ON THE REFINEMENT OF DISCRETIZATION FOR OPTIMAL CONTROL PROBLEMS

Julien LAURENT-VARIN * **Frédéric BONNANS** **
Nicolas BEREND *** **Christophe TALBOT** ****
Mounir HADDOU †

* *Projet SYDOCO, INRIA Rocquencourt B.P. 105 78153 Le Chesnay Cedex France* julien.laurent-varin@inria.fr
** *Projet SYDOCO, INRIA Rocquencourt B.P. 105 78153 Le Chesnay Cedex France* frederic.bonnans@inria.fr
*** *ONERA Long-Term Design and System Integration Department BP 72 - 29 avenue de la Division Leclerc 92322 Chatillon Cedex* Nicolas.Berend@onera.fr
**** *CNES : launcher directorate Rond-point de l'Espace - 91023 Evry Cedex* christophe.talbot@cnes.fr
† *Maître de Conférences UMR 6628-MAPMO B.P. 6759 45067 Orléans cedex 2*
mounir.haddou@labomath.univ-orleans.fr

Abstract: In the framework of future launchers design studies, CNES and ONERA are studying possible improvements of tools and methods for trajectory optimization. Trajectory optimization of reusable launch vehicle (RLV) is a very complex task calling for a versatile tool, which should be able to address - either simultaneously or separately - ascent and reentry trajectory phases. Improvement can be made in comparison with existing methods and tools regarding issues such as the global processing of ascent and reentry phases, or specific constraints of reentry phases. In this context, a first candidate for a new optimization method is currently being studied. This method is an adapted interior point algorithm, associated with a Runge-Kutta discretization scheme of the optimal control problem. In this paper we will focus on the presentation of the RK method to use, the error estimation and the mesh refinement policy to use in order to have an optimal complexity and then a fast algorithm.

Keywords: Optimal control, Discretization, Trajectories, Aerospace trajectories.

INTRODUCTION

This paper [1] deals with so-called transcription numerical methods for optimal control problems; we refer to (Betts, 1998) for a survey of numerical optimal control. Our motivation is the need of better algorithms for optimizing trajectories for reusable launch vehicles (RLV), where several paths are to be optimized simultaneously. Indeed, for RLVs improvements are needed in various fields of optimization : in time of calcu-

[1] Submitted to the 16^{th} IFAC Symposium on Automatic Control in Aerospace

lation, in path constraint treatment and in global trajectories mission resolution.

In this paper we concentrate on the strategy of refinement of the discretization mesh.

We choose an interior point algorithm associated with a RK discretization scheme of the optimal control problem. The RK method will be detailed in Part 1. Then we choose to minimize the number of points of this discretization for the first moment of optimization procedure, and when we get closer to the solution, we refine the mesh in order to reach the solution as fast as possible. An error estimate is then required and a refinement procedure is needed too. Part 2 of this paper will present this aspect of the method. Finally we will apply our method to an ascent trajectory of an academic problem inspired from the Venture Star concept. We end with numerical experiments.

1. DISCRETIZATION OF OPTIMALITY CONDITIONS

To compute the solution of the continuous optimality condition we need to discretize them. We have chosen the Runge-Kutta method.

1.1 Optimal control and symplectic integration

In order to certify the optimality we have to consider good co-state with a controlled error. We want to choose the RK coefficients so that the integration has a pre-determined order, which characterizes the method's precision. The order of the method can be assessed from the coefficients through equations called "order conditions".

In his paper (Hager, 2000) Hager gives explicit conditions (up to order four) for optimality systems deriving from Runge-Kutta discretization of the state equation. The order conditions for Hager's partitioned RK methods can be computed by noting that these methods are symplectic (Hairer and Wanner, 1996).

Consider the following unconstrained optimal control problem :

$$\begin{cases} \text{Min } \Phi(y(T)); \\ \dot{y}(t) = f(u(t), y(t)), \quad t \in [0, T]; \\ y(0) = y^0. \end{cases} \qquad (P)$$

Indeed, the interior point methodology reduces constrained optimal control problem to unconstrained ones (see Wright in (Wright, 1997; Wright, 2001; Wright and Jarre, 1999; Laurent-Varin *et al.*, 2003)).

First order necessary optimality conditions are :

$$\begin{cases} \dot{y}(t) = f(u(t), y(t)), \\ \dot{p}(t) = -f_y(u(t), y(t))^T p(t), \\ p(T) = \Phi'(y(T)), \quad y(0) = y^0, \\ 0 = f_u(u(t), y(t))^T p(t). \end{cases} \qquad (OC)$$

We have to integrate a Hamiltonian system that can be discretized by a partitioned Runge-Kutta method, but we can also discretize the initial problem and then work on the discrete one.

$$\begin{cases} \text{Min } \Phi(y_N); \\ y_{k+1} = y_k + h_k \sum_{i=1}^{s} b_i f(u_{ki}, y_{ki}), \\ y_{ki} = y_k + h_k \sum_{j=1}^{s} a_{ij} f(u_{kj}, y_{kj}), \\ y_0 = y^0. \end{cases} \qquad (DP)$$

We omit $k = 0, \ldots, N-1, \quad i = 1, \ldots, s$. After some algebraic computation and the hypotheses of $b_i \neq 0$ (see (Hager, 2000)) we exactly obtain a specific discretization of the continuous optimality conditions (OC) :

$$\begin{cases} y_{k+1} = y_k + h_k \sum_{i=1}^{s} b_i f(u_{ki}, y_{ki}), \\ y_{ki} = y_k + h_k \sum_{j=1}^{s} a_{ij} f(u_{kj}, y_{kj}), \\ p_{k+1} = p_k + h_k \sum_{i=1}^{s} \hat{b}_i f_y(y_{ki}, u_{ki})^T p_{ki}, \\ p_{ki} = p_k + h_k \sum_{j=1}^{s} \hat{a}_{ij} f_y(y_{kj}, u_{kj})^T p_{kj}, \\ 0 = f_u(y_k, u_k)^T p_k, \\ 0 = f_u(y_{ki}, u_{ki})^T p_{ki}, \\ y_0 = y^0, \qquad p_N = \Phi'(y_N). \end{cases} \qquad (DOC)$$

Where : $\hat{b} = b$ and $\hat{a}_{ij} = \dfrac{b_i b_j - b_j a_{ij}}{b_i}$ for all i, j.

This partitioned RK method happens to be symplectic : (Hairer and Wanner, 1996, Theorem 4.6). In particular the following diagram commutes, when we use the above discretization.

$$\begin{array}{ccc} (P) & \xrightarrow{\text{discretization}} & (DP) \\ {\scriptstyle \text{optimality conditions}} \downarrow & & {\scriptstyle \text{optimality conditions}} \downarrow \\ (OC) & \xrightarrow{\text{discretization}} & (DOC) \end{array} \qquad (D)$$

See the presentation of Runge-Kutta and symplectic methods in the books (Hairer *et al.*, 2002; Hairer *et al.*, 1993).

The order r' of the scheme (DOC) is equal to the one r of (DP) if $r \leq 2$, but may be smaller if $r > 2$. Hager gives the four additional conditions for having $r = r'$ when $r = 3$ and 4.

Finally we obtain a set of non-linear equations to solve. In order to solve these equations, we will use a classical Dogleg method (see (Conn *et al.*, 2000)).

2. ERROR ANALYSIS AND MESH REFINEMENT

Whereas optimal control softwares commonly achieve error estimation on primal variables without caring about dual state (see (Betts, 2001)). We show here how, with a symplectic method, to establish a refinement policy in the resolution process with a good error on co-state and then a certificate of optimality. Indeed, large error on integration of the co-state could lead to control strategy that is far from being optimal. This analysis is, however, restricted for the time being to unconstrained problems.

Then, we have to solve a set of equations obtained by discretization of the optimality conditions. But, the discretization produces an error compared with the exact solution. Moreover the non-perfect Newton resolution generates another error compared with the solution. In this part, we would like to analyze the distance to the exact solution in order to know when we have to refine the mesh during the resolution process while minimizing the complexity of the algorithm.

We should notice that the number of operations at each iteration of Newton is proportional to the number of discretization points.

2.1 Newton method and discretization

2.1.1. Newton method Consider the following non-linear equation to be solved with a Newton method : $G(x) = 0$. The distance between the solution x given by the Newton method, and the exact solution $\bar{x}$ may be estimated by the Newton step :

$$x - \bar{x} = G'(x)^{-1}G(x) + o(\|x - \bar{x}\|^2).$$

We can assess the value of this estimate by looking at the ratio of reduction of the norm of G after the previous Newton step. Of course this estimate is meaningless if $G'(x)$ is not invertible or x is far from $\bar{x}$.

2.1.2. Exact optimality conditions We have to solve an unconstrained optimal control problem for which we need to solve the optimality conditions of the following type :

$$\begin{cases} \dot{y} = H_p(y,u,p), \ \dot{p} = -H_y(y,u,p), \\ 0 = H_u(y,u,p), \\ 0 = F(y(0),p(0),y(T),p(T)). \end{cases} \qquad (1)$$

Denote $x = (y,p)$ and let $\phi(x_0)$ be the exact flow associated to the integration of the Hamiltonian system from 0 to T, the control being eliminated by the implicit functions theorem applied to the condition : $0 = H_u(y,u,p)$. Thus, the two points boundary value problem could be written with F as :

$$F(x(0), \phi(x(0))) = 0.$$

2.1.3. Discretization As seen before, system (1) is discretized with a symplectic RK method. Consider the numerical flow ϕ_h of the discretized system. With notations of (DOC), $\phi_h(y_0,p_0)$ is equal to (y_N,p_N).

Denote by $h \in \mathbb{H}$ the discretization, where $\mathbb{H}$ is the set of discretization, defined as follows :

Definition 1. $\mathbb{H}$ is the set of positively valued finite sequences summing to T, i.e; $h \in \mathbb{H}$ if and only if :

- $\exists n_0 \in \mathbb{N}$, $\forall n$, $n \geq n_0 \Leftrightarrow h_n = 0$;
- $\sum_{n=0}^{n_0} h_n = T$.

For Runge-Kutta type methods, it is possible to estimate the principal term error, i.e. to evaluate a function ψ_h such that :

$$\phi(x) - \phi_h(x) = \psi_h(x) + o(\psi_h(x)). \qquad (2)$$

2.1.4. From discrete resolution to the exact continuous solution Since the exact flow ϕ is not known, we solve the following approximate problem :

$$F(x_h(0), \phi_h(x_h(0))) = 0. \qquad (3)$$

Denote $\mathcal{F}(x) = F(x, \phi(x))$, $\mathcal{F}_h(x) = F(x, \phi_h(x))$, and let $\bar{x}$, $\bar{x}_h$ be zeroes of $\mathcal{F}$ and $\mathcal{F}_h$, respectively. We have :

$$\mathcal{F}_h(x) - \mathcal{F}(x) = \Delta_h(x) + o(\psi_h(x)),$$

where $\Delta_h(x) := -\partial_2 F(x, \phi_h(x))(\psi_h(x))$.

Now, let us estimate the distance from the point x when $\mathcal{F}_h(x)$ is small, to the solution $\bar{x}$ of the continuous problem.

$$x - \bar{x} = \mathcal{F}_h'^{-1}(x)\mathcal{F}_h(x) - \mathcal{F}_h'^{-1}(x)\Delta_h(x) \\ + o(\|x - \bar{x}\|) + o(\|\Delta_h(x)\|).$$

The principal term can be viewed as a sum of two components :

- $\mathcal{F}_h'^{-1}(x)\mathcal{F}_h(x)$ which estimates $x - \bar{x}_h$ and
- $-\mathcal{F}_h'^{-1}(x)\Delta_h(x)$ which estimates $\bar{x}_h - \bar{x}$

Then we can estimate the error due to the Newton resolution and discretization. In the next section we concentrate on the refinement of discretization

2.2 Discretization refinement

We assume in this part that the integration method used is a p order method, i.e. the local error on an interval h_k is of the form $C_k h_k^{p+1}$. Hence, if we split the interval h_k into q_k subintervals of size h_k/q_k, then the new estimate error is $q_k C_k (h_k/q_k)^{p+1} = q_k^{-p} C_k h_k^{p+1}$. Moreover, we would like to minimize the number of points to add in order to have an optimal complexity of the algorithm. In the sequel we use the sum of estimated local errors as an estimate of the global error, i.e. $\psi_h(x)$. Then, let us set the problem of reducing the (estimated) global error to a given threshold E with a minimal number of additional time steps. The problem will be to find $q = (q_1, q_2, ..., q_N)$ in $\mathbb{N}^N$ of the following problem :

$$\operatorname*{Min}_{q \in \mathbb{N}^N} \sum_{k=1}^{N} q_k; \quad \sum_{k=1}^{N} \frac{e_k}{q_k^p} \le E \qquad (INP)$$

where $e_k := C_k h_k^{p+1}$. Note that this modelization is only asymptotically correct i.e., if h_k is too large, the error is not of the form e_k/q_k^p.

2.3 Optimal refinement

In this part, let us slightly generalize the problem by denoting (IP) the following integer program :

$$\operatorname*{Min}_{q \in \mathbb{N}^N} \sum_{k=1}^{N} q_k; \quad \sum_{k=1}^{N} e_k f(q_k) \le E \qquad (IP)$$

with $f : \mathbb{N} \to \mathbb{R}_+$ convex and decreasing.

The optimal refinement problem corresponds to $f(x) = 1/x^p$. Obviously, if $\lim_\infty f = 0$, then problem (IP) is feasible. Let us show how to compute the solution of (IP).

Definition 2. Let us define the *maximal marginal gain* g and the *maximal gain index* k_g which expresses the maximum error reduction obtained by adding only one point. We have :

$$g(q) := \max_k e_k \left(f(q_k) - f(q_k + 1) \right) \qquad (4)$$

Algorithm 1. Primal Algorithm

For $k = 1, \ldots, N$ do $q_k := 1$. **End for**
While $\sum_{k=1}^{N} e_k f(q_k) > E$ do
  Compute k_g, the maximal gain index.
  $q_{k_g} := q_{k_g} + 1$.
End While

We can now prove that this algorithm leads to the optimal solution of (IP) with the following proposition :

Proposition 3. Assume (IP) feasible, then the algorithm finds the solution of problem (IP) with $O((R + N) \log N)$ operations, where R is the number of new points.

Because of a question of space the demonstration does not appear on this paper.

3. APPLICATION TO A SINGLE STAGE LAUNCHER TRAJECTORY

We take the example of a single-stage RLV, like the Venture Star concept. For simplification purpose, we choose to consider the problem in two dimensions, with a perfectly round and non-rotating earth, no-J_2 gravity and an exponential atmosphere. The vehicle is expected to reach a given altitude with a given horizontal velocity, while minimizing the propellant consumed during the mission (the thrust and the mass flow-rate are constant). The control variable is the pitch θ, and we assume that the thrust is collinear to the vehicle's axis (so the pitch defines the angle between the thrust and the horizontal axis). Denote h the altitude, V the velocity, T is the thrust, θ the pitch, ϕ the latitude and α the angle of attack.

3.1 Problem characteristics

Table 1. Boundary conditions

	Initial conditions		Final conditions	
h	0	[m]	250	[km]
ϕ	0	[deg]	free	
V	100	[m/s]	7.75	[km/s]
γ	90	[deg]	0	[deg]

The launcher lifts-off vertically at altitude zero. We set an initial vertical speed of 100 m/s in order to avoid singularity issues. At the end of the mission, the vehicle should reach an altitude of 250 km with a horizontal velocity corresponding to the injection into a circular orbit with this altitude.

3.1.1. Thrust force We choose a motor with constant mass flow rate and Isp, and hence constant thrust. Numerical values, summarized in Table 2, include the total initial mass m_0,

Table 2. Thrust characteristics

m_0	=	10^6	[kg]
Isp	=	420	[s]
$(T/W)_0$	=	1.2	
q	=	$m_0 (T/W)_0/\text{Isp}$	[kg/s]
	=	2857.1	[kg/s]
T	=	$q g_0 \text{Isp}$	[N]
	=	$1.18 10^7$	[N]

specific impulse Isp, mass flow rate q, thrust T, and initial thrust-to-weight ratio $(T/W)_0$.

3.1.2. Aerodynamic forces Denote the aerodynamic forces D and L for drag and lift respectively. We have :

$$D = C_D Q S_{ref} \; ; \; L = C_L Q S_{ref},$$

where the aerodynamic coefficient C_D and C_L depend on angle of attack $\alpha = \theta - \gamma$, dynamic pressure $Q := \frac{1}{2}\rho V^2$ and reference surface of the launcher S_{ref}. We choose :

$$C_L = C_L^0 \alpha \; ; \; C_D = C_D^0 + k C_L^2,$$

with numerical values from Table 3.

Table 3. Aerodynamic characteristics

C_L^0	=	1.8	k	=	0.3	
C_D^0	=	0.13	S_{ref}	=	460	[m^2]

3.2 Dynamics of problem

The free final time T_f is a parameter of the problem. We choose to consider it as an additional state variable (with a null dynamic). Using the normalized time $s = t/T_f$ (belonging to $[0,1]$), we can use the following equations of dynamics :

$$\begin{aligned}
\frac{\mathrm{d}h}{\mathrm{d}s} &= T_f V \sin\gamma, \\
\frac{\mathrm{d}\phi}{\mathrm{d}s} &= T_f \frac{V}{h+R_T}\cos\gamma \\
\frac{\mathrm{d}V}{\mathrm{d}s} &= T_f \left[\frac{T}{m}\cos\alpha - \frac{D}{m} - g\sin\gamma\right], \\
\frac{\mathrm{d}\gamma}{\mathrm{d}s} &= T_f \left[\frac{T}{mV}\sin\alpha + \frac{L}{mV} + \left(\frac{V}{h+R_T} - \frac{g}{V}\right)\cos\gamma\right], \\
\frac{\mathrm{d}T_f}{\mathrm{d}s} &= 0.
\end{aligned}$$

The control variable θ appears implicitly in α and then in L and D. We should notice that physical quantities are scaled in order not to have numerical problem. Scaled values were chosen such as variables stay in $[-10, 10]$ (error estimation is then scaled too).

3.3 Cost function

Since the ejected mass-flow rate is constant, the objective of minimizing the propellant consumed is equivalent to minimizing the flight time, so we take the parameter T_f as the cost function.

3.4 Numerical experience

We compare two different refinement algorithms applied to the ascent trajectory problem. The first one is a simple heuristics whereas the second one solves (INP). The two algorithms are initialized by an equidistant discretization mesh of 20 points.

Table 4. Algorithm 1

Iteration	Number of points	Error estimation	New points
1	20	4.01591	5
2	25	1.57963	7
3	32	0.78918	9
4	41	0.42241	12
13	419	0.00437117	125
14	544	0.00244748	163
15	707	0.00147779	212
16	919	0.000899956	X

Table 5. Algorithm 2

Iteration	Number of points	Error estimation	New points
1	20	4.01591	768
2	788	0.0011063	52
3	840	0.000998684	X

Algorithm 2. First algorithm

Solve non-linear equations
While error estimation $> E$
  Add new points in the middle of 30% of the largest error intervals
  Re-solve non-linear equations
End While

Algorithm 3. Second algorithm

Solve non-linear equations
While error estimation $> E$
  Use optimal policy describe in *Algorithm* 1
  Re-solve non-linear equations
End While

The Pitch θ, angle of attack α and path angle γ evolutions obtained after applying one or other algorithm are given in Figure 2. We can clearly see that α is bigger than in usual ascent trajectory because it is not constrained in our problem. We can see that Algorithm 2 ends with 840 points in 2 refinement steps compared to algorithm 1 which ends with 919 in 15 refinement steps. The target of the resolution is 0.001 (Physical quantities are scaled in order to manipulate reduced variables evoluting in $[-10, 10]$).

CONCLUSION

This paper presents a preliminary work on an discretized optimization method for solving RLV trajectory optimization problems. First we have studied a specific RK method called symplectic and its interesting properties for optimal control. Secondly we have combined an estimation of error discretization and the resolution of an integer linear programming problem in order to obtain the best refinement policy we can. Finally

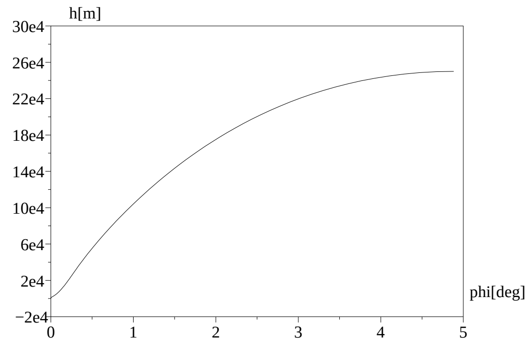

Fig. 1. Trajectory

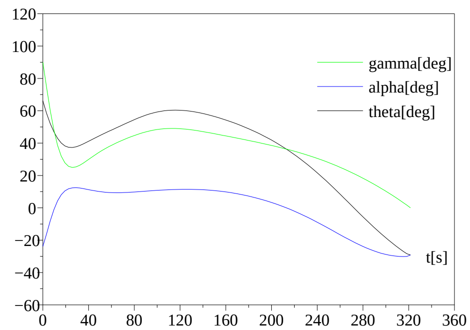

Fig. 2. Angle of attack (α), Pitch (θ) and Flight path angle (γ)

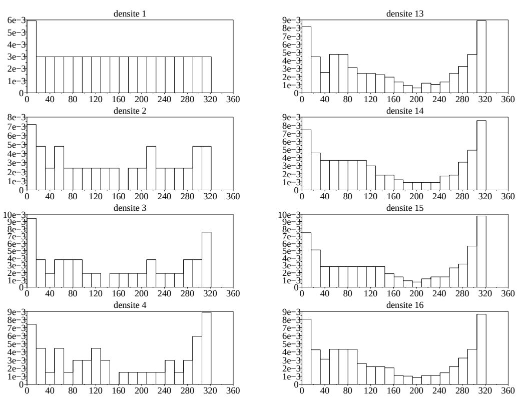

Fig. 3. Algorithm 1 - Mesh evolution

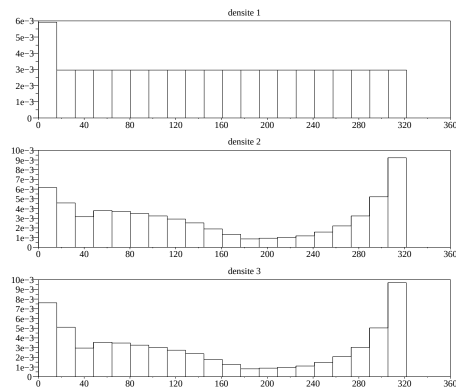

Fig. 4. Algorithm 2 - Mesh evolution

we apply this method to an ascent trajectory optimization.

This first results seems to be very promising about refinement policy. We now have to work on reentry trajectory optimization and then multiphase problem.

REFERENCES

Betts, J.T. (1998). Survey of numerical methods for trajectory optimization. *AIAA J. of Guidance, Control and Dynamics* **21**, 193–207.

Betts, J.T. (2001). *Practical methods for optimal control using nonlinear programming*. Society for Industrial and Applied Mathematics (SIAM). Philadelphia, PA.

Conn, A.R., N.I.M. Gould and P.L. Toint (2000). *Trust-region methods*. MPS/SIAM Series on Optimization. SIAM. Philadelphia, PA.

Hager, W. (2000). Runge-Kutta methods in optimal control and the transformed adjoint system. *Numer. Math.* **87**(2), 247–282.

Hairer, E. and G. Wanner (1996). *Solving ordinary differential equations. II*. second ed.. Springer-Verlag. Berlin.

Hairer, E., C. Lubich and G. Wanner (2002). *Geometric numerical integration*. Springer-Verlag. Berlin.

Hairer, E., S. P. Nørsett and G. Wanner (1993). *Solving ordinary differential equations. I*. second ed.. Springer-Verlag. Berlin.

Laurent-Varin, J., N. Bérend, F. Bonnans, M. Haddou and C. Talbot (2003). A preliminary interior point algorithm for solving optimal control problems. 5^{th} International Conference on Launcher Technology, Madrid, Spain. preliminary version on `http://www-rocq.inria.fr/sydoco/`.

Wright, S.J. (1997). *Primal-dual interior-point methods*. Society for Industrial and Applied Mathematics (SIAM). Philadelphia, PA.

Wright, S.J. (2001). On the convergence of the Newton/log-barrier method. *Mathematical Programming* **90**, 71–100.

Wright, S.J. and F. Jarre (1999). The role of linear objective functions in barrier methods. *Mathematical Programming* **84**, 357–373.