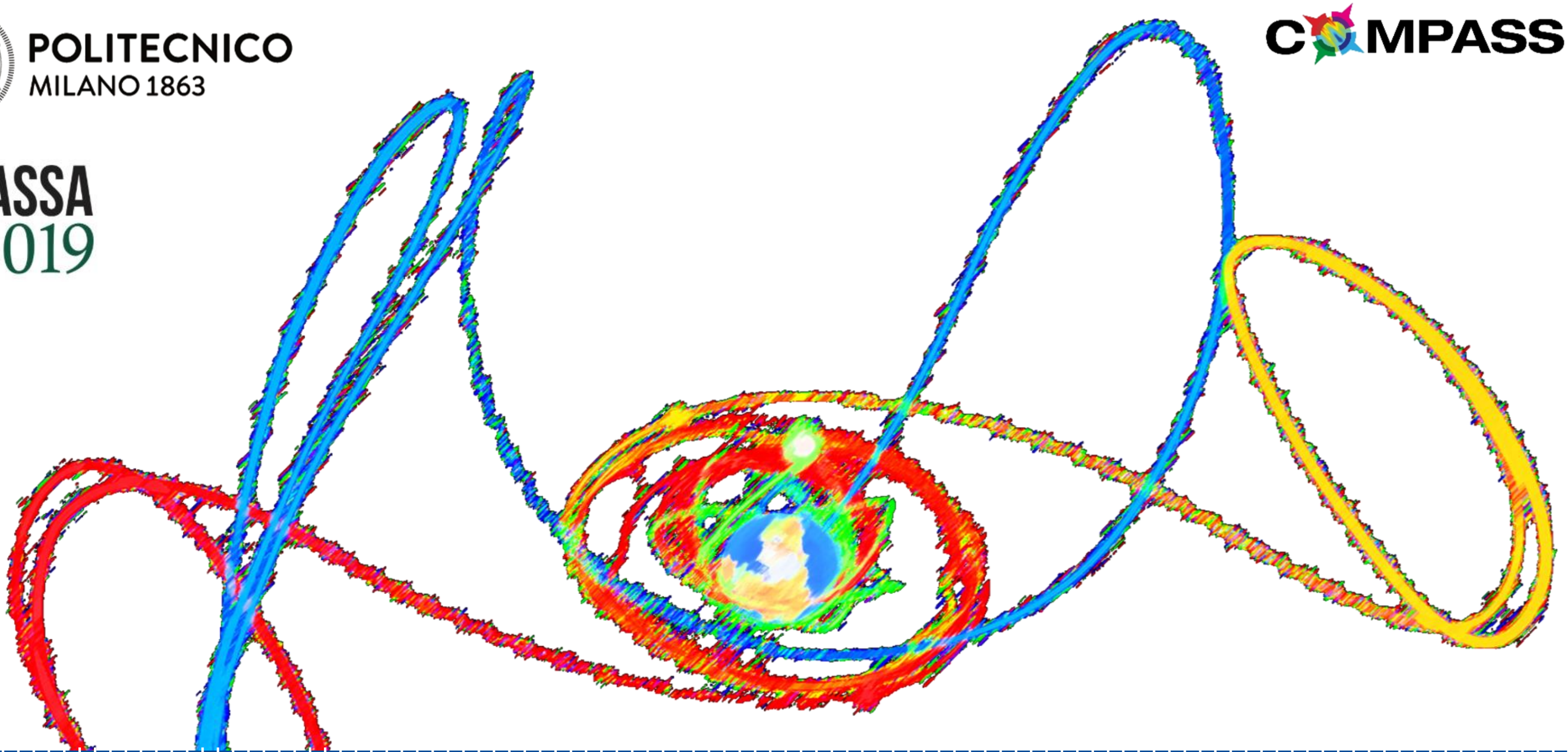
POLITECNICO MILANO 1863

KePASSA 2019

COMPASS erc



Introduction to a Keplerian-Orbital-Element-based optimisation approach via Differential Dynamic Programming

Marco Nugnes and Camilla Colombo

4th International Workshop on Key Topics in Orbit Propagation Applied to Space Situation Awareness (KePASSA)

Logroño, 24-26 April 2019

Outline

- Review of Differential Dynamic Programming (DDP)
- Orbital-Element-based DDP modelling
- Results
- Conclusion and future work

REVIEW OF DIFFERENTIAL DYNAMIC PROGRAMMING (DDP)

Differential Dynamic Programming

Introduction

- Differential Dynamic Programming (DDP) is one of the most recent techniques used for solving nonlinear optimal control problems.
- It is based on Bellman's Principle of Optimality which states [1]:

"An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision."

- This principle can be mathematically expressed using the following formulations

$$-\frac{\partial V(x,t)}{\partial t} = \min_u [L(x,u,t) + \langle V_x(x,t), f(x,u,t)\rangle],$$

where V is the optimal value function defined as:

$$V(x,t) = \int_{t_0}^{t} L(x,u,t)\, dt + F\big(x(t_f), t_f\big)$$

Differential Dynamic Programming

Main principle

- The previous partial differential equation is unsolvable analytically.
- The difficulty of the numerical solution lies in the high dimensionality of the equation.
- Differential Dynamic Programming tries to overcome the curse of dimensionality associated to Bellman's Principle of Optimality [2].
- The main idea is to apply Bellman's Principle of Optimality in the neighbourhood of a nominal, nonoptimal trajectory of the system.

Differential Dynamic Programming

Problem statement

- The problem to be solved is the following: considering a trajectory defined by $x_0, x_1, \dots, x_N$, which satisfies:

$$x_{i+1} = f(x_i, u_i, t_i)$$

- Determine the control vectors $u_0, u_1, \dots, u_{N-1}$ minimising the cost function:

$$\hat{V} = \sum_{i=1}^{N-1} L(x_i, u_i, t_i) + F(x_N)$$

- Subject to the vector equality constraint:

$$\vartheta(x_N) = 0$$

Differential Dynamic Programming

Constraint consideration

- The equality constraint is embedded in the value function using a set of Lagrange multipliers k

$$V(x_i, k, t_i) = \sum_{i=1}^{N-1} L(x_i, u_i, t_i) + F(x_N) + k^T \vartheta(x_N)$$

- The value function satisfies Bellman's Principle of Optimality, which in the discrete case is:

$$V(x_i, k, t_i) = \min_{u_i}[L(x_i, u_i, t_i) + V(x_{i+1}, k, t_{i+1})]$$

- It is possible to rewrite the previous expression in terms of displacements $\delta x_i, \delta x_{i+1}, \delta k$ in the following way:

$$V\left(\bar{x}_i + \delta x_i, \bar{k} + \delta k, t_i\right) = \min_{u_i}\left[L(\bar{x}_i + \delta x_i, u_i, t_i) + V\left(\bar{x}_{i+1} + \delta x_{i+1}, \bar{k} + \delta k, t_{i+1}\right)\right]$$

Differential Dynamic Programming

Taylor expansions

- It is assumed that $\delta x_i, \delta x_{i+1}, \delta k$ are small enough so that each term can be rewritten using Taylor expansions around the nominal trajectory terminated at second-order terms.

$$V\left(\bar{x}_i + \delta x_i, \bar{k} + \delta k, t_i\right) = \ a^i + \bar{V}^i + \ V_x^i \delta x_i + \ V_k^i \delta k + \frac{1}{2}\delta x_i^T V_{xx}^i \delta x_i + \delta x_i^T V_{xk}^i \delta k + \frac{1}{2}\delta k^T V_{kk}^i \delta k$$

$$L(\bar{x}_i + \delta x_i, t_i) = \ L^i + \ L_x^i \delta x_i + \frac{1}{2}\delta x_i^T L_{xx}^i \delta x_i$$

$$\delta x_{i+1} = \left(f^i(\bar{x}_i, u_{i,} t_i) - \bar{f}^i(\bar{x}_i, \bar{u}_{i,} t_i)\right) + \ f_x^i \delta x_i + \frac{1}{2}\delta x_i^T f_{xx}^i \delta x_i$$

$$V\left(\bar{x}_{i+1} + \delta x_{i+1}, \bar{k} + \delta k, t_{i+1}\right) = \ a^{i+1} + \ \bar{V}^{i+1} + \ V_x^{i+1} \delta x_{i+1} + \ V_k^{i+1} \delta k + \frac{1}{2}\delta x_{i+1}^T V_{xx}^{i+1} \delta x_{i+1} + \delta x_{i+1}^T V_{xk}^{i+1} \delta k + \frac{1}{2}\delta k^T V_{kk}^{i+1} \delta k$$

- Where $a = V \ - \ \bar{V}$ represents the difference between the optimal and nominal value function respectively.

Differential Dynamic Programming

Minimisation process

- Replacing everything in Bellman's Principle of Optimality, an expression to be minimised with respect to the control u_i is get.
- The minimisation is performed in two stages:
  - First, set $\delta x_i = 0$ and $\delta k = 0$ to compute the value of a pseudo-optimal control u^*
  - Perform a Taylor expansion of the previous terms about the pseudo-optimal control with $u_i = u^* + \delta u_i$
  - The second stage is accomplished when the new expansion is minimised with respect to δu_i leading to the following feedback control law:

$$\delta u_i = \beta_1 \delta x_i + \beta_2 \delta k$$

$$\beta_1 = -\Delta^{-1}\left[H_{ux}^i + f_u^{i^T} V_{xx}^{i+1} f_x^i + \left(f^i - \bar{f}^i\right)^T V_{xx}^{i+1} f_{ux}^i\right] \qquad \beta_2 = -\Delta^{-1} f_u^{i^T} V_{xk}^{i+1}$$

$$\Delta = H_{uu}^i + f_u^{i^T} V_{xx}^{i+1} f_u^i + \left(f^i - \bar{f}^i\right)^T V_{xx}^{i+1} f_{uu}^i \qquad H^i = L^i + V_x^{i+1} f^i$$

Differential Dynamic Programming

Difference equations

- Substituting all the expressions in Bellman's Principle of Optimality and equating the coefficients of like powers of δx_i and δk, the following difference equations are obtained [3]:

$$a^i = a^{i+1} + H^i - \bar{H}^i + \tfrac{1}{2}\left(f^i - \bar{f}^i\right)^T V_{xx}^{i+1}\left(f^i - \bar{f}^i\right) \qquad a^N = 0$$

$$V_x^i = H_x^i + \left(f^i - \bar{f}^i\right)^T V_{xx}^{i+1} f_x^i \qquad V_x^N = F(\bar{x}_N) + \bar{k}\vartheta_x(\bar{x}_N)$$

$$V_k^i = V_k^{i+1} + \left(f^i - \bar{f}^i\right)^T V_{xk}^{i+1} \qquad V_k^N = \vartheta^T(\bar{x}_N)$$

$$V_{xk}^i = f_x^{i^T} V_{xk}^{i+1} - \beta_1^T \Delta \beta_2 \qquad V_{xk}^N = \vartheta_x^T(\bar{x}_N)$$

$$V_{kk}^i = V_{kk}^{i+1} - \beta_2^T \Delta \beta_2 \qquad V_{kk}^N = 0$$

$$V_{xx}^i = H_{xx}^i + f_x^{i^T} V_{xx}^{i+1} f_x^i + \left(f^i - \bar{f}^i\right)^T V_{xx}^{i+1} f_{xx}^i - \beta_1^T \Delta \beta_1 \qquad V_{xx}^N = F_{xx}(\bar{x}_N) + \bar{k}^T \vartheta_{xx}(\bar{x}_N)$$

- These difference equations should be integrated backward starting from the set of final conditions.

Differential Dynamic Programming

Equality constraint check

- After the minimisation of the cost function in the unconstrained case, it is necessary to check if the equality constraint is satisfied.
- If this does not occur a new value for the Lagrange multipliers is computed by using the maximisation of the value function which leads to this formulation:

$$\delta k^T = -\varepsilon {V_{kk}^{0}}^{-1} \vartheta(x_N)$$

- The parameter $0 < \varepsilon \leq 1$ is introduced to modulate the size of the δk produced by the previous equation.
- The Lagrange multipliers are updated and a new stage of the algorithm can start until the equality constraint is respected and the cost is minimised.

ORBITAL-ELEMENT-BASED DDP MODELLING

Orbital-Element-Based DDP

Introduction

- The main idea is to apply the Differential Dynamic Programming algorithm to problems where the state is defined in terms of orbital elements (for example classical Keplerian elements).
- In this way it is possible to get a more physical interpretation of the problem since it is easy to understand how the orbit is changing and what its shape is.
- Moreover, it is interesting to compare how the DDP works for different state representations and check which are the positive and negative aspects associated to the two formulations.

Orbital-Element-Based DDP

Dynamics definition

- The first requirement for the DDP algorithm is a formulation of the Dynamics in terms of Keplerian orbital elements.
- For this purpose, Gauss' variational equations have been considered, in such a way that they can be applied also for non-conservative effects [4].

$$\frac{da}{dt} = \frac{2a^2v}{\mu}a_t$$

$$\frac{de}{dt} = \frac{1}{v}\left[2(e+\cos f)a_t - \frac{r}{a}\sin f\, a_n\right]$$

$$\frac{di}{dt} = \frac{r\cos(\omega+f)}{h}a_h$$

$$\frac{d\Omega}{dt} = \frac{r\sin(\omega+f)}{h\sin i}a_h$$

$$\frac{d\omega}{dt} = \frac{1}{ev}\left[2\sin f\, a_t + \left(2e+\frac{r}{a}\cos f\right)a_n\right] - \frac{r\sin(\omega+f)\cos i}{h\sin i}a_h$$

$$\frac{df}{dt} = \frac{h}{r^2} - \frac{1}{eh}\left[2\sin f\, a_t + \left(2e+\frac{r}{a}\cos f\right)a_n\right]$$

Orbital-Element-Based DDP

Updated dynamics

- It is better to express each term explicitly as a function of the six orbital elements and replace the perturbing accelerations with the control components.

$$\frac{da}{dt} = 2\sqrt{\frac{a^3(1+2e\cos f + e^2)}{\mu(1-e^2)}}\,u_t$$

$$\frac{de}{dt} = \sqrt{\frac{a(1-e^2)}{\mu(1+2e\cos f+e^2)}}\left[2(e+\cos f)u_t - \frac{(1-e^2)\sin f}{1+e\cos f}u_n\right]$$

$$\frac{di}{dt} = \sqrt{\frac{a(1-e^2)}{\mu}}\frac{\cos(\omega+f)}{1+e\cos f}u_h$$

$$\frac{d\Omega}{dt} = \sqrt{\frac{a(1-e^2)}{\mu}}\frac{\sin(\omega+f)}{(1+e\cos f)\sin i}u_h$$

$$\frac{d\omega}{dt} = \frac{1}{e}\sqrt{\frac{a(1-e^2)}{\mu(1+2e\cos f+e^2)}}\left[2\sin f\,u_t + \left(2e+\frac{1-e^2}{1+e\cos f}\cos f\right)u_n\right] - \sqrt{\frac{a(1-e^2)}{\mu}}\frac{\sin(\omega+f)\sin i}{(1+e\cos f)\cos i}u_h$$

$$\frac{df}{dt} = \sqrt{\frac{\mu}{a^3(1-e^2)^3}}(1+e\cos f)^2 - \frac{1}{e}\sqrt{\frac{a(1-e^2)}{\mu(1+2e\cos f+e^2)}}\left[2\sin f\,u_t + \left(2e+\frac{1-e^2}{1+e\cos f}\cos f\right)u_n\right]$$

Orbital-Element-Based DDP

Constraint definition

- After the assessment of the dynamics, it is necessary to express also the equality constraint in terms of orbital elements.
- The easiest way is to define a final condition given in terms of orbital elements and consider the following implicit equations:

$$\vartheta_1 = a - a_f = 0$$
$$\vartheta_2 = e - e_f = 0$$
$$\vartheta_3 = i - i_f = 0$$
$$\vartheta_4 = \Omega - \Omega_f = 0$$
$$\vartheta_5 = \omega - \omega_f = 0$$
$$\vartheta_6 = f - f_f = 0$$

- However, these equations imply that there is a rendez-vous, since it is imposed an equality constraint both on the position and the velocity vectors.

POLITECNICO MILANO 1863

RESULTS

Results

Test case definition

- An interplanetary transfer to Mars has been considered to test the algorithm.
- The problem is defined both in Cartesian formulation and Keplerian elements version in order to compare the results.
- Moreover, the problem will be first solved in the unconstrained case and then the constraint will be added (for which optimal results are still not available).

Results

Cartesian formulation

- Dynamics

$$\begin{cases} \dot{r}_x = v_x \\ \dot{r}_y = v_y \\ \dot{r}_z = v_z \\ \dot{v}_x = -\dfrac{\mu}{\left(r_x^2 + r_y^2 + r_z^2\right)^{3/2}} r_x + u_x \\ \dot{v}_y = -\dfrac{\mu}{\left(r_x^2 + r_y^2 + r_z^2\right)^{3/2}} r_y + u_y \\ \dot{v}_z = -\dfrac{\mu}{\left(r_x^2 + r_y^2 + r_z^2\right)^{3/2}} r_z + u_z \end{cases}$$

- Final condition: Mars orbit

$$\begin{cases} x_f = x_{Mars} \\ y_f = y_{Mars} \\ z_f = z_{Mars} \end{cases}$$

- Constraint definition

$$\theta = \begin{cases} r_{x_N} - x_f = 0 \\ r_{y_N} - y_f = 0 \\ r_{z_N} - z_f = 0 \end{cases}$$

- Cost function

$$V = \sum_{j=0}^{N-1} \left\| u_j \right\|^2 + k^T \vartheta$$

Results

Theoretical considerations

- The first step of the minimisation procedure of the algorithm consists in finding the pseudo-optimal control u^*. It is given by the minimum of this expression:

$$\min_{u_i}\left[L^i + a^{i+1} + \bar{V}^{i+1} + V_x^{i+1}\left(f^i - \bar{f}^i\right) + \frac{1}{2}\left(f^i - \bar{f}^i\right)^T V_{xx}^{i+1}\left(f^i - \bar{f}^i\right)\right]$$

- Since the previous quantity is at a minimum when evaluated at u^*, its first derivative with respect to u_i should be zero.

$$H_u^i + \left(f^i - \bar{f}^i\right)^T V_{xx}^{i+1}\left(f^i - \bar{f}^i\right) = 0$$

- The derivative of the Hamiltonian is further explored:

$$H_u^i = L_u^i + V_x^{i+1} f_u^i$$

- Only the last three equations in the equations of motion have the derivative with respect to the control different from 0.

Results

Theoretical considerations

- The term V_x^{i+1} is simply equal to the vector of Lagrangian multiplier.

$$V_x^{i+1} = k^T$$

- Analysing the difference equations for the V_{xx}^{i+1} and the coefficient β_1

$$V_{xx}^i = H_{xx}^i + f_x^{i^T} V_{xx}^{i+1} f_x^i + \left(f^i - \bar{f}^i\right)^T V_{xx}^{i+1} f_{xx}^i - \beta_1^T \Delta \beta_1$$

$$\beta_1 = -\Delta^{-1} \left[H_{ux}^i + f_u^{i^T} V_{xx}^{i+1} f_x^i + \left(f^i - \bar{f}^i\right)^T V_{xx}^{i+1} f_{ux}^i\right]$$

- H_{ux}^i is identically equal to 0 and also H_{xx}^i is identically equal to 0 for the position constraint problem, since its formulation is given by:

$$H_{xx}^i = L_{xx}^i + V_x^{i+1} f_{xx}^i$$

- The f_{xx}^i matrix has non-zero elements corresponding to the derivatives of the last three equations of the dynamics but these derivatives multiply null terms.
- The result is that $V_{xx}^i = 0$ and since the final condition is equal to 0, this term is also identically equal to 0.

Results

Practical consequences

- This result has effects also on the evaluation of the coefficient β_1 which becomes always equal to 0.
- This means that the first part of the optimisation process considering $\delta k = 0$ occurs in a single iteration.
- This property is lost if a constraint on the velocity is introduced or if the constraint is expressed into a different form (for example with a quadratic form).

Results

Orbital elements formulation

- Dynamics: Gauss' variational equations
- Initial condition: satellite Keplerian elements

$$kep_0 = [a_0, e_0, i_0, \Omega_0, \omega_0, f_0]$$

- Final condition: Mars orbital elements

$$kep_f = [a_{Mars}, e_{Mars}, i_{Mars}, \Omega_{Mars}, \omega_{Mars}, f_{Mars}]$$

- Constraint definition

$$\vartheta = \begin{cases} a_N - a_{Mars} = 0 \\ e_N - e_{Mars} = 0 \\ i_N - i_{Mars} = 0 \\ \Omega_N - \Omega_{Mars} = 0 \\ \omega_N - \omega_{Mars} = 0 \\ f_N - f_{Mars} = 0 \end{cases}$$

- Cost function

$$V = \sum_{j=0}^{N-1} \left\| u_j \right\|^2 + k^T \vartheta$$

Results

Theoretical considerations

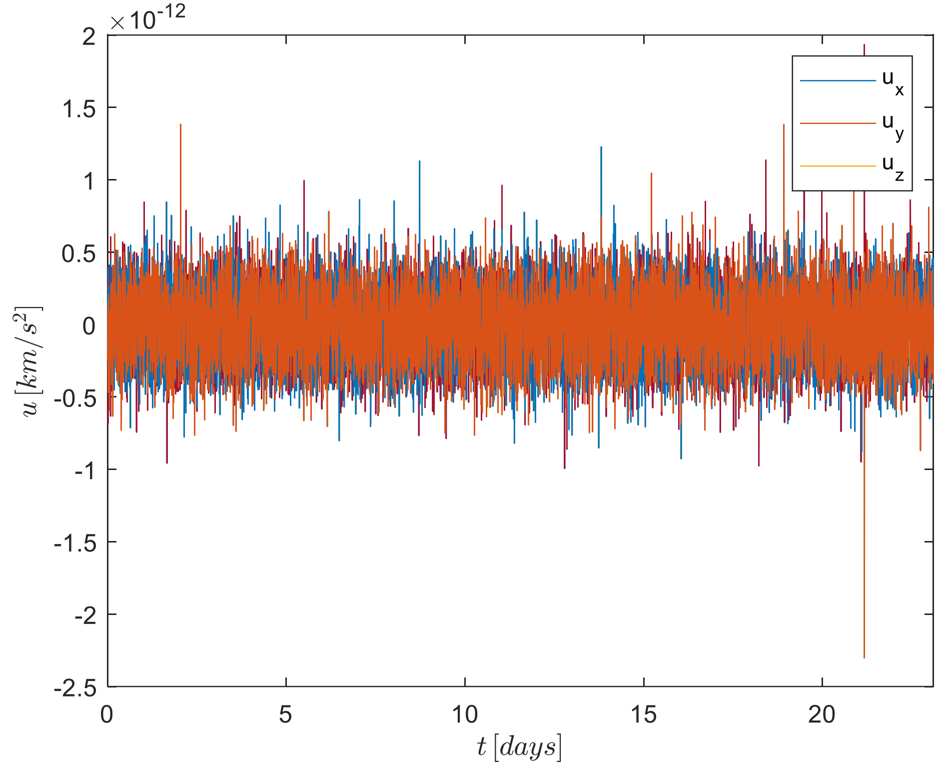
- The only constraint that can be neglected is the last one on the true anomaly, in order to achieve the requirement to be on the final orbit (even if the position may be different from the one desired).
- This means that the minimum number of constraints to be considered is 5 and all the properties defined for the Cartesian representation are lost.
- Moreover, one of the main difficulties in using the Keplerian representation is to establish a proper tolerance for the constraint achievement since the semi-major axis has a different order of magnitude with respect to the other elements.

Results

Unconstrained problem

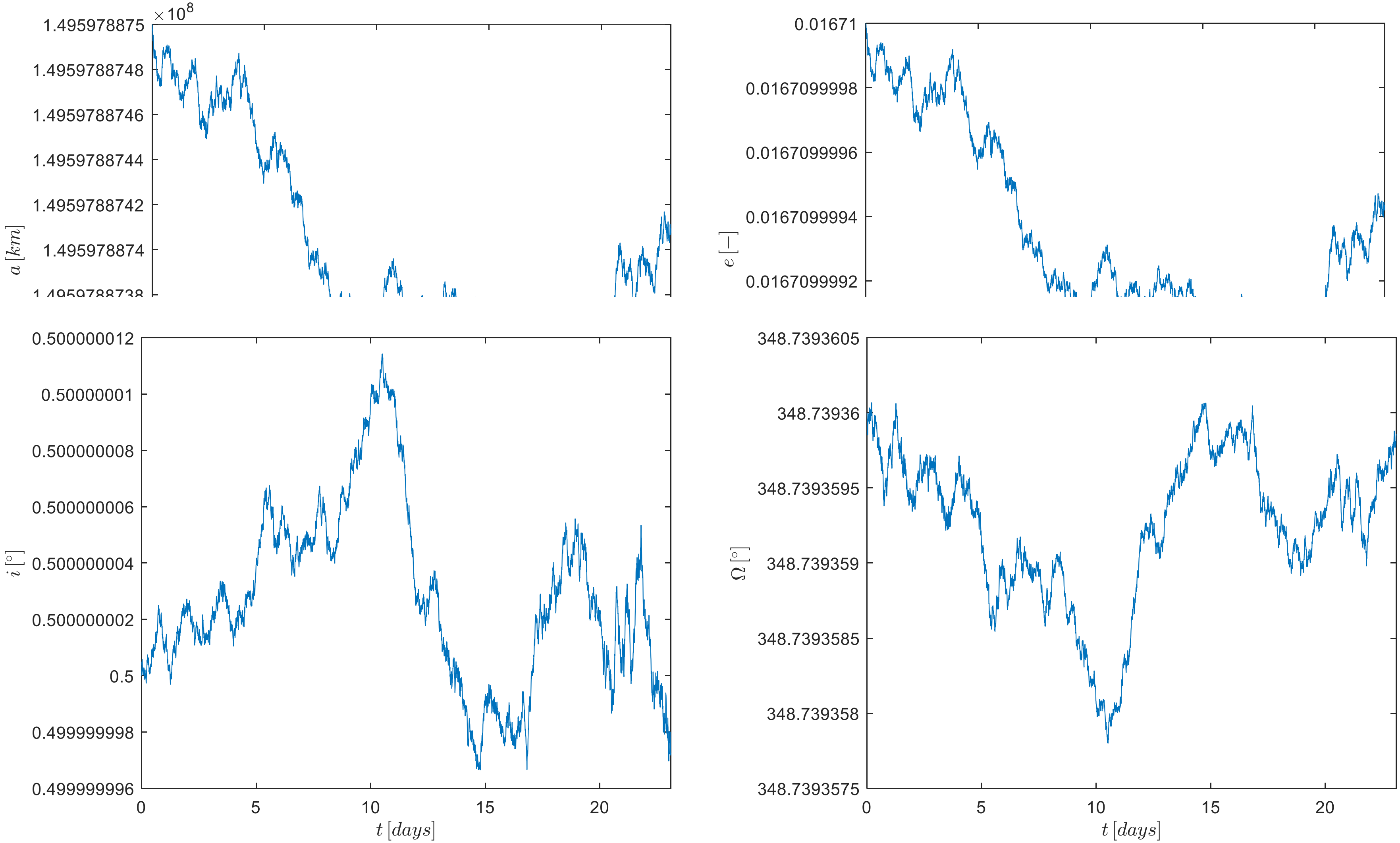
- Both for the Cartesian representation and the Keplerian version, the unconstrained problem gives as optimal solution $u = 0$.

- This solution is correct since the system of equations defined by the partial derivatives of the Hamiltonian with respect to the control becomes a homogeneous system.



Results

Unconstrained problem: elements variation



Conclusion and future work

- An introduction to a Keplerian-Orbital-Element-based optimisation approach via Differential Dynamic Programming has been investigated.
- The algorithm procedure has been kept the same, considering only a new formulation for the dynamics (Gauss' variational equations) and for the equality constraints.
- Cartesian representation shows numerical advantages in the unconstrained part of the optimisation.
- New formulations of the constraints will be considered to get better convergence.
- This kind of procedure suits particularly for low-thrust applications since it considers a continuous control thrust.
- A further improvement of the technique will be the introduction of state transition matrix in terms of Keplerian elements to map the partial derivatives.

Bibliography

[1] Bellman, R., E., *Dynamic Programming*, Princeton University Press, Princeton, 1957.

[2] Jacobson, D., H., and Mayne, D., Q., *Differential Dynamic Programming,* Elsevier, New York, 1970.

[3] Gershwin, S.,B., and Jacobson, D., H., "A discrete-time differential dynamic programming algorithm with application to optimal orbit transfer", *AIAA Journal*, Vol. 8, No. 9, 1970, pp. 1616-1626. https://doi.org/10.2514/3.5955

[4] Battin, R., H., *An Introduction to the Mathematics and Methods of Astrodynamics,* Revised Edition, AIAA, Education Series, Reston, Virginia, 1999.

POLITECNICO
MILANO 1863

COMPASS erc

This project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No 679086 – COMPASS)

Introduction to a Keplerian-Orbital-Element-based optimisation approach via Differential Dynamic Programming

Marco Nugnes, marco.nugnes@polimi.it

Camilla Colombo, camilla.colombo@polimi.it

@COMPASS_ERC

www.compass.polimi.it