Using switching detection and variational equations for the shooting method

Pierre Martinon[1,*], Joseph Gergaud[1]

[1] *martinon,gergaud@enseeiht.fr, ENSEEIHT-IRIT, UMR CNRS 5505*

This is a preprint of an article submitted for publication in Optimal Control Applications and Methods, Copyright (c) 2006 John Wiley & Sons, Ltd.

ACKNOWLEDGEMENT

The authors wish to thank Professor Ernst Hairer (Department of Mathematics, University of Geneva) for his advices and suggestions, as well as the reviewers for their remarks and questions.

SUMMARY

We study in this paper the resolution by single shooting of an optimal control problem with a bang-bang control involving a large number of commutations. We focus on the handling of these commutations regarding the precise computation of the shooting function and its Jacobian. We first observe the impact of a switching detection algorithm on the shooting method results. Then, we study the computation of the Jacobian of the shooting function, by comparing classical finite differences to a formulation using the variational equations. We consider as an application a low thrust orbital transfer with payload maximization. This kind of problem presents a discontinuous optimal control, and involves up to 1800 commutations for the lowest thrust. Copyright © 2000 John Wiley & Sons, Ltd.

KEY WORDS: Shooting method, Jacobian evaluation, switching detection, variational equations

1. Context

The context of this work is the numerical resolution, by a single shooting method, of an optimal control problem with a discontinuous (bang-bang) optimal control. This large number of commutations causes two main problems. First, how to determine the correct control structure, i.e. the number and location of commutations. This matter, which is not the object of the present paper, was solved in our case by the use of continuation methods, as recalled briefly

*Correspondence to: ENSEEIHT-IRIT, UMR CNRS 5505, 2 rue Camichel, 31000 Toulouse, France

Contract/grant sponsor: This work was partially funded by the CNES†, contract R&T 02/CNES/0257/00

in Section 1.2.

Then, these discontinuities cause numerical difficulties with the shooting function, and require careful handling if one wants to solve the problem precisely. We focus here on two points, namely the computation of the shooting function itself, and of its Jacobian required for the shooting method. The first point involves the detection of commutations during the integration of the initial value problem needed to compute the shooting function. The second point deals with the use of variational equations to compute the Jacobian, compared to traditional finite differences.

These points are illustrated with numerical experiments on a low thrust orbital transfer with minimization of the fuel consumption, which can present up to 1800 commutations.

1.1. Shooting method

We begin with a brief presentation of the single shooting method, which is part of the *indirect* methods, and is based on Pontryagin's Maximum Principle (we refer readers interested in these methods to [17, 8, 4, 19, 7] for instance). We recall that *direct* methods, on the other hand, typically involve the partial or total discretization of the problem, and then use various approaches (SQP and interior point techniques for instance) to solve the resulting optimization problem. Direct methods are thus supposed to be robust, but the counterpart is a relatively low precision, and a huge problem size depending on the discretization stepsize used. This makes these methods ill-suited to some particular cases, such as the problems studied here, which present a bang-bang control structure with a huge number of commutations.

Back to the indirect methods, single shooting consists in finding a zero of the *shooting function* associated with the original problem. There is no discretization, even if the method still involves an integration of the system in some way. It is a fast and high precision method, but often requires a good initial guess: as they typically consist in applying a Newton solver to the shooting function, the convergence radius may be quite small, depending on the problem. This is particularly true for problems involving commutations, or worse, state constraints or singular arcs (both are not in the scope of this paper), as it is often necessary to have a good knowledge of the structure of the optimal control. In our case, we use a continuation approach to obtain a suitable initial point, as described later.

We consider a general optimal control problem in the Bolza form

$$(P)\begin{cases} Min\ g(t_0, x(t_0), t_f, x(t_f)) + \int_{t_0}^{t_f} l(t,x,u)\ dt & Objective \\ \dot{x} = f(t,x,u) & Dynamics \\ u \in U & Admissible\ Controls \\ \psi_0(t_0, x(t_0)) = 0 & Initial\ Conditions \\ \psi_1(t_f, x(t_f)) = 0 & Terminal\ Conditions \end{cases}$$

Notation/Remark: for clarity, the time t will often be omitted in the formulas, except in the ambiguous cases.

We use here and in all the following the notations: $x \in \mathbf{R}^n$ for the state, $u \in \mathbf{R}^m$ for the control, U the compact convex set of admissible controls, and f for the state dynamics. We

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

assume that U is a compact convex subset of $\mathbf{R}^m$ and that the initial and final times t_0 and t_f are fixed. We introduce the costate p, of same dimension as the state x, and define the **Hamiltonian** by

$$\mathcal{H}(t,x,p,u) = l(t,x,u) + (p|f(t,x,u)).$$

Remark: we thus assume in all the following that we are in the ***normal*** *case, meaning that the costate p_0 associated to the integral objective l is non zero, and can be made equal to 1, by dividing the whole costate p by p_0...*

The Pontryagin's Maximum Principle then states that, under the assumptions:
• $\exists$ (x,u) feasible for (P), with x absolutely continuous and u measurable.
• f and l are continuous with respect to u and C^1 with respect to t and x.
• g, ψ_0, ψ_1 are C^1 with respect to x.

Let $(\overline{x}, \overline{u})$ be an optimal pair for (P), then

(i) $\exists\ \overline{p} \neq 0$ absolutely continuous such that we have the Hamiltonian system

$$\begin{cases} \dot{\overline{x}} = \frac{\partial \mathcal{H}}{\partial p}(t,x,p,u) \\ \\ \dot{\overline{p}} = -\frac{\partial \mathcal{H}}{\partial x}(t,x,p,u) \end{cases}$$

(ii) $\overline{u}$ is solution of $Min_{w\in U}\mathcal{H}(t,x,p,w)$ ae in $[t_0,t_f]$.

(iii) "Transversality conditions": $\exists$ $(\overline{\mu}_0, \overline{\mu}_1)$ such that

$$(TC)\begin{cases} \psi_0(x(t_0)) = 0 \\ \overline{p}(t_0) = -\frac{\partial \Phi}{\partial x_0}(t_0, \overline{x}(t_0), t_f, \overline{x}(t_f), \overline{\mu}_0, \overline{\mu}_1) \\ \\ \psi_1(x(t_f)) = 0 \\ \overline{p}(t_f) = \frac{\partial \Phi}{\partial x_f}(t_0, \overline{x}(t_0), t_f, \overline{x}(t_f), \overline{\mu}_0, \overline{\mu}_1) \end{cases}$$

with

$$\Phi : (t_0, x_0, t_f, x_f, \mu_0, \mu_1) \mapsto g(t_0,x_0,t_f,x_f) + (\psi_0(t_0,x_0)|\mu_0) + (\psi_1(t_f,x_f)|\mu_1)$$

Remark: this is why indirect methods are sometimes referred to as necessary condition *methods.*

Now we denote $y = (x,p)$ and φ the state-costate dynamics derived from the Hamiltonian system. We assume here that the expression of the optimal control given by the necessary conditions (Hamiltonian minimization) is actually a function, noted γ, ie $\overline{u(t)} = ArgMin_{w\in U}\mathcal{H}(t,x,p,w) = \gamma(t,y)$. Solving (P) is equivalent to solving the following Boundary Value Problem[‡]

$$(BVP)\begin{cases} \dot{y} = \varphi(t,y,\gamma(t,y)) & ae\ in\ [t_0,t_f] \\ c_0(t_0,y(t_0)) = 0 & Boundary\ Conditions\ at\ t_0 \\ c_1(t_f,y(t_f)) = 0 & Boundary\ Conditions\ at\ t_f \end{cases}$$

[‡]or more precisely, Two Point Boundary Value Problem $(TPBVP)$, as the boundary conditions apply only at t_0 and t_f.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

Note: these Boundary Conditions c_0 and c_1 correspond to the Transversality Conditions mentioned above, that contain the Initial and Terminal conditions of (P) in addition to the constraints on the costate p.

It is possible to integrate $y = (x, p)$ if we set the value of $y(t_0)$, and we then obtain the following Initial Value Problem

$$(IVP) \begin{cases} \dot{y} = \varphi(t, y, \gamma(t, y)) & ae\ in\ [t_0, t_f] \\ y(t_0) = \zeta & Initial\ Value \end{cases}$$

We introduce now an application called the **shooting function**, which basically maps the initial value ζ to the value of the Boundary conditions at t_f for the corresponding solution of (IVP). In practice, the initial conditions ψ_0 of the problem (P) already give a part of $y(t_0)$, so the unknown of the shooting function is reduced to the "missing" part, that we note z. A frequent situation is when the initial conditions determine the initial state $x(t_0)$, therefore z is actually the initial costate $p(t_0)$. Then the value of the shooting function is given by the boundary conditions at t_f for the solution $y(\cdot, z)$ of (IVP) corresponding to the initial value $y(t_0, z) = (x_0, z)$

$$(Shooting\ function) \qquad S : z \mapsto c_1(y(t_f, z)).$$

Finding a zero of the shooting function S is then equivalent to the resolution of (BVP), and therefore also gives a solution of (P). The "shooting method" thus consists in solving the equation $S(z) = 0$, as summarized below:

Optimal Control Problem (P)	$\longrightarrow$	Boundary Value Problem (BVP)	$\longrightarrow$	Initial Value Problem (IVP) and Shooting function S

We consider now the discontinuous case, when the optimal control presents switchings (or commutations). More precisely, the Hamiltonian minimization gives

$$\begin{array}{lll} u = \gamma_1(t, y) & , & if\ \psi(t, y) < 0 \\ u = \gamma_2(t, y) & , & if\ \psi(t, y) > 0 \\ u = \Gamma(t, y) & , & if\ \psi(t, y) = 0 \end{array}$$

with $\Gamma(t, y)$ a subset of U, and ψ the switching function, whose zeros correspond to the commutations of the optimal control. We assume that the set of switching times t such that $\psi(t, y) = 0$ is finite. We then obtain $y(\cdot, z)$ as solution of the initial value problem

$$(IVP)_{disc} \begin{cases} \dot{y} & = \varphi_1(t, y) & , & if\ \psi(t, y) < 0 \\ \dot{y} & = \varphi_2(t, y) & , & if\ \psi(t, y) > 0, \\ y(t_0, z) & = (x_0, z) & & \end{cases}$$

We would like here to highlight some difficulties related to the single shooting method in presence of a large number of commutations, namely the evaluation of the shooting function and its Jacobian.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

1.2. The orbital transfer problem

We study here an orbital transfer problem, originally submitted by CNES[§]. We consider a satellite with a mass of 1500 kg and low thrust electro-ionic propulsion (with thrusts ranging from 10 Newtons to 0.1 Newton). We want to transfer it from a strongly elliptic, slightly inclined orbit, to a circular, equatorial, geostationary orbit. The objective is to maximize the payload, i.e., to minimize the fuel consumption during the transfer (NB: this is not a minimum-time problem). This kind of problems was studied for instance in [5, 16], and this specific family of problems in [9, 10, 18, 23].

Model

We consider that the forces applied to the satellite are the Earth attraction (central force) and the engine thrust:

$$\ddot{r} = -\mu \frac{r}{|r|^3} + \frac{T}{m}$$

with r the position vector in $\mathbf{R}^3$, T the thrust also in $\mathbf{R}^3$ (ie the control), m the mass of the satellite, μ the gravitational constant of the Earth ($\mu = \mathcal{G} m_T$, with $\mathcal{G}$ the universal gravitational constant and m_T the mass of the Earth), and $|\cdot|$ the Euclidean norm.

State

For a better numerical stability, we use orbital parameters instead of Cartesian coordinates to define the state variables, to avoid the strong oscillations that would result from the high number of revolutions. These parameters define the shape of the orbit and the position of the satellite on it, as illustrated on Figure 1 (where w is the true anomaly, Ω_n the ascending node longitude, ω the argument of perigee and i the inclination with respect to the equatorial plane). More precisely, the new state variables are:

- Orbit parameter P
- Eccentricity vector (e_x, e_y), in the orbit plane, oriented towards perigee
- Rotation vector (h_x, h_y), in the equatorial plane, collinear to the intersection of orbit and equatorial planes
- True longitude $L = \Omega_n + \omega + w$
- Mass m

The eccentricity and rotation vectors are given by (e is the orbit eccentricity):

$$\begin{array}{lll} e_x = e\ cos(\Omega+\omega) & , & e_y = e\ sin(\Omega+\omega) \\ h_x = tan(i/2)\ cos\Omega & , & h_y = tan(i/2)\ sin\Omega \end{array}$$

Control

The normalized three-dimensional control u such that $T = T_{max}\, u$ is expressed in the moving reference frame attached to the satellite, as radial thrust q, transverse thrust s and normal

[§]Centre National d'Etudes Spatiales

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

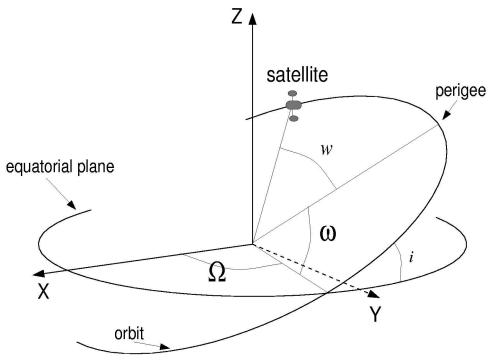

Figure 1. Orbital parameters.

thrust w, as shown on Figure 2.

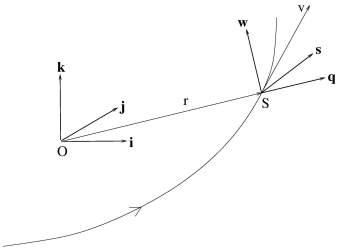

Figure 2. Three-dimensional control.

Optimal Control Problem

If we note $x = (P, e_x, e_y, h_x, h_y, L)$, with T_{Max} the maximal thrust we have the following formulation of the problem:

$$(P)\begin{cases} Min \int_{t_0}^{t_f} |u(t)|\ dt & \\ \dot{x} = f_0(x) + \frac{T_{Max}}{m} B(x)u & \dot{m} = -\beta T_{Max}\ |u| \\ |u| \leq 1 & \\ x(t_0) = (11625, 0.75, 0, 0.0612, 0, \pi) & m(t_0) = 1500 \\ x(t_f) = (42165, 0, 0, 0, 0, free) & m(t_f)\ is\ free \\ t_0 = 0 & \\ t_f\ is\ fixed & \end{cases}$$

with

$$f_0(x) = \sqrt{\frac{\mu}{P}} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & \frac{W^2}{P} \end{pmatrix}^T$$

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

and

$$B(x) = \sqrt{\frac{P}{\mu}} \begin{pmatrix} 0 & 2P/W & 0 \\ sinL & cosL + (e_x + cosL)/W & -Ze_y/W \\ -cosL & sinL + (e_y + sinL)/W & Ze_x/W \\ 0 & 0 & (CcosL)/(2W) \\ 0 & 0 & (CsinL)/(2W) \\ 0 & 0 & Z/W \end{pmatrix}$$

where

$$W = 1 + e_x cosL + e_y sinL \quad , \quad Z = h_x sinL - h_y cosL \quad , \quad C = 1 + h_x^2 + h_y^2$$

The physical constants μ and β are respectively $398600.47 km^3 s^{-2}$ and $0.05112 km^{-1} s$.

Remark: the presence of the norm of the control in the mass dynamic would lead to a non differentiable problem with a direct method.

Switching function and Control

We note p the costate associated to x, and p_m the costate associated to m, and define the **switching function** ψ:

$$\psi(x, m, p, p_m) = 1 - \beta T_{max} p_m \ - \ \frac{T_{Max}}{m} |B(x)^t p|.$$

We now make two assumptions (that we numerically check, see [23]):
- $B(x)^t p$ is non-zero on $[t_0, t_f]$
- There is no **singular arc**, ie ψ does not vanish on any finite interval.

The application of Pontryagin's Maximum Principle then leads to the following expression of the optimal control

$$\begin{cases} u = -\frac{B(x)^t p}{|B(x)^t p|} & if \ \psi(x, m, p, p_m) < 0 \\ u = 0 & if \ \psi(x, m, p, p_m) > 0 \end{cases}$$

We can see that this control has a bang-bang structure, as its norm switches between 0 and 1 at zeros of the switching function ψ. The two cases define the two dynamics φ_1 and φ_2.

Shooting function

In this context, we can prove that the shooting function S is smooth on a suitable open subset of $\mathbf{R}^n$. More precisely, if we consider a security zone, corresponding to the fact that the satellite does not come too close to Earth, and does not use all of its fuel (*note: the following definitions use the Cartesian coordinates, the state being decomposed as (position,velocity,mass)*):

$$Let \ A = \{x = (r, v, m) \in \mathbf{R}^7 | r > \rho_0 > 0 \ , \ m > M_0 > 0\}$$

where ρ_0 is a safety minimal altitude and M_0 the minimal final mass (for instance the mass of the satellite without fuel).

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

Let us now define

$$\begin{array}{lll} \Omega = \{z \in \mathbf{R}^n | & x(t,z) \in A, \quad \forall t & \\ & p_v(t,z) \neq 0, \quad \forall t & \\ & \psi(y(t,z))^2 + (p_r(t,z)|p_v(t,z))^2 \neq 0, \quad \forall t & \\ & \psi(y(0,z)) \neq 0 \; and \; \psi(y(t_f,z)) \neq 0 & \} \end{array}$$

Then we have
i) for $z \in \Omega$, $t \mapsto \psi(y(t,z))$ is continuously differentiable, and finitely many switching times t_i exist such that $\psi(y(t_i,z)) = 0$.
ii) for every commutation, the flow is transverse to the commutation surface, and goes through the surface
iii) Ω is an open subset of $\mathbf{R}^n$ ans S is C^∞ on Ω.

Proof. see [9].

Continuation approach

It is well known that shooting methods are quite sensible to the initial guess, and it is indeed the case for this family of problems. We therefore use continuation approaches to obtain a suitable starting point (we refer readers interested in continuation methods to [1, 2, 3]). The principle is to find a suitable homotopy that connects the problem to an easier one, and then to follow the zero path of this homotopy, from the easy problem to the original one. In our case, we choose for the "easy" problem the same transfer with minimization of an "energy" criterion, namely $\int_{t_0}^{t_f} |u(t)|^2 \, dt$. For our homotopy, we take the shooting function corresponding to the orbital transfer with the following objective, parametrized by $\lambda \in [0,1]$:

$$J_\lambda = \int_{t_0}^{t_f} \lambda |u(t)| + (1-\lambda)|u(t)|^2 \, dt.$$

The resulting perturbed problem (P_λ) has a strictly convex Hamiltonian (with respect to u), with a continuous optimal control, and is much easier to solve than $(P) = (P_1)$. The path following requires some precautions, but this approach provided us with a good initial point for the single shooting method, with **no a priori assumption** on the structure of the optimal control, see [9, 10, 23, 18].

Remark: For a better numerical behavior, we also use longitude instead of time as the integration variable. The resulting problem is non-autonomous, with a fixed final longitude (ie "rendez-vous") and free final time. Along with the choice of the orbital parameters, this leads to a very good numerical stability of the system. This allows us to solve the problem with a single shooting method, without having to resort to multiple shooting.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

2. Methods

Now let us go back to some specific difficulties for the shooting method in this context. First, the integration of $(IVP)_{disc}$ itself can be tricky, and typically requires a good variable step integrator. Then, the use of variable step can hinder the computation of the Jacobian by classical finite differences (cf [6, 11]). We propose to study the behavior of some practical ways to deal with these difficulties.

2.1. Switching detection

First, instead of letting the variable step integrator deal with the discontinuities of the right hand side on its own, we try to detect the switchings during the integration of $(IVP)_{disc}$. The principle of the detection is described below, and assumes that the integration method used is able to provide a "dense output", ie a cheap approximation of y on each integration interval. This is typically done by some polynomial interpolation, see [11] for a thorough description of dense outputs available for many integration methods.

At integration time t, compute the switching function $\psi(t, y)$.

If the sign of ψ has changed since the previous step, a commutation has occurred (actually, an odd number of commutations, but most often one in practice).

Then we locate the commutation by dichotomy, by using the dense output of the integrator to provide y and ψ. Then we perform the control switch, store the sign of ψ, and restart the integration back from the commutation time.

Else the integration goes on normally.

This algorithm is quite simple, but effective. The main issue is the risk of skipping one or several commutations during an integration step. An easy improvement of the detection method is to perform additional sign checks at a fixed number of intermediate points, and not only at the end of the step. If these additional checks bring some new commutations, then the number of intermediate checkpoints should be increased until no new commutations appear. Besides, with a variable step integrator, we can also expect the stepsize control mechanism to reject a step that would have missed commutations. In practice, for the orbital transfer studied in this paper, setting 10 intermediate checkpoints appears to be sufficient to catch all commutations (up to 1800 for the longest transfer with the lowest thrust).

2.2. Jacobian evaluation - variational equations

The first way to compute the Jacobian of the shooting function is simply to use finite differences. We use for the shooting method the nonlinear solver HYBRD (see code references on page 20), which performs finite differences with a step of

$$h_j = \sqrt{\epsilon}|x_j|,$$

where ϵ is the error on the shooting function evaluation (for the numerical experiments, we set ϵ to the relative error of the integrator). This method will be noted "external differentiation"

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

(END) in the following. The drawback of this method is that when used with a variable step integrator, the integration steps can vary at the points where S is evaluated for the finite differences, which can impair the approximation of the Jacobian (see [6], [11] p. 200).

Then, another possibility is to use the variational equations to compute the Jacobian of the shooting function. We first recall the smooth case, when we consider the derivative with respect to the initial condition of the system

$$(IVP)\begin{cases} \dot{y}(t) = \varphi(t, y(t)) \\ y_i(t_0) = y_i^0 \quad \text{for } i = 1, \dots, n \\ y_{i+n} = z_i \quad \text{for } i = 1, \dots, n. \end{cases}$$

If we note $y(\cdot, z)$ the solution of (IVP), we know that $\frac{\partial y}{\partial z_i}(t_f, z)$ is solution of the variational system

$$(VAR)_j\begin{cases} \dot{Y}_j(t) = \frac{\partial \varphi}{\partial y}(t, y(t))Y_j(t) \\ Y_j(t_0) = \begin{pmatrix} 0 & \cdots & 1 & \cdots & 0 \end{pmatrix}^T \end{cases}$$

It is then possible to compute the Jacobian of the shooting function by integrating (VAR) along with (IVP). We will use two methods to compute the right hand side of (VAR). First, we can just obtain $\frac{\partial \varphi}{\partial y}$ by automatic differentiation (see code references on page 20). Another possibility is to approximate the right hand side by finite differences (cf [11] p.201):

$$\dot{Y}(:, j) \approx \frac{1}{h}\left(\varphi(y + hY(:, j)) - \varphi(y)\right)$$

We will use the notations VARad and VARfd to distinguish these two methods.

Remark on the error orders (see [11]): For the external differentiation (END), the error of the finite differences with a step of h on the shooting function is of size $O(Tol/h) + O(h)$, where Tol is the tolerance of the integrator used for (IVP). Therefore, it seems natural to chose $h \approx \sqrt{Tol}$, to obtain an error of size $O(\sqrt{Tol})$.

For the variational formulations, the Jacobian is obtained from the integration of the system (VAR). If we use automatic differentiation for the right hand side (VARad), then the error is of size Tol. If finite differences are used (VARfd), the error is of size $O(h) + O(macheps/h)$, where $macheps$ is the machine precision. Choosing $h \approx \sqrt{macheps}$ gives an error of size $O(\sqrt{macheps})$ on the right hand side of (VAR), for a final error on the Jacobian of size $O(Tol + \sqrt{macheps})$. Thus for $Tol \geq \sqrt{macheps}$, the method VARfd is as precise as VARad.

A simple example: We illustrate these different methods on a simple example:

$$(\mathcal{P}_\varepsilon)\begin{cases} Min \int_0^2 (|u(t)| - \varepsilon(\ln|u(t)| + \ln(1 - |u(t)|)))dt \\ \dot{x}(t) = -x(t) + u(t) \\ |u(t)| < 1 \\ x(0) = x^0 = 0 \\ x(2) = x^f = 0.5 \end{cases}$$

We consider $\varepsilon = 0.01$, and use `Matlab`'s `ode45` to integrate the corresponding (IVP), with a tolerance of $Tol = 10^{-4}$. The solution is $z^* = -0.2715$, and the corresponding trajectory and

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

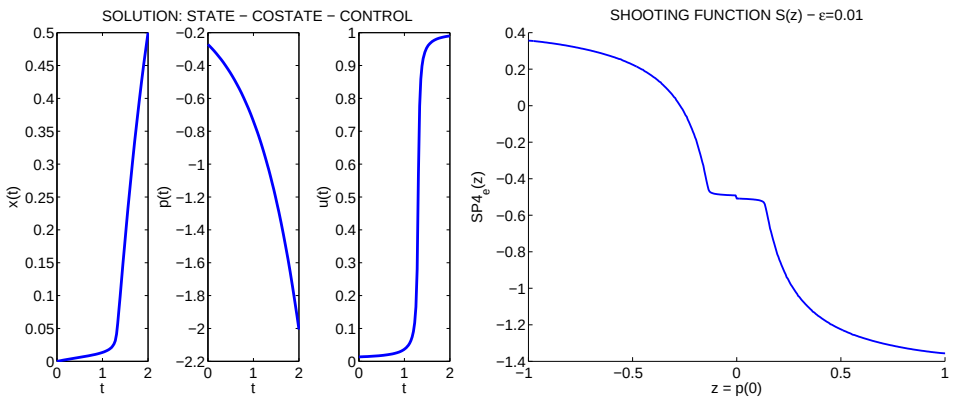

Figure 3. Trajectory and control at the solution $z^* = -0.2715$, and shooting function S on $[-1, 1]$.

control are shown on Figure 3, along with the shooting function values for $z \in [-1, 1]$.

We trace on Figure 4 the derivative of the shooting function $\frac{\partial S}{\partial z}$, as computed by the external differentiation (with a finite difference step of Tol and $\sqrt{Tol}$), and the two variational formulations VARad and VARfd. According to the formulas recalled above, we expect the error on the Jacobian to be of size $O(1)$ and $O(10^{-2})$ for the two END, and $O(10^{-4})$ for both VARad and VARfd. The graphs confirm that the first END ($h = Tol$) is the worst of all, followed by the second END ($h = \sqrt{Tol}$), the two variational methods VARad and VARfd being the most accurate, and of similar precision.

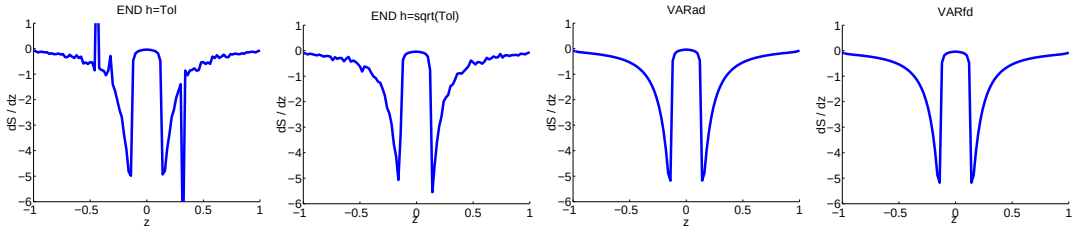

Figure 4. Illustration of the various Jacobian computation methods.

Note: The two peaks observed on S' at $z \approx +/-0.14$ correspond to a strong variation (with respect to z) of the control values near $t_f = 2$, which result in a strong variation of $S(z)$.

2.3. Variational system - Discontinuous case

Let us now describe the use of the variational equations in presence of commutations. For a fixed z, let us denote by $t_0 < t_1 < \ldots < t_q < t_f$ the q switching times, and decompose the shooting function $S(z) = S_q \circ \ldots \circ S_0(z)$, with the applications

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

$$\begin{array}{rcl} S_0 : \Omega \subset \mathbf{R}^n & \longrightarrow & \Omega_1 \subset \mathbf{R} \times \mathbf{R}^{2n} \\ z & \longmapsto & (t_1(z), y(t_1(z), z)), \end{array} \qquad \begin{array}{rcl} S_q : \Omega_q \subset \mathbf{R} \times \mathbf{R}^{2n} & \longrightarrow & \Omega_{q+1} \subset \mathbf{R}^{2n} \\ (t_i, y_i) & \longmapsto & y(t_{i+1}(t_i, y_i), t_i, y_i). \end{array}$$

$$\textit{and} \quad \begin{array}{rcl} S_i : \Omega_i \subset \mathbf{R} \times \mathbf{R}^{2n} & \longrightarrow & \Omega_{i+1} \subset \mathbf{R} \times \mathbf{R}^{2n} \\ (t_i, y_i) & \longmapsto & (t_{i+1}(t_i, y_i), y(t_{i+1}(t_i, y_i), t_i, y_i)), \end{array} \quad i = 1, \ldots, q-1$$

The implicit function theorem gives the derivatives of the switching time $t_{i+1}(t_i, y_i)$

$$t'_{i+1}(t_i, y_i) = \left(\frac{\partial \psi}{\partial t}(t, y(t, t_i, y_i)) + \frac{\partial \psi}{\partial y}(t, y(t, t_i, y_i)).\dot{y}(t, y(t, t_i, y_i)) \right)^{-1}$$
$$\left(\frac{\partial \psi}{\partial y}(t_{i+1}, y(t_{i+1}, t_i, y_i))\ Y_{t_i}(t_{i+1}) \quad \frac{\partial \psi}{\partial y}(t_{i+1}, y(t_{i+1}, t_i, y_i))\ Y_{y_i}(t_{i+1}) \right)$$

Here $Y_{t_i}(.) = \frac{\partial y}{\partial t_i}(., t_i, y_i)$ and $Y_{y_i}(.) = \frac{\partial y}{\partial y_i}(., t_i, y_i)$ are solutions of the variational systems (*we note in the following φ_i the right hand side on $]t_i, t_{i+1}[$, which can be either φ_1 or φ_2.*)

$$(VAR)_{t_i} \begin{cases} \dot{Y}(t) = \frac{\partial \varphi_i}{\partial y}(t, y(t)) Y(t) \\ Y(t_i) = -\varphi_i(t_i, y_i) \end{cases} \textit{ and } (VAR)_{y_i} \begin{cases} \dot{Y}(t) = \frac{\partial \varphi_i}{\partial y}(t, y(t)) Y(t) \\ Y(t_i) = I \end{cases}$$

Note: as the variational system is linear with respect to Y, we have $Y_{t_i} = -Y_{y_i} \varphi_i(t_i, y_i)$, so only Y_{t_i} needs to be integrated in practice.

We then have for S_i

$$S'_i(t_i, y_i) = \begin{pmatrix} \frac{\partial t_{i+1}}{\partial t_i}(t_i, y_i) & \frac{\partial t_{i+1}}{\partial y_i}(t_i, y_i) \\ \frac{\partial t_{i+1}}{\partial t_i}(t_i, y_i) \varphi_i(t_{i+1}, y_{i+1}) + Y_{t_i}(t_{i+1}) & \varphi_i(t_{i+1}, y_{i+1}) \frac{\partial t_{i+1}}{\partial y_i}(t_i, y_i) + Y_{y_i}(t_{i+1}) \end{pmatrix}$$

And for S_0 and S_q, we have

$$S'_0(z) = \begin{pmatrix} \frac{\partial t_1}{\partial z}(z) \\ \varphi_0(t_1(z), y_1(t_1(z), t_0, z)) \frac{\partial t_1}{\partial z}(z) + \frac{\partial y}{\partial z}(t_1(z), t_0, z) \end{pmatrix}$$

and

$$S'_q(t_q, y_q) = \begin{pmatrix} \frac{\partial y}{\partial t_q}(t_f, t_q, y_q) & \frac{\partial y}{\partial y_q}(t_f, t_q, y_q) \end{pmatrix}.$$

This gives the derivatives of the shooting function

$$S'(z) = S'_q \circ \ldots \circ S'_0(z)$$

(note: a similar approach can be found for instance in [15]). As in the smooth case, the partial derivatives in the formulas can be computed either by automatic differentiation, or by finite differences, and we will keep the notations VARad and VARfd in the following.

An improved formulation: However, one might notice that the above formula actually computes $Y_{y_i}(.) = \frac{\partial y}{\partial y_i}(., t_i, y_i)$ on each sub-arc, which implies integrating a variational system of dimension $2n \times 2n$. The point is, we only need the derivatives with respect to the unknown $z \in \mathbf{R}^n$ for the Jacobian of the shooting function. We will see that it is not necessary to compute the full derivatives with respect to y_i, but only half of it, corresponding to the components of z.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

We will now assume for the reasoning that the unknown z corresponds to the n last components of $y = (x, p)$. This means that the unknown of the shooting function is the initial costate, which is often the case from the application of the PMP. In other cases, it would be sufficient to rearrange the components of y to put the n components corresponding to z at the end.

We consider the integration of y on the interval $[0, T]$ from the initial value y_0, with a commutation occurring at $\tau \in]0, T[$. We assume here that the system is autonomous (it is always possible to transform a non-autonomous system into an autonomous one by adding a state component corresponding to the independent variable, of derivative equal to 1), and therefore replace the notation $y(t, t_0, y_0)$ by $y(t - t_0, y_0)$.

Assuming the dynamic is φ_1 before τ and φ_2 afterward, the integrated value of y at T, denoted $I(y_0)$, is given by:

$$I(y_0) = y_2(T - \tau, y_1(\tau, y_0)) \quad , \quad \textit{with } \tau = \tau(y_0) \textit{ determined by } \psi(y_1(\tau, y_0)) = 0.$$

With the notations $y_\tau = y_1(\tau, y_0)$, $y_T = y_2(T - \tau, y_\tau)$, $Y_1(t, y_0) = \frac{\partial y_1(t,y_0)}{\partial y_0}$ and $Y_2(t, y_0) = \frac{\partial y_2(t,y_0)}{\partial y_0}$, we can write the derivatives as

$$I'(y_0) = Y_2(T - \tau, y_\tau)(Y_1(\tau, y_0) + \varphi_1(y_\tau)\tau'(y_0)) - \varphi_2(y_T)\tau'(y_0)$$

where $\tau'(y_0)$ can be obtained like before by the implicit function theorem:

$$\tau'(y_0) = -\frac{\frac{\partial \psi}{\partial y}(y_\tau)Y_1(\tau, y_0)}{\frac{\partial \psi}{\partial y}(y_\tau)\varphi_1(y_\tau)} \qquad (*)$$

If we integrate the variational system until the commutation, taking $T = \tau^+$ in the formula above gives the derivatives at the commutation (as $Y_2(\tau^+ - \tau, y_\tau) = I$)

$$I'(y_0) = Y_1(\tau, y_0) + (\varphi_1(y_\tau) - \varphi_2(y_{\tau^+}))\tau'(y_0)$$

Instead of reinitializing the integrated Y to I, we perform the update

$$Y \leftarrow Y + (\varphi_1(y_\tau) - \varphi_2(y_{\tau^+}))\tau'(y_0).$$

Thus Y is actually equal to $\frac{\partial y}{\partial y_0}(\tau^+, y_0)$, and we have crossed the commutation while keeping the correct derivatives. We can then continue the integration of the variational system until the next commutation, and so on until we reach the prescribed final time. At this point the integrated Y holds the value of $\frac{\partial y}{\partial y_0}(t_f, y_0)$, which immediately gives the Jacobian of the shooting function S.

A first advantage of this second formulation is to get rid of the composition matrix products that appear in $S'(z) = S'_q \circ \ldots \circ S'_0(z)$. Moreover, it is now easy to see that the update formula above is still valid if we replace $Y = \frac{\partial y}{\partial y_0}$ by $Y = \frac{\partial y}{\partial p_0}$ and $\tau'(y_0)$ by $\frac{\partial \tau}{\partial p_0}$. And it is clear that taking $Y = \frac{\partial y}{\partial p_0}$ in $(*)$ precisely gives the value of $\frac{\partial \tau}{\partial p_0}$ instead of $\tau'(y_0)$. Recalling the assumption that the unknown of the shooting function is the initial costate, we therefore only need to integrate the variational system for $Y = \frac{\partial y}{\partial p_0}$, which is of dimension $2n \times n$, half than for the first formulation. This improved formulation gives similar results to the previous one, but is about twice faster, which could be expected. The numerical results presented in the following correspond to this faster algorithm.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

Remark on the complexity: With this second formulation, using finite differences to approximate the right hand side of the variational system has the same complexity as using finite differences to compute the Jacobian of the shooting function.

Finite differences to compute $S'(z)$ require $n+1$ evaluations of S, which corresponds roughly to $(n+1) \times N_{steps}$ evaluations of the dynamic φ, if we assume that the $n+1$ calls to S nearly use the same number of integration steps.

Finite differences to compute the right hand side of the variational system require $n+1$ evaluations of φ, thus leading to the same total complexity if the variational system nearly requires the same number of integration steps then the evaluation of S (ie the integration of $(IVP)_{disc}$).

This is obviously true with a fixed step integrator, but must be taken into account with a variable step integrator, as integrating the larger variational system will usually require more steps than the simple initial value problem for S. This is problematic, as we compute the derivatives of a function that numerically slightly differs from S, the y component being integrated with more steps than when we just compute the value of S. A way to correct this behavior is to force the stepsize control algorithm of the integrator to ignore all the components corresponding to Y, and take only into account the $2n$ components of y. This is easily done in practice if the integrator accepts vector (instead of scalar) tolerances.

3. Numerical results

We examine now the numerical results of the various methods described above, applied to the orbital transfer problem. The numerical experiments in the following use the DOPRI5 integrator from E. Hairer and G. Wanner (see [11]), a 5th order Dormand-Prince embedded Runge-Kutta method with a 4th order dense output.

To begin with, we show on Figure 5 the solution for the transfer with a maximal thrust of 10 Newtons (*Notes: The "+" signs on the state/costate graphs indicate the final conditions*). From left to right, we have the following graphs, with the time as abscissa:
- the state $x \in \mathbf{R}^8$: orbital parameters P, e_x, e_y, h_x, h_y, L, mass m and final time t_f[¶].
- in the middle column, the corresponding costate
- in the right column, the three components (q, s, w) and norm $|u|$ of the control.

It can be seen that the evolution of the state variables is quite smooth, due to the use of orbital parameters instead of Cartesian coordinates, and also to the low thrust of the propulsion. As brutal corrections (like in the impulsional case) are not possible, the cheapest way to move the satellite to its final orbit seems to slowly and regularly correct the orbit along each revolution (more precisely at the apogee, see the trajectories on Figures 6 and 7 below).

[¶]The original problem with free final time was reformulated as a fixed final time one via the usual transformation $t = t_f.s$, with $s \in [0, 1]$. This adds the components t_f and p_{t_f} to the state and costate vectors respectively.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

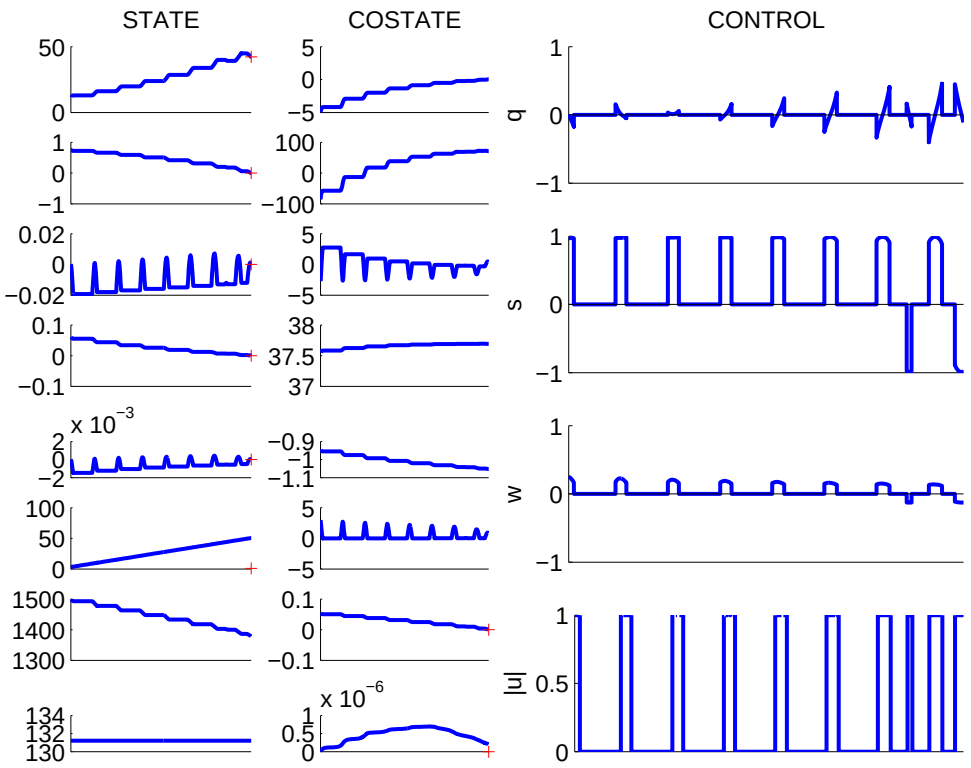

Figure 5. Orbital transfer: solution for a 10N thrust (abscissa is time for all graphs).

In particular, the inclination regularly decreases until the end of the transfer, so it does not seem optimal to correct the inclination first and then proceed with a coplanar transfer, for instance. As expected, the control is bang-bang, with a succession of full thrust and drifting arcs, the commutations being clearly visible on the graph showing the control norm.

We see now on Figure 6 the trajectory projected on the equatorial plane, with the thrust and drifting arcs shown in red (darker) and green respectively. We observe that the thrust arcs are located at the apogees, and also at the last perigees, which may be linked to the fact that at the end of the transfer the orbit is nearly circular, so the distinction between apogee and perigee becomes thin. According to the criterion (consumption minimization), we can assume that thrust arcs occur where corrections to the orbit are the most effective.

For this thrust of 10N, the optimal control presents only 18 switchings, whereas the transfer for 0.1N (on Figure 7) has about 1800 switchings, but with a qualitatively similar trajectory and control structure. It actually happens that this family of problems is extremely regular with respect to the maximal thrust, with several empirical laws that are extremely well verified, most importantly $Thrust(N) \times t_f \approx C^{te}$, see [23] (a similar law was first observed for the minimum time problems, see [22]). A similar law holds for the total number of switchings, and

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

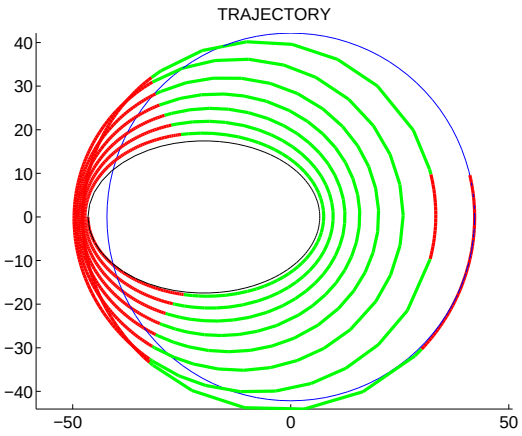

Figure 6. Orbital transfer: trajectory for a 10N thrust.

also for the total number of integration steps (for a given integration method).

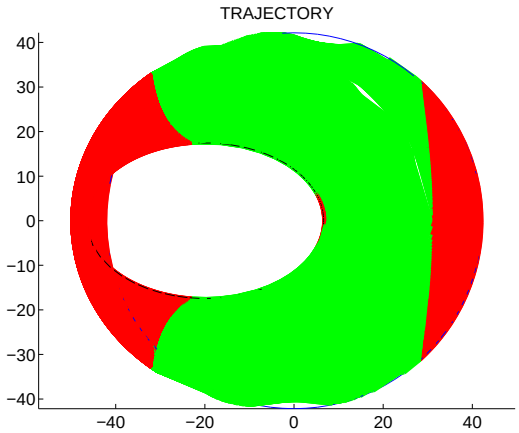

Figure 7. Orbital transfer: trajectory for a 0.1N thrust.

In the following, we will show the numerical results of the switching detection and variational equations methods on this family of problems. Integration is performed by the DOPRI5 code, with absolute and relative tolerances set to $10^{-8}/10^{-6}$. The same tests, conducted with lower $(10^{-6}/10^{-4})$ or higher $(10^{-12}/10^{-10})$ precision, give qualitatively similar results. The only notable difference is that the basic external differentiation, relatively to the other methods, performs less badly with a high precision, and much worse with a low precision.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

3.1. Switching detection

We begin the tests with the switching detection method, by comparing the results of the shooting algorithms with normal integration, and with the detection algorithm described on page 8. In both cases, the Jacobian of the shooting function is computed by the HYBRD solver, by finite differences (external differentiation). Tables I and II show the results of the shootings with and without switching detection. The legend of the columns is:
- T_{Max}: maximal thrust in Newtons (ranges from 10N to 0.1N)
- Nf (Njac): number of shooting function S (resp. Jacobian) evaluations by the HYBRD solver
- Time: computational time in seconds (on a 2.4GHz Celeron)
- $|S|$: norm of the shooting function at the solution
- Objective: objective value (ie fuel consumption) at the solution
- Steps, Rejected, ratio: total integration steps, rejected steps and total/rejected ratio

Table I. Shooting results without switching detection

T_{Max}	Nf (Njac)	Time	$\|S\|$	Objective	Steps	Rejected	ratio
10N	61 (5)	2	$1.87\ 10^{-6}$	121.21	846	411	0.49
5N	72 (5)	7	$4.20\ 10^{-6}$	121.58	1683	813	0.48
1N	59 (4)	30	$1.06\ 10^{-5}$	121.77	8383	4136	0.49
0.5N	102 (7)	92	$5.53\ 10^{-4}$	121.69	16506	8083	0.49
0.2N	48 (3)	116	$2.49\ 10^{-4}$	121.71	41360	20274	0.49
0.1N	75 (5)	327	$2.42\ 10^{-4}$	121.70	82684	40435	0.49

Table II. Shooting results with switching detection

T_{Max}	Nf (Njac)	Time	$\|S\|$	Objective	Steps	Rejected	ratio
10N	24 (1)	1	$1.71\ 10^{-14}$	121.21	167	33	0.20
5N	121 (8)	5	$1.55\ 10^{-7}$	121.58	310	64	0.21
1N	47 (3)	10	$1.59\ 10^{-8}$	121.78	1523	396	0.26
0.5N	98 (7)	41	$9.35\ 10^{-7}$	121.69	2982	765	0.26
0.2N	79 (5)	78	$5.58\ 10^{-5}$	121.71	7615	1909	0.25
0.1N	33 (2)	65	$3.61\ 10^{-7}$	121.70	15455	3995	0.26

First, we observe a better convergence with the detection, as indicated by the norm of the shooting function at the solution, and notice a strong decrease of the execution time, the detection method being two to four times faster than the standard integration. The objective value at the solution is almost identical, and the solution themselves are quite close (the scaled solution vectors are identical up to four digits at 10N, and two digits at 0.1N). We note that the number of function and Jacobian evaluations does not seem to be really affected by the detection.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

The improvement is actually due to the great impact of the detection on the integration steps, with a reduction of the total steps by a factor 5 roughly, and of the rejected steps by a factor 10, the rejected step ratio falling from 50% (!) to 25%. These values are shown on Figure 8, with a logarithmic scale for the number of total steps, and in abscissa the thrust with an inverted logarithmic scale, the shortest transfer (10N) being on the left and the longest (0.1N) on the right. We can check that the empirical relation (for this problem family) $Thrust(N) \times Steps \approx C^{te}$ is numerically well verified. This relation is a consequence of the $Thrust(N) \times t_f \approx C^{te}$ law mentioned previously, and holds for different integration methods (eg the well known RKF45), of course with different constants.

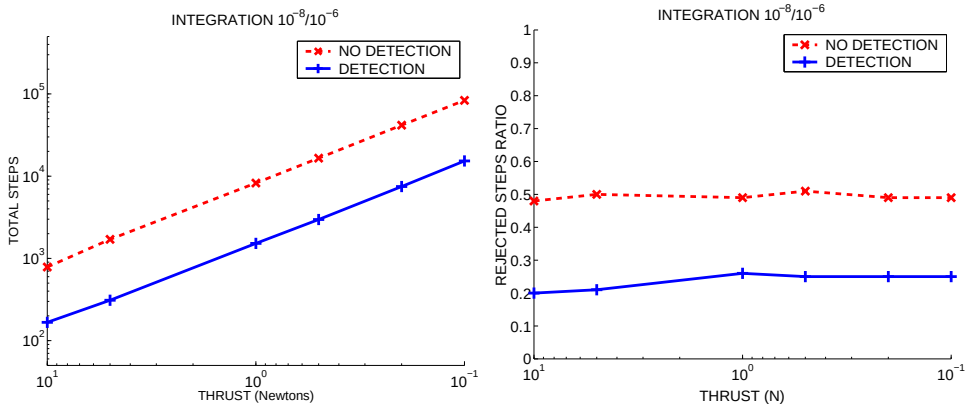

Figure 8. Total integration steps, and rejected steps ratio.

Figure 9 illustrates the integrated trajectory (ie the accepted steps) for the 0.1N transfer, and the reduction of the number of steps is clearly visible. We notice that the steps are more clustered on the "left" and "right" parts of the orbit, which corresponds to the fact that thrust arcs are located around the apogees, and perigees at the end of the transfer. It seems natural that less steps are needed to integrate the drifting arcs in the "middle" of the graphs.

This is confirmed by looking at the integration stepsize for the 10N transfer, as shown on Figure 10 (plain line for standard integration, bold line with switching detection, with a logarithmic scale for the stepsize). In the standard case, we observe three stepsize ranges:
- relatively large steps ($[0.5, 1]$ roughly), corresponding to the drifting (no thrust) arcs.
- smaller steps (about 10^{-1}), corresponding to the thrust arcs at the apogees and last perigees.
- extremely small steps (near 10^{-6}), just before the commutations (a lot of rejected steps also occur here, as the integrator reduces the stepsize until the error due to the commutation crossing is acceptable).

With the switching detection method, we see that the stepsize on the thrust and drifting arcs is nearly unaffected, whereas the very small steps at the commutations have almost disappeared. The switching detection therefore effectively does its job, and clearly improves

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

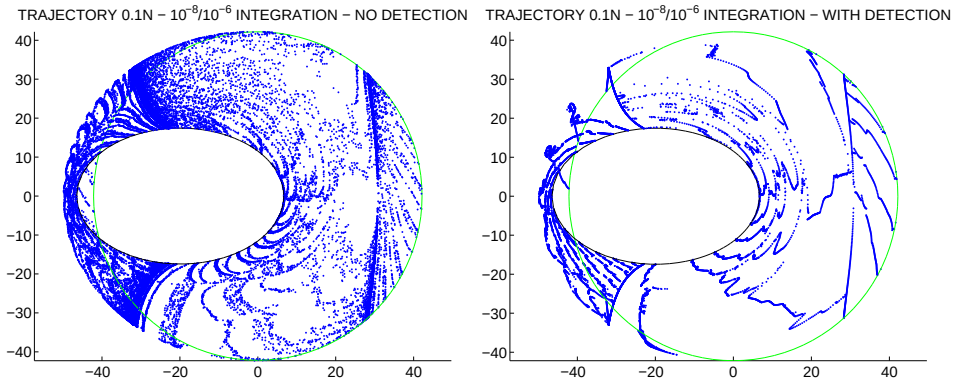

Figure 9. Trajectories for the 0.1N transfer.

the integration of $(IVP)_{disc}$ for the computation of the shooting function. It should be kept in mind that this method's performance probably depends on the dense output of the integrator, which should be cheap and reasonably accurate (for the description of the dense output of DOPRI5 see [11], page 192).

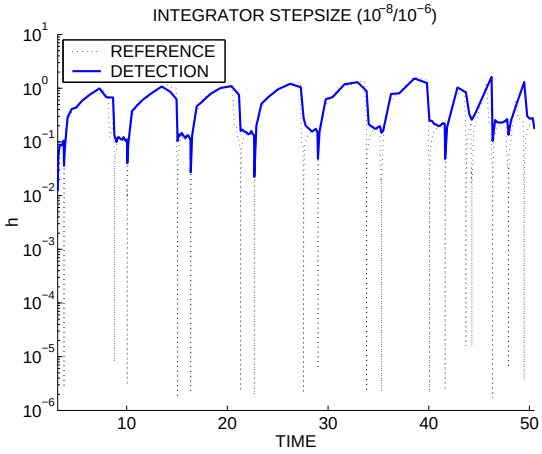

Figure 10. Integration stepsize for the 10N transfer.

3.2. Variational equations

We compare now the two variational formulations VARad and VARfd to the switching detection method ENDc. We recall that both variational formulations actually also include the switching detection algorithm. The initial point and settings are the same as before. For the two variational formulations VARad and VARfd, we use a variant of HYBRD (namely HYBRJ), whose only difference with HYBRD is that it takes the user-supplied Jacobian instead of approximating it by finite differences. We indicate here on Table III the norm of the shooting

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

Table III. Shooting results - Switching detection alone and variational equations

$10^{-8/6}$	$ENDc$		$VARad$		$VARfd$	
T_{Max}	Time	$\|S\|$	Time	$\|S\|$	Time	$\|S\|$
10N	1	$1.71\ 10^{-14}$	1	$3.26\ 10^{-14}$	1	$2.73\ 10^{-14}$
5N	5	$1.55\ 10^{-7}$	5	$6.13\ 10^{-14}$	2	$3.57\ 10^{-13}$
1N	10	$1.59\ 10^{-8}$	12	$2.84\ 10^{-8}$	19	$2.50\ 10^{-8}$
0.5N	41	$9.35\ 10^{-7}$	93	$3.63\ 10^{-6}$	22	$6.89\ 10^{-7}$
0.2N	79	$5.58\ 10^{-5}$	85	$1.21\ 10^{-4}$	56	$6.29\ 10^{-6}$
0.1N	65	$3.61\ 10^{-5}$	114	$4.16\ 10^{-5}$	57	$4.00\ 10^{-5}$

function at the solution and the execution times.

The three formulations converge in all cases, and we note that the convergence is qualitatively similar for ENDc and the VAR methods. At the solutions, the objective value and the rejected step ratios are very close for the three formulations. These tests do not show any significant difference in the number of Jacobian evaluations between the three methods. Concerning the execution times, we see that the automatic differentiation (VARad) is slower than the other two. The times for ENDc and VARfd are close, in accordance with the remark on complexity on page 13, and the variational formulation seems a bit faster, which might indicate a better Jacobian approximation.

We now have a look at the Jacobian computed by the various formulations at the initial point of the shooting. We show on Figure 11 the Frœbinius norm of the difference between the Jacobian of each formulation. More specifically, we trace the following distances: END to ENDc, END to VARad, ENDc to VARad, and VARfd to VARad.

We see that the two variational formulations are the closest ones, which is not surprising. Their distance ranges from 10^{-4} to 10^{-2}, and grows when the thrust decreases, probably because the transfer time and number of commutations increase. The next shorter distance is between ENDc and VARad, between 10^{-1} and 1 roughly. The distance between ENDc and VARfd is obviously of the same order, which is why it was omitted on the graph. On the other hand, the greatest distance is between END and the other three formulations (same remark as before, the END-VARfd has been omitted, and END-ENDc is very close to END-VARad...).

Remark: taking the formulas for the continuous case recalled on page 9, they would give an error size of $O(\sqrt{Tol})$ for the two END methods and $O(Tol)$ for the two VAR methods, with $Tol = 10^{-6}$. Although these results do not hold for the discontinuous case, it is interesting to note that the difference between the Jacobians of the END and VAR methods seems actually close to $\sqrt{Tol} = 10^{-3}$.

Concerning the Jacobian numerical computation, it seems reasonable to assume that the variational methods provide the most accurate value, followed by the external differentiation

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

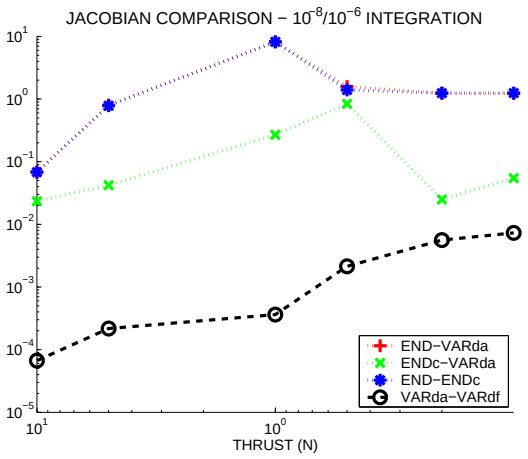

Figure 11. Distance between the Jacobian - 10N to 0,1N transfers.

with switching detection (ENDc), the basic external differentiation (END) being the furthest one. But while the improvement given by the switching detection is quite noticeable on the shooting results, the use of the variational equations does not seem to further improve the convergence. The approximation of the Jacobian by finite differences, when coupled with the switching detection, is here sufficient to ensure a good convergence of the quasi-Newton solver.

Conclusion

As a conclusion, we observe that on this family of problems with a large number of commutations, a switching detection method relying on the integrator's dense output can significantly improve both the convergence and the execution time of the shooting method. Besides, we see that using the variational equations to obtain the Jacobian of the shooting function achieves the same qualitative results in terms of convergence, with a slight improvement in terms of execution times for the finite differences formulation (VARfd).

Codes references:

The code (shooting method and homotopy) used for all the presented numerical experiments is available at `http://enseeiht.fr/~martinon/simplicial`. See [18] for a complete description. The package includes the third party codes described below.

The integrator used for the experiments of this paper is DOPRI5 from E.Hairer and G.Wanner (see [11]), available at `http://www.unige.ch/~hairer/software.html`.

The nonlinear solver for the shooting method is HYBRD/HYBRJ from Burton s. Garbow, Kenneth e. Hillstrom, Jorge j. More (see [12, 13, 14]), available at `http://netlib.bell-labs.com/netlib/minpack/hybrd.f.gz`.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls

For the automatic differentiation, we used TAPENADE from INRIA (http://www-sop.inria.fr/tropics/tapenade.html, see [20, 21] for more details).

REFERENCES

1. E. Allgower and K. Georg. Simplicial and continuation methods for approximating fixed points and solutions to systems of equations. *Siam review*, 22(1):28–85, 1980.
2. E. Allgower and K. Georg. *Numerical Continuation Methods*. Springer-Verlag, Berlin-Heidelberg-New York, 1990.
3. E. Allgower and K. Georg. Piecewise linear methods for nonlinear equations and optimization. *Journal of Computational and Applied Mathematics*, 124:245–261, 2000. Special Issue on Numerical Analysis 2000: Vol. IV: Optimization and Nonlinear Equations.
4. U.M. Ascher, R.M. Mattheij, and R.D. Russel. *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*. Prentice Hall, 1988.
5. M.C. Bartholomew-Biggs, L.C.W. Dixon, S.E. Hersom, and Z.A. Maany. The solution of some difficult problems in low-thrust interplanetary trajectory optimization. *Optimal Control Applications and Methods*, 9:229-251, 1998.
6. H. Bock. Numerical treatment of inverse problems in chemical reaction kinetics. In W. Jiger K. Ebert, P. Deuflhard, editor, *Modeling of Chemical Reaction Systems*, pages 102–125. Springer, Berlin, 1981.
7. J.F. Bonnans. The shooting algorithm for optimal control problems: a review of some theoretical and numerical aspects. Technical report, Université d'El Manar (Tunis), 2002. Lectures notes, DEA de Mathématiques Appliquées de l'ENIT.
8. L. Cesari. *Optimization theory and application. Problems with ordinary differential equations.* Springer-Verlag, New York, 1983.
9. J. Gergaud and T. Haberkorn. Homotopy method for minimum consumption orbit transfer problem. *Control, Optimization and Calculus of Variations*, 12(2):294–310, 2006.
10. J. Gergaud, T. Haberkorn, and P. Martinon. Low thrust minimum-fuel orbital transfer: an homotopic approach. *Journal of Guidance, Control and Dynamics*, 27(6):1046–1060, 2004.
11. E. Hairer, S.P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I: Non-stiff Problems*, volume 8 of *Springer Series in Comput. Mathematics*. Springer-Verlag, Berlin-Heidelberg-New York, 1993. Second Revised Edition.
12. M.J.D. Powell. *A Hybrid Method for Nonlinear Equations*, in *Numerical Methods for Nonlinear Algebraic Equations*, 87–114. P. Rabinovitz, Gordon and Breach, London, 1970.
13. B.S. Garbow, H.E. Hillstrom and J.J. More. *Implementation Guide for MINPACK-1.* Technical Report ANL-80-68, Argonne National Laboratory, 1980.
14. J.J. More, B.S. Garbow and H.E. Hillstrom. *User Guide for MINPACK-1.* Technical Report ANL-80-74, Argonne National Laboratory, 1980.
15. C.Y. Kaya and J.L. Noakes, Computations and Time-Optimal Controls. *Optimal Control Applications and Methods*, 17:171-185, 1996.
16. H.J. Oberle and K. Taubert, Existence and Multiple Solutions of the Minimum-Fuel Orbit Transfer Problem. *Journal of Optimization Theory and Applications*, 95(2):243-262, 1997.
17. L. Pontriaguine, V. Boltianski, R. Gamkrelidze, and E. Michtchenko. *Théorie Mathématique des Processus Optimaux*. Editions Mir, Moscou, 1974.
18. P. Martinon. *Numerical resolution of optimal control problems by a Piecewise Linear continuation method*. PhD thesis, Institut National Polytechnique de Toulouse, 2005.
19. J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. Springer-Verlag, Berlin-Heidelberg-New York, 1983.
20. L. Hascoët. *TAPENADE: a tool for Automatic Differentiation of programs*. Proceedings of 4^{th} European Congress on Computational Methods, ECCOMAS'2004, Jyvaskyla, Finland, 2004.
21. L. Hascoët and V. Pascual. *TAPENADE 2.1 user's guide*. Technical Report 0300, INRIA, 2004. (http://www.inria.fr/rrrt/rt-0300.html).
22. T.L. Cong. *Contrôle optimal et transfert en temps minimal*. PhD thesis, Institut National Polytechnique de Toulouse, 1999.
23. T. Haberkorn. *Transfert orbital à poussée faible avec minimisation de la consommation: résolution par homotopie différentielle*. PhD thesis, Institut National Polytechnique de Toulouse, 2004.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using ocaauth.cls