Indirect approach for optimal control in Orekit

Romain Serra

October 2024

1 Introduction

This document explains the foundations of Orekit (version 12.2) implementation of indirect optimal control for space trajectories. The library provides the user with a model for the so-called adjoint variables as defined in Pontryagin's Maximum Principle (PMP), along with a solver for these optimality conditions for fixed-time, orbit-to-orbit problems, in the form of a single shooting algorithm. To know more about optimal control theory and their application in astrodynamics, the reader is encouraged to check out the literature, for example [3, 2, 1].

2 Adjoint equation

The state vector $\mathbf{x}$ is made of the position vector $\mathbf{r}$, the velocity $\mathbf{v}$ and the mass m. Let $\mathbf{T}$ be the thrust force. Let us assume that the exhaust speed is constant, so that the mass equation is:

$$\dot{m}(t) = -\alpha |\mathbf{T}|_2, \tag{1}$$

where $\alpha \geq 0$. Given a set of external accelerations $\mathbf{a}_1, \ldots, \mathbf{a}_n$, according to Newton's second law, the equations of motion read:

$$\begin{aligned} \dot{\mathbf{r}}(t) = & \quad \mathbf{v}(t) \\ \dot{\mathbf{v}}(t) = & \quad \frac{\mathbf{T}(t)}{m(t)} + \sum_i \mathbf{a}_i(t, \mathbf{r}(t), \mathbf{v}(t)) \end{aligned} \tag{2}$$

Let us adopt a Lagrange form of the cost function J:

$$J = \int_{t_0}^{t} L(t, \mathbf{x}, \mathbf{T}) d\tau \tag{3}$$

Boundary conditions (at t_0 and t_f) aside, the minimization of J under the differential equations (1-2) defines an optimal control problem. Following the PMP's framework and discarding the so-called abnormal case (see [3] for definition), the Hamiltonian H is:

$$H = -L + < \mathbf{p}_r, \mathbf{v} > + < \mathbf{p}_v, \frac{\mathbf{T}}{m} + \sum_i \mathbf{a}_i > -\alpha |\mathbf{T}|_2 p_m, \tag{4}$$

where $\mathbf{p}_r = (\mathbf{p}_r, \mathbf{p}_v, p_m)$ is the adjoint vector. The optimal solution $\mathbf{x}^*$, $\mathbf{p}^*$ maximizes this quantity as H^* and $\mathbf{p}^*$ satisfies $\frac{d\mathbf{p}}{dt} = -\frac{\partial H^*}{\partial \mathbf{x}}$.

Assuming that $L = L(t, m, |\mathbf{T}|_2)$, then it comes that the thrust must be aligned with the adjoint velocity i.e. $|\mathbf{p}_v|_2\mathbf{T} = |\mathbf{T}|_2\mathbf{p}_v$. One can then rewrite the Hamiltonian as:

$$H = -L(t, m, |\mathbf{T}|_2) + < \mathbf{p}_r, \mathbf{v} > + |\mathbf{p}_v|_2 \frac{|\mathbf{T}|_2}{m} + < \mathbf{p}_v, \sum_i \mathbf{a}_i > -\alpha |\mathbf{T}|_2 p_m \quad (5)$$

Because the cost function does not depend directly on the position and velocity vectors, the differential equations for them are already known and are actually linear:

$$\begin{aligned} \dot{\mathbf{p}}_r &= -< \mathbf{p}_v, \sum_i \frac{\partial \mathbf{a}_i}{\partial \mathbf{r}} > \\ \dot{\mathbf{p}}_v &= -\dot{\mathbf{p}}_r - < \mathbf{p}_v, \sum_i \frac{\partial \mathbf{a}_i}{\partial \mathbf{v}} > \end{aligned} \quad (6)$$

In Orekit, the overall logic of Eq. (6) is encoded in *(Field)CartesianAdjoint-DerivativesProvider*. The individual contributions to it (and to the Hamiltonian) from the different accelerations $\mathbf{a}_i$ are in the implementation of the interface *(Field)-CartesianAdjointTerm*. The native ones cover attraction from a point-mass body (central or not, as a third body or not) and the J_2 effect, as well as non-inertial forces (in a rotating frame).

2.1 Energy cost

From now on, let us assume that L is proportional to the squared Euclidean norm of thrust, with typically a scaling factor $\frac{1}{2}$, but the exact form can vary. It is sub-optimal regarding fuel consumption but solutions are easier to find. Next, let us go over the three versions implemented in Orekit, as inheritors of *AbstractCartesianEnergy*. Note that the interface *CartesianCost* is generic and can be built upon to emulate any user-defined function.

2.1.1 Negligible mass flow

If $\alpha = 0$, the mass is constant and can be dropped from the state vector, as well as the corresponding adjoint variable. It is then more convenient to choose the control vector as the acceleration $\mathbf{u} = \frac{\mathbf{T}}{m}$ rather than the force, so that $L = \frac{1}{2}|\mathbf{u}|$. Plugging this into Eq.(4), it becomes:

$$H = -\frac{1}{2}|\mathbf{u}|_2^2 + < \mathbf{p}_v, \mathbf{u} > + < \mathbf{p}_r, \mathbf{v} > + < \mathbf{p}_v, \sum_i \mathbf{a}_i > \quad (7)$$

The maximization readily gives $\mathbf{u} = \mathbf{p}_v$. Note that an advantage of this choice of control vector is that this relationship does not depend on the mass at all. Another one is that the no-control initial guess $\mathbf{p} = 0$ can be used .

2.1.2 Unbounded thrust

Let us go back to $\alpha \neq 0$ and the control vector being $\mathbf{T}$, so that $L = \frac{1}{2}|\mathbf{T}|_2^2$. The Hamiltonian is a second order polynomial w.r.t. the thrust magnitude. Let us define the switching function $S = \frac{|\mathbf{p}_v|_2}{m} - \alpha p_m$, so that:

$$H = -\frac{1}{2}\left(|\mathbf{T}|_2 - S\right)^2 + \frac{1}{2}S^2 + < \mathbf{p}_r, \mathbf{v} > + < \mathbf{p}_v, \sum_i \mathbf{a}_i > \quad (8)$$

When S is negative, then maximizing H is equivalent to $|\mathbf{T}|_2 = 0$. Otherwise, $|\mathbf{T}|_2 = S$, so all in. It is now possible to derive the adjoint mass rate. Both cases boil down to the same equation:

$$\dot{p}_m = |\mathbf{T}|_2 \frac{|\mathbf{p}_v|_2}{m} - < \mathbf{p}_v, \sum_i \frac{\partial \mathbf{a}_i}{\partial m} > \quad (9)$$

In Orekit, to maintain propagation accuracy, events detectors are included internally to properly handle the singularities when the switches occur. This is why *CartesianCost* extends *EventDetectorsProvider*.

2.1.3 Bounded thrust

Let us now consider that the thrust magnitude has an upper bound $\bar{T}$. In this case, let us define the control vector as $\mathbf{T}/\bar{T}$. The maximization of the Hamiltonian is identical to the previous analysis, except when $S > \bar{T}$, in which case $|\mathbf{T}|_2 = \bar{T}$.

3 Shooting method

The PMP also gives so-called transversality equations, linked to the boundary conditions. If the terminal Cartesian variables are fixed, then the corresponding adjoint variables are free. If the terminal mass is free, then the adjoint mass must vanish. Thus for fixed times (t_0, t_f), fixed initial state $(\mathbf{x}_0, m_0)$ and fixed terminal Cartesian vector $\mathbf{x}_f$, then the only non-trivial transversality equation is $p_m(t_f) = 0$. The shooting approach consists in iterating on the value of $\mathbf{p}(t_0)$ from a guess via differential correction until the equations are satisfied. The single version uses propagation from t_0 to t_f (as opposed to splitting the interval for robustness). Orekit tackles this case with a simple Newton-Raphson update, as implemented in *NewtonFixedBoundaryCartesianSingleShooting*. Note that its ancestor class *AbstractIndirectShooting* is very generic and can be leveraged upon for custom use. The reader can refer to the *FixedBoundarySingleShooting* file in the Orekit tutorials repository for examples. In particular, it is shown how the solver can be used in a sequence, adding up constraints, using the obtained adjoint vector as initialization for the next one, starting with the cost from 2.1.1 and a no-control guess $\mathbf{p} = 0$. Note that the shooting method works with both *Orbit*-based (with *CARTESIAN OrbitType*) and *AbsolutePVCoordinates* numerical propagation.

References

[1] M. Cerf. Optimization techniques ii: Discrete and functional optimization. In *Optimization Techniques II*. EDP Sciences, 2023.

[2] G. Colasurdo and L. Casalino. Indirect methods for the optimization of spacecraft trajectories. *Modeling and Optimization in Space Engineering*, pages 141–158, 2013.

[3] E. Trélat. Optimal control and applications to aerospace: some results and challenges. *Journal of Optimization Theory and Applications*, 154:713–758, 2012.