Hampath – on solving optimal control problems by indirect and path following methods

STRUCTURAL DYNAMICAL SYSTEMS : Computational Aspects SDS2010

O. Cots

in collaboration with J.-B. Caillau and J. Gergaud

June 2010
Motivation

- LEO - GOE transfer: $P_0 = 11.625$ Mm, $e_0 = 0.75$ et $i_0 = 7^\circ$
- Initial mass: $m_0 = 1500$ kg
- Low thrust: $T_{\text{max}} = 0.1$ N
- Costs
  - Minimization of $t_f$
  - Maximization of the final mass
  - Minimization of the “energy”
Equations and results

\( (P) \) \(
\begin{align*}
\min \int_0^{t_f} |u| dt & \iff \max m(t_f) \\
\dot{r} &= v \\
\dot{v} &= -\frac{\mu_0}{|r|^3} r + T_{\text{max}} \frac{u}{m} \\
\dot{m} &= -\beta T_{\text{max}} |u| \\
|u| &\leq 1
\end{align*}
\)

Initial and final conditions
\( t_f \) free

\(|u|:\) euclidian norm

**Figure:** Optimal trajectory and norm of the optimal control for \( T_{\text{max}} = 10\text{N} \)
Issue  The control $u$ is discontinuous (bang-bang solution)

$\Rightarrow$  (BVP) with discontinuities

Idea  Regularize and converge to the problem.
1. Optimal control overview

2. Differential homotopy method - hampath

3. Back to the orbital transfer

4. Conclusion
Problems solved

\( P_{\lambda} \) \left\{ \begin{array}{l}
\text{Min} \quad g(x(t_f), \lambda) + \int_{0}^{t_f} f^0(x(t), u(t), \lambda) dt \\
\dot{x}(t) = f(x(t), u(t), \lambda) \quad , \quad u(t) \in U \subset \mathbb{R}^m \quad , \quad \lambda \in \mathbb{R} \\
b_0(x(0), \lambda) = 0 \in \mathbb{R}^{n_0} \quad , \quad n_0 \leq n \\
b_f(x(t_f), \lambda) = 0 \in \mathbb{R}^{n_1} \quad , \quad n_1 \leq n
\end{array} \right. 

**Definition**

*The Hamiltonian function is*

\[
H : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \times \mathbb{R}^- \times \mathbb{R}^n \longrightarrow \mathbb{R} \\
(x, u, \lambda, p^0, p) \quad \mapsto \quad H(x, u, \lambda, p^0, p) \\
H(x, u, \lambda, p^0, p) = p^0 f^0(x, u, \lambda) + < p, f(x, u, \lambda) >
\]
Hypothesis

- Normal case: \( p^0 \neq 0 \Rightarrow p^0 = -1 \)

- All functions are smooth

- The maximization of the Hamiltonian gives a smooth function
  \[ u(x, p, \lambda) = \operatorname{argmax}_{w \in U} H(x, w, \lambda, p) \]

Definition

*We call true Hamiltonian the function*

\[ H_r(x, \lambda, p) = H(x, u(x, p, \lambda), \lambda, p) \]
Example

\[
(P_\lambda) \begin{cases}
\min \int_0^1 u^2 dt \\
\dot{x} = -\lambda x + u \\
x(0) = -1 \\
x(1) = 0
\end{cases}
\]

The Hamiltonian is \( H(x, u, \lambda, p) = -u^2 + p(-\lambda x + u) \),

The maximization of the Hamiltonian gives \( u(x, p, \lambda) = p/2 \),

the true Hamiltonian is \( H_r(x, \lambda, p) = \frac{p^2}{4} - \lambda xp \)

and the *Principle of the Maximum of Pontryagin* (PMP) does not give conditions on \( p(0) \) nor \( p(1) \) (because \( x(0) \) and \( x(1) \) are known).
Shooting method

\[ (P_\lambda) \]

\[ (PMP) \]

\[ (BVP_\lambda) \]

\[ \begin{align*}
\dot{x} &= -\lambda x + \frac{p}{2} \\
\dot{p} &= \lambda p \\
x(0) + 1 &= 0 \\
x(1) &= 0
\end{align*} \]

\[ (\text{Shooting method}) \]

\[ S(x(0), p(0), \lambda) = \begin{pmatrix} x(0) + 1 \\ x(1) \end{pmatrix} = 0 \]

with \( x(1) \) obtained from integration (it depends on \( x(0), p(0) \) and \( \lambda \)).
Generally

$$(P_\lambda)$$

(PMP)

$$\begin{cases}
(\dot{x}, \dot{p}) = \tilde{H}(x, \lambda, p) = \left( \frac{\partial H_r(x, \lambda, p)}{\partial p}, -\frac{\partial H_r(x, \lambda, p)}{\partial x} \right)

b_0(x(0), p(0), \lambda) = 0 \in \mathbb{R}^n

b_f(x(t_f), p(t_f), \lambda) = 0 \in \mathbb{R}^n
\end{cases}$$

(Shooting method)

$$S(x_0, p_0, \lambda) = \begin{pmatrix} b_0(x_0, p_0, \lambda) \\ b_f(x_f, p_f, \lambda) \end{pmatrix} = 0 \quad (S : \mathbb{R}^{2n+1} \to \mathbb{R}^{2n})$$

with $(x_f, p_f) \equiv (x(t_f, x_0, p_0, \lambda), p(t_f, x_0, p_0, \lambda)) = \exp_{t_f} \tilde{H}(x_0, \lambda, p_0)$
1 Optimal control overview

2 Differential homotopy method - hampath

3 Back to the orbital transfer

4 Conclusion
Introduction to the homotopy

We want to compute the zeros path of the homotopic function

\[ S : \mathbb{R}^{2n} \times [0, 1] \rightarrow \mathbb{R}^{2n} \]

\[(z_0, \lambda) \mapsto S(z_0, \lambda)\]

\(S\) is nonlinear and smooth. Assuming that 0 is a regular value for \(S\), the solution set of \(S(c) = 0\) is a smooth curve.

Path following techniques:

- Prediction-Correction (Allgower and Georg. [2])
- Path following by SVD (Dieci and al. [8])
- hampath for optimal control [4]
Assuming that \( c(s) = (z_0(s), \lambda(s)) \) is such as

1. \( c(0) = (z_0^0, 0) \)
2. \( S(c(s)) = 0 \)
3. \( S'(c(s)) \) of full rank
4. \( \dot{c}(s) \neq 0 \)

then \( \dot{c}(s) = T(S'(c(s))) \) is determined by

1. \( S'(c(s))\dot{c}(s) = 0 \)
2. \( |\dot{c}(s)| = 1 \)
3. \( \det \begin{pmatrix} S'(c(s)) & t\dot{c}(s) \\ \end{pmatrix} \) is of constant sign
Differential algorithms


- **hampath** [4] uses DOPRI5 from E. Hairer and G. Wanner [6] [7], for the numerical integration of (without any correction) :

\[
(IVP) \begin{cases}
    \dot{c}(s) = T(S'(c(s))) \\
    c(0) = (z_0^0, 0)
\end{cases}
\]

Until \( s_f \) such as \( \lambda(s_f) = 1 \) (dense output).
\[ S(z_0, \lambda) = bc(z_0, z(t_f, z_0, \lambda), \lambda) \]

with

\[ bc(z_0, z_f, \lambda) = (b_0(z_0, \lambda), b_f(z_f, \lambda)) \]

\[ \Rightarrow S'(z_0, \lambda) = \left[ \frac{\partial bc}{\partial z_0} + \frac{\partial bc}{\partial z_f} \frac{\partial z}{\partial z_0} + \frac{\partial bc}{\partial z_f} \frac{\partial z}{\partial \lambda} \right] \]

- **hampath** uses TAPENADE [5] (from INRIA) for the automatic differentiation: \( \frac{\partial bc}{\partial z_0}, \frac{\partial bc}{\partial z_f} \) and \( \frac{\partial bc}{\partial \lambda} \) computation.

- **hampath** generates the variational equations to compute \( \frac{\partial z}{\partial z_0} \) and \( \frac{\partial z}{\partial \lambda} \).
\[ S'(z_0, \lambda) \]

\[ \frac{\partial z}{\partial z_0} (t_f, z_0, \lambda) \text{ is solution of:} \]

\[
\begin{aligned}
    (VAR) \quad 
    \begin{cases}
        \dot{Y}(t) = \frac{\partial \tilde{H}}{\partial z}(z(t, z_0, \lambda), \lambda)Y(t) \\
        Y(0) = I_{2n}
    \end{cases}
\end{aligned}
\]

with \( \frac{\partial \tilde{H}}{\partial z} \) and \( \tilde{H} = (\frac{\partial H_r}{\partial p}, -\frac{\partial H_r}{\partial x}) \) computed by automatic differentiation (TAPENADE).

Recall: \( z = (x, p) \) and \( H_r \) is the true Hamiltonian.

- Same thing for \( \frac{\partial z}{\partial \lambda} (t_f, z_0, \lambda) \) and \( \frac{\partial \tilde{H}}{\partial \lambda} \).
ssolve and expdhvfun methods

\[
\begin{align*}
\dot{y}(t) &= \frac{\partial H}{\partial z} y(t) \\
y(0) &= y_0
\end{align*}
\]

expdhvfun integrates the variational equations, with any initial condition. It is used to check the second order optimality conditions (cf [3] et [1]).

ssolve uses HYBRJ (Minpack library) as NLE solver. Besides hampath integrates the variational equations so that this diagram commutes (DOPRI5 has been modified):

- (IVP) \xrightarrow{\text{Numerical integration}} S(z_0, \lambda)
- (VAR) \xrightarrow{\text{Numerical integration}} \frac{\partial S}{\partial z_0}(z_0, \lambda)
User implementation (example)

\( bc(z_0, z_f, \lambda) \)

```
Subroutine bcfun(t0, tf, n, z0, zf, hf, lpar, par, nbc, bc)
  !Variable declarations
  bc = (/ z0(1)+1 , zf(1)/)
end subroutine bcfun
```

\( H_r(z, \lambda) \)

```
Subroutine hfun(t, n, z, lpar, par, H)
  !Variable declarations
  x = z(1)
  p = z(2)
  lambda = par(1)
  H = p**2/4 - lambda * x * p
end subroutine hfun
```

Available MATLAB functions after compiling

- `ssolve` to solve the problem at \( \lambda \) fixed
- `hampath` to follow the zeros path
- `expdhvfun` to check the optimality of the solution
Global diagram

\[
bcfun \\
bc(z_0, z_f, \lambda)
\]

\[
hfun \\
H_r(z, \lambda)
\]

\[
dbc \\
\vec{H} \\
\left( \frac{\partial H_r}{\partial p}, -\frac{\partial H_r}{\partial x} \right)
\]

\[
S(z_0, \lambda)
\]

\[
S'(z_0, \lambda)
\]

\[
T(S'(z_0, \lambda))
\]

\[
ssolve \\
shooting method
\]

\[
MATLAB functions \\
Available for use
\]

\[
hampath \\
continuation method
\]

\[
expdhvfun \\
variational equations
\]

FORTRAN 90

TAPENADE

DOPRI5

VAR

DOPRI5

HYBRJ

O. Cots (IMB Bourgogne)  Hampath  June 2010  19 / 29
1. Optimal control overview
2. Differential homotopy method - hampath
3. Back to the orbital transfer
4. Conclusion
Homotopy chosen

\[
\begin{align*}
\text{Min} & \quad \int_0^{t_f} |u| dt \\
\downarrow & \\
\text{Min} & \quad (1 - \lambda) t_f + \lambda \int_0^{t_f} |u| - (1 - \lambda) (\log |u| + \log(1 - |u|)) dt
\end{align*}
\]

- For \( \lambda = 0 \) the final time \( t_f \) is minimized;
- For \( \lambda = 1 \) the final mass is maximized;
- For \( \lambda \in [0, 1) \) the control \( u(x, p, \lambda) \) is smooth.
Zeros path - $\lambda$ vs costate
The control

$|u|$ for $\lambda \in \{0, 0.01, 0.05, 0.1, 0.25, 0.5, 0.75, 0.9, 0.99, 0.999\}$

$\lambda = 0$

$\lambda = 0.9$
The control

\[ \lambda = 0.999 \]

O. Cots (IMB Bourgogne)

Hampath

June 2010

23 / 29
Locally optimality (assuming $t_f$ fixed)

$$exp_{t,x_0,\lambda} : \mathbb{R}^n \longrightarrow \mathbb{R}^n$$

$p_0 \longmapsto x(t, x_0, p_0, \lambda)$

$(x_0, p_0)$ is a locally optimal solution

$$\Leftrightarrow \forall t \in (0, t_f] \quad \text{rank}(\frac{d}{dp_0}exp_{t,x_0,\lambda}(p_0)) = n$$

$$\Leftrightarrow \forall t \in (0, t_f] \quad \frac{\partial x}{\partial p_0}(t, x_0, p_0, \lambda) \in \mathbb{R}^{n \times n} \text{ is of full rank}$$

for $\lambda \in \{0, 0.01, 0.05, 0.1, 0.25, 0.5, 0.75, 0.9, 0.99, 0.999\}$
1. Optimal control overview

2. Differential homotopy method - hampath

3. Back to the orbital transfer

4. Conclusion
Points to remember

**hampath**

- is used in the case of smooth functions;
- permits to treat discontinuous optimal problems by smooth regularization;
- computes accurately the jacobian \( S'(z_0, \lambda) \);
- is easy to use since it has a MATLAB interface.
Other applications

Quantum mechanic problems

The particles are controlled by laser fields and interact with the environment.

- homotopy on physical parameters (characteristics of the particle).
- regularization by homotopy (to be smooth).

3 bodies problem

The satellite goes from the Earth to the moon.

- homotopy on the influence of the moon.
- homotopy on the maximal thrust.
- ...

Discontinuous case

**hampath** can be applied with smooth functions and permits to have a good approximation of a discontinuous problem.

- Associate the code SHOOT PACKAGE [9] (shooting method for the discontinuous case) with **hampath**.
- Study the second order optimality conditions in the discontinuous case (no complete results for now).

Continuation method and bifurcations

**hampath** computes at each step the jacobian $S'(z_0, \lambda)$ used to find the tangent vector. We want to integrate the work of DIECI AND AL. [8] to have a path following by smooth SVD.

- Update of the decomposition of the jacobian.
- Detection of bifurcation points.
References

A. Agrachev and Y. Sachkov.

E.L. Allgower and K. Georg.

J.-B. Caillau B. Bonnard and E. Trélat.

J.B. Caillau, O. Cots, and J. Gergaud.
HAMPATH [apo.enseeiht.fr/hampath](http://apo.enseeiht.fr/hampath).

A. Dervieux, L. Hascoet, and V. Pascal.

E. Hairer, S.P. Nørsett, and G. Wanner.

E. Hairer and G. Wanner.

M. G. Gasparo L. Dieci and A. Papini.
Path following by svd.

P. Martinon and J. Gergaud.
SHOOT PACKAGE [apo.enseeiht.fr/shoot](http://apo.enseeiht.fr/shoot).