SCOP: a Sequential Constraint-free Optimal control Problem algorithm

Grégory Rousseau, Quang Huy Tran, Delphine Sinoquet

SCOP:
a Sequential Constraint-free Optimal control Problem algorithm

Grégory Rousseau, Quang Huy Tran and Delphine Sinoquet

Abstract— **In this paper, we propose a new computational algorithm for optimal control problems with a scalar state subject to simple-bound constraints. Unlike multiple shooting algorithms, the number of active constraints does not need to be known in advance. This method is also much faster than dynamic programming. Convergence is proved for two particular cases. SCOP is applied for an optimization problem of fuel consumption for a hybrid vehicle.**

I. INTRODUCTION

Optimal control problems involving dynamic systems and state boundary constraints can be solved following various approaches. These approaches can be mainly divided into two families: (i) direct methods, which assume that the optimal control problem has been discretized in time and space, and which lead to classical but often large-sized nonlinear optimization problems; (ii) indirect methods ([6]), which take advantage of Pontryagin's maximum principle at the continuous level, using shooting algorithms.

While direct methods are well suited for non-analytic models, their main drawback is the computational time. A good example is the Dynamic Programming algorithm (DP) ([6], [8]). Nevertheless DP can be easily extended to take state constraints into account.

Besides, Pontryagin's principle gives optimality conditions which are often solved by a shooting method. However, the presence of state constraints is a tremendous obstacle, and requires the use of a multiple shooting algorithm. Furthermore, the active state constraints should be known ([3]).

This contribution presents optimal control problems with a bounded scalar state. We devise a numerical method that does not require any a priori knowledge on active constraints. Proof of convergence is given for two applications: a hybrid vehicle and an elastic rope. Finally, we carry out a numerical test in order to show that the method still works fine on a realistic case: this application is the optimization of torque split between the engine and the electric motor of a hybrid vehicle with respect to fuel consumption with constraints on the state of charge of the battery ([1], [2], [9]).

Département Mathématiques Appliquées, Institut Français du Pétrole, 1 et 4 avenue de Bois-Préau, 92852 Rueil-Malmaison Cedex, FRANCE
Corresponding author: `Gregory.Rousseau@ifp.fr`

II. DESCRIPTION OF THE METHOD

A. Statement of the problem

Let us consider the abstract optimal control problem

$$(\mathscr{P}) \qquad \min_{u(\cdot)\in\mathfrak{U}} \int_0^S L(z(s), u(s), s)\, ds \tag{1}$$

subject to

$$\frac{dz}{ds}(s) = f(z(s), u(s), s) \tag{2a}$$
$$z(0) = \zeta_0 \tag{2b}$$
$$z(S) = \zeta_S \tag{2c}$$
$$z(s) \geq z_{\min} \tag{2d}$$
$$z(s) \leq z_{\max}, \tag{2e}$$

where $z(\cdot) \in \mathbb{R}$, $u(\cdot) \in \mathfrak{U}$, and $s > 0$ stand for the state, the control and the evolution variable. We assume that

$$z_{\min} < \min\{\zeta_0, \zeta_S\} \leq \max\{\zeta_0, \zeta_S\} < z_{\max}. \tag{3}$$

B. The SCOP method

Our idea consists in solving problem $(\mathscr{P})$ by a sequence of unconstrained subproblems $(\mathscr{Q}_k)$, defined as

$$(\mathscr{Q}_k)[s_k, Z_k] \qquad \min_{u(\cdot)\in\mathfrak{U}} \int_0^{s_k} L(z_k(s), u(s), s)\, ds \tag{4}$$

subject to

$$\frac{dz_k}{ds}(s) = f(z_k(s), u(s), s) \tag{5a}$$
$$z_k(0) = \zeta_0 \tag{5b}$$
$$z_k(s_k) = Z_k, \tag{5c}$$

where (s_k, Z_k) are two parameters meant to evolve until s_k reaches some "correct" value. The overall algorithm reads

---

Algorithm 1. SCOP

---

data: $k = 0$, $s_0 = S$, and $Z_0 = \zeta_S$;
begin
  Solve $(\mathscr{Q}_0)[s_0, Z_0]$
  while $\exists s \in]0, s_k[\mid z_k(s) \leq z_{\min}$ **or** $z_k(s) \geq z_{\max}$ **do**
    Define $\Delta_k(s) = \max\{z_{\min} - z_k(s), z_k(s) - z_{\max}\}$
    Determine $s_{k+1} = \mathrm{argmax}_{s\in[0,s_k]} \Delta_k(s)$
    Compute $Z_{k+1} = \Pi_{[z_{\min}, z_{\max}]}(z_k(s_{k+1}))$
    Solve $(\mathscr{Q}_{k+1})[s_{k+1}, Z_{k+1}]$
  end
end

The symbol $\Pi_{[z_{\min},z_{\max}]}(Z)$ denotes the truncation operator

$$\Pi_{[z_{\min},z_{\max}]}(Z) = \begin{cases} z_{\min} & \text{if } Z < z_{\min} \\ z_{\max} & \text{if } Z > z_{\max}. \end{cases} \tag{6}$$

Our claim is that if

$$\forall s \in [0, s_k], \quad z_{\min} \le z_k(s) \le z_{\max} \tag{7}$$

occurs at some intermediate iteration k or at the limit $k \to \infty$, then $z_k(s)$ coincides with the optimal trajectory $z(s)$ of problem $(\mathscr{P})$ over the interval $[0, s_k]$.

The benefit of this method lies in the fact that each unconstrained problem $(\mathscr{Q}_k)$ can be solved either analytically or by a single shooting algorithm via a constraint-free Pontryagin's principle.

III. Pontryagin's principle for a problem without state constraint

For later use, let us recall Pontryagin's principle for problem $(\mathscr{P})$ in which the constraints (2d)–(2e) have been left out. Let

$$H(z, u, s, p) = L(z, u, s) + p(s)f(z, u, s) \tag{8}$$

be the Hamiltonian, considered as a function of four variables. Then, the optimal process satisfies

$$\frac{dz}{ds}(s) = \quad H_p(z(s), u(s), s, p(s)) \tag{9a}$$

$$\frac{dp}{ds}(s) = -H_z(z(s), u(s), s, p(s)) \tag{9b}$$

$$u(s) = \operatorname{argmin}_{v\in\mathfrak{U}} H(z(s), v, s, p(s)). \tag{9c}$$

When there is no constraint on the control u, e.g. for $\mathfrak{U} = \mathbb{R}$, condition (9c) can be expressed, at least formally, as

$$0 = H_u(z(s), u(s), s, p(s)). \tag{10}$$

IV. Analysis for a hybrid vehicle model

In this section, we set $s \equiv t$, $z(s) \equiv x(t)$, $S = T$ and $Z_k \equiv X_k$. Consider the optimal control problem

$$(\mathscr{H}) \qquad \min_{u(\cdot)\in\mathbb{R}} \int_0^T \alpha^2(t)u^2(t)\,dt \tag{11}$$

subject to

$$\dot{x}(t) = -\alpha(t)(1 - u(t)) \tag{12a}$$

$$x(0) = \chi_0 \tag{12b}$$

$$x(T) = \chi_0 \tag{12c}$$

$$x(t) \le x_{\max}. \tag{12d}$$

This problem corresponds to a simplified model of torque split for a hybrid vehicle powered by an engine and an electric motor (see section VI for a detailed description of a realistic hybrid vehicle). The integrand $\alpha^2 u^2$ represents the fuel consumption to be minimized, $-\alpha(1-u)$ is the battery law, and $x(t)$ is the state of charge of the battery at time t. Note that $x(0) = x(T) = \chi_0 < x_{\max}$. We assume that $\alpha(t)$ is positive, differentiable and strictly increasing with respect to $t \in [0, T]$. For convenience, we introduce

$$\beta(t) = \frac{1}{t}\int_0^t \alpha(\tau)\,d\tau, \qquad \text{for } t > 0, \tag{13a}$$

$$\beta(0) = \alpha(0). \tag{13b}$$

It is straightforward to check that β is an increasing function of $t \in [0, T]$.

Lemma 4.1: There exists a unique $\bar{t} \in]0, T[$ such that

$$\alpha(\bar{t}) = \beta(T) \tag{14}$$

Proof: This follows from the intermediate value theorem and from the strictly increasing property of α. ■

First, we write down the exact solution of $(\mathscr{H})$.

Proposition 4.1: Let

$$\bar{x} = \chi_0 + \bar{t}[\beta(T) - \beta(\bar{t})] \tag{15}$$

where $\bar{t}$ is defined in Lemma 4.1.

1) If $x_{\max} > \bar{x}$, then the optimal trajectory is given by

$$x(t) = \chi_0 + t[\beta(T) - \beta(t)]. \tag{16}$$

In this case, the constraint $x(t) \le x_{\max}$ is never active.

2) If $x_{\max} \le \bar{x}$, then there exists a unique $t^* \in]0, T]$ such that, for $t \in [0, t^*]$, the trajectory is given by

$$x(t) = \chi_0 - \frac{p_0^*}{2}t - \int_0^t \alpha(\tau)d\tau, \tag{17}$$

where (t^*, p_0^*) is the unique solution to the system

$$x_{\max} - \chi_0 = -\frac{p_0^*}{2}t^* - \int_0^{t^*} \alpha(\tau)\,d\tau, \tag{18a}$$

$$\alpha(t^*) = -\frac{p_0^*}{2}. \tag{18b}$$

Proof: **Case 1**. Applying (9)–(10) to the Hamiltonian

$$H(x, u, t, p) = \alpha^2 u^2 - p\alpha(1 - u), \tag{19}$$

we end up with

$$\dot{p} = 0, \qquad 2\alpha^2 u + p\alpha = 0, \tag{20}$$

from which we infer that

$$p(t) = p_0, \qquad u(t) = -\tfrac{1}{2}\tfrac{p_0}{\alpha(t)}. \tag{21}$$

Integration with respect to time of the dynamic law

$$\dot{x}(t) = -(\alpha + \tfrac{1}{2}p_0) \tag{22}$$

yields

$$x(t) = \chi_0 - \frac{p_0}{2}t - \int_0^t \alpha(\tau)\,d\tau. \tag{23}$$

Invoking the final condition $x(T) = \chi_0$, we obtain (16). Since

$$\dot{x}(t) = \beta(T) - \alpha(t), \qquad \ddot{x}(t) = -\dot{\alpha}(t) < 0, \tag{24}$$

we see that the only critical point $\bar{t} \in [0, T]$ achieves a maximum for $x(\cdot)$, i.e.,

$$\max_{t\in[0,T]} x(t) = x(\bar{t}) = \bar{x}, \tag{25}$$

where $\bar{x}$ is defined in (15). Therefore, if $x_{\max} > \bar{x}$, the optimal trajectory is (16).

Case 2. If $x_{\max} < \bar{x}$, let t^* be the contact point where the trajectory hits the boundary. From the continuity of the Lagrange multipliers established in [3] by Bonnans and Hermant, we derive the matching conditions

$$x(t^*) = x_{\max} \tag{26a}$$
$$\dot{x}(t^*) = 0. \tag{26b}$$

For $t < t^*$, the optimal trajectory $x(\cdot)$ is of the form (23), in which we write the co-state as p_0^* instead of p_0 for clarity. Plugging (23) into (26), we obtain (18).

Let us eliminate p_0^* from (18a)–(18b), so as to have

$$t^*\alpha(t^*) - \int_0^{t^*} \alpha(\tau)\, d\tau = x_{\max} - x_0. \tag{27}$$

From (27), existence and uniqueness of t^* can be obtained by arguing that the function

$$\psi(t) = t\alpha(t) - \int_0^t \alpha(\tau)\, d\tau \tag{28}$$

is strictly increasing and that

$$\psi(0) < x_{\max} - x_0 < \psi(T). \tag{29}$$

This completes the proof. ■

We now proceed to investigate the behavior of SCOP. It is obvious that if $x_{\max} > \bar{x}$, then once $(\mathscr{Q}_0)[t_0 = T, X_0 = \chi_0]$ has been solved, the trajectory $x_0(\cdot)$ is the optimal one (16) and the algorithm stops.

Theorem 4.1: For $x_{\max} \leq \bar{x}$, the sequence $\{t_k\}_{k\geq 0}$ produced by SCOP is well-defined, strictly decreasing and converges to t^* as $k \to \infty$.

This sequence is governed by the implicit recursion

$$\alpha(t^{k+1}) = \frac{x_{\max} - x_0}{t_k} + \beta(t_k) \tag{30}$$

and can be interpreted as a fixed point approximation to the continuous relation

$$\alpha(t^*) = \frac{x_{\max} - x_0}{t^*} + \beta(t^*), \tag{31}$$

which is a consequence of (27).

Proof: **Implicit recursion**. Except for $X_0 = \chi_0$, we have $X_k = x_{\max}$ for $k \geq 1$, as long as the algorithm goes on. According to (23),

$$x_k(t) = \chi_0 - \frac{p_k}{2} t - \int_0^t \alpha(\tau)\, d\tau, \tag{32}$$

at the k-th step. By construction of Algorithm 1, we have

$$x_k(t_k) = x_{\max} \tag{33a}$$
$$\dot{x}_k(t_{k+1}) = 0. \tag{33b}$$

Plugging (32) into (33) leads to

$$x_{\max} - x_0 = -\frac{p_k}{2} t_k - \int_0^{t_k} \alpha(\tau)\, d\tau \tag{34a}$$
$$\alpha(t_{k+1}) = -\frac{p_k}{2}, \tag{34b}$$

from which we deduce (30) by eliminating p_k.

Existence and uniqueness. We are going to show that if $t^k > t^*$, then t^{k+1} exists and $t^{k+1} > t^*$, so that by induction the whole sequence can be generated.

Equation (30) is guaranteed to have a unique solution as soon as

$$\alpha(0) < \frac{x_{\max} - x_0}{t_k} + \beta(t^k) < \alpha(t^k)$$

The left inequality is obvious, since $\alpha(0) = \beta(0) \leqslant \beta(t^k)$ (and β is an increasing function). As for the right inequality, it is equivalent to

$$x_{\max} - x_0 < \psi(t^k), \tag{35}$$

where ψ was introduced in (28). However, we have seen in (27) that

$$x_{\max} - x_0 = \psi(t^*). \tag{36}$$

Because ψ is strictly increasing,

$$\psi(t^*) < \psi(t^k) \Leftrightarrow t^* < t^k, \tag{37}$$

and this equivalence ensures existence and uniqueness of $t^{k+1} \in]0, t^k[$ for $t^k > t^*$.

To prove that $t^{k+1} > t^*$, we resort to the auxiliary function

$$\Phi_k(\theta) = t^k \alpha(\theta) - \int_0^{t^k} \alpha(\tau) d\tau. \tag{38}$$

Since α is strictly increasing, Φ_k is strictly increasing too. Thus, we simply have to prove that

$$\Phi_k(t^*) < \Phi_k(t^{k+1}). \tag{39}$$

This amounts to

$$t^k\alpha(t^*) - \int_0^{t^k} \alpha(\tau)\, d\tau < x_{\max} - x_0 \tag{40a}$$
$$= t^*\alpha(t^*) - \int_0^{t^*} \alpha(\tau)\, d\tau, \tag{40b}$$

where the last equality is due to (27). Now, (40b) can be re-written under the form

$$(t^k - t^*)\alpha(t^*) < \int_{t^*}^{t^k} \alpha(\tau)\, d\tau, \tag{41}$$

which holds true because α is strictly increasing.

Convergence. Since $\{t^k\}$ is decreasing and bounded from below by t^*, it has a limit $\tilde{t}$. Passing to the limit in the recursion (30), we get the desired result, namely $\tilde{t} = t^*$. ■

V. Analysis for an elastic rope

In this section, the notations are switched to $s \equiv x$, $z(s) \equiv y(x)$, $S \equiv 1$ and $Z_k \equiv Y_k$. Consider the control problem

$$(\mathscr{E}) \qquad \min_{u(\cdot)\in\mathbb{R}} \int_0^1 [\tfrac{1}{2}u^2(x) + gy(x)]\,dx \tag{42}$$

subject to

$$y'(x) = u(x) \tag{43a}$$
$$y(0) = 0 \tag{43b}$$
$$y(1) = 0 \tag{43c}$$
$$y(x) \geq -h. \tag{43d}$$

This problem corresponds to an elastic rope fixed at its endpoints and bending under a uniform gravity force g. The integrand $\frac{1}{2}u^2 + gy$ represents the potential energy to be minimized in order to find the equilibrium vertical position $y(x)$ as a function of abscissa x, under the constraint due to the level $-h < 0$ of the floor on which the rope can lie.

Proposition 5.1: The exact solution of $(\mathscr{E})$ is given by the following rule.

1) If $h > \frac{1}{8}g$, then the equilibrium position is

$$y(x) = -\frac{1}{2}g\,x(1-x). \tag{44}$$

In this case, the constraint $y(x) \geq -h$ is never active.

2) If $h \leq \frac{1}{8}g$, let

$$x^* = \sqrt{\frac{2h}{g}}. \tag{45}$$

Then, for $x \in [0, x^*]$, the equilibrium position is

$$y(x) = -\sqrt{2hg}\,x + \frac{1}{2}gx^2. \tag{46}$$

Proof: The proof follows the same steps as in the previous section, first by considering the unconstrained case, then by expressing the matching condition at the contact point. We refer readers to [4] for further details. ■

We now address the question of convergence for SCOP. It is obvious that if $h > \frac{1}{8}g$, then once $(\mathscr{Q}_0)[x_0 = 1, Y_0 = 0]$ has been solved, the trajectory $y_0(\cdot)$ is the optimal one (44) and the algorithm stops.

Theorem 5.1: For $h \leq \frac{1}{8}g$, the sequence $\{x_k\}_{k\geq 0}$ produced by SCOP is well-defined, strictly decreasing and converges to x^* as $k \to \infty$.

This sequence is governed by the explicit recursion

$$x^{k+1} = \frac{1}{2}\left[x_k + \frac{2h}{gx_k}\right] \tag{47}$$

and can be interpreted as a fixed point approximation to the continuous relation

$$x^* = \frac{1}{2}\left[x^* + \frac{2h}{gx^*}\right] \tag{48}$$

that is a consequence of (45).

Proof: Except for $Y_0 = 0$, we have $Y_k = -h$ for $k \geq 1$, as long as the algorithm goes on. As a consequence of Pontryagin's principle (9)–(10), the optimal trajectory is of the form

$$y_k(x) = -p_k x + \frac{1}{2}gx^2 \tag{49}$$

at the k-th step. By construction of Algorithm 1, we have

$$y_k(x_k) = -h \tag{50a}$$
$$y_k'(x_{k+1}) = 0. \tag{50b}$$

Plugging (49) into (50) leads to

$$-h = -p_k x_k + \tfrac{1}{2}gx_k^2 \tag{51a}$$
$$0 = -p_k + gx_{k+1}, \tag{51b}$$

from which we deduce (47) by eliminating p_k. It is now an easy algebra exercise to check that the sequence $\{x_k\}_{k\geq 0}$ is decreasing and converges to x^*. ■

It is remarkable that the sequence (47) is the classical Babylonian algorithm that computes the square-root of the positive real number $2h/g$. Fig. 1 and 2 display the SCOP results compared to those obtained with a multiple shooting algorithm which needs to know the active constraints.

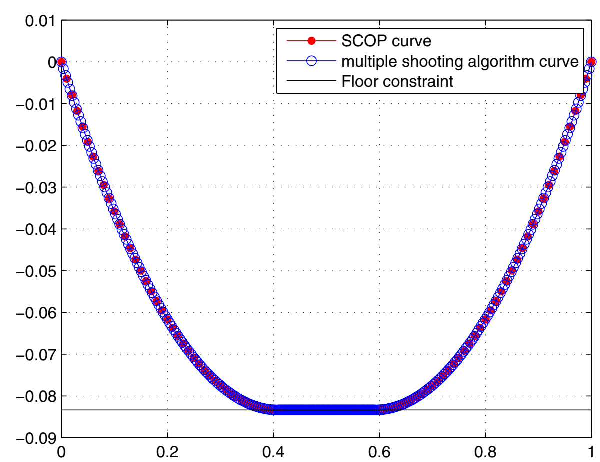

Fig. 1. Elastic line trajectory constrained by the floor level

The two trajectories superimpose perfectly. Fig. 2 illustrates that SCOP retrieves the exact position of x^*.

VI. Numerical results for a realistic hybrid vehicle

In this section, SCOP algorithm is applied to a realistic model of a hybrid vehicle. The engine is characterized by its static fuel consumption map, depending on engine speed ω and delivered torque T_{eng}. For this application, a quadratic polynomial is used

$$L(\omega, T_{\text{eng}}) = \sum_{i,j=0}^{2} K_{ij}\omega^i T_{\text{eng}}^j.$$

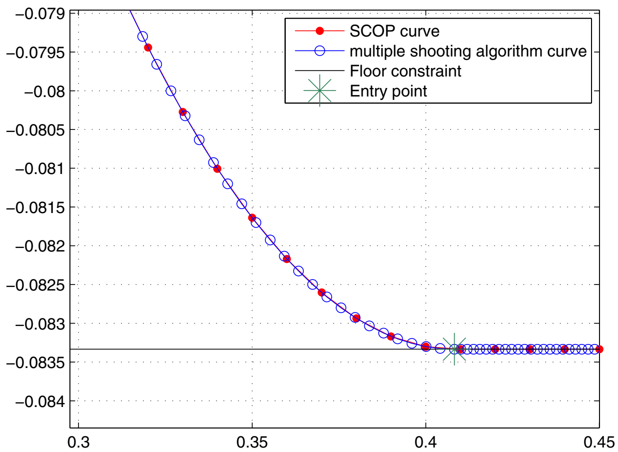

Fig. 2. Elastic line trajectory constrained by the floor level (zoom)

The electric motor is characterized by its torque $T_{\rm mot}$, and is fed by a battery. The state variable x, the state of charge of the battery, is defined by

$$\dot{x}(t) = -\frac{\omega(t)T_{\rm mot}(t)K'}{U_{\rm batt}(t)n_{\rm capa}} = -K\omega(t)T_{\rm mot}(t), \tag{52}$$

with $U_{\rm batt}$ the battery voltage, assumed to be constant, $\omega(t)$ the electric motor speed, K' a scaling constant, and $n_{\rm capa}$ the nominal capacity of the battery. Then the state x depends on the electric power.

In this problem, a prescribed vehicle cycle is imposed (the Artemis Urban cycle for instance, see [7] for more details), and supplies the speed $\omega(t)$ and torque $T_{\rm rq}(t)$ of the vehicle. We assume that both the engine and the electric motor have the same rotation speed ω, and that the requested torque $T_{\rm rq}$ must be provided according to $T_{\rm rq} = T_{\rm eng} + T_{\rm mot}$.

Let us introduce the torque split control $u(t)$, defined as

$$u(t)T_{\rm rq}(t) = T_{\rm eng}(t) \tag{53a}$$
$$(1-u(t))T_{\rm rq}(t) = T_{\rm mot}(t). \tag{53b}$$

Because of maximum and minimum electric motor torque (during battery regeneration), and of maximum engine torque, the control $u(t)$ is constrained by values $u_{\min}$ and $u_{\max}$ depending on the speed $\omega(t)$.

The resulting optimal control problem is thus similar to problem (1–2) with

$$L(u(t),t) = \sum_{i,j=0}^{2} K_{ij}\omega^i(t)T_{\rm rq}^j(t)u^j(t) \tag{54}$$

and

$$f(u(t),t) = -K\omega(t)T_{\rm rq}(t)(1-u(t)). \tag{55}$$

We apply SCOP algorithm with the following expression of u_k^*, solution of $(\mathscr{Q}_k)$, deduced from (9):

$$u_k^*(t) = -\frac{\displaystyle\sum_{i=0}^{2} K_{i1}\omega^i(t) + p_k K\omega(t)}{\displaystyle 2\sum_{i=0}^{2} K_{i2}\omega^i(t)T_{\rm rq}(t)}. \tag{56}$$

Fig. 3 shows for prescribed driving cycle (Artemis Urban cycle) (a) the requested speed of the vehicle, (b) the corresponding engine speed (depending on the vehicle characteristics), (c) the requested torque, that can be provided by the engine, or by the electric motor, or both.

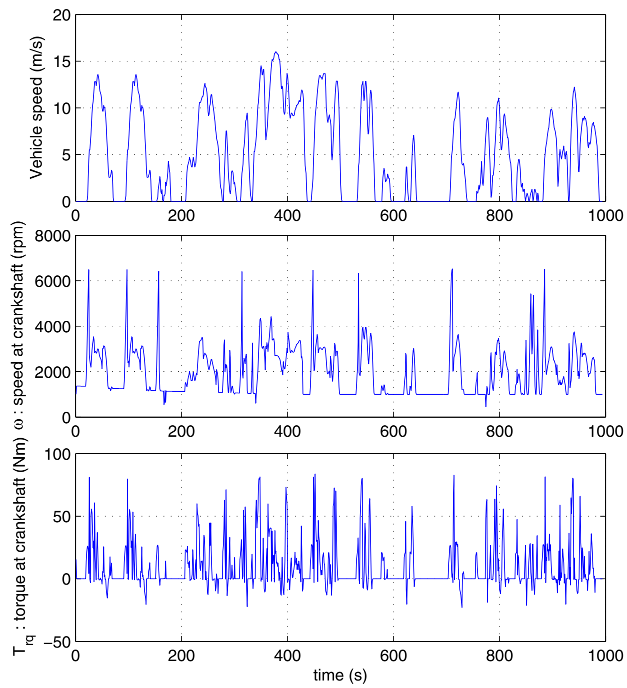

Fig. 3. Vehicle speed requested by Artemis Urban cycle (top) / requested engine speed (middle) / requested torque (bottom)

Fig. 4 shows a comparison between the state trajectory obtained with SCOP, and those obtained with a classical dynamic programming algorithm.

The trajectories are very close to one another. However, the computational times are not the same: the dynamic programming algorithm needs 135s for $dx = 0.5$, while SCOP needs only about 1s to obtain the optimal trajectory. One can also notice that the dynamic programming trajectory converges to SCOP trajectory, as the state step size decreases and tends to 0.

An interesting result is that the analytic expression of the optimal trajectory (56) can be used to build a real-time control strategy parameterized by p (see [1], [2], [9]). For instance, an efficient control is ECMS (Equivalent fuel Consumption Minimization Strategy), where a parameter needs to be defined to determine the equivalence between fuel consumption and battery energy. This parameter is proportional to p.

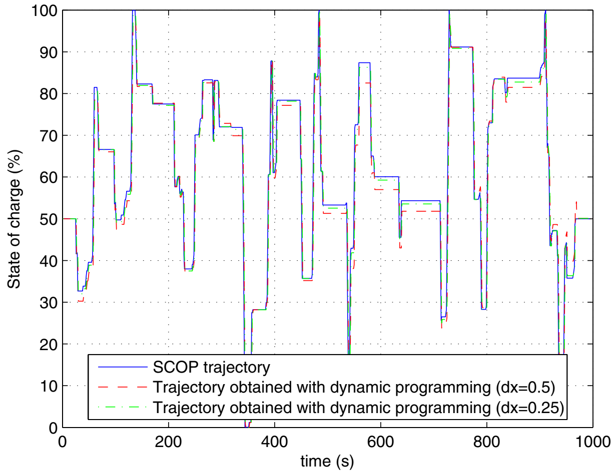

Fig. 4. Optimal state trajectories obtained with DP and SCOP (empty battery : $x_{\min} = 0$, full battery : $x_{\max} = 100$).

VII. CONCLUSIONS AND FUTURE WORKS

In this paper, we have proposed an original method to solve optimal control problems with state constraints. A proof of convergence, in some particular cases, has been established. Although the proof of convergence needs some strong assumptions, SCOP has been successfully applied to more general cases.

The first application is the energy minimization of a rope fixed at its endpoints. This problem can also be solved by using a multiple shooting algorithm, however for this algorithm the shape of the trajectory (with number of junction points) needs to be known.

The second presented application is the optimization of fuel consumption of a hybrid vehicle. In this problem, it is often impossible to know the number of active constraints and when a state constraint becomes active. Classically a dynamic programming (DP) algorithm is used to solve this kind of problem. SCOP allows to retrieve the optimal trajectory of state-constrained optimal problems in a very small computational time, about 100 times faster than DP. Thus the presented applications of SCOP illustrate its potentiality compared to classical methods. However, the range of applicability of the method remains to be clarified.

VIII. ACKNOWLEDGMENTS

The authors wish to thank Pierre Rouchon and Audrey Hermant for several stimulating discussions.

REFERENCES

[1] G. Rousseau, D. Sinoquet, and P. Rouchon, Constrained Optimization of Energy Management for a Mild-Hybrid Vehicle, *Oil & Gas Science and Technology - Rev. IFP*, Vol. 62, pp. 623-634, 2007

[2] L. Guzzella and A. Sciarretta, *Vehicle Propulsion Systems: Introduction to Modeling and Optimization*, Springer; 2005.

[3] J. F. Bonnans and A. Hermant. Well-Posedness of the Shooting Algorithm for State Constrained Optimal Control Problems with a Single Constraint and Control, *SIAM Journal on Control and Optimization*, 46(4),1398-1430, 2007.

[4] J. F. Bonnans and A. Hermant. Stability and Sensitivity Analysis for Optimal Control Problems with a First-order State Constraint, to appear.

[5] R.F. Hartl, S.P. Sethi, and R.G. Vickson. A survey of the maximum principles for optimal control problems with state constraints. *SIAM Review*, 37:181-218,1995.

[6] A. E. Bryson and Y. Ho, *Applied Optimal control*, Hemisphere Publishing Corp., Washington, 1975.

[7] M. André, The ARTEMIS European driving cycles for measuring car pollutant emissions, *Science of The Total Environment*, Vol. 334-335, pp. 73-84

[8] D.P. Bertsekas, *Dynamic programming and optimal control*, Athena scientific, Belmont, 2001 Vol. 334-335, pp. 73-84

[9] A. Sciarretta and L. Guzzella and M. Back, 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

[10] F. Bonnans and P. Rouchon, *Commande et optimisation de systèmes dynamiques* , Éditions de l'École polytechnique, 2005,