in practice only values for a discretized time $t_i \in [t_0, t_f]$ is available. Here, we are interested by the final values of the state variables, hence we define a function:

$$VSF(iref, t_0, t_f) := (E(t_f), i_m(t_f), \Omega(t_f), pos(t_f)) \in I\!R^4,$$

all the computations are performed using function VS which solves the system of differential equations (2).

The main idea of this work is to subdivide the cycle of time $[0, t_f]$ into P subintervals. In each step of time $[t_{p-1}, t_p]$ with $p \in \{1, \cdots, P\}$ ($t_p = p \times \frac{t_f}{P}$), we apply a reference current $iref_p$ which takes values in $[-150, 150]$ in order to directly satisfy the constraint on the state variable of Problem (1).

Thus, we focuss on the resolution of the following optimization problem:

$$\left\{ \begin{array}{l} \min_{iref \in [-150,150]^P} \sum_{k=1}^{P} E_k \\ u.c. \\ \\ (E_k, i_k, \Omega_k, pos_k) := VSF(iref_k, t_{k-1}, t_k) \\ (E_0, i_0, \Omega_0, pos_0) = (E^0, i_m^0, \Omega^0, pos^0) \in I\!R^4 \\ (i_P, \Omega_P, pos_P) \in \mathcal{T} \subseteq I\!R^3 \end{array} \right. \quad (3)$$

Problem (3) is a good approximation of the initial problem (1) which generates just a few number of variables: P. In fact, we use a current regulator system to control the vehicle; this is also interesting in itself for a future implementation of the system in the car.

III. Dedicated Branch and Bound Algorithm

For the moment, we are not able to solve exactly the global optimization problem (3), thus we need to discretize also the possible values for the reference current: $iref \in \{-150, -150 + s, -150 + 2 \times s, \cdots, 150\}^P$; we will take integer values for s which divide exactly $[-150, 150]$. Therefore, the set of solution becomes finite and could be enumerated. Nevertheless, if we want to have a good approximation for the resolution of the global optimization problem (3) we have to discretize into small steps and the finite set of possible points becomes rapidly too huge to be entirely enumerated in a reasonable CPU-time.

The idea is then to use a Branch and Bound algorithm in order to not explore all the finite set of solutions. For using such an algorithm, we have to elaborate a technique to compute bounds for the four main parameters: $E_k, i_k, \Omega_k, pos_k$ over a box $IREF \subseteq \{-150, -150 + s, -150 + 2 \times s, \cdots, 150\}^P$ and for given t_0 and t_f. In order to be more efficient, in a previous step, we compute 4 matrices: $M_E, M_i, M_\Omega, M_{pos}$ where the columns corresponds to values when $iref$ is fixed with $i_m^{t_0} = iref$ and the lines provides values for the entities when a speed Ω^{t_0} is given (we discretize also the possible values of the speed). For example $m_E(i, j)$ represents the value of the energy which is consummated during a step of time $t_p - t_{p-1}$ when $iref$ is equal to the jth components of $\{-150, -150 + s, -150 + 2 \times s, \cdots, 150\}$ with $i_m^{t_0} = iref$ and the ith discretized value for the speed, the other initial values are taken equal to 0: i.e., $E^{t_0} = pos^{t_0} = 0$.

When a box $IREF$ is considered, we can compute bounds for E, i, Ω and pos by computing the integer sets I and J of the indices corresponding to the possible values of the speed at the previous step and the possible values of $iref$. Then, we have to compute the bounds which correspond to the minimal and maximal values of $m_E(i, j), m_i(i, j), m_\Omega(i, j), m_{pos}(i, j)$ with $(i, j) \in I \times J$. To obtain the final value for E and pos, we have to sum all the lower and upper bounds. The rest of the Branch and Bound algorithm that we develop is simple and uses the following classical principle: (i) subdivision into two (distinct) parts of the enumerate set $IREF$ (which represents the possible values for $iref$); (ii) the upper bound is updated by taking the middle of the box $IREF$ if the constraints are satisfied and if its value is better than the previous one (we start with $+\infty$); (iii) we branch following the heuristic of lowest lower bound of the energy.

IV. Numerical Experiments

To illustrate our method, we simulated it for a displacement of 100 meters, and a cycle $t_f = 10$ seconds: $(i_m(0), \Omega(0), pos(0)) = (0, 0, 0)$; $(i_m(t_f), \Omega(t_f), pos(t_f)) \in \mathcal{T} = I\!R \times I\!R \times \{100\}$.

For the resolution of the VSF function, we used the Runge-Kutta integrator at the order 4, with a step of time equal to 10^{-3}, simulated on MatLab 7 on a standard PC Laptop with 2GB of RAM. The parameters for our code are fixed to $P = 5$, $s = 5$, $0.1km/h$ for the step of the discretization of the speed (to compute the matrices M) and $\Delta = 1$. Thus, we obtain the exact (for the discretized problem) solution $iref^* = (150, 90, 25, -15, -110)$ corresponding to the minimal value $E^*(10) = 24430.21J$. Moreover, we have $pos^*(10) = 100.03m$. The CPU-time is about $966s$ corresponding to 201830 iterations of the Branch and Bound algorithm. This long CPU-time strongly depends on the parameter s and also P which is understandable for a Branch and Bound code (the complexity of such an algorithm depends on $\left(\frac{2 \times 150}{s} + 1\right)^P$). Thus, if we take $s = 10$, we

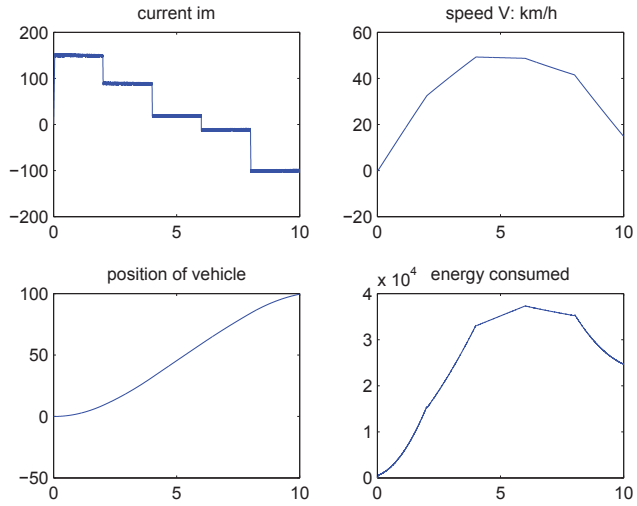

Fig. 1. Solution for a final position of $100m$ in $10s$.

obtain the following results: $iref^* = (150, 90, 20, -10, -100)$ corresponding to the minimal value $E^*(10) = 24589.90J$ with a position of $100.15m$ in only $17.62s$ for 31090 iterations. This solution is represented in the following figure: Therefore an idea to obtain much more precise solutions, is simply to run the Branch and Bound code iteratively by defining more and more precise zones around the previous exact solutions and by increasing parameter P and decreasing s. We remark that the current i_m remains trapped around $iref$ with respect to the tolerance Δ. The values of u switches many times between -1 and $+1$; this is due to the fact that the current in the motor increases too quickly (average of $3A$ every $10^{-3}s$). Note that the final speed is not equal to zero because the final time is too short. Moreover, we remark that the curve of the energy decreases at the end of the cycle because this corresponds to the phase of deceleration with a negative period for the reference current $iref$.

V. Conclusion

In this paper, we show an original way based on discretization and a Branch and Bound algorithm to solve a hard global optimization problem which is an approximation of an optimal control problem. In a future work, we want to improve the efficiency of our Branch and Bound algorithm. Furthermore, we are interested by the resolution of Problem (3) directly by computing bounds.

References

[1] A. Sciarreta, L. Guzzella, *Control of Hybrid Electric Vehicles - A Survey of Optimal Energy-Management Strategies*, IEEE Control Systems Magazine, Vol. 27, N. 2, pp. 60–70, 2007.

[2] C. Musardo, G. Rizzoni, Y. Guezennec, B. Staccia, *A-ECMS: An Adaptive Algorithm for Hybrid Electric Vehicle Energy Management*, European Journal of Control, N. 11 (4–5), pp. 509–524, 2005.

[3] J. Bernard, S. Delprat, T.M. Guerra, F. Buechi, *Fuel Cell Hybrid Vehicles: Global Optimization based on Optimal Control Theory*, International Review of Electrical Engineering, 1, 2006.