Progress in Flight Dynamics, Guidance, Navigation, and Control **10** (2018) 159-176
https://doi.org/10.1051/eucass/201810159

OPTIMIZATION OF THE LAUNCHER ASCENT TRAJECTORY LEADING TO THE GLOBAL OPTIMUM WITHOUT ANY INITIALIZATION: THE BREAKTHROUGH OF THE HAMILTON–JACOBI–BELLMAN APPROACH

E. Bourgeois[1], O. Bokanowski[2,3], H. Zidani[4], and A. Désilles[4]

[1]CNES Launcher Directorate, DLA/SDT/SPC
Paris 75012, France

[2]Laboratoire J.-L. Lions, Université Pierre et Marie Curie
Paris Cedex 75252, France

[3]UFR de Mathématiques, Site Chevaleret, Université Paris-Diderot
Paris Cedex 75205, France

[4]Unité des Mathématiques Appliquées (UMA), Ensta ParisTech
828 Bd des Maréchaux, Palaiseau Cedex 91762, France

The resolution of the launcher ascent trajectory problem by the so-called Hamilton–Jacobi–Bellman (HJB) approach, relying on the Dynamic Programming Principle, has been investigated. The method gives a global optimum and does not need any initialization procedure. Despite these advantages, this approach is seldom used because of the difficulties of computing the solution of the HJB equation for high dimension problems. The present study shows that an efficient resolution is found. An illustration of the method is proposed on a heavy class launcher, for a typical GEO (Geostationary Earth Orbit) mission. This study has been performed in the frame of the Centre National d'Etudes Spatiales (CNES) Launchers Research & Technology Program.

ACRONYMS

GEO	Geostationary Earth Orbit
GTO	Geostationary Transfer Orbit
HJB	Hamilton–Jacobi–Bellman
PMP	Pontryagin Maximum Principle

Article available at https://www.eucass-proceedings.eu or https://doi.org/10.1051/eucass/201810159

NOMENCLATURE

C_i	State target
f	Launcher dynamics
K	Set of admissible trajectories
l	Latitude with respect to Equator [rad]
L	Longitude with respect to Greenwich meridian [rad]
r	Geocentric altitude [km]
t	Time
t_1	Beginning of the exo-atmospheric phase
t_f	Combustion duration
U	Set of admissible controls
v	Launcher velocity magnitude [m/s]
X_0	Set of initial state
y_{ux}	A trajectory satisfying the flight dynamics
γ	Launcher velocity slope with respect to local horizon [rad]
χ	Launcher velocity azimuth with respect to instantaneous trajectory plane [rad]

1 INTRODUCTION

A brief explanation of the HJB approach is proposed, associated assets justifying the motivation to use this method to optimize a launcher trajectory are described. An implementation and associated challenges are then presented. In the end, the HJB algorithm is operated to optimize the trajectory of a heavy class launcher, for a typical GEO mission; results are analyzed with regard to a solution obtained through a shooting method.

For any acronym in the following text, please refer to ACRONYMS at the beginning of the paper. For any mathematical notation in the following text, please refer to NOMENCLATURE at the beginning of the paper.

2 PRINCIPLE OF THE HAMILTON–JACOBI–BELLMAN APPROACH

2.1 Mapping of Optimization Methods and Associated Issues

Most of optimization methods rely on necessary condition of optimality. In the field of control optimization, let us quote the Pontryagin Maximum Prin-

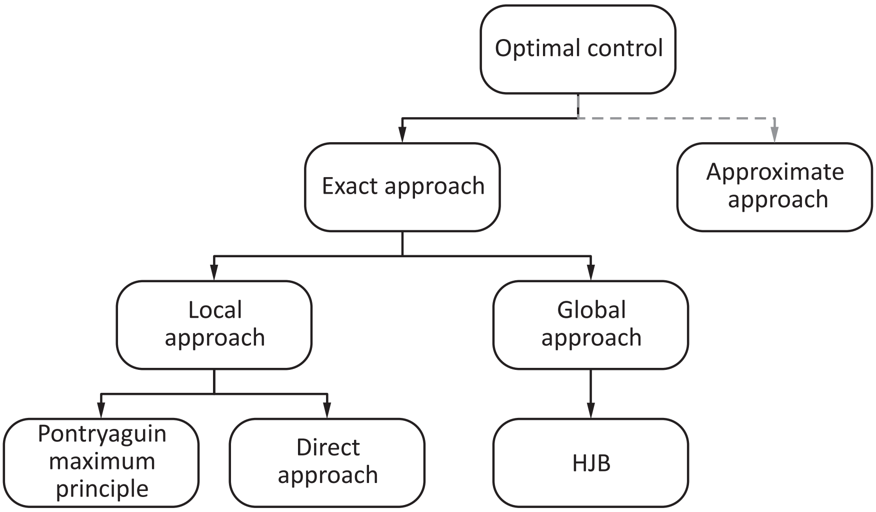


Figure 1 Illustration of classes of optimization approaches

ciple (PMP) [1], which relies on the use of relationship between real and dual states, and Direct Approach [2], which can be declined in various discretization strategies of the control and of the state. Approximate approaches can also be considered to simplify the algorithm implementation by looking for the optimal solution in a restrained control shape family but leading to a suboptimal solution (Fig. 1).

Algorithms derived from exact optimization approaches previously mentioned can be used to find an optimal solution but it should be emphasized that there is no guaranty that this solution is the global optimal one because these approaches deal only with necessary conditions of optimality.

Besides, most of these algorithms are typically operated through iterative processes; as a consequence, it raises the question of initialization and convergence. Finding a good initial guess (in the sense that it would lead to a solution of good quality at the end of the iteration process) can be quite tricky, in particular, when dealing with duality and when the area of convergence is restricted. Efforts have also to be put on the iterative process, so as to get an optimal solution in a reasonable time. A detailed presentation of optimization algorithm families can be found in [3].

With regard to local methods, the HJB approach has two major assets. First, the background theory assures to obtain the global solution when it exists. Second, the implementation of the HJB approach does not require any iterative process, freeing the engineer from tricky tasks that are initialization and convergence. These great assets justify the motivation for the efforts made jointly by CNES and Ecole Nationale Supérieure des Techniques Avancées on applying HJB approach to optimize launcher trajectories.

2.2 Elements Related to the Hamilton–Jacobi–Bellman Approach: Discrete Problems

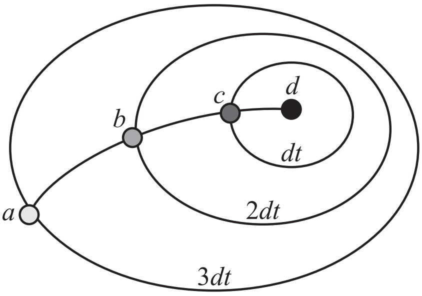


Figure 2 Illustration of the Dynamic Programming Principle

Let us consider the following problem: look for the path starting at point a and allowing reaching point d in minimum time (Fig. 2).

The dynamic is such that at any time, the time of arrival depends only on the current state and on the control we choose; in particular, it is considered that the history of dynamic state prior to the current state has no explicit influence on the future state. In this frame, the core idea at the origin of Dynamic Programming [4] may be expressed in this term: if the optimal trajectory from point a to point d goes through point b, the portion of this trajectory linking point b to point d is also the quickest path between these latter points. If it was not, there would exist a quicker trajectory between points a and d. Now considering the quickest trajectory between points b and d, this smaller optimization problem can be solved if we know that this trajectory goes through point c, and so on.

This simple-looking idea is at the origin of a powerful approach that can be used to solve an optimal control problem: starting from the target, we look for the points being reachable in a given time dt. The locus of this point may be understood as a level curve which is memorized. Starting from this curve, the location reachable in a given time dt is determined; this new level curve represents the locus of points reachable in $2dt$ and is memorized. This computation is running until the farthest level curve crosses point a. Then, this propagation phase, which is time reverse propagated, stops.

Now, starting from point a, let us use the level curves which have been successively memorized during the propagation phase to determine the quickest path, the time flowing in direct sense from 0 to dt, then from dt to $2dt$, and so on until reaching point d. This is the reconstruction phase. Let us notice that not only the best path starting from point a can be reconstructed, but all the paths starting from any point included inside the level curve crossing point a.

2.3 Elements Related to the Hamilton–Jacobi–Bellman Approach: Continuous Problems

As previously described, the Dynamic Programming approach proposes to solve an optimal control problem by considering it as a part of a more global family of optimal control problems: to find the optimal path from point a to point d, one may find the optimal path from point b to point d, and so on. The HJB equation extends this approach, expressed in discrete domain, to continuous optimal control problem. As explained hereafter, solving HJB equation leads to a "value function" that characterizes the optimal control problem family mentioned before and that can be determined as the solution of a first order nonlinear Partial Differential Equation which dimension is related to the number of variables involved in the problem.

Let us consider the following problem:

$$\left.\begin{array}{l} \text{Minimize } t_f \\ \text{with } \begin{cases} y'_{ux}(t) = f(y_{ux}(t), u(t)), t \in [t_1, t_f]\,; \\ y_{ux}(t_1) \in X_0\,; \end{cases} \\ \qquad t_f \geq 0\,, u(t) \in U \text{ for any } t \in [0, t_f]\,; \\ \qquad y_{ux}(t_f) \in C_i \end{array}\right\} \tag{1}$$

where f: $R_d \times A \to R_d$ is the dynamics; X_0 is the set of initial states; $C_i \subset R_d$ is the target; t_f is the combustion duration; y_{ux} is the trajectory satisfying f; t_1 is the beginning of the exo-atmospheric phase; and U is the set of admissible controls.

Let us recall some theoretical results concerning the HJB approach for problem (1). In this section, we assume that f satisfies some classical assumptions: f is a continuous function and for every $x \in R_d$, the set $f(x, A)$ is closed and convex. There exists $c_0 \geq 0$ so that $\sup_{a\in A} |\, f(\xi, a) \,| \leq c_0(1 + |\xi|)$. Moreover, for every $R > 0$, there exists $L_R > 0$, such that:

$$\forall \xi, z \in B(0, R)\,, \quad \sup_{a\in A} |\, f(\xi, a) - f(z, a) \,| \leq L_R |\xi - z|\,. \tag{2}$$

Consider the minimal time function T which associates to any point $x \in R_d$ the minimal time needed to reach the target with an admissible trajectory y_{ux} solution of (1) and satisfying $y_{ux}(\theta) \in K$ (the set of admissible trajectories):

$$T(x) := \inf \{t \geq 0\,, \exists u \in L_\infty((0, t); U)\,; y_{ux}(t) \in C_i\,;\ \ y_{ux}(\theta) \in K \ \ \forall \theta \in [0, t]\}\ .$$

Many works have been devoted to the regularity of the minimum time function T. When $K \equiv R_d$ and under some local metric properties around the target, the function T is the unique continuous viscosity solution of an HJB equation.

Here, without assuming any controllability assumption at the boundary of the target and neither at the boundary of K, the function T may be discontinuous. Indeed, if, for $x \in R_d$, no trajectory y_{ux} reaches the target C_i or if any trajectory leaves K before reaching the target, we set $T(x) = +\infty$. Nevertheless, it can be shown that T is low semicontinuous and its epigraph can be characterized by a Lipshitz function ϑ.

To do so, let us first consider a Lipschitz continuous function ϑ_0: $R_d \to R$ such that

$$\vartheta_0(x) \leq 0 \Leftrightarrow x \in C_i\,. \tag{3}$$

In particular, one has:

$$\operatorname{cap} t(0) = C_i = \{x, \vartheta_0(x) \leq 0\}\ .$$

Consider the value function u associated to the Mayer problem with final cost ϑ_0:

$$u(x,t) := \inf\left\{\vartheta_0(y_{ux}(t)), u \in U, y_{ux}(\theta) \in K\ \ \forall \theta \in [0,t]\right\}\ . \tag{4}$$

The capture basin is characterized by

$$\operatorname{cap} t(t) = \{x, u(x,t) \leq 0\}\ .$$

However, function u is a value function of a state constrained problem and we are still faced with the problem of characterizing this value function if no controllability assumption is made. To overcome this difficulty, let us consider another Lipschitz continuous function g: $R_d \to R$ such that

$$g(x) \leq 0 \Leftrightarrow x \in K\,. \tag{5}$$

Then, let us consider the control problem:

$$\vartheta(x,t) :\, = \inf\left\{\max(\vartheta_0(y_{ux}(t)), \max_{\theta\in[0,t]} g(y_{ux}(\theta))), u \in U\right\}\ . \tag{6}$$

Problem (6) has no "explicit" state constraint. In fact, in this new setting, the term $\max_{\theta\in[0,t]} g(y_{ux}(\theta))$ plays the role of a penalization that a trajectory y_{ux} would pay if it violates the state constraints. In Theorem 3.2, it will be shown that the advantage of considering (6) is that ϑ can now be characterized as the unique continuous solution of an HJB equation.

Theorem 3.1 (characterization of the capture basin): assume (2), let ϑ_0 and g be Lipschitz continuous functions defined, respectively, by (3) and (5). Let u and ϑ be the value functions defined, respectively, by (4) and (6). Then, for every $t \geq 0$, one has the following:

(i) the capture basin is given by:

$$\operatorname{cap} t(t) = \{x, u(x,t) \leq 0\} = \{x, \vartheta(x,t) \leq 0\}\,;$$

(ii) if $\vartheta(x,t) < 0$ and $K^{\circ} = \{x, g(x) < 0\}$, then $u(x,t) < 0$ and on $[0,t]$, there exists an admissible trajectory y_{ux} that never touches the boundary ∂K.

Proposition 3.1: assume (2); the minimal time function T: $R_d \to R + \cup\{+\infty\}$ is Lipschitz semicontinuous.

Moreover, one has:

$$T(x) = \inf\{t \geq 0, \vartheta(x,t) \leq 0\}\,,$$

with ϑ being the value function defined in (6) where ϑ_0 and g are any Lipschitz functions satisfying, respectively, (3) and (5).

The use of a level-set approach is a standard way to determine the minimal time function of unconstrained control problems; we generalize it to the case when the time control problem is in the presence of state constraints. Our formulation also allows obtaining the capture basins. As mentioned before, the function ϑ can be characterized as the unique solution of a Hamilton–Jacobi equation. More precisely, let us consider the Hamiltonian:

$$H(x,p) := \max_{u \in U}(-f(x,u)p)\,.$$

Theorem 3.2: assume (2) and that ϑ_0 and g are Lipschitz continuous. Then, ϑ is the unique continuous viscosity solution of the HJB variational inequality (obstacle problem):

$$\left.\begin{aligned} \min(\partial_t \vartheta + H(x, \nabla\vartheta), \vartheta - g(x))\,, \quad & t > 0\,,\ x \in R_d\,; \\ \vartheta(x,0) = \max(\vartheta_0(x), g(x))\,, \quad & x \in R_d\,. \end{aligned}\right\} \tag{7}$$

Here, g is the “fictitious cost” that a trajectory would pay if it leaves K; it comes from the presence of the sup-norm $\max_{\theta \in [0,t]} g(y_{ux}(\theta))$ in the cost function which defines ϑ (see (6)).

The HJB equations (7) are set in the whole space $R_+ \times R_d$, with $d = 6$; nevertheless, in order to perform computations, a finite domain is used on which the HJB equations (7) are discretized. For this, a uniform space grid is chosen; the time step is chosen constant for simplicity. Advanced numerical technics are then applied to efficiently evaluate the approximation of the function ϑ which corresponds to the propagation phase previously described.

Once the function ϑ is computed everywhere, the minimal time function is computed as

$$T(x_i) := \inf\{t_n, t_n \geq 0, \vartheta(t_n, x_i) \leq 0\}\,.$$

This corresponds to the reconstruction phase: it is now possible to get the optimal feedback control law and the corresponding optimal trajectory, typically by using a classical second-order Runge–Kutta scheme.

2.4 Challenges Related to the Implementation of the Hamilton–Jacobi–Bellman Approach

Main issues related to HJB approach are due to the strong dependence of computation efforts (in terms of memory size and computation time) to the number of dimensions of the optimization problem to solve: these issues are due to the fact that the HJB equation is a partial differential equation which has to be solved in terms of state and time which all are discretized. Besides, in the frame of the HJB approach, parameters behave like additional states (associated to steady dynamics), which increase the number of dimensions to consider.

When dimension of the problem increases (typically, above dimension 3), numerical computation becomes very challenging. This is what Robert Bellman describes as "the curse of dimensionality;" for a typical three-dimensional (3D) trajectory such as the one presented here after, 12 dimensions have to be considered (see section 4); considering a very rude discretization of 10 points in each dimension (decreasing the computation accuracy), we still get $10e^{12}$ points to handle. In the frame of launcher trajectory optimizations, efforts have been pushed on dealing with high-dimension problems.

3 APPLICATION OF THE HAMILTON–JACOBI–BELLMAN APPROACH TO LAUNCHER TRAJECTORY OPTIMIZATION

Having in mind the issues previously mentioned, a formulation of the launcher trajectory optimization has been proposed in order to replace this high-dimension problem by a set of smaller ones, without losing the global optimality. The principle is to consider a cost function which is separable in time or, in other words, a cost function which value depends at each time only on the current state (and, naturally, on the control) and not on the history of previous states. In such a frame, if the problem to optimize relies on control and parameters occurring at different phases, it is possible to cut the global problem in smaller ones only ruled by current "active" control and/or parameters. Each subproblem can be solved sequentially, while the global optimality is maintained thanks to the combining effort of memorization operated at the end of each subpropagation phase and to the matching operated between the subproblems during the global reconstruction phase.

3.1 Presentation of the Reference Problem

As a consequence, a formulation of the trajectory optimization problem suitable for HJB computations implies firstly a precise characterization of the flight sequential and of the variables to optimize (control, parameters).

A mission toward GEO with an "Ariane5-like" launcher is considered. The choice of this mission is justified by satellite market considerations and also because it includes several elements which are challenging for the HJB approach implementation: the state domain is large (roughly 36 000 km around the Earth, required launcher increase of speed of more than 10 000 m/s), the mission duration is long (more than 5 h), and the number of parameters is also important as detailed hereafter.

The objective is to optimize the launcher trajectory so as to maximize the payload mass injected on the GEO. The configuration of the launcher relies on two solid propulsion boosters fired on ground burning simultaneously, a cryogenic lower stage ignited on ground and still operating after booster jettisoning and a cryogenic upper stage. The engine of the upper stage can be ignited several times in flight; the total combustion duration is spread along among the different burns. In addition, the perigee altitude of the transfer orbit reached at the end of the first upper stage boost should be at least 180 km. For all the engines, the profile of thrust magnitude is preset on ground; the trajectory is commanded through the orientation of thrust.

The launcher is operated from French Guyana; the flight starts with a vertical takeoff, goes on with a pitch over maneuver at a constant angle rate which intensity can be optimized, and is followed by a atmospheric gravity turn phase during which the launcher thrust is steered according to a null aerodynamic angle of attack path, so as to limit aerodynamic loads on launcher structures. The azimuth of launch characterizing roughly the geographical direction of the plane in which occurs the atmospheric flight has also to be optimized.

The atmospheric phase ends with the booster jettisoning; above begins the exo-atmospheric phase during which the thrust orientation is fully optimized, under constraints. The upper stage will inject the satellite on the GEO thanks to two boosts; respective combustion durations and associated ballistic phase length have to be optimized.

3.2 Formulation of the Hamilton–Jacobi–Bellman Approach

From the above description of the launcher flight sequential, we can remark that the atmospheric flight is totally controlled by the initial parameters that are the pitch over rate and the launch azimuth, the orientation of the thrust being then constrained by the gravity turn phase. Naturally, other parameters such as the ballistic phase duration also have an effect on the cost function but do not act explicitly during the atmospheric phase (recall that the combined contributions of these parameters will be taken into account during the reconstruction phase, assuring the global optimality of the HJB solution).

For the atmospheric flight having low dimensions in terms of parameters, which totally determine the control, an efficient way to reduce computation time is to manage atmospheric and exo-atmospheric separately. Taking also into account the fact that the atmospheric flight is quite short, this part of the flight can be handled by computing and storing in memory a set of atmospheric trajectories associated to a range of set of values for tilting angle rates and launch azimuths.

Exo-atmospheric flight can also be subdivided into different phases, considering that the lower cryogenic stage exo-atmospheric flight phase is explicitly controlled only by the orientation of the thrust. The same consideration can be made during the two propelled phases of the upper stage, while the ballistic phase is controlled by its duration, no engine thrust being available by definition.

The management of different successive exo-atmospheric subproblems is handled by starting from the GEO, which is perfectly known, and then performing a time-reverse propagation.

Additionally, it is assumed that the second upper stage boost is an impulsive one; as it can be seen during numerical computation, this hypothesis is quite weak: considering the engine mass flow rate and the amount of propellant determined thanks to HJB, the second boost lasts less than a minute, which is quite negligible with regards to the total upper stage combustion time and the duration of the ballistic phase.

Making this hypothesis allows to nicely simplify the management of different successive subproblems: the final boost being impulsive, the GTO necessarily intersects the GEO; consequently, starting from the final GEO, several sets of kinematic conditions corresponding to different orbit intersection configurations are considered. To each of them corresponds an amount of propellant required for getting from the GTO to the GEO, this amount being determined thanks to space mechanics equations.

Then, the first HJB problem is introduced during which all the sets of kinematic conditions are time reverse propagated, the only control of this problem being the ballistic phase duration. The propagation from a discretize state to another is performed by computing the function ϑ for the nodes in the neighborhood of the current state, the values at the different nodes being then stored in memory. At the end of the propagation, the value of the function ϑ is known everywhere on the grid; that corresponds to get the level map drawn in Fig. 2. The second HJB problem is then considered from the end of the first upper stage boost to the beginning of the exo-atmospheric phase. In this subproblem, controls are successively the thrust orientation of the upper and lower stage. The dynamic states at the interfaces of all the subproblems are handled thanks to multidimensional interpolations minimizing the loss of information occurring at the encounter of the different dynamic state "wave fronts" generated during the propagation phases.

The optimal trajectory and associated control are finally determined by performing the reconstruction among the trajectories resulting from the matching of different subproblems. The state being discretized, this reconstruction corresponds to follow the nodes which have the lowest value for the function ϑ.

From the above description of the launcher flight sequential, we can remark that the payload mass is not only the criterion to maximize but it intervenes also as a parameter which influences the dynamics of all the flight phases. In order to reduce the dimension of all the subproblems, the above process is run considering the payload mass as a constant. The criterion to be minimized in the HJB problem is the combustion duration which corresponds to the pro-

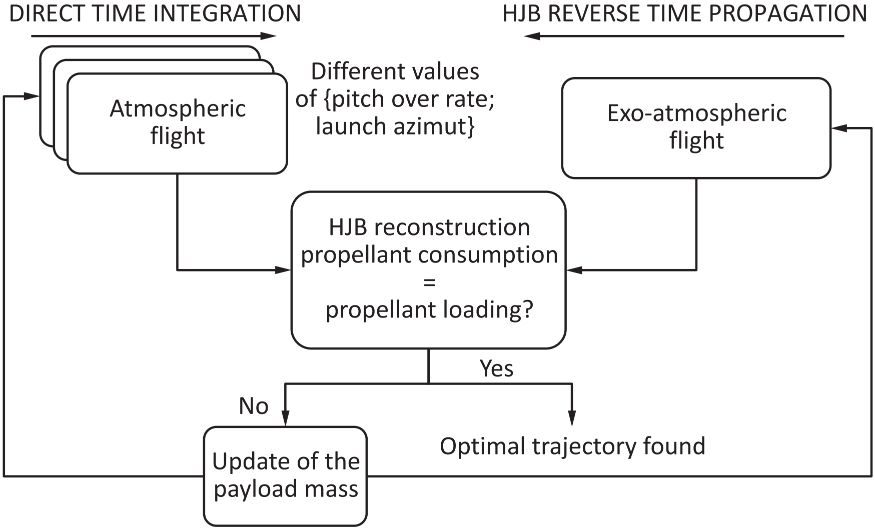

DIRECT TIME INTEGRATION
HJB REVERSE TIME PROPAGATION
Atmospheric flight
Different values of {pitch over rate; launch azimut}
Exo-atmospheric flight
HJB reconstruction propellant consumption = propellant loading?
Yes
No
Update of the payload mass
Optimal trajectory found

Figure 3 Implementation of the HJB approach

pellant consumption; this formulation allows expressing the launcher trajectory optimization problem as a minimum time problem which matches the notion of reachability inherent to the HJB approach. At the end of the reconstruction, if the optimal trajectory does not use all the available propellant, it means that the launcher could bring a heavier payload: the above process is run another time with an increased value of the payload mass. The process goes on until payload mass variations tend to be negligible. This scheme preserves the properties of the HJB approach assuring to reach the global optimal solution (Fig. 3).

4 NUMERICAL RESULTS

The launcher trajectory optimization toward GEO is solved by applying the HJB described previously. The characteristics of the launcher are given in Table 1.

Let us precise some elements about the flight dynamics considered: the launcher is considered to respond instantaneously to the control (thrust orientation); in particular, inertia dynamics around the center of gravity is considered to be negligible, the launcher is a point on which are concentrated the weight, thrust, and aerodynamics forces. The trajectory is considered in 3D space: the state includes three positions, three velocities, and scalar parameters: pitch over rate, launch azimuth, ballistic phase duration, allocation of propellant consumption among the two upper stage boosts, and payload mass. This leads to 11 dimensions to which the time (which is also discretized) should be added.

The trajectory is optimized with two independent algorithms: one relying on a shooting method, the other corresponding to the HJB implementation strategy previously described. The trajectory optimized by the shooting method is named "referenced trajectory:" the associated numerical code is validated and representative of CNES trajectory/performance studies is performed in standard launcher engineering analysis.

Before optimizing, a trajectory is simulated using the flight dynamics implemented in HJB algorithm, but with the control associated to the reference

Table 1 Launcher characteristics

Stage	Propellant loading, T	Mean thrust, kN	Mean combustion duration, min
Booster (2x)	240	3800	2
1	175	1350	10
2	28	180	10

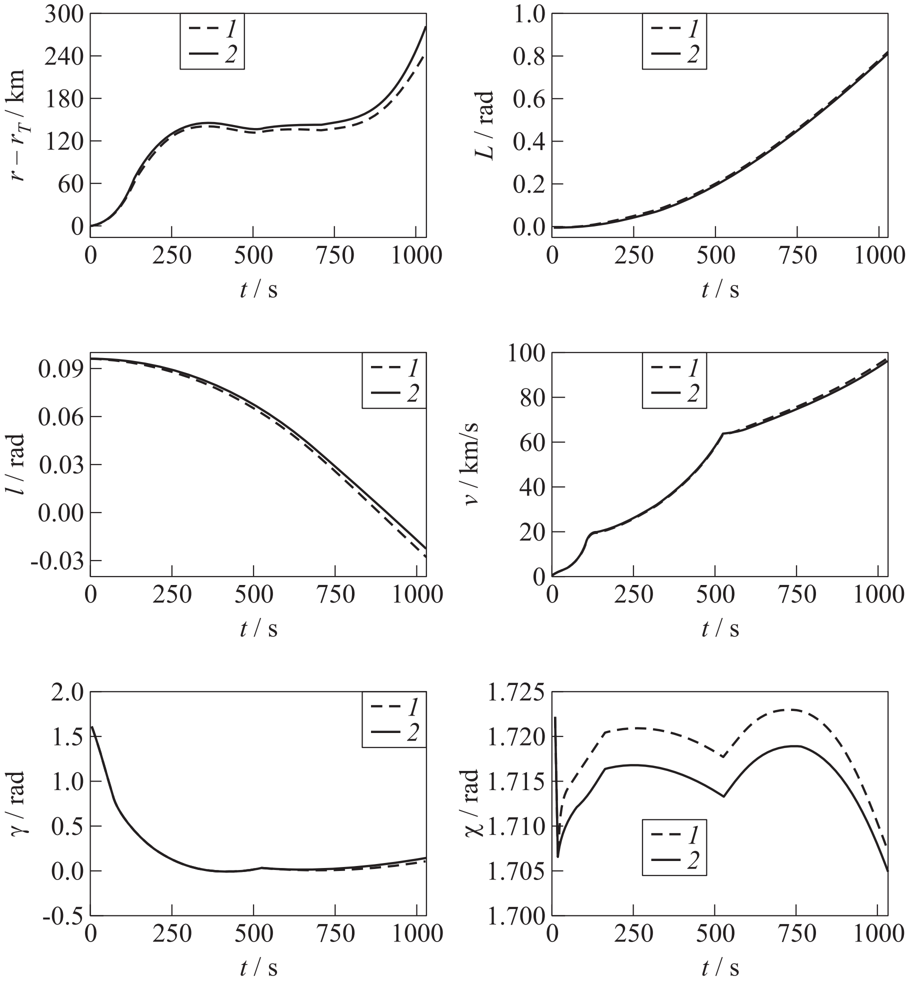


Figure 4 Comparison of reference (*1*) and simulated (*2*) trajectories

trajectory. This trajectory is compared to the reference trajectory (please refer to section NOMENCLATURE for the definition of the plotted parameters) (Fig. 4).

The two trajectories are quite close, which validate the flight dynamics implemented in the HJB algorithm. The small gaps will be investigated in future activities.

Now, the first HJB optimization is performed in order to minimize the upper stage propellant consumption, with payload mass fixed to the value maximized

Table 2 Trajectory characteristics

Approach		Launch azimut, deg/East	Pitch over rate, deg/s	Propellant consumption of the upper stage, kg		Payload mass, kg	Propellant remaining in tanks after GEO injection, kg
				First boost	Second boost		
PMP (reference trajectory)		−7.50	0.92	19 746	4 754	5 307	0
HJB	Opt1	−7.50	0.92	18 795	5 071	5 307	634
	Opt2	−7.45	0.97	18 704	5 097	5 307	699
	Opt3	−7.51	0.92	19 746	4 754	5 632	0

Table 3 Grid sizes considered

Grid	Size ($r \times L \times l \times v \times \gamma \times \chi$)
0	$20 \times 5 \times 20 \times 15 \times 5 \times 5$
1	$40 \times 10 \times 40 \times 30 \times 10 \times 5$
2	$60 \times 15 \times 60 \times 45 \times 15 \times 5$

with the reference trajectory and targeting the GTO of the reference trajectory. The first optimization is performed (hereafter, named "opt1") with launch azimuth and pitch over rate coming from the reference trajectory and then another optimization with these parameters optimized (hereafter, named "opt2") (Table 2).

The comparison of the reference and opt1 trajectories shows that HJB algorithm allows identifying a more efficient trajectory in the sense that it requires less propellant to reach the same GTO, with the same payload mass. In addition, the comparison of the opt1 and opt2 trajectories shows a moderate update of the optimal parameter values.

But it may be understood that HJB opt1 and opt2 trajectories are computed on a given grid (as previously explained) and that the accuracy of this grid is still not completely satisfying (Table 3).

The size of Grid 0 is determined by putting a fine discretization on parameters that show the quickest evolution along the trajectory; it is typically the case for the evolutions of the altitude and of the norm of the velocity vector. The total size of the grid remains limited in order to get first results quickly. For this first preliminary study, Grids 1 and 2 are refined just by increasing arbitrarily the mesh density, the discretization remaining evenly distributed in each state dimension. For further studies, it could be worth trying to improve the distribution of the nodes in the state space in order to increase the numerical accuracy while keeping a reasonable computation time. It could be also use-

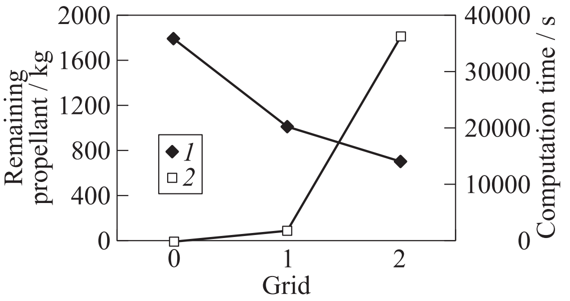


Figure 5 Impact of the grid refinement on results and computation duration: *1* — remaining propellant; and *2* — computation duration

ful to elaborate an assessment of the computational accuracy for a given grid (Fig. 5).

While refining the grid from 0 to 2, it can be seen that propellant remaining in tanks after GEO injection tends to decrease. A more refined grid would allow determining more accurate results but Grid 2 is a compromise between accuracy and computation time which increases quickly. Increasing the accuracy of the HJB algorithm is one axis of improvement expressed hereafter. Let us note also that a rough grid already gives preliminary tendencies for a very short computation time. For the moment, it should be kept in mind that the gaps between reference and HJB trajectories rely at least partly on the grid accuracy.

Finally, the last trajectory is optimized (named "opt3"), corresponding to the initial problem presented before: to maximize the payload mass on the GEO, while optimizing all controls and parameters. The computations were performed on a $50 \times 15 \times 50 \times 30 \times 15 \times 3$ grid (computation lasted for 2 h 30 min for a fixed payload mass). To get the optimal trajectory, 14 h are required (Fig. 6).

The reference and opt3 trajectories have a global good consistence, the latter trajectory showing an extra amount of payload mass of 300 kg. Gaps between the two solutions may result from several causes (in addition to previous identified elements) that will be further investigated; in particular, the second boost in HJB computation being considered as impulsive, trajectory losses due to gravity and incidence are not taken into account, which tends to overestimate the launcher performance. The perturbation of Earth gravity field is not considered in the dynamics handled by HJB algorithm, which tends to overestimate the altitude of the apogee of the GTO and so the launcher performance. These points motivate some future improvements presented hereafter.

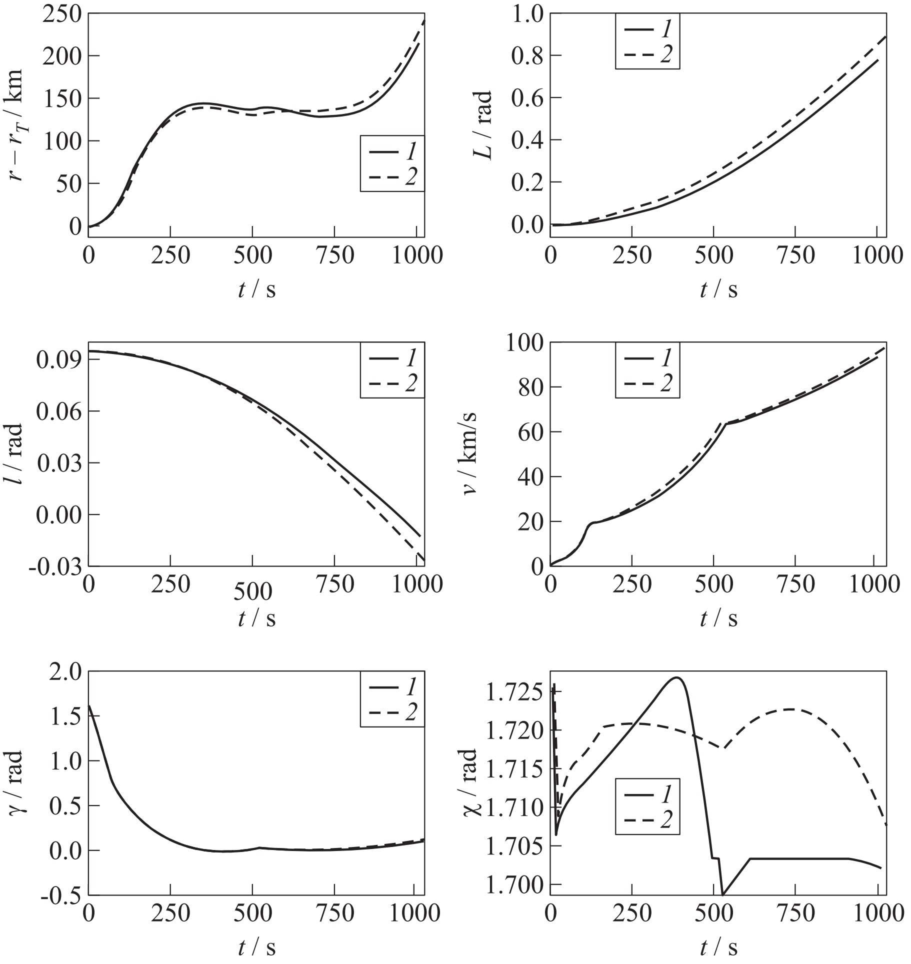
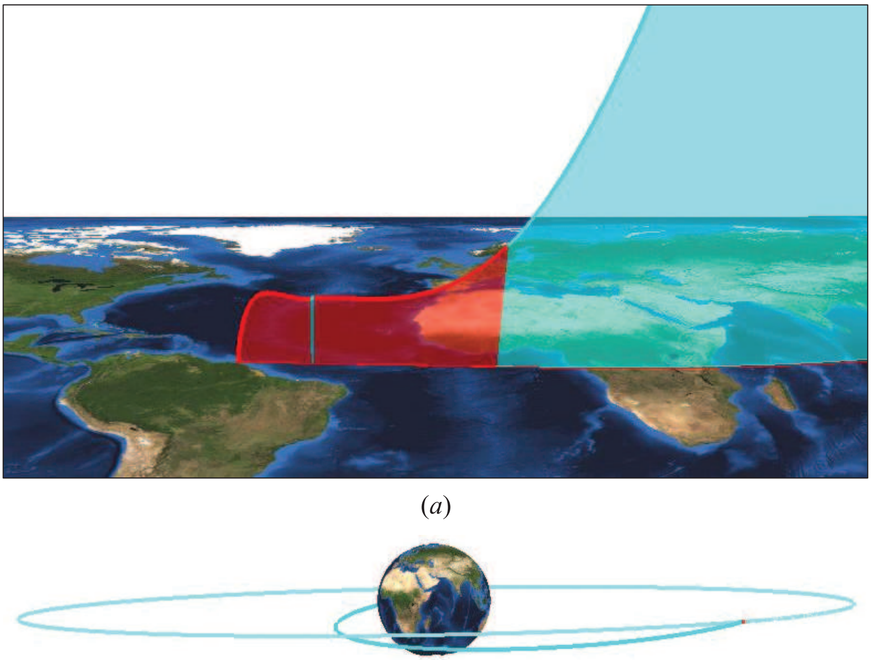


Figure 6 The opt3 trajectory profiles: *1* — reference; and *2* — HJB

The HJB computation of trajectory opt3 lasts roughly 10 h. While being a nonnegligible figure, it should be emphasized that the process is fully automatic, freeing engineers from time consuming efforts such as initialization and convergence tasks. Moreover, this duration is small enough to allow computation in "bash mode" during nonworking hours and to have results available at the beginning of the next working session.

In Fig. 7, there are given some illustrations of the trajectory profile, from launch base to final GEO (in red are the propelled phases and in blue are the ballistic phases).



(*a*)

(*b*)

Figure 7 Trajectory profiles from launch until the upper stage first boost cuts off (*a*) and of the upper stage second boost (*b*)

5 IMPROVEMENTS

In order to reduce the computation time, an implementation relying of parallel computation is going to be developed, taking benefit of the power offered by a computer network organized through a cluster.

The corresponding increased calculation capacity may also be used to refine the grid used to discretize the dynamic state. The postprocessing of the GEO trajectory shows that the dynamic evolution is not evenly distributed among the grid which suggests refining the node density in some areas. An adaptive mesh may also be envisaged to perform automatically the grid improvement.

Another way of improvement would be to consider the HJB approach as a way to improve local indirect approach: indeed, the gradient of the level set function corresponds to the dual state considered in the PMP. The HJB approach could be used to provide a correct estimate of the dual state at the beginning of the trajectory, feeding the shooting method classically derived from the expression of transversality relationships. The HJB approach would be used with a not too much refined grid to speed up calculations; then the local approach would only be used to converge quickly toward the global optimum.

Some efforts may also be made on introducing other trajectory constraints that should be useful for using HJB algorithm in engineering studies.

REFERENCES

1. Pontryagin, L. S., V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko. 1962. *The mathematical theory of optimal processes*. Interscience. 360 p.
2. Betts, J. T. 2001. *Practical methods for optimal control using nonlinear programming*. 1st ed. Advances in design and control ser. Society for Industrial and Applied Mathematics. Vol. 3. 190 p.
3. Betts, J. T. 1998. Survey of numerical methods for trajectory optimization. *J. Guid. Control Dynam.* 21(4):193–207.
4. Bellman, R. 1957. *Dynamic programming*. Princeton, NJ: Princeton University Press. 367 p.
5. Cristiani, E., and P. Martinon. 2010. Initialization of the shooting method via the Hamilton–Jacobi–Bellman approach. *J. Optimiz. Theory App.* 146(2):321–346.
6. Bokanowski, O., E. Bourgeois, A. Désilles, and H. Zidani. 2015. Global optimization approach for the climbing problem of multi-stage launchers. Prepint on HAL.