Trajectory optimization of launcher with Hamilton-Jacobi-Bellman approach.

J. Laurent-Varin[*1], O.Bokanowski*[*2],E.Cristiani*[*3] and H.Zidani*[*3]*
[*1] *CNES, Rond point de l'espace, 91000 EVRY, France*
[*2] *Laboratoire J.-L. Lions, Université Pierre et Marie Curie 75205 Paris Cedex*
[*3] *ENSTA & INRIA-Saclay (projet commands), 32 Bd Victor, 75739 Paris*

Abstract

In the framework of CNES/DLA research activities, and in collaboration with ENSTA (H. Zidani) and Uiversity Denis Diderot (O. Bokaowski), we have studied the applicability of Hamilton-Jacobi approach for trajectory optimization of space launcher Ariane 5.
Because of the well known "curse of dimension", dynamic programming principle had been considered as un-usable method of solving space launcher trajectory problems. In view of some recent advances on numerical schemes for partial differential equations (PDE) and the increase of computer's power, the use of dynamical programming becomes possible by integration of Hamilton-Jacobi-Bellmann (HJB) equation. Recall that this method has the great advantage to give access to the global optimum and to obtain a feed-back control low.
We will present a simplified 4-dimensional climbing problem and the main ideas of the method. We will also discuss some perspectives: combination of our method with parallel calculus, the use of HJB approach in order to obtain a good initialization of the shooting method.
This method is patent pending.

1. Introduction

During mission analysis, or launcher development, computation of space launcher trajectory is in the middle of all processes. It is essential to get the optimal trajectory for various missions. In a non convex setting, classical iterative methods compute local optimal solution. Moreover, the initialization of these methods is (very often) difficult. In the contrary, Hamilton-Jacobi-Bellman (HJB) approach allows to compute the global optimal solution and does not need any initialization process, but this approach needs huge computation capabilities.
In this part, we focus on advantage and drawback of this approach.

1.1 Advantages

The classical methods used for trajectory optimization of space launchers are those based on total discretization of the problem or on shooting algorithms (using Pontryagin's principle), see for instance for an overview of the popular algorithms. These methods are know to be very accurate. However, their use presents several difficulties. The first obvious difficulty comes from the fact that the optimization problem is non-convex and then the classical methods are not able to avoid local minima, especially for shooting methods that have the reputation to have a small convergence radius. The initialization of the methods is then very hard. Moreover, it is often required to have a good knowledge of the global behavior of the solution before calculating it (existence of singular arcs, number of commutations and so on).

An interesting by-product of the HJB approach is the synthesis of the **optimal control in feedback form**. Once the HJB equation is solved, for any starting point, the reconstruction of the optimal trajectory can be performed in real time.

An other main advantages of this kind of method is that it leads to the **global optimum** of the problem and is not perturbed by local optimum.

The HJB approach is based on the dynamic programming principle of Bellman and consists in characterizing the value function of the control problem as a unique solution of a nonlinear Partial Differential Equation (PDE - see paragraph 3). This PDE can be solved by **a non iterative method**. Then, we never fall into a non-convergent algorithm which fall into an infinite loop.

Furthermore thanks to this approach, **no initialization** is required.

In classical optimization method, constraints on state and control can introduce complexity of analysis. Especially for shooting method where a co-state should be computed, and where discontinuity of co-state can appear at activation of some state constraint. In the HJB approach, the state constraints can help to reduce the domain of integration of the PDE, which implies also a reduction of the computation time. In fact **constraints** (state, control or mixed) **help the algorithm**.

1.2 Drawbacks

The big and first disadvantage of the method is that we have to solve a PDE that have the dimension of the state system plus one for the time. Then a basic discretization and standard integration lead to an explosion of computation complexity. This problem is well known since the first formulation of dynamical principle.

In order to reduce the complexity, we suggest some new tools to obtain a n efficient and fast algorithm for solving the HJB equation :

- a new discretization scheme
- a specific sparse storage method
- and a model reduction of the initial control problem.

An other drawback of this approach is that, the solution obtained is as precise as the discretization, then we can not get a very accurate solution. In order to improve the accuracy of the solution, we will interface HJB solution to initialize precise method like shooting method.

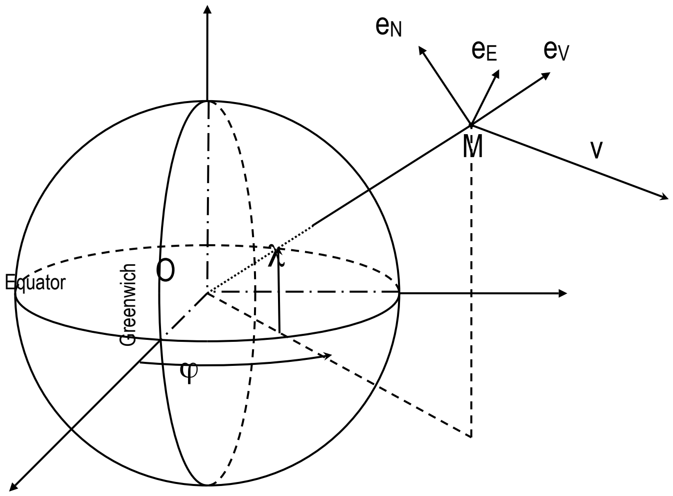
In the next part, we will focus on the state reduction of the initial problem.

2. Reduction of full model

2.1 Full dimensional problem

We focus on the trajectory of space launcher with some stages like Ariane 5. The aim of the launcher is to steer a payload onto an orbit like GTO ones. In our study, we are looking for the trajectory of the center of gravity of the launcher, and we are looking for the command law that maximize the mass of the final payload, or we will look for the trajectory that minimize the consumption if a payload is already defined.

If we conceder the full 3D motion of the center of gravity of the launcher the dynamic of the system is of dimension 6 for position and velocity and 1 more for the mass.



O :	centre of earth
M :	centre of gravity of launcher
(e_E, e_N, e_V) :	Local Est, Nord, Vertical reference frame
r = OM =	radius of position
λ :	latitude
φ :	longitude
v :	norm of velocity
γ :	path angle
χ :	azimuth of velocity

The state of the system is then:

- the position in radius, longitude and latitude, with
- the velocity described by its norm, the path angle and the azimuth, and
- the mass of the launcher.

The control variable is the attitude of the launcher that can described by the angle of attack in the vertical plan and horizontal one. Then the dynamic of the full 6D problem is given by :

$$
\begin{cases}
\dfrac{dr}{dt} = v\cos\gamma\,; \\
\dfrac{d\lambda}{dt} = \dfrac{v}{r}\sin\gamma\,\cos\chi; \\
\dfrac{d\varphi}{dt} = \dfrac{v}{r}\sin\gamma\,\dfrac{\sin\chi}{\cos\lambda}; \\
\dfrac{dv}{dt} = -g(r)\cos\gamma - \dfrac{F_D(r,v)}{m(t)} + \dfrac{F_T(r,v)}{m(t)}\cos\alpha + \Omega^2 r\cos\gamma\,; \\
\dfrac{d\gamma}{dt} = \sin\gamma\,\left(\dfrac{g(r)}{v} - \dfrac{v}{r}\right) - \dfrac{F_T}{vm(t)}\sin\alpha - \Omega^2\dfrac{r}{v}\sin\gamma - 2\Omega; \\
\dfrac{d\chi}{dt} = f(v,\gamma,\chi,r,\varphi,\lambda,m,\Omega); \\
\dfrac{dm}{dt} = -q.
\end{cases}
\tag{1}
$$

F_D is the drag force, F_T is the thrust, Ω is the rotation velocity of earth and f is a complex function that correspond to the dynamic of the azimuth of velocity.

2.2 Reduction of problem

If we consider a plan trajectory we can focus on the main parameters: radius, norm of velocity and angle of attack. The command variable being the angle of attack in the vertical plan.

$$
\begin{cases}
\dfrac{dr}{dt} = v\cos\gamma\,; \\
\dfrac{dv}{dt} = -g(r)\cos\gamma - \dfrac{F_D(r,v)}{m(t)} + \dfrac{F_T(r,v)}{m(t)}\cos\alpha + \Omega^2 r\cos\gamma\,; \\
\dfrac{d\gamma}{dt} = \sin\gamma\,\left(\dfrac{g(r)}{v} - \dfrac{v}{r}\right) - \dfrac{F_T}{vm(t)}\sin\alpha - \Omega^2\dfrac{r}{v}\sin\gamma - 2\Omega; \\
\dfrac{dm}{dt} = -q.
\end{cases}
\tag{2}
$$

The longitude and azimuth variables can be assumed as null. The reduced model still contain the main difficulties of introducing energy in the system via propulsion. The out of plan maneuver is neglected. For more detail see [R6].

3. HJB solver

In this section, we present the Hamilton-Jacobi-Bellman approach for a general optimal control problem, we will underline the advantages of this approach, and the difficulties to pass trough in order to find the optimum of the optimization problem.

3.1 Preliminaries

The trajectory optimization we are interested in enters in a general setting of optimal control problems:

$$(P)\begin{cases} T(x) := \textit{minimize}\, t_f, \\ \text{with } \begin{cases} \dot{y}_x(t) = f(t_f - t,\ y_x(t), \alpha(t)), t \in [0,T], \\ y_x(0) = x, \end{cases} \\ t_f \geq 0, \alpha(t) \in A \text{ for all } t \in [0,T], \\ y_x(t_f) \in C,\ g(y_x(t), \alpha(t)) \leq 0 \text{ for all } t \in [0,T] \end{cases} \tag{3}$$

with :

- y_x the state depending of the initial state x,
- α the control,
- f the dynamic of the system,
- g the constraint function.

In order to adjust problem formulation (1) into the form (3), we will look for the set of initial state that can reach the final orbit.

3.2 PDE formulation

Problem (3) leads to the computation of the following Hamilton-Jacobi-Bellman equation :

$$\begin{cases} \vartheta_t(t,x) + \max\limits_{a \in A, g(x,a) \leq 0} \left\{ -f(t,x,a).\nabla_x \vartheta(t,x) \right\} = 0 \\ \vartheta(0,x) = \Phi(x) \end{cases} \tag{4}$$

And the solution of this equation coincides with the value function of the problem:

$$\Phi(y) = \begin{cases} 0 \text{ if } y \in C \\ 1 \text{ else} \end{cases}, \text{ and } \begin{cases} \vartheta(t,x) := \textit{minimize}\, \Phi(y_x(t)) \\ \text{with } \begin{cases} \dot{y}_x(s) = f(t-s,\ y_x(s), \alpha(s)), s \in [0,t], \\ y_x(0) = x, \end{cases} \\ \alpha(s) \in A \text{ for all } s \in [0,t], \\ g(y_x(s), \alpha(s)) \leq 0 \text{ for all } s \in [0,t] \end{cases}$$

In the equation (4), ϑ_t is the derivative with respect to the time variable, and $\nabla_x \vartheta$ is the gradient of the function ϑ with respect to variable x.

The function $\vartheta(t,x)$ is called accessibility function and takes null value if it exist a trajectory y_x which reach C before the time t and starting from the initial position x, and takes unit value otherwise. The set $\Omega_t := \{x \in \Re^d \mid \vartheta(t,x) = 0\}$ is called catch basin Ω_t of the problem P. For the catch basin Ω_t we associate the front Γ_t which is the boundary of Ω_t.

3.3 Resolution

The HJB equation is solved by the Ultra Bee scheme (and) which is known for its non-diffusive property. To speed up the computation and save memory, we use the storage technique developed in . This storage method consists in storing in a special dynamic data structure only a subset of the grid nodes at each time step and compute the solution only at these nodes. This technique allows to have a fast and efficient local algorithm which concentrates, at each time step, the numerical effort in a small tube.

4. Numerical application

In our setting, the target is a part of GTO orbit (in the state space (r,v,γ)). In the following figure, we compare the optimal solution computed by HJB approach by the one obtained by shooting method. As we can see, the trajectory has the same global behavior than the one obtained by shooting method. The main difference comes from the fact that the HJB approach seems to find an other point on the GTO.

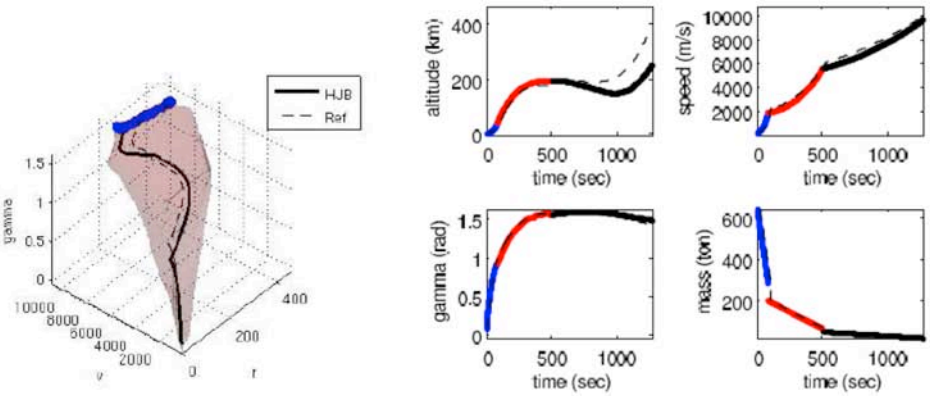


Figure 8: Test 3, Approximation of the optimal trajectory on a grid of $200 \times 200 \times 75$ nodes. CPU time = 57 min.

Further investigation is still in progress.

Conclusion

Hamilton-Jacobi-Bellman approach, when used with suitable discretization scheme and an efficient storage method, gives good results in very reasonable cpu time. In order to increase the accuracy of this method and to speed the computation, parallel calculus method should be coupled to our algorithm. An other issue of our algorithm is the exploitation of HJB solution for initializing a (more accurate) local method, which usually needs a good initialization of the iterative process, as the well known shooting methods.

References

[R1] M. Bardi and I. Capuzzo-Dolcette. *Optimal Control and viscosity solutions of Hamilton-Jacobi-Bellman equations*. Birkhäuser Boston, 1997.

[R2] G. Barles. *Solutions de viscosité des equations de Hamilton-Jacobi*, volume 17 of *Mathématiques et applications*. Springer, Paris 1994.

[R3] O. Bokanowski, N. Megdich, and H. Zidani. Convergence of a non-monotone scheme for Hamilton-Jacobi-Bellman equations with discountinuous data. To appear in Num. Mathematik

[R4] J. T. Betts. Practical methods for optimal control using nonlinear programming. Society for Industrial and Applied Mathematics, Philadelphia, 2001.

[R5] H. J. Pesch. A practical guide to the solution of real-life optimal control problems. Control and Cybernetics, 23:7-60, 1994.

[R6] O. Bokanowski, E. Cristiani, J. Laurent-Varin and H. Zidani. Hamilton-Jacobi-Bellman approach for climbing problem for heavy launcher. Submitted