POLITECNICO DI MILANO

Facoltà di Ingegneria Industriale

Corso di Laurea Specialistica in Ingegneria Spaziale

SYSTEMATIC STATE DEPENDENT RICCATI EQUATION OPTIMAL CONTROL APPLIED TO A RENDEZ-VOUS AND DOCKING PROCEDURE

Relatore: Prof. Franco BERNELLI ZAZZERA

Tesi di Laurea di:
Yann DARMAILLAC 764703

Anno Accademico 2011 – 2012

Acknowledgements

I would like to dedicate some words to several persons without who this thesis would never be possible.
Firstly, I have to thank the persons who supported me and pushed me to dedicate my studies to space engineering, including my teachers, my family and my friends.
I would also like to thank my universities for their double degree agreement I have got the great chance to benefit, which is reaching its end with this work.
Concerning the following study, I would like to thank my supervisor Franco Bernelli who helped me in the choice of my subject and led me during this work, my friends who reviewed my work and everybody who intervened in many different ways.

Table of Contents

List of figures

List of tables

List of abbreviations

SDRE	State Dependent Riccati Equation
LQR	Linear Quadratic Regulator
LVLH	Local Vertical – Local horizontal reference frame
ESA	European Space Agency
NASA	National Aeronautics and Space Administration
ATV	Automated Transfer Vehicle
ISS	International Space Station
SDC	State Dependent Coefficient
NQR	Nonlinear Quadratic Regulator

Abstract

The linear optimal control theory brought a lot in many fields, by providing proven methods that can be applied to several systems, the class of linear/linearized systems, in order to reach an optimal behavior according to the requirement defined by the control designer. During the last two decades, several studies aimed to develop similar methods that can be used for a larger class of systems, by taking in account the non linearities. This is the principle of the State Dependent Riccati Equation method.
The first step of this method is to give a parameterization of the control problem, that is to say to go from a system where the governing equations are known to a specific factorized structure, with a matrix multiplying a vector composed by the parameters to control. This matrix is not easy to obtain and generally implies to manipulate the equations to find a proper expression. Besides, this matrix is not unique, fact which imposes a selection process.
The aim of this study is to propose a systematic method which takes a set of equations to produce a parameterization, without the need to rewrite the equations in a proper way. For that, the approximation of the system equations was considered. Indeed, by having a polynomial expression of the equations, the factorization, even if non unique, is really easy. Thus a large part of this study was to evaluate the loss of information due to this approximation and justify the relevancy of the systematic method. A second part was dedicated to the selection process among the obtained parameterization in order to give to the control designer the values he will need to choose among the different possibilities.
Among the different space problems, the space navigation is already subjected to research for SDRE control application, including rendezvous and docking procedures. The systematic method had been implemented in a SDRE rendezvous solver developed during a previous thesis, in order to prove the efficiency of the method by comparing the performances of the systematic parameterization with the ones obtained by the exact parameterization considered in the solver.
Unfortunately, the simulations showed that this specific problem was not the best application since the impact of the parameterization is negligible face to the control law to follow. Because of that, any parameterization, even null, was reaching the same performances. But in any case, the systematic method code, justified with theoretical considerations, is developed and can be tested on another control problem.

Keyword: Nonlinear control, State-Dependent Riccati Equation, SDRE, SDC parameterization, Optimal control, systematic control method, rendezvous.

Sommario

I metodi di controllo ottimo per sistemi lineari sono stati molto importanti in diversi campi, dando la possibilità di controllare una grande classe di sistemi in modo ottimale, facendo in modo cioè che il sistema segua la prestazione imposta dal progettista. Gli ultimi decenni hanno visto lo sviluppo di tanti metodi adatti per sistemi più complessi tenendo in conto anche le non linearità. Questo è anche l'obiettivo del metodo 'State Depedent Riccati Equation".
Il primo passo consiste nella parametrizzazione del problema di controllo, è necessario cioè di trasformare le equazioni del sistema in una forma fattorizzata: in forma matriciale il sistema viene rappresentato da una matrice moltiplicata per il vettore dei parametri da controllare.
Il procedimento per trovare questa matrice non è banale: si devono manipolare le equazioni per ottenere un espressione utile allo scopo, inoltre, questa matrice non è unica, e risulta quindi necessario un processo di selezione.
Lo scopo di questo studio è quello di sviluppare un metodo automatico per ottenere una parametrizzazione da un sistema di equazioni, senza bisogno di ulteriori manipolazioni del sistema. Per questo, la scelta è quella di approssimare queste equazioni: ottenere un espressione polinomiale permette una fattorizzazione facile, pur non garantendo l'unicità.
Gran parte di questo lavoro è stato quindi dedicato alla valutazione dell'errore commesso a causa dell'approssimazione, e quindi di misurare la validità di questo metodo. La seconda parte del lavoro è stata incentrata sulla necessità di trovare metodi che permettessero di scegliere la parametrizzazione più adatta allo scopo, tra le diverse possibilità presenti.
Il metodo SDRE ha trovato applicazione nel campo dell'ingegneria spaziale nell'ambito della navigazione, in particolare è stato implementato per le procedure di rendezvous; un metodo automatico è stato sviluppato e implementato nell'algoritmo SDRE e applicato poi a una missione di rendezvous, nell'ambito di un'altra tesi. In questo modo è possibile misurare le prestazioni in termini di controllo del modello ottenuto con il metodo automatico, e fare quindi il confronto con le prestazioni ottenute con la parametrizzazione.
Le simulazioni hanno però mostrato che questa soluzione non era ideale per questo scopo, essendo l'effetto della dinamica trascurabile rispetto al controllo necessario per seguire la traiettoria prevista. Il codice è comunque disponibile, implementato e giustificato teoricamente; può quindi essere sfruttato per un altri tipi di problemi di controllo.

Parole chiave: Controllo non lineare, State-Dependent Riccati Equation, SDRE, SDC parametrizzazione, Controllo ottimale, metodo automatico di controllo, rendezvous.

Chapter 1

Introduction

1.1 Automatic control

The introduction of linear algebra in automatics brought some powerful results in term of system control. One of the main is to have developed theories assessing the feasibility of a controller and giving systematic design methods, not for a specific system, but for a large class of systems, the linear systems. Then, without any loss of generality, a deep knowledge of the system is not anymore required as long as it is known that the system respects few specific conditions. It is even possible to prepare algorithms working directly on the system equations to produce a systematic controller for which the efficiency is assured by the theory.

Among the control methods rising from the Linear control theory, the so called optimal control, or LQR method, allows designing a regulator according to some chosen/given criteria to optimize, a great advantage for an engineering problem.

Unfortunately, this theory concerns only linear systems, then any more complex system needs to be modeled, and linearized. This linearization means a loss of information in general and induces a different behavior between the model and the real system, especially when the nonlinearities are not really negligible. Last decade saw a growing interest in the adaptation of the linear theory for non linear-system in order to develop better controller usable thanks to the on-board computation power available nowadays.

The State Dependant Riccati Equation method is a trade off from the last studies on this subject, using a similar way to represent the system, but this time with matrices not anymore constant. This can be done by finding a factorization of the system, called quasi-linearization. But this process is not that easy, it requires researches and manipulations of the system equations.

That is why more and more researches are trying to define and test systematic procedures that can be used on a large class of problems without any loss of generality.

1.2 The rendezvous problem

Rendezvous and docking procedures are a key point in space operations, firstly due their necessity for in-space assembly, refueling and in general for manned missions, but also because they are among the most critical phases that one can find during a space mission. Indeed, a bad management of rendezvous and docking can simply lead to the incapacity to perform the contact between the two vehicles, or even worst, to a destructive collision. Unlike general space operations, dealing with standard celestial mechanics, those procedures are computed in the framework of the relative dynamics. Since the first operation of this kind, performed in the early 1966 by NASA, between a manned Gemini capsule with another unmanned vehicle, different methods and improvements were developed. One was to provide to some vehicle the capacity to perform those operations in an automatic way. This led to the first Russian automatic docking in 1967, to the new generation ATV performing the ISS refueling.

1.3 Objectives and scope

This study aims to find a systematic SDRE method that will be applied to a rendezvous procedure, by defining an algorithm taking the relative dynamics equation set to provide an approximated parameterization that can be used by a SDRE controller.
The model obtained will have to be compared to the real system behavior to assess its validity and to understand its limitations.
Finally the resulting model will be implemented in a rendezvous simulation on MATLAB®, developed during another thesis last year.
This would permit to measure the validity of this systematic method by applying it to a real case.

After a brief introduction to the linear control theory, Chapter 2 will present all the fundamentals of the SDRE method and the methods to implement it, in order to have the elements required to explain in Chapter 3 the design of the systematic method developed, from theoretical concepts to the used algorithms.

The application of the above mentioned method requires firstly setting the relative dynamics equations in Chapter 4, and then to explain what is a rendezvous procedure in Chapter 5.
Chapter 6 gives the model of the control problem suggested, and the evaluation of the compliancy of the systematic method to this problematic.
In conclusion, chapter 7 will be dedicated to the implementation of the obtained parameterization in the SDRE solver developed last year by another student.

Chapter 2

SDRE control technique

A lot of control problem can be faced with a linear controller. But on some complex systems, regarding to the non linearity of their governing equations, good performance can be obtained only by a non linear controller, widely more complex than the standard linear control problem due to the lack of a systematic design method. In the field of the nonlinear control problem, the 90s saw the rise of an optimal control based extended linearization method, the SDRE method, enabling to exploit the results of the LQR theory. This approach of the nonlinear quadratic regulator problem, NQR, can be seen as an extension of the linear case where the system matrices would not be anymore constants, without any approximation on the system to control. This chapter will firstly remind the fundamentals of the LQR theory [3] before to present the principles of the SDRE method developed by Cloutier, D'souza and Mracek [4] and summarized in [5] and [6].

2.1 Linear quadratic regulator

2.1.1 General framework

Considering a real linear/linearized time-invariant system, the standard analytical form to represent it is the following:

$$\dot{x} = Ax + B_u u + B_d d \qquad x(0) = x_0 \tag{2.1}$$

$$y = C_y x + D_{yu} u + D_{yd} d + D_{yr} r \tag{2.2}$$

$$z = C_z x + D_{zu} u + D_{zd} d \tag{2.3}$$

where the first differential equation represents the system dynamics under the disturbances d and the control u, while the second and third algebraic equations

are the measurement output y and the performance output z, r being the measurement noise. Under the time-invariant and linear system assumptions, all the matrices in those equations are constant and real.
The effect of the disturbances and noises won't be considered, simplifying the system:

$$\dot{x} = Ax + Bu \tag{2.4}$$
$$z = Cx + Du \tag{2.5}$$

The LQR principle is to control the dynamics and obtain a wished state using a linear state-feedback control u that respects a certain criterion, function of the problem requirement: the minimization of a quadratic cost function:

$$J(x_0, x_f, u) = \frac{1}{2}\left(x_f^t P_f x_f + \int_0^{t_f} (z^t W_{zz} z + u^t W_{uu} u) dt \right) \tag{2.6}$$

The matrices W_{uu} and W_{zz}, being symmetrical, are the design parameters to impose specific performances according to the project requirement.
On the other hand, the control law is written as a linear function of the state error, the difference between the command and the real state:

$$u = K(x_c - x) \tag{2.7}$$

where x_c is the command to follow and K the gain matrix. This K matrix is the key of the LQR method: it is the parameter assuring the minimization of (2.6) and the control of (2.4).

2.1.2 Minimization process

In order to obtain the expression of this parameter, (2.6) shall be rewritten as a function of the state and the command by introducing the expression of z, given in (2.5), in the cost function. Thus the following general form is obtained:

$$J = \frac{1}{2}\left(x_f^t P_f x_f + \int_0^{t_f} (x^t Q x + u^t R u + 2x^t S u) dt \right) \tag{2.8}$$

where the three real matrices Q, R and S are symmetrical. Using the Lagrange parameters method, the constraint $\lambda^t(Ax + Bu - \dot{x})$ shall be added in the

previous expression, λ^t being the Lagrange multiplier. Introducing the Hamiltonian, the criteria to minimize becomes:

$$J = \frac{1}{2}\left(x_f^t P_f x_f + \int_0^{t_f} (H(u,x,t,\lambda) - \lambda^t \dot{x}) dt \right) \tag{2.9}$$

$$H = x^t Q x + u^t R u + 2x^t S u + \lambda^t (Ax + Bu) \tag{2.10}$$

Finally, integrating per party the last term of the integral:

$$\int_0^T \lambda^t \dot{x} dt = [\lambda^t \dot{x}]_0^{t_f} - \int_0^T \dot{\lambda}^t x dt$$

Then the criterion to minimize becomes:

$$J = \frac{1}{2}\left(x_f^t P_f x_f + [\lambda^t x]_0^{t_f} + \int_0^{t_f} \left(H(u,x,t,\lambda) - \dot{\lambda}^t x\right) dt \right) \tag{2.11}$$

From the variational form of this criterion, the following set of equations, representing the minimization process, is obtained:

$$\begin{cases} \dfrac{\partial H}{\partial u} = 0 \\ \dfrac{\partial H}{\partial x} = -\dot{\lambda}^t \\ \dfrac{\partial H}{\partial \lambda} = \dot{x} \\ \dot{\lambda}^t(t_f) = 2P_f x\left(t_f\right) \end{cases} \tag{2.12}$$

The solution is obtained thanks to the expression of the Hamiltonian given in (2.10), reducing the problem to the determination of the pair (x, λ):

$$\begin{cases} u = -R^{-1}S^t x - \dfrac{1}{2} R^{-1} B^t \lambda \\ \dot{\lambda} = -2(Q - SR^{-1}S^t)x - (A^t - SR^{-1}B^t)\lambda \\ \dot{x} = (A - BR^{-1}S^t)x - \dfrac{1}{2} BR^{-1}B^t \lambda \\ \dot{\lambda}^t(t_f) = 2P_f x\left(t_f\right) \end{cases} \tag{2.13}$$

To make this set consistent, the assumption that **R is positive-definite** is required. The problem can be rewritten in the Hamiltonian sub-matrices form:

$$\begin{cases} \dot{x} = \bar{A}x - \frac{1}{2}BR^{-1}B^t\lambda \\ \dot{\lambda} = -2\bar{Q}x - \bar{A}^t\lambda \end{cases} \qquad \bar{A} = A - BR^{-1}S^t \qquad \bar{Q} = Q - SR^{-1}S^t \tag{2.14}$$

Unfortunately, x being defined with an initial condition and λ with a final condition, this system is pretty difficult to solve.

But adding the assumption that the pair $(\bar{A}, B)$ **is controllable** and the pair $(\bar{C}, \bar{A})$ **is observable**, with $\bar{C}$ a factorization of $\bar{Q}$ one can prove that:

$$\lambda(t) = 2P(t)x(t) \tag{2.15}$$

with P a $n\ x\ n$ matrix with the final condition: $P(t_f) = P_f$. It is important to note that the factorization $\bar{Q} = \bar{C}^t\bar{C}$ implies that $\bar{\boldsymbol{Q}}$ **is positive-semidefinite**.

2.1.3 Riccati Equation

Now, the solution of the minimization is reduced to the determination of this matrix P, called the Riccati matrix, the command u being obtained from $\frac{\partial H}{\partial u} = 0$ and (2.15), with the expression:

$$u(t) = R^{-1}\left(S^t + B^tP(t)\right)(x_c(t) - x(t)) \tag{2.16}$$

To obtain this matrix, the derivation of (2.15) introduced in (2.14) leads to the so called nonlinear differential Riccati equation:

$$P\bar{A} + \bar{A}^tP - PBR^{-1}B^tP + \bar{Q} = -\dot{P} \tag{2.17}$$

Facing a differential problem with a matrix as a variable, the solution is still complex to determine. Restricting the problem to the infinite horizon H∞, that is to say $t_f = \infty$, the previous equation is simplified:

$$P\bar{A} + \bar{A}^t P - PBR^{-1}B^t P + \bar{Q} = 0 \tag{2.18}$$

the so called Algebraic Riccati Equation (ARE), which admits at least one solution if the system is controllable/observable, and a unique one that stabilize the system in closed loop.

2.1.4 Summary

To conclude, for a linear/linearized system in the infinite horizon case, if:

- The pair $(\bar{A}, B)$ is controllable and the pair $(\bar{C}, \bar{A})$ is observable, with $\bar{C}$ a factorization of $\bar{Q}$,
- $\bar{Q}$ and R being respectively symmetrical positive-semi definite/definite,

it exists a unique solution u controlling the system and minimizing the cost function (2.6). This solution is expressed in (2.16).

2.2 Nonlinear quadratic SDRE regulator

2.2.1 The NQR problem

As showed before, the LQR method is a good way to control a linearized system by minimizing a criterion defined by the designer according to the requirement. But this restriction of linearization means a loss of information, a less accurate model from which the controller is designed. To avoid this, the nonlinear quadratic regulator approach is required, then facing the following system to control:

$$\begin{gathered} \dot{x}(t) = f\big(x(t)\big) + g\big(x(t), u(t)\big) \\ x(0) = x_0 \end{gathered} \tag{2.19}$$

The NQR method as an analog aim than the LQR, that is to say to define a control law u minimizing the criterion (2.6), for each x on the control domain Ω. To simplify the approach, the cross term will be neglected, that is to say S considered null, in the infinite horizon case:

$$J(x_0, u) = \frac{1}{2}\int_0^{\infty} (x^t Q x + u^t R u) dt \tag{2.20}$$

with the same conditions on Q and R, both symmetric and respectively positive-semi definite and positive-definite. Using the same development than in the previous section, analog equations are found for the minimization process:

$$\begin{cases} \dfrac{\partial H}{\partial u} = 0 = Ru + \dfrac{\partial g^t}{\partial u}(x,u)\lambda \\ \dfrac{\partial H}{\partial x} = -\dot{\lambda}^t = Qx - \dfrac{\partial f^t}{\partial x}(x)\lambda + \dfrac{\partial g^t}{\partial x}(x,u)\lambda \\ \dfrac{\partial H}{\partial \lambda} = \dot{x} = f(x) + g(x,u) \end{cases} \tag{2.21}$$

Then the control law is defined in an identical way than (3.16):

$$u(t) = -R^{-1}\frac{\partial g^t}{\partial u}\big(x(t), u(t)\big)\lambda(t) \tag{2.22}$$

where the dependence on u in both sides of the equation will be noted.

This is the general NQR problem considering constant weight matrices and only state dependant dynamics.

2.2.2 The SDRE method

From this point, the SDRE method is bringing a way to solve (2.21) to obtain the control law. The first step is to perform a quasi-linearization of the dynamics, thanks to a factorization process on Ω:

$$\begin{aligned} f(x) &= A(x)x \\ g(x,u) &= B(x,u)u \end{aligned} \tag{2.23}$$

Among the literature, this procedure is called extended linearization, apparent linearization or SDC parameterization [4], giving a linear-like structure of the system (2.19). The resulting factorization is clearly not unique when the state dimension is superior to 1.

This process requires working on the functions f and g to write them in a form easily factorable. Unfortunately this implies generally a human intervention since it is needed to have each part of f and g proportional to at least one of the state vector's components, so by mathematical manipulation or using some assumption on the system needing to be controlled.

This is one the main point of this study, define a process to obtain this factorization in a systematic way, and will be theoretically discussed in the next section. But it is know from [5] that if f is continuously differentiable on Ω, then such a factorization can be found.
Thanks to the expression (2.23), the SDRE method mimics the LQR method presented before, by seeking for a costate with the form:

$$\lambda(t) = P\big(x(t), u(t)\big)x(t) \tag{2.24}$$

As for the LQR, the controllability and observability of the pairs (A, B) and (C, A) is required, but for each $x \in \Omega$. The first equation of the minimization set, the, control law can then be written as:

$$u(x) = -R^{-1}(B^t(x,u)P(x,u)x - \sum_{i=1}^{k} u_i \left(\frac{\partial B_{1\to m,i}}{\partial u}(x,u)\right)^t P(x,u)x) \tag{2.25}$$

The derivation of the relation (3.24) reintroduced in the set (3.21) gives a Riccati-like equation:

$$\begin{aligned} P(x,u)A(x) + A^t(x)P(x,u) - P(x,u)B(x,u)R^{-1}B^t(x,u)P(x,u) + Q \\ = -\left[\dot{P}(x,u) + \sum_{i=1}^{k} x_i \left(\frac{\partial A_{1\to m,i}}{\partial x}(x,u)\right)^t P(x,u)\right. \\ + \sum_{i=1}^{k} u_i \left(\frac{\partial B_{1\to m,i}}{\partial x}(x,u)\right)^t P(x,u)) \\ \left. - P(x,u)B(x,u)R^{-1} \sum_{i=1}^{k} u_i \left(\frac{\partial B_{1\to m,i}}{\partial u}(x,u)\right)^t P(x,u))\right] \end{aligned} \tag{2.26}$$

By assuming the derivatives of the SDC are negligible, the previous expression is largely simplified [6]. The resulting formula is the SDRE. Thus the solution obtained will represent only a suboptimal control, the precision depending on the validity of the previous assumption. The SDRE system to solve is:

$$P(x,u)A(x) + A^t(x)P(x,u) - P(x,u)B(x,u)R^{-1}B^t(x,u)P(x,u) + Q = 0 \tag{2.27}$$

$$u(x) = -R^{-1}(B^t(x,u)P(x,u)x \tag{2.28}$$

This is the system generally implemented in SDRE solver instead of the previous expression given in (2.26). It is easy to see that the assumption made previously is a measure of the non-linearity. Indeed by having derivatives small compare to the original matrix is having a system not far from being linear, for which the derivative is exactly zero since the matrix A is constant. That is why for highly non linear systems, the expression (2.26) can be used, if the computing time is not too high.

2.2.3 SDRE control design

As showed before, a quasi-linearization of the dynamics is giving the opportunity to extend the quadratic regulation problem to nonlinear systems which are respecting the following conditions [5]:

- The dynamics f is continuously differentiable on Ω.
- The origin 0 is an equilibrium point of the system: $f(0) = 0$
- The design parameters Q and R are respectively positive-definite and semi definite.
- The pair $(A(x), B(x,u))$ is controllable and the pair $(C, A(x))$ is observable on Ω, with C a factorization of Q.

Then for a given parameterization $(A(x), B(x,u))$, the existence of a unique matrix $P(x,u)$ solving the SDRE is guaranteed, and so a unique law u(x), given in (2.28), controlling the system. This SDC parameterization being not unique, it represents another set of design parameters to control the real system described in (2.19).

In addition, the results given in the previous part can be extended to design parameters which are also function of x, under the hypothesis that the derivate of $Q(x)$ can be neglected and the condition 3 is respected and the pair $(C(x), A(x))$ is observable for each $x \in \Omega$ [5].

2.2.4 Power series formulation

Even with the previous simplification, some significant difficulties can be encountered with the SDRE method, especially when the SDC are depending on both state and control law because of the dependence in u on both sides of (2.19). A way to overpass those difficulties is to use a power series formulation of the Riccati matrix, as developed by Wernli and Cook in 1974 [7]:

$$P(x,u,\varepsilon) = \sum_{i=0}^{\infty} \varepsilon^j L_j(x,u,\varepsilon) \tag{2.29}$$

where ε is a temporary variable. This variable is also used to decompose A and B in two parts separating the linear part from the rest:

$$A(x) = A_0 + \varepsilon \frac{A_1(x)}{\varepsilon}$$
$$B(x,u) = B_0 + \varepsilon \frac{B_1(x,u)}{\varepsilon}$$

Those three expressions are then reintroduced in the state dependent Riccati equation and then gathered by power of ε. This brings the following infinite set of equations:

$$L_0 A_0 + A_0^t L_0 - L_0 B_0 R^{-1} B_0^t L_0 + Q = 0 \tag{2.30}$$

$$L_1 \overline{A_0} + \overline{A_0^t} L_0 + \frac{1}{\varepsilon}[L_0 A_1 + A_1^t L_0 - L_0 (B_0 R^{-1} B_1^t + B_1 R^{-1} B_0^t) L_0] = 0 \tag{2.31}$$

$$L_j\overline{A_0} + \overline{A_0^t}L_j - \sum_{i=1}^{j-1} L_i B_0 R^{-1} B_0^t L_{j-i} + \frac{1}{\varepsilon}\left[L_{j-1}A_1 + A_1^t L_{j-1} - \sum_{i=0}^{j-1} L_i (B_0 R^{-1} B_1^t + B_1 R^{-1} B_0^t) L_{j-1-i}\right] - \frac{1}{\varepsilon^2}\sum_{i=0}^{j-2} L_i B_1 R^{-1} B_1^t L_{j-2-i} = 0 \quad (2.32)$$

where $\overline{A_0} = A_0 - B_0 R^{-1} B_0^t L_0$, which is a constant matrix. Equation (2.30) is a standard Riccati equation easily solved. The next equation is a Lyapunov equation with Q non constant. This equation can be written in the form:

$$\hat{A}_0 vect(L_1(x,\varepsilon)) = -vect(Q_1(x,u,\epsilon))$$

$$\hat{A}_0 = Kron(I_n, \overline{A_0}) + Kron(\overline{A_0}, I_n)$$

$$Q_1(x,u,\epsilon) = \frac{1}{\varepsilon}[L_0 A_1 + A_1^t L_0 - L_0(B_0 R^{-1} B_1^t + B_1 R^{-1} B_0^t) L_0]$$

with $Kron(A,B)$ being the Kronecker product of A and B. The solution can be easily expressed in a closed form:

$$vect\big(L_1(x,\varepsilon)\big) = -\hat{A}_0^{-1} vect(Q_1(x,u,\epsilon)) \quad (2.33)$$

The same procedure can be applied for $j > 2$ to obtain all the $L_j(x,u,\varepsilon)$, up to a certain order N, the order of the power expansion. Then, the expressions like (2.33) shall be transformed into a matrix form to be gathered, giving the suboptimal control law u:

$$u(x) = -R^{-1}B^t(x,u)\sum_{j=0}^{N} \varepsilon^j L_j(x,u,\varepsilon)x \quad (2.34)$$

It is important to note that u is not depending on the temporary variable, since from (2.30), (2.31) and (2.32) is can be easily seen that $L_j(x,u,\varepsilon) \propto \frac{1}{\varepsilon^j}$.

The previous expression of the control law can still represent a difficulty, as said before, due to the dependence in u of B and L_j. An alternative can be the expression of u as a serie solved iteratively online. Although this opportunity will

not be considered, for a matter of simplification B will be supposed depending only on the state, not on the control. Thus:

$$u(x) = -R^{-1}B^{t}(x)\sum_{j=0}^{N}\varepsilon^{j}L_{j}(x,\varepsilon)x \tag{2.35}$$

2.3 Conclusion

All the precedent considerations gave a method to find a suboptimal control law for a nonlinear system respecting some conditions. The last process to define is the choice of a parameterization of the dynamics thanks to a quasi-linearization. If this parameterization is respecting the conditions described previously, a unique control law can be found, representing a suboptimal control of the system. To find this law, a power series expansion can be used by introducing a temporary variable. Most of the computation can be done offline since the elements of the expansion are expressed in a closed form. The evaluation of the control law given in (2.35) is then reduced to the online evaluation of the L_j.

The design of the control is then ruled by: the weight matrices Q and R, the choice of the parameterization, and the order of the power series expansion. This last parameter is not that evident to set, a higher order does not mean a better optimality since the method is supplying a suboptimal control.

Chapter 3

Systematic parameterization

The previous section set the fundamentals concerning the development of a nonlinear controller optimizing a criterion defined by the designer or the project's requirement. When a good parameterization is obtained, respecting few conditions, then a unique control law can be computed. But the way to find this parameterization is not that easy. One can find a way, by using mathematical manipulations as identities, to express the system's governing equations with at least one of the state vector's components appearing in each term of the numerator. But this can't be performed by a machine. Some assumptions can be made also, neglecting some parts of the equation, which needs as well a human intervention. Some researches published in the early 90's tried to defined systematic ways to obtain a parameterization of some governing equations to introduce them in a SDRE solver.

3.1 An example of systematic parameterization

As demonstrated by R.W.Bass in 1991 [8], a systematic parameterization exists under two assumptions:

If $f: \Omega \to R^n \in C^1(\Omega)$ and $f(0) = 0$, then $A(\boldsymbol{x}) = \int_0^1 \frac{\partial f}{\partial \boldsymbol{x}}\Big|_{x=\lambda \boldsymbol{x}} d\lambda$ where λ is a dummy variable used only for integration purpose.

Indeed, using the function $\tilde{f}(\lambda): R \to R^n \triangleq f(\lambda \boldsymbol{x})$ one can notice that

$$f(\boldsymbol{x}) = \tilde{f}(1) = \tilde{f}(0) + \int_0^1 \frac{d\tilde{f}(\lambda)}{d\lambda} d\lambda$$

But $\tilde{f}(0) = 0$ and by composition: $\frac{d\tilde{f}(\lambda)}{d\lambda} = \left(\frac{\partial f}{\partial x}\Big|_{x=\lambda x}\right) \boldsymbol{x}$ it is clear that:

$$f(\boldsymbol{x}) = \int_0^1 \frac{\partial f}{\partial \boldsymbol{x}}\Big|_{x=\lambda x} d\lambda \, \boldsymbol{x} = A(\boldsymbol{x})\boldsymbol{x}$$

But such a parameterization is not so much convenient, taking in account than an integration process is required to evaluate the SDC matrices, so rising the computation time dramatically, because this procedure does not obviously provide a closed from of the matrix. The following will present an alternative systematic procedure to obtain the parameterization, developed for this study.

3.2 Parameterization process

The main idea behind the developed method lays on the fact that the SDRE control method is a suboptimal control, so an approximation of the governing equations does not mean an obvious loss of performances. To refresh the minds on this point, the previous section set the assumption of a negligible derivative of the SDC matrices: a good approximation of the equations can totally be a better assumption, creating a lower divergence from optimality.
Then, the governing equations can be approximated as a system easily factorable and quick to compute online: an expansion of the equations in a polynomial expression.

3.2.1 Taylor expansion of a multivariate function

Different processes can be used to obtain such a polynomial. For a numerical model, a polynomial interpolation would be a good choice, but when a symbolic form of the equations is available a Taylor expansion can be performed efficiently. Indeed, the Taylor's theorem for a multivariate function [9] shows that, for $f: \Omega \to R^n \in C^{k+1}(\Omega)$, if $[\boldsymbol{a}, \boldsymbol{a}+\boldsymbol{x}] \subset \Omega$:

$$f(\boldsymbol{a}+\boldsymbol{x}) = \sum_{p=0}^{k} \frac{1}{p!} D_{\boldsymbol{x}}^{p} f(\boldsymbol{a}) + \int_0^1 D_{\boldsymbol{x}}^{k+1} f(\boldsymbol{a}+s\boldsymbol{x}) \, \frac{(1-s)^k}{k!} \, ds \tag{3.1}$$

$$\text{with} \quad D_{\boldsymbol{x}}^{p} f(\boldsymbol{a}) = \sum_{|\alpha|=p} \frac{p!}{\alpha!} \frac{\partial^p f}{\partial \boldsymbol{x}^{\alpha}} (\boldsymbol{a}) \boldsymbol{x}^{\alpha},$$

where α is a multi-index notation. One can prove that the integral rest is

negligible compare to $\|x\|^k$ [10], giving the so called Taylor-Young formula. Then, applying the Taylor-Young formula in a neighborhood of zero for a function respecting $f(0) = 0$, a Taylor expansion can be computed at an order up to k if $f \in C^k(\Omega)$, giving the following polynomial expression

$$f(\boldsymbol{x}) = \sum_{p=0}^{k} \sum_{|\alpha|=p} \frac{1}{\alpha!} \frac{\partial^p f}{\partial \boldsymbol{x}^\alpha}(0)\boldsymbol{x}^\alpha + O(\|\boldsymbol{x}\|^k) \tag{3.2}$$

Thus, it appears that the regularity properties of the studied equations are defining the level of precision achievable with this method. When the SDC factorization proposed in the introduction was applicable and exact for $f \in C^1(\Omega)$, the Taylor expansion gives an approximation where the precision for such a class of system is not assured, since at an order 1, the divergence between the function and the polynomial is high.

On the other hand, the simplicity for the obtained expression, easily factorable is a better alternative for functions quite regular, that is to say $f \in C^k(\Omega)$ with k sufficiently high, when the rest is negligible. Indeed, the Taylor expansion process is something really well known, and a lot of efficient algorithm are available, for example in MATLAB thanks to the function:

$$taylor(fun., expansion\ variables, expansion\ point).$$

One can notice that the application of the Taylor's Theorem with integral rest for $k = 0$ is exactly method given in the introduction of this section.

3.2.2 Sensibility analysis

A general NQR problem was considered so far. From now, by matter of simplicity and regarding the direct application considered for this study, the governing equations will be supposed to depend only on the state, and being linear in term of control. Then, the matrix B is reduced to a constant expression that will not need any processing. Now the attention is focalized on the dynamics part.

The precision of the Taylor expansion of the dynamics needs to be evaluated. This is not an easy problem for a MIMO system and can't be totally systematic since the parameters and plots of main interests are totally depending on the system under study, and need a human judgment. But the method proposed by this study suggests and gives some tools to evaluate this precision:

- the computation of the relative and absolute error between the expansion and the real dynamics on different surfaces parameterized by two or three variables, for different expansion orders.
- the computation of the relative and absolute error on the whole control domain thanks to a Monte-Carlo simulation.

The plots of those errors according to relevant parameters can give an idea of the precision and help to make a decision about the required order or the validation of the obtained expansion.

3.2.3 Horner scheme

By performing a Taylor expansion for each equation of the system, a vector of polynomials is obtained. It is now required to factorize this vector according to the different variables in order to produce a matrix of factors. The chosen option is to use a Horner scheme, since this factorization method aims to optimize the computation time by reducing the number of multiplications and additions required to evaluate a polynomial. Indeed, for a univariate polynomial, the Horner scheme is the following transformation:

$$P(x) = \sum_{i=0}^{k} a_i x^i \rightarrow H(P) = a_0 + x\big(a_1 + x(a_2 + \cdots . + x(a_{n-1} + a_n x) \ldots)\big) \quad (3.3)$$

Then the evaluation of the polynomial is resumed to a list of affine functions that can be computed quickly:

$$\begin{aligned} b_n &= a_n \\ b_{n-1} &= a_{n-1} + b_n x \\ &\vdots \\ b_0 &= a_0 + b_1 x \end{aligned} \quad (3.4)$$

where b_0 is the value of the polynomial in x. The computation process is made of n multiplication and n additions, which is the minimum amount of operations required for the evaluation of an n-order polynomial [13].

Getting back to the definition of a SDRE systematic parameterization method, the space of variables is larger than one. Thus the Horner scheme needs to be adapted. The process considered is a partial multivariate Horner scheme, expressing the polynomials as:

$$\begin{aligned} H(P) &= x_1 q_{x_1}(x_1, \dots, x_n) + x_2 q_{x_2}(x_2, \dots, x_n) + \cdots + x_n q_{x_n}(x_n) + a_0 \\ q_{x_i}(x_{i\to n}) &= a_{i1}(x_{i+1\to n}) + x_i \left(a_{i2}(x_{i+1\to n}) + x_i \left(a_{i3}(x_{i+1\to n}) + \cdots \right) \right) \end{aligned} \tag{3.5}$$

With $x_1, \dots, x_n$ being the variables ordered by factorization position: P is firstly factorized by x_1 to obtain an expression with the form $x_1 q_1 + r_1$, where q_1 is the quotient having a Horner structure for x_1, and r_1 the rest, function of the variables $x_2, \dots, x_n$. The next step is to factorize r_1 by x_2 and so on, arriving at the end to the expression (3.5).

This process is applied to the whole set of equations and the quotients are gathered in a factor matrix, the SDC parameterization:

$$A(x_1, \dots, x_n) = \begin{pmatrix} q_{1,x_1} & \cdots & q_{1,x_n} \\ \vdots & \ddots & \vdots \\ q_{n,x_1} & \cdots & q_{n,x_n} \end{pmatrix} \tag{3.6}$$

The matrix obtained is clearly not unique [11], since the order of factorization is determinant: for example, the factorization of the expression $xy(x + y)$ will give $[x(xy + y\text{^}2)\ 0]$ according to $[x\ y]$, or $[0\ y(x\text{^}2 + xy)]$ according to $[y\ x]$.

The space of the possible combinations has a cardinal of $(n!)^n$, n being the size of the state vector. It is also important to notice that this matrix is ordered to be multiplied by the vector $(x_1, \dots, x_n)$, which is the factorization sequence. If required, this sequence can be permuted to recover the original state vector, which means to reorder the matrix A as well.

The denomination “partial multivariate” lays on the fact that each coefficient of the factor matrix as a Horner-like structure for a single variable. A multivariable scheme would imply to have all the coefficients $a_{ji}(x_{i+1}, \dots, x_n)$ with a Horner-like structure for all the variables. This process being longer and not so relevant for a systematic parameterization method, a partial multivariate scheme has been preferred.

A procedure creating the factor matrices for all the possible factorization sequences has been developed in MATLAB, in order to propose a selection method proposed in the following.

3.2.4 Parameterization selection

The final step is to define the criterion to select the good parameterization. A systematic way to choose the SDC parameterization is to evaluate the controllability of the system thanks to the state dependant controllability (resp. observability) matrix $M_c = [B\ A(x)B(x,u)\ A(x)^2B(x,u) \dots A^{n-1}(x)B(x,u)]$, by computing its determinant, $det(M_c)$. For a MIMO system, this matrix is a non square matrix for which the determinant is not defined, even if the controllability is still assured by having $rank(M_c) = n$. In this case $\sqrt{det(M_c^t . M_c)}$ can be evaluated [5]. This is the criterion used in this study, the best parameterization being the one maximizing this value.
Concerning the evaluation of this criterion, a Monte-Carlo simulation is processed to have an idea of the model’s global behavior on the control domain, for each factorization sequence. An optimal criterion would be the maximization of the controllability, but this notion is not that evident on a multidimensional domain. Thus several parameters are considered:

- the minimum and maximum values,
- the mean value
- the variance

Those are as much parameters that an engineer can consider to make a choice concerning the factorization sequence.

3.2.5 Parameterization validation

The last step of the procedure is to check if the accuracy of the approximation is relevant compare to the sub-optimality assumption inherent to the SDRE control method. The derivative of the parameterization is required, and can be compared to the derivatives of the real equations. The following expression is computed on the control domain thanks to a Monte-Carlo simulation:

$$dA(\boldsymbol{x}) = \left\| \sum_{i=1}^{k} x_i \left(\frac{\partial A_{1\to n,i}}{\partial \boldsymbol{x}}(\boldsymbol{x}) \right)^t \right\|$$

$$\text{with } \frac{\partial A_{1\to n,i}}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{\partial A_{1,i}}{\partial x_1} & \cdots & \frac{\partial A_{1,i}}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial A_{n,i}}{\partial x_1} & \cdots & \frac{\partial A_{n,i}}{\partial x_n} \end{bmatrix} \tag{3.7}$$

If this function is negligible in front of $\|A(\boldsymbol{x})\|$, then the control quality provided by the SDRE controller are good. If the absolute error between the modeled system and the real equations has the same order of magnitude, then the approximation is totally justified and the controller can reach the same close-to-optimality performances than an exact SDC parameterization.

3.3 Developed method summary

Now that the main considerations and steps performed by the systematic method are set, it is possible to present the algorithms used in the codes developed for this study. It is important to know that those codes have been designed for governing equations being dependant only on the state, the control part being linear and depending only on the command. Indeed the output given at the end of the process is an SDC matrix $A(\boldsymbol{x})$, the matrix B being inserted by the user. This is due to the planned application. But the process can be easily upgraded to compute $B(\boldsymbol{x}, u)$ from a function $g(\boldsymbol{x}, u)$.

The process begins by the insertion of the governing equations and the state vector inside the code, the regularity of the equations being tested previously. It is also required to insert the domain on which the accuracy will be tested,

containing at least the domain of control. Finally, a first order of expansion needs to be assessed.

From that, the codes after running are providing the Taylor expansion and the errors, in Cartesian and spherical.
Once the analysis is performed by the controller designer and validated, the Horner schemes comparison can be run. The code is constructing the parameterization for a predefined set of possible factorization sequences, by default all the possibilities.

From a specific factorization sequence, it defines a dummy vector, for example $(x_1, x_2, x_3) = (var3, var1, var2)$, and the equations are rewritten as a function of the new dummy variables and the factorization is processed for each equation starting by the first variable up to the last one:

$$r_{i0} = T_i(x_1, \dots, x_n)$$

$$for\ j = 1..n: \quad r_{ij}(x_{j+1}, \dots, x_n) = r_{ij-1}(x_j = 0, \dots, x_n) \qquad (3.8)$$
$$q_{ij}(x_j, \dots x_n) = horner\left(\frac{r_{ij-1} - r_{ij}}{x_j}\right)$$

Each q_{ij} is placed in the matrix A at the right position according to the factorization sequence: using the previous example, $q_{11}(x_1, \dots) = q_{11}(var3, \dots) = A_{13}$. Once the matrix A is computed, the dummy variables are replaced by the original state vector, and the statement $A(\boldsymbol{x})\boldsymbol{x} = T$ is checked, assuring to have a correct parameterization. Then the controllability values are computed and stored in vectors where one line represents one factorization sequence. When all the sequences are tested, the controllability performances can be compared and the designer can make a choice regarding the project requirement.

The systematic parameterization method is summarized in the following diagram, for which the colors represent the systematic steps and the designer interventions:

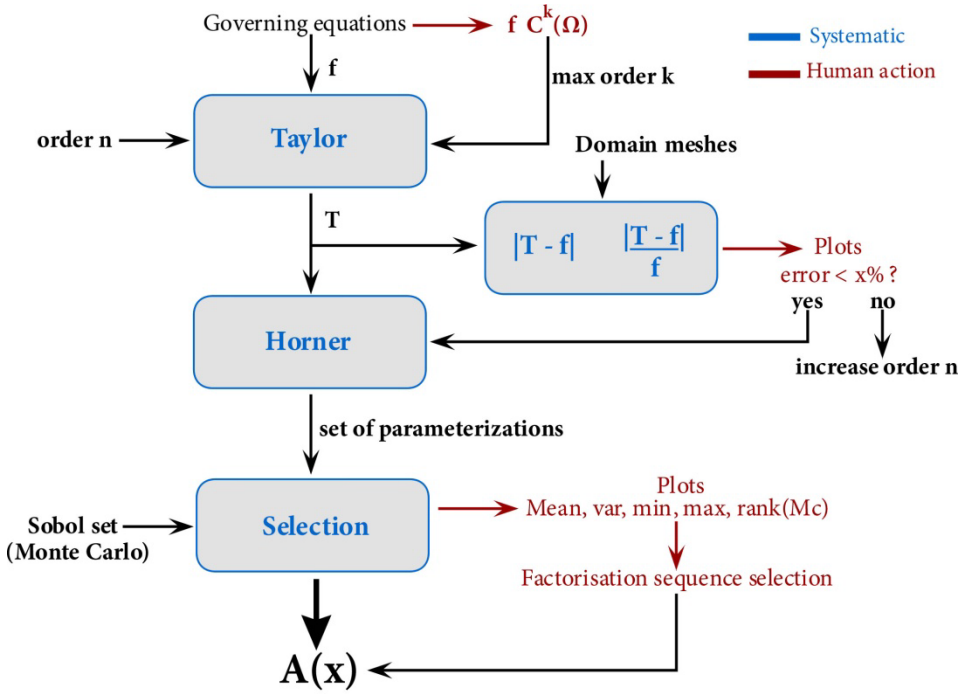


Figure 1: systematic parameterization method mind map

This is the structure of the codes developed on MATLAB, The Horner box being the main step of the parameterization, the next diagram is representing its algorithm implemented for this part of the method:

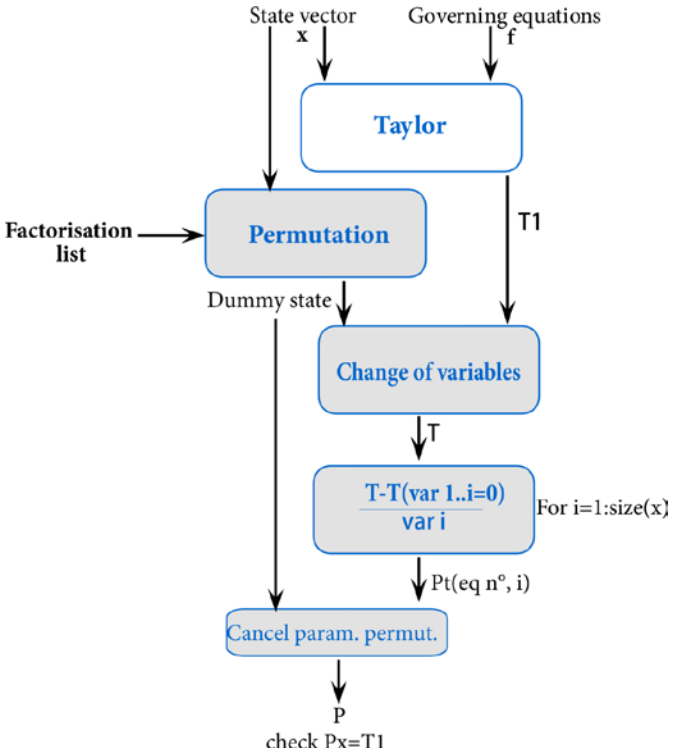


Figure 2: Horner scheme algorithm

To conclude, the codes are producing from a set of equations and the definition of the control domain some plots and values that lead the designer to the choice of a specific parameterization for which the expression can be exported to be implemented in an SDRE solver. Obviously the selection process implies to make some decision from the designer and can't be systematic without the risk to lose an important phenomenon, but the elements needed to make this choice are produced systematically.

Chapter 4

Relative dynamics

This section presents the steps to obtain the relative dynamics equations from the standard celestial mechanics, using the Cartesian coordinates. Considering the framework of a rendezvous and docking maneuver, the relative dynamics gives a set of equations describing the movement of the approaching body, the pursuer, relatively to a second one, the target to dock to, when both are orbiting round a third body, the main attractor. The gravitational interactions between the pursuer and the target as considered as negligible compare to the main attractor gravity field, and then the resulting orbits are Keplerian. This section will follow the approach to the relative dynamics problem developed by H.Schaub, J.L.Junkins in [1].

4.1 Reference frame

In order to develop the relative dynamics, it is firstly required to introduce an adequate reference frame. The target is identified by its position through the vector $\boldsymbol{R}(t)$, while the pursier position is denoted by $\boldsymbol{r}(t)$. To obtain the relative equations in a simple form, the target's Local-Vertical Local Horizontal (LVLH) reference frame, known as the Hill frame and denoted O in the following, is used:

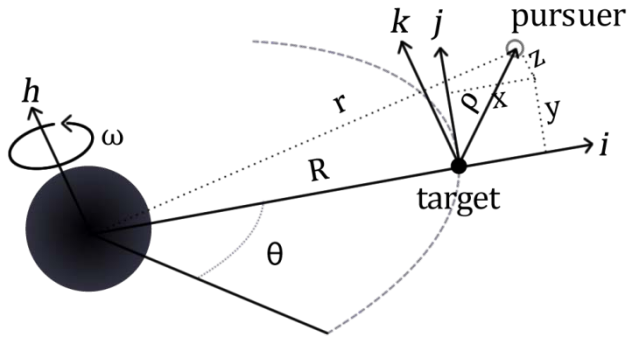


Figure 3: Hill reference frame

defined as $O = (C_t, \boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k})$ where:

- C_t is the center of mass of the target body
- **i** is opposed to Nadir
- **j** is oriented according to the target tangential speed
- **k** is perpendicular to the orbit

This reference frame is obtained from the Attractor centered reference frame, noted I in the following and considered as inertial, by the following relation:

$$R_I^O = \begin{pmatrix} cos\,\theta & sin\,\theta & 0 \\ -sin\,\theta & cos\,\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos\,i & sin\,i \\ 0 & -sin\,i & cos\,i \end{pmatrix} \begin{pmatrix} cos\,\Omega & sin\,\Omega & 0 \\ -sin\,\Omega & cos\,\Omega & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{4.1}$$

where Ω is the right ascension, i the inclination and θ is the true anomaly of the target's orbit.

The set of Cartesian parameter $\boldsymbol{\rho}(t) = (x(t), y(t), z(t))$ is describing the relative position of the pursuer in the rotating Hill reference frame, (x, y) representing the motion in the target's orbital plan.

4.2 Relative dynamics equations

Considering the target and assuming no perturbation, so only subject to the gravity field generated by the main attractor, its equations of motion in the Hill frame are simply given by:

$$\ddot{\boldsymbol{R}} = \left(\ddot{R} - R\dot{\theta^2}\right)\widehat{\boldsymbol{R}} = -\frac{\mu}{R^2}\widehat{\boldsymbol{R}} \tag{4.2}$$

with μ being the gravitational parameter of the attractor, the first equality falling from the definition of the angular velocity vector of the Hill frame O relatively to inertial frame I:

$$\boldsymbol{\omega}_I^O = \dot{\theta}\boldsymbol{k} \tag{4.3}$$

Now considering the pursuer, from geometrical consideration the position relatively to the main attractor can be rewrite as:

$$\boldsymbol{r} = \boldsymbol{R} + \boldsymbol{\rho} = (R + x)\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k} \tag{4.4}$$

and then the equations of motions in the Hill frame for the pursuer are given by:

$$\ddot{\boldsymbol{r}} = -\frac{\mu}{r^3}\begin{Bmatrix} R+x \\ y \\ z \end{Bmatrix} \tag{4.5}$$

Those equations are just the general celestial dynamics equations, expressed in the Hill reference frame, used as a base to obtain the relative equations.
On the other hand, using equation (4.3) and (4.4), the second derivative of the pursuer position can also be expressed as:

$$\begin{aligned} \ddot{\boldsymbol{r}} = \Big(\big(\ddot{R}+\ddot{x}\big) - 2\dot{y}\dot{\theta} - \ddot{\theta}y - \dot{\theta}^2(R+x)\Big)\boldsymbol{i} \\ + \big(\ddot{y} + 2\dot{\theta}\big(\dot{R}+\dot{x}\big) + \ddot{\theta}(R+x) - \dot{\theta}^2 y\big)\boldsymbol{j} \\ + \ddot{z}\boldsymbol{k} \end{aligned} \tag{4.6}$$

Then, using the fact that the orbit angular momentum $\boldsymbol{h} = R^2\dot{\theta}\boldsymbol{k}$ is constant for a Keplerian motion, its first derivative shows:

$$\begin{aligned} \dot{h} = 2R\dot{R}\dot{f} + R^2\ddot{f} = 0 \\ \Rightarrow \ \ddot{\theta} = -\frac{2\dot{R}}{R}\dot{\theta} \end{aligned} \tag{4.7}$$

Now, introducing the expression of the target radius acceleration found in (4.2) and the equation (4.7) in the set of equations (4.6), the pursuer acceleration vector is reduced to:

$$\begin{aligned} \ddot{\boldsymbol{r}} = \left(\ddot{x} - 2\dot{\theta}(\dot{y} - y\frac{\dot{R}}{R}) - x\dot{\theta}^2 - \frac{\mu}{R^2}\right)\boldsymbol{i} \\ + \left(\ddot{y} + 2\dot{\theta}\left(\dot{x} - x\frac{\dot{R}}{R}\right) - \dot{\theta}^2 y\right)\boldsymbol{j} + \ddot{z}\boldsymbol{k} \end{aligned} \tag{4.8}$$

Finally, equating (4.5) and (4.8), the exact non linear equations of the non perturbed relative dynamics can be expressed thanks to the system of equations:

$$\begin{aligned} \ddot{x} = 2\dot{\theta}(\dot{y} - y\frac{\dot{R}}{R}) + x\dot{\theta}^2 + \frac{\mu}{R^2} - \frac{\mu}{r^3}(R+x) \\ \ddot{y} = -2\dot{\theta}\left(\dot{x} - x\frac{\dot{R}}{R}\right) + \dot{\theta}^2 y - \frac{\mu}{r^3}y \\ \ddot{z} = -\frac{\mu}{r^3}z \end{aligned} \tag{4.9}$$

4.3 Case of circular orbits

Assuming that we are facing a circular orbit for the target, those equations can be simplified. Indeed, having a constant radius implies:

$$\ddot{R} = \dot{R} = 0$$

$$\ddot{R} - R\dot{\theta}^2 = R\omega^2 = -\frac{\mu}{R^2}$$

with $\dot{\theta} = cst = \omega$, which reduce the relative dynamics equations to the following system:

$$\begin{aligned} \ddot{x} &= 2\omega\dot{y} + \omega^2(R + x) - \frac{\mu}{r^3}(R + x) \\ \ddot{y} &= -2\omega\dot{x} + \omega^2 y - \frac{\mu}{r^3}y \\ \ddot{z} &= -\frac{\mu}{r^3}z \end{aligned} \tag{4.10}$$

This last system of equations is the exact non linear non perturbed relative dynamics, assuming a circular orbit for the target.
To give a more general aspect to those equations, an adimensionalization can be performed, scaling distances by R and times by $\frac{1}{\omega}$ as suggested in [2]. The resulting system is:

$$\begin{aligned} \ddot{x} &= 2\dot{y} - (1 + x)(\frac{1}{r^3} - 1) \\ \ddot{y} &= -2\dot{x} - y(\frac{1}{r^3} - 1) \\ \ddot{z} &= -\frac{1}{r^3}z \end{aligned} \tag{4.11}$$

This set of equations will be the one considered for the development of this study: The acceleration of the pursuer relatively to the target can be obtained by solving this system, proceeding a scaling after to recover the dimensions. For a matter of simplicity, this set will be called ‘relative dynamics equations’ in the following.

Chapter 5

Rendezvous procedures

Refueling, crew renewal or space captures are as much space operations needing a safe approach between two vehicles to assure their contact. Those operations are the so called rendezvous and docking/berthing procedures. No unique method exists for the design of such an operation, several strategies can be considered. Each mission has its specific rendezvous profile. This section will not go in detail concerning those strategies, but will give general considerations of the rendezvous problem.

5.1 The rendezvous phases

In general, five different phases are required to achieve the contact between two vehicles. Firstly an adequate launch is required. By defining an opportune launch window and operating some corrective maneuvers in the early phase of the ascension, a preliminary orbit can be reached by achieving an inclination and right ascension inside the required margins, near from the target's orbit one.

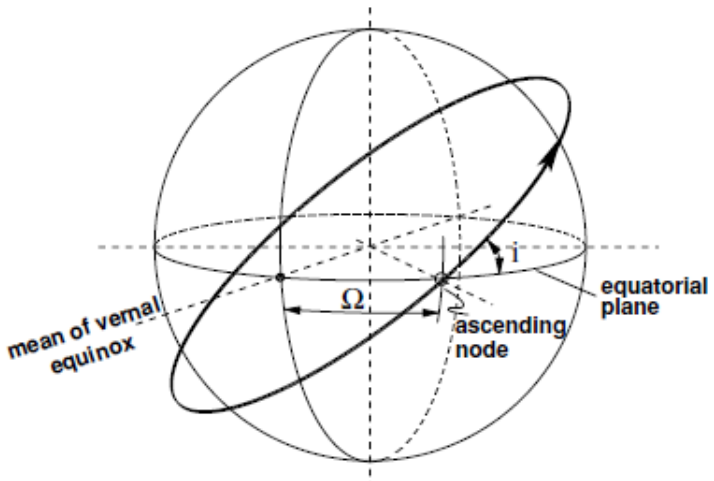


Figure 4: the initial orbit

A lower orbit is chosen in general, having a smaller Δv required. But in some case the opposite can be preferred, changing the whole approach strategy.

Secondly, from this orbit slightly different from the target's one, the time to be in phase with the target can be controlled by successive apogee or perigee rising.

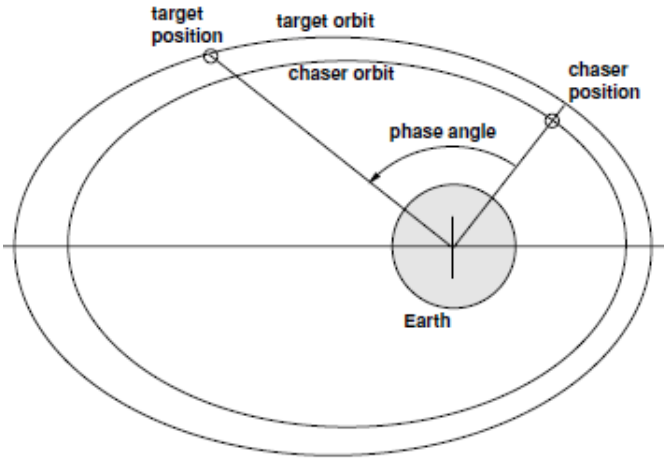


Figure 5: Phasing

Those two first phases, performed with an Earth-centered traditional celestial mechanics approach, are bringing the pursuer into an entry gate, at a distance from the target going from few thousands of kilometers to hundreds of kilometers.

The next phases are properly the rendezvous phase, starting the operations in the relative dynamics framework. An appropriate reference frame would be used, analog to the target's LVLH frame:

- R-bar, on the Earth's center-Target's center of mass direction
- V-bar, according to the target's velocity vector
- H-bar, according to the momentum

The far rendezvous phase allows going from the entry gate to a close vicinity point and reach a final state vector compatible with a close approach, and this according to the mission timeline projected. It can start when the distance with the target is sufficiently small to have the communication of the relative position between both vehicles. Several strategies can be used for this maneuver, some are presented on Fig.6. A positioning accuracy going from 100m to 10m is generally projected during the requirement definition.

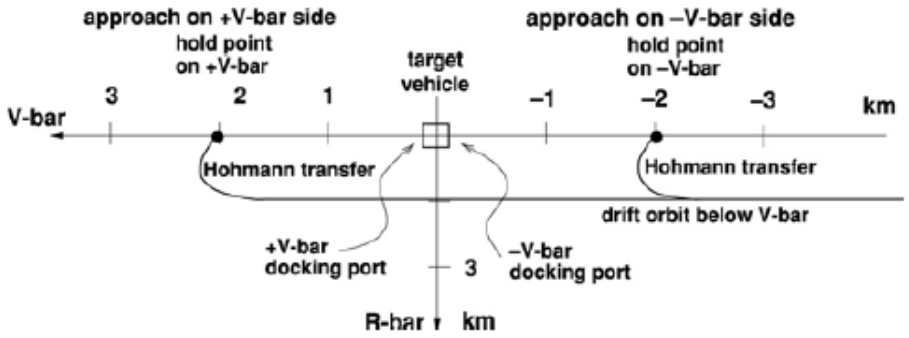


Figure 6: Far rendezvous phase

The closed rendezvous phase starts at this end point. With few impulses, the pursuer will be brought to the final corridor (the closing) going to the capture point (the final approach). At this time, all the out-of-plane or in-plane errors had been corrected and the mission timeline is synchronized. The impulses just aim to reduce the in-plane distance to the target. Even if the relative positioning is the main parameter, a ground station is constantly monitoring several parameters, for safety reasons. When the acquisition of the final corridor is done, position, speed, attitude and angular rates are compatible with the final maneuver for contact. If not, the control system can't be enabled, as well for safety reasons, to avoid collisions. This phase can start at few kilometers, the strategy depending on the initial distance. The Fig.7 is proposing five different strategies, with different pros and cons, the selection process being done according to the mission requirement and the target vehicle configuration.

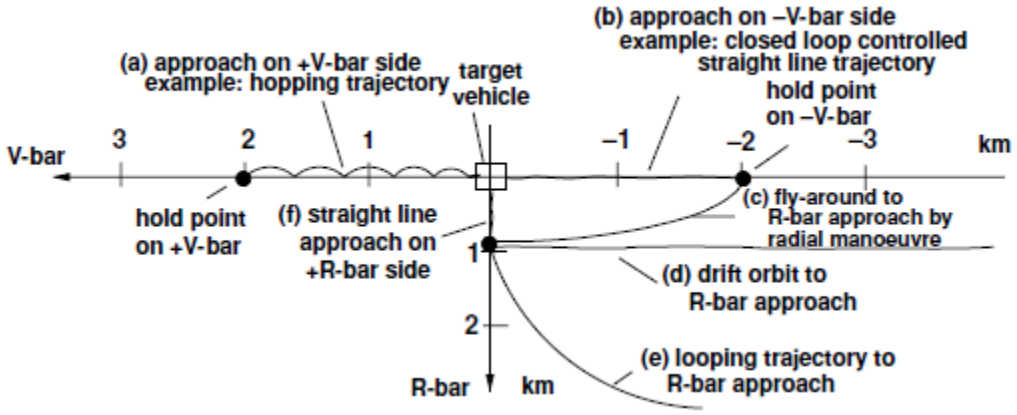


Figure 7: Close rendezvous phase

The final approach is generally a straight or quasi straight trajectory for the acquisition of the entry corridor, to safety cone of half angle round 10-15°.

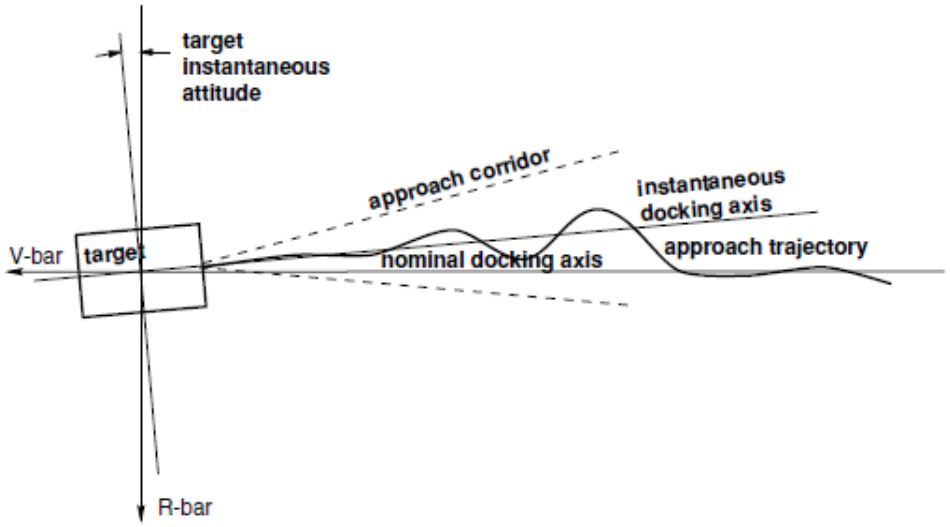


Figure 8: Final approach

Finally, the mating is the structural process assuring the contact between the two vehicles, or a capture can be performed if the target/pursuer is having a berthing system. Last tests and validations are performed to close the rendezvous procedure.
All the phases briefly presented before are summarized in Fig.9:

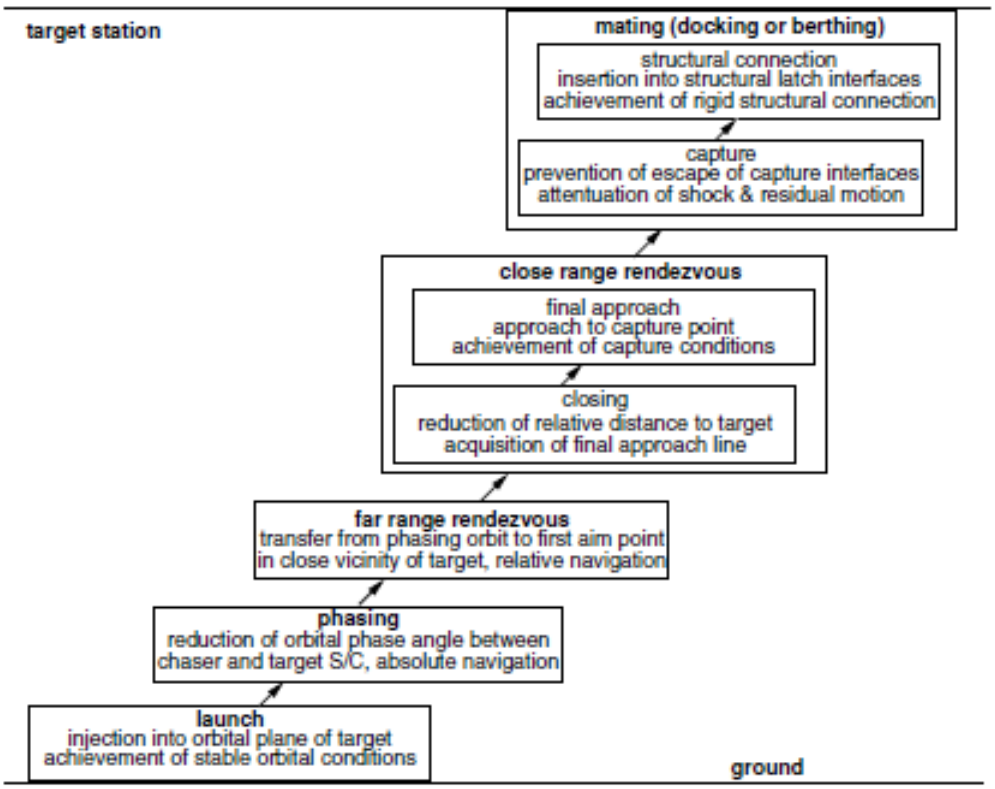


Figure 9: Rendezvous missions' phases

5.2 Reference rendezvous trajectory

As explained before, there are several strategies to perform a rendezvous, depending on the mission requirement and the target/pursuer configuration. During this study, the target is on a LEO trajectory, at 400km of height. The launch and phasing are supposed already performed, letting the pursuer with the following state vector:

$$X_0 = \begin{Bmatrix} 1.02 \\ 3500 \\ 0.1 \\ -1{,}74 \\ 1{,}44 \\ -5{,}96.10^{-5} \end{Bmatrix} \text{(m, m/s)}$$

From this point, the far and close range rendezvous can be performed, with an H-bar and two V-bar maneuvers that are computed to reach the final acquisition corridor, at 11m from the target. A straight maneuver is then necessary for the final approach. Those maneuvers are computed as Hohmann transfers. The final trajectory is the following:

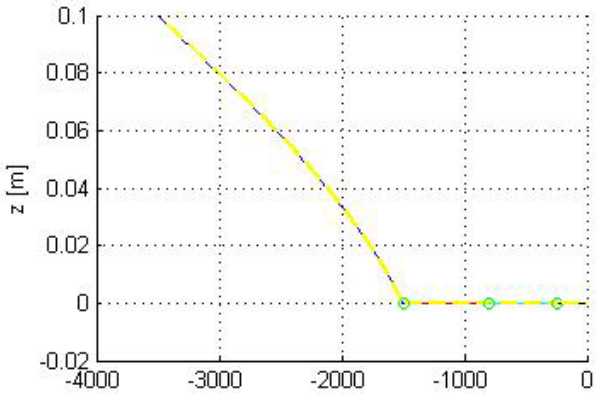


Figure 10: Reference trajectory in YZ plan

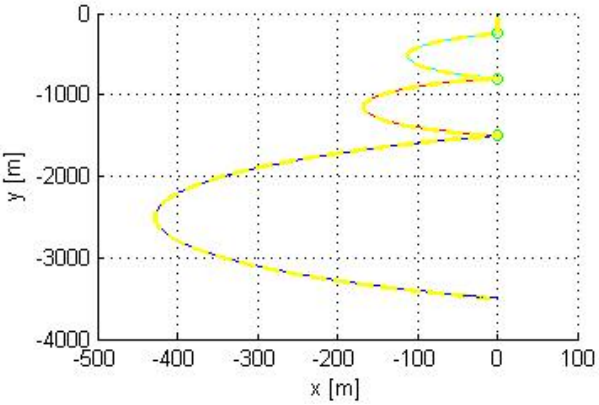


Figure 11: Reference trajectory in XY plan

The reference trajectory is composed of a vector indicating the position according to the timeline, and a vector giving the velocity computed to perform the Hohmann transfers. The solver is also taking in account some attitude control maneuvers (not studied for this work), giving a timeline with seven steps, representing a phase during almost 4800 seconds, so 1h20.

Chapter 6

Problem modeling

The considerations presented so far give the framework of a real case operational problem: apply a certain control method to follow a pre-defined trajectory between two vehicles subjected to a main attractor's gravity field. To set the final control problem, the system performing this mission, the pursuer, needs to be defined:
The pursuer is a spacecraft considered as a punctual constant mass, not subjected to perturbations, having a set of thrusters providing a linear control capability.
Being a punctual mass, no torques can be applied, then the attitude problem is not considered, and in this case the linear control capability is obvious.
If torques were present, the problem could still be decoupled in attitude and position, by performing an attitude correction before each positioning impulsion to align the thrusters to the main target's LVLH axis, imposing just a timeline sufficiently flexible to add those attitude control maneuvers.
The pursuer's behavior in term of relative dynamics is known, described by the set (4.11). Adding the control forces, the following NQR problem is obtained:

$$\boldsymbol{X} = \begin{Bmatrix} x \\ y \\ z \\ u \\ v \\ w \end{Bmatrix}; \boldsymbol{U} = \begin{Bmatrix} u_x \\ u_y \\ u_z \end{Bmatrix}; J(\boldsymbol{X_0}, \boldsymbol{U}) = \frac{1}{2}\int_0^\infty (\boldsymbol{X}^t Q \boldsymbol{X} + \boldsymbol{U}^t R \boldsymbol{U}) dt$$

$$\dot{\boldsymbol{X}} = f(\boldsymbol{X}) + g(\boldsymbol{U}) = \begin{Bmatrix} u \\ v \\ w \\ 2v - (1+x)(\frac{1}{r^3} - 1) \\ -2u - y(\frac{1}{r^3} - 1) \\ -\frac{1}{r^3} z \end{Bmatrix} + \begin{Bmatrix} 0 \\ 0 \\ 0 \\ u_x \\ u_y \\ u_z \end{Bmatrix} \tag{6.1}$$

To implement a systematic SDRE controller in order to control this system, it is necessary to check the compatibility of this problem with the conditions presented at the beginning of this study.

6.1 Systematic SDRE conditions compliancy

The first condition implies a dynamics that is sufficiently regular on the control domain. From the dynamics expression, it can be seen that the point $(-1,0,0)$ is a singularity, due to the fact that $r = 0$. But this point is the coordinate of the main attractor, so out of the domain physically used. By derivation of those equations, a new set depending on $\frac{1}{r^3}$, $\frac{1}{r^5}$ and polynomials in x, y, z is found. Then the relative dynamics is clearly $C^\infty(R^3/\{-1,0,0\})$. Obviously, the effective domain where the control will be used is far away smaller, since the far rendezvous range starts at few kilometers from the target. From this fact, it appears that the choice for the order of the Taylor expansion order is not limited.

The second condition, zero being an equilibrium point, is as well respected. This is obvious for the two last equations, the first one going to zero since $r = 1$ in zero.

The third condition is independent from the system, being a condition on the weight matrices. So the compliance to this condition is up to the designer and will be discussed later, during the presentation of the simulations.

The last condition is the most difficult to measure. Firstly, it requires defining a SDC parameterization of the state equations given in (6.1), which respects $(A(x), B(x,u))$ being controllable and $(C, A(x))$ being observable. As explained in methodology for the systematic SDRE control, this aspect is the criterion to select the parameterization among the n^n possibilities, so it is checked during the automatic process.

By matching the above mentioned conditions, the relative dynamics control problem is totally compatible with the systematic SDRE method considered in this study. The following will be dedicated to the SDC parameterization the will be implemented in the SDRE solver.

6.2 Exact SDC Parameterization

To have a better approach to measure the accuracy and the good behavior of the systematic method, a comparison with a parameterization representing the real equations is important. For this specific problem, an exact SDC parameterization can be found. Indeed, taking the dynamic equations, one can find:

$$\begin{cases} u \\ v \\ w \\ -x\left(\frac{1}{r^3}-1\right)+2v-\left(\frac{1}{r^3}-1\right) \\ -y\left(\frac{1}{r^3}-1\right)-2u \\ -z\frac{1}{r^3} \end{cases}$$

It appears that a term in the fourth equation shows no evident factorization. But using some identities falling from the Newton's binomial theorem:

$$\begin{aligned} \frac{1}{r^3}-1=\frac{1-r^3}{r^3}=-\frac{(r-1)(r^2+r+1)}{r^3}=-\frac{(r^2-1)(r^2+r+1)}{r^3(r+1)} \\ =-\big((2+x)x+yy+zz\big)\frac{(r^2+r+1)}{r^3(r+1)} \end{aligned} \tag{6.2}$$

This is a common SDC parameterization admitted for the relative dynamics, the one initially used in the SDRE solver in which the systematic model will be tested. Its expression is given by:

$$A(\boldsymbol{X})=\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1-\frac{1}{r^3}-(2+x)\frac{r^2+r+1}{r^3(r+1)} & y\frac{r^2+r+1}{r^3(r+1)} & z\frac{r^2+r+1}{r^3(r+1)} & 0 & 2 & 0 \\ 0 & 1-\frac{1}{r^3} & 0 & -2 & 0 & 0 \\ 0 & 0 & -\frac{1}{r^3} & 0 & 0 & 0 \end{pmatrix} \tag{6.3}$$

This expression is quite heavy, multiplying and dividing polynomials in r.

6.3 Systematic SDC parameterization

It is now sufficient to introduce the equations written in (6.1) in the code developed for the systematic SDRE to obtain a matrix of polynomial. But before to talk about the parameterization, let's focus on the accuracy of the expansion. This study was done with the standard MATLAB order for Taylor, 4, the main variables restricted generally to the triplet (x,y,z). The first aim is to understand the general limits before to check for the compatibility with the rendezvous trajectory described before.

6.3.1 Accuracy on specific plans

The first approach considered was to evaluate the absolute error $R_a = |T_{f_i} - f_i|$ and the relative error $R_r = \left|\frac{T_{f_i} - f_i}{f_i}\right|$, on a uniform mesh for a set of two variables, the others being set to 0. This was applied only to the 3 last equations f_4, f_5 and f_6 (representing the accelerations a_x, a_y, a_z), since the others are totally linear.
The color scale in the following diagrams is limited to 0.1, where a black area means an error higher than 10%. Thus, by identifying those black areas, the accuracy limits can be easily understood.
Firstly, for the XY plan ($z = 0$), the errors concerning a_x and a_y are plotted on Fig.12 (Relative on the left, absolute on the right). In this case, a_z and T_{a_x} are both equal to zero on the whole plane, making any plot useless.

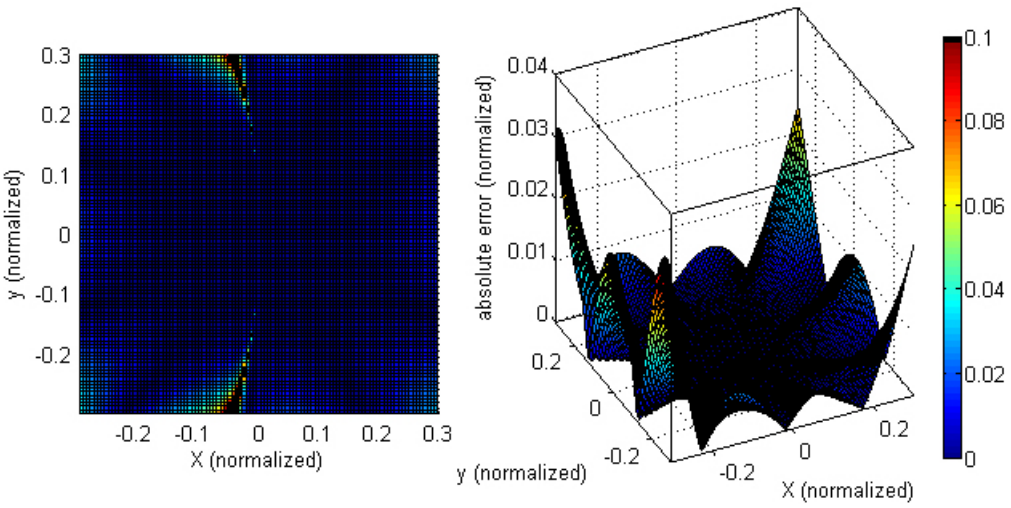


Figure 12.a: ax errors on XY plan (left: relative, right: absolute)

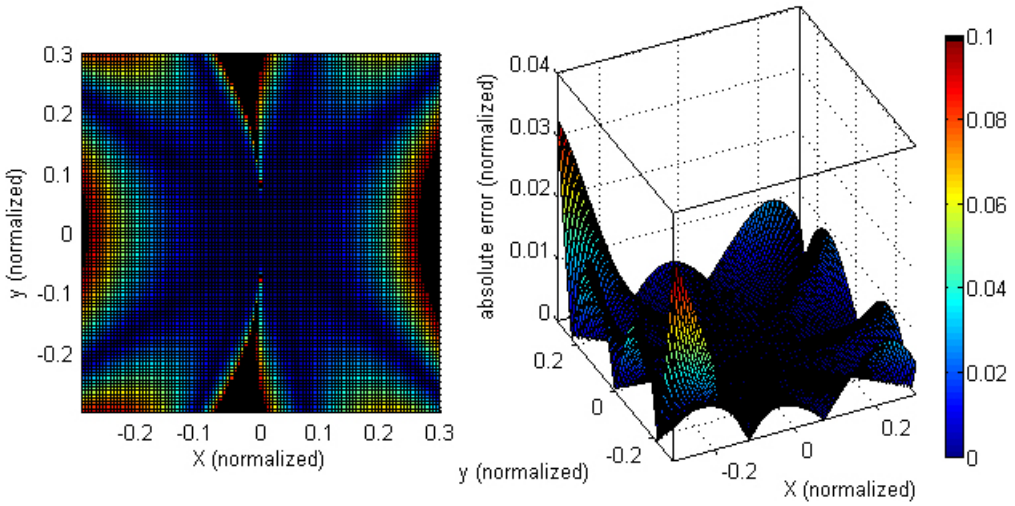


Figure 12b: ay errors on XY plan (left: relative, right: absolute)

The first interesting point is the presence of those black triangles on the relative error diagrams. They are due to the singularities of the relative error, when the real acceleration f_i is going to zero. Indeed, the absolute error does not show any singularities in those areas, which shows that the approximation is not going to extreme values. This phenomenon can also be described analytically:

$$\frac{1}{r^3} - 1 = 0 \Leftrightarrow r^3 = 1 \Leftrightarrow -2x = x^2 + y^2 = \rho^2$$
$$x = -\frac{\rho^2}{2}; y = \pm\sqrt{\rho^2 - \frac{\rho^4}{4}} \tag{6.4}$$
$$\tan\theta = \frac{y}{x} = \pm\sqrt{\frac{4}{\rho^2} - 1}$$

where ρ is the distance from the target and θ the angle with R-bar. This result shows the existence of curves where accelerations are null, being areas of singularity for the relative error if the Taylor expansion is not exactly zero. And this is related to the order of the expansion: indeed the number of poles is limited for a polynomial expression, when the real function is continuously equal to zero on the curves described by (6.4). The Taylor expansion can't fit those curves, creating a bad precision in the neighborhood: those triangle shapes. But this concern the relative error, the absolute error stays framed and small.

Except those limited areas, the relative error is small in a long range. Restraining

the domain to $\rho \leq 0.1$ will assure a good accuracy everywhere, smaller than 2%. This sphere is containing the trajectory to follow, since this radius represents 10% of the distance between the target and the main attractor, far away more than a far rendezvous maneuver maximum distance. In this case the accuracy is sufficient even if the approach angle is the continuity of the black curves.

Concerning the XZ plan the results are similar for a_x. However a_z is not subjected to theses singularities depicted before, since the annulations of a_z are happening only when z=0, cancelling as well the Taylor expansion. Since $y = 0$ on this plan, a_y is null on the whole mesh.

Finally, for the YZ plan, the three accelerations are not null, since $x = 0$ doesn't imply $a_x = 0$. But $x = 0$ is also implying that $r^3 = 1$ is impossible, so the lines previously described aren't present. It will be noted that a_y is still having the worst accuracy (10% of error reached at $\rho \leq 0.28$).

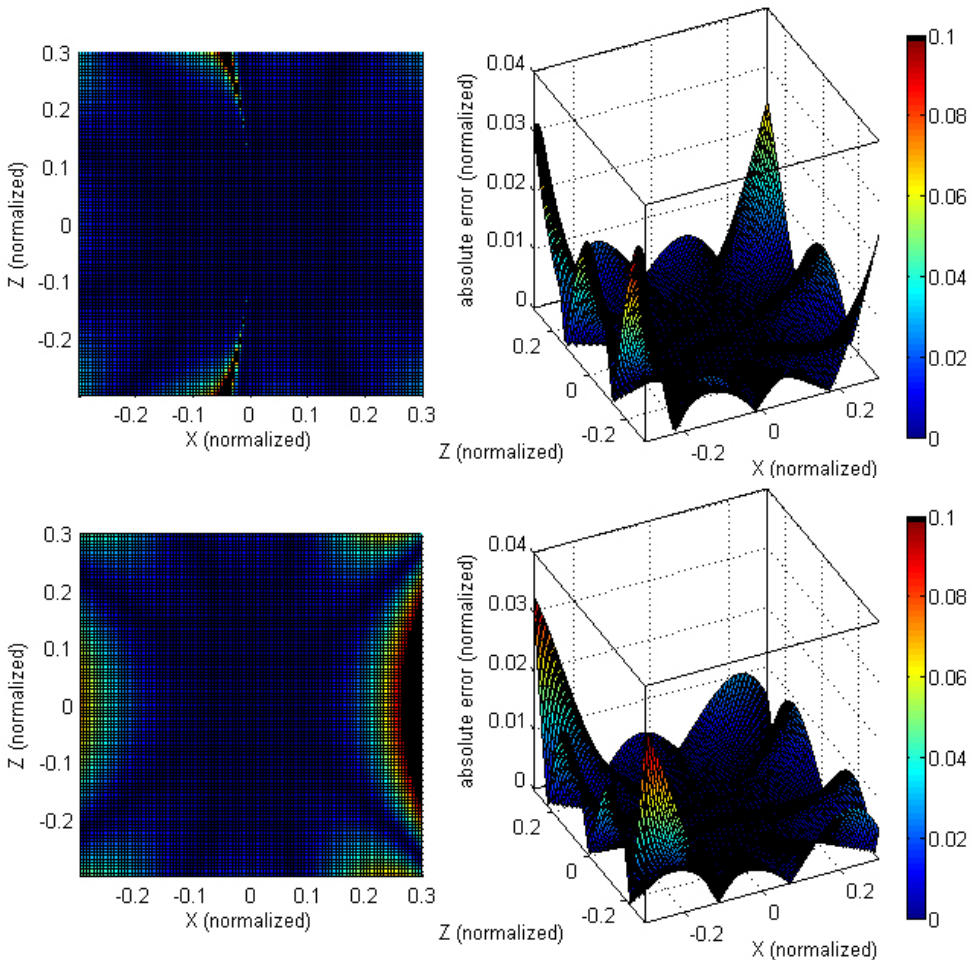


Figure 13: ax(up) and az(bottom) error on XZ plan

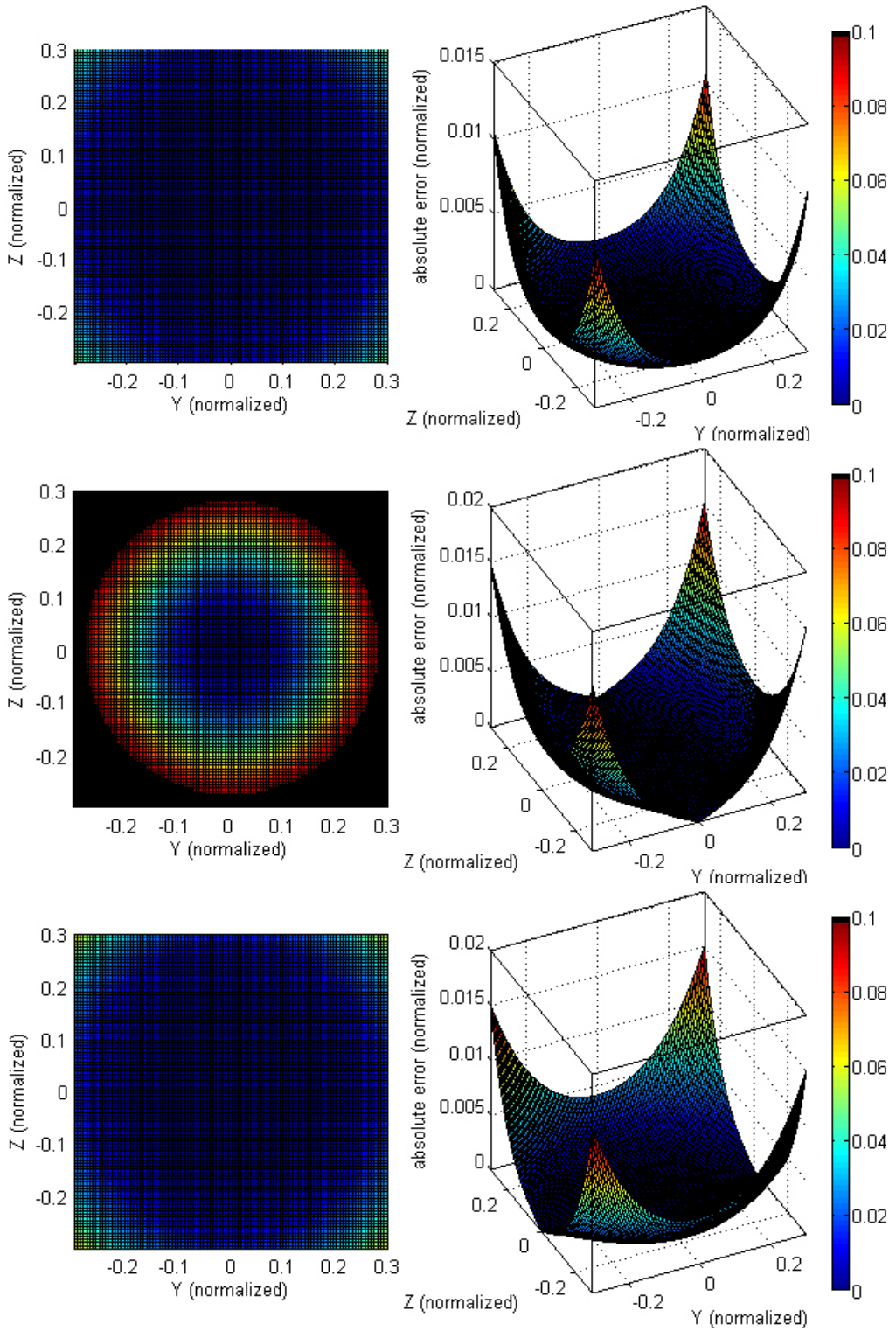


Figure 14: ax(up), ay(center) and az(bottom) error on YZ plan

To conclude, the accuracy is actually pretty good on the main Cartesian plans, the relative error reaches 10% only for large distances from the origin, and can be restricted to 2% for a maximum distance $\rho = 0.1$, since the loss of accuracy near to the zero acceleration points is reduced. Now the accuracy should be studied considering 3D surfaces even if the same kind of results can be expected.

For that, a change of coordinates is necessary; the spherical coordinates being convenient to test all the points at a given distance ρ.

6.3.2 Accuracy in 3D

The passage to spherical coordinates for different ρ gives a good idea of the error on whole domain. By defining a mesh on phi and theta, the transformation:

$$\begin{aligned} x &= \rho \cos\varphi \sin\vartheta \\ y &= \rho \sin\varphi \sin\vartheta \\ z &= \rho \cos\vartheta \end{aligned} \tag{6.5}$$

the errors in spherical coordinates are obtained. This time the analysis will only consider the relative error. The color code used before remains the same.

These diagrams present the ax acceleration results for three cases: $\rho = 0.1$; $\rho = 0.2$ and $\rho = 0.3$. The black curves previously depicted are visible in 3D, creating a kind of corona having an angle θ with the x axis (6.4).

This angle is depending on the distance, as well as the error magnitude. It can be seen that in general the error is constant for a given solid angle round the x axis, having the coordinates phi=0 or 180 and theta=90. For low radius, θ is round 90° or 270°, so near to the approach angle. Fortunately, $\rho = 0.1$ shows a low error at those angles.

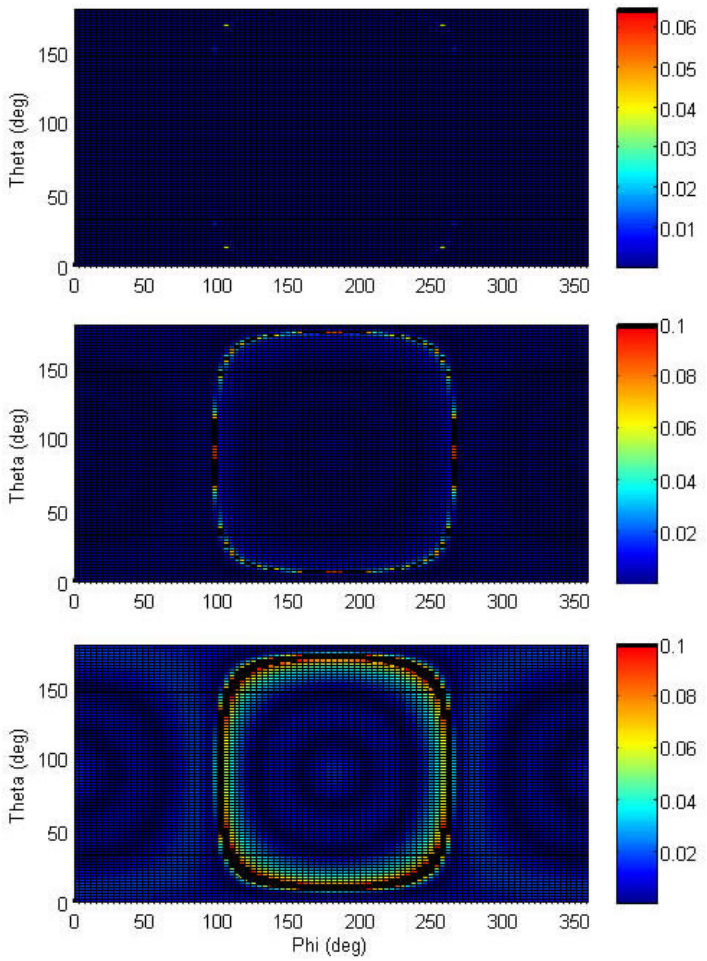


Figure 15: ax spherical relative error

The same analysis can be performed for the ay acceleration. They have exactly the same characteristics, but with a worse impact. This is due to the frequent annulations of ay, implying a larger number of poles than ax. The expansion order being the same for both, the Taylor of ay is fitting less, since can't be annulated sufficiently. But for the radius previously presented, the error is negligible.

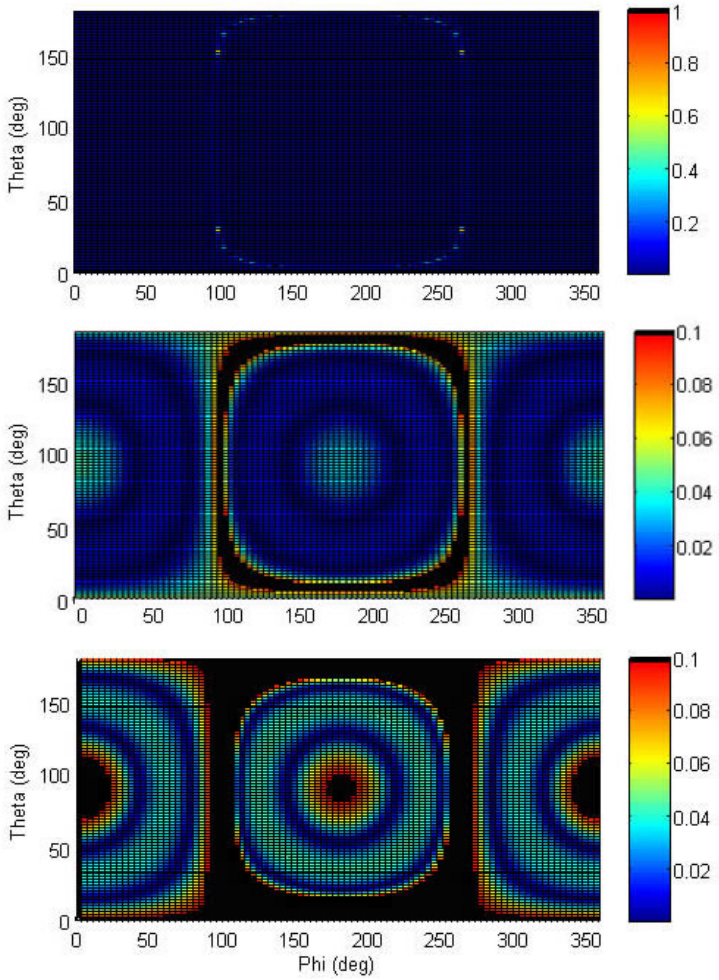


Figure 16: ay spherical relative error

Finally for az, it is found that the error is far away less a problem, which can be explained with the same consideration than for ay: az is going to zero only when z=0, which traduces a low number of poles, so a better expansion.

.

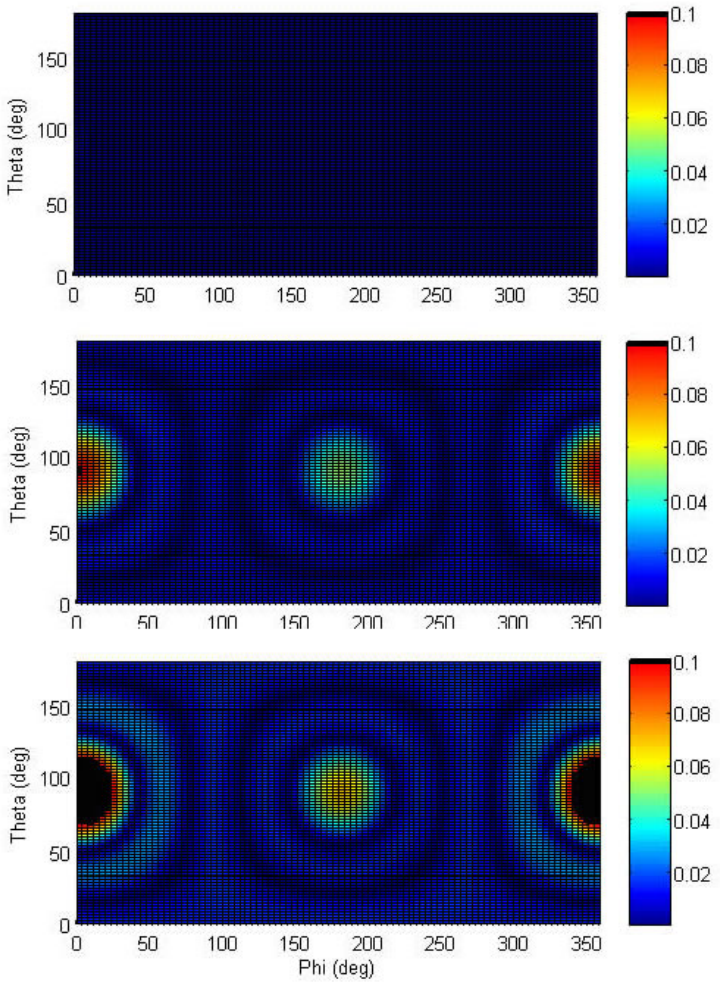


Figure 17: az spherical error

This analysis is confirming the accuracy expectation that can be guessed from the study on the Cartesian plans. The corona depicted is giving the first limit for the expansion. In any case at a distance of 0.1, the approximation is really good.

To have an easier 3D representation, the following picture plots a sphere of radius ρ augmented by the ay relative error (worst case), for the same three different ρ.

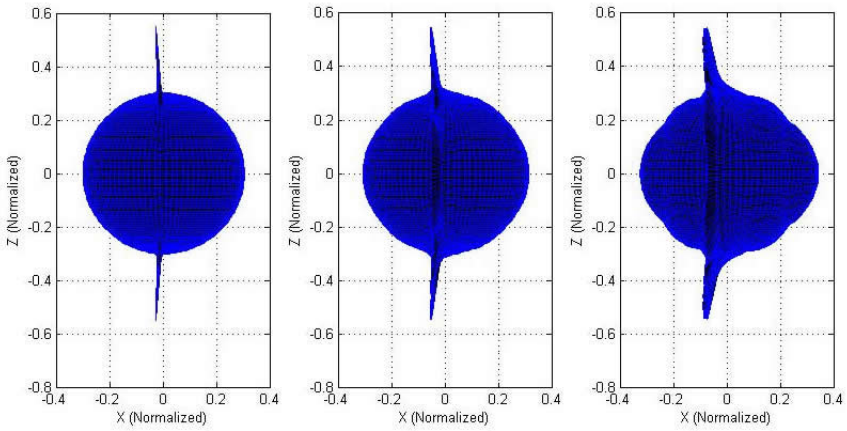


Figure 18: 3D relative error

The symmetry of revolution is easily seen, as well as the dependence of the error on the solid angle with the x axis and the distance from the target.

6.3.3 Velocity influence

Only the influence of the triplet (x,y,z) has been tested. The part depending on (u,v,w) is largely simpler, since linear and uncoupled. In order to measure the effect of a non zero velocity on the previous diagrams, the worst case is considered, that is to say ay on the XY and YZ plans for u and ax on XY for v. Here the max distance is set to ρ=0.4.

For the first one, the velocity u is obviously changing the lines ay=0, since $r^3 = 1$ is not anymore the solving equation but $2u = -y\left(\frac{1}{r^3} - 1\right)$. As explained before those lines are the main areas of loss of accuracy, then their displacement is the key point. Considering five cases, u= 0 ; 0.05 ; 0.1 ; 0.5 ; 1, the following diagram is showing that

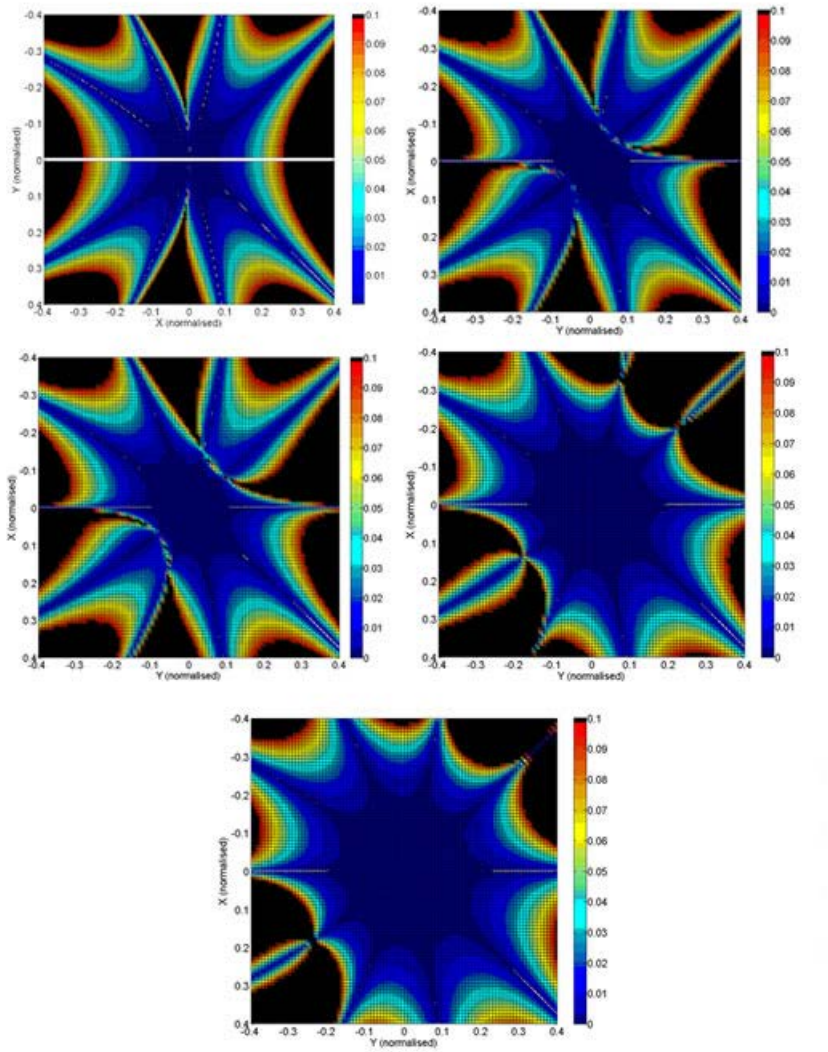


Figure 19: Influence of the velocity u on ay on XY plan (up left to right bottom)

theses curves are rotated and actually providing a better accuracy, the black areas starting being like folded on themselves. It appears that the influence of u is more a benefit. It is as well true for ay on YZ (here u=1): a better accuracy is obtained.

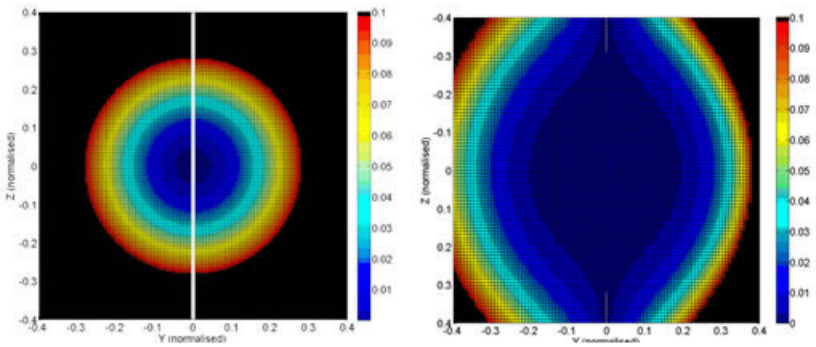


Figure 20: Influence of the velocity u on ay on YZ plan

Finally, for the variable v, so concerning only the ax acceleration, considered here on the XY plan, the black areas are not anymore rotated but more translated. Indeed for v= -1, -0.5, 1, the accuracy is not so much changed.

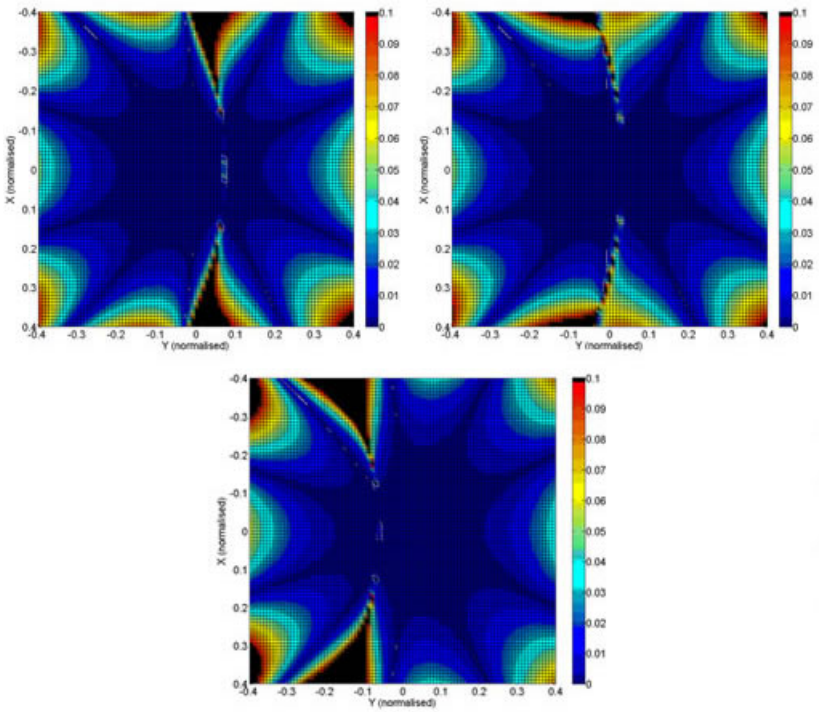


Figure 21: Influence of the velocity v on ax

All those diagrams are just confirming that the key of the problem is more concerning the position than the velocities. The accuracy is still really good, especially for a ρ lower than 0.1.

6.3.4 Influence of the order

A last parameter can be evaluated: the influence of the expansion order. The previous study was performed for the standard MATLAB Taylor order, 4. Now, considering the previous worst case on the XY plan, having an order changing from 2 to 6:

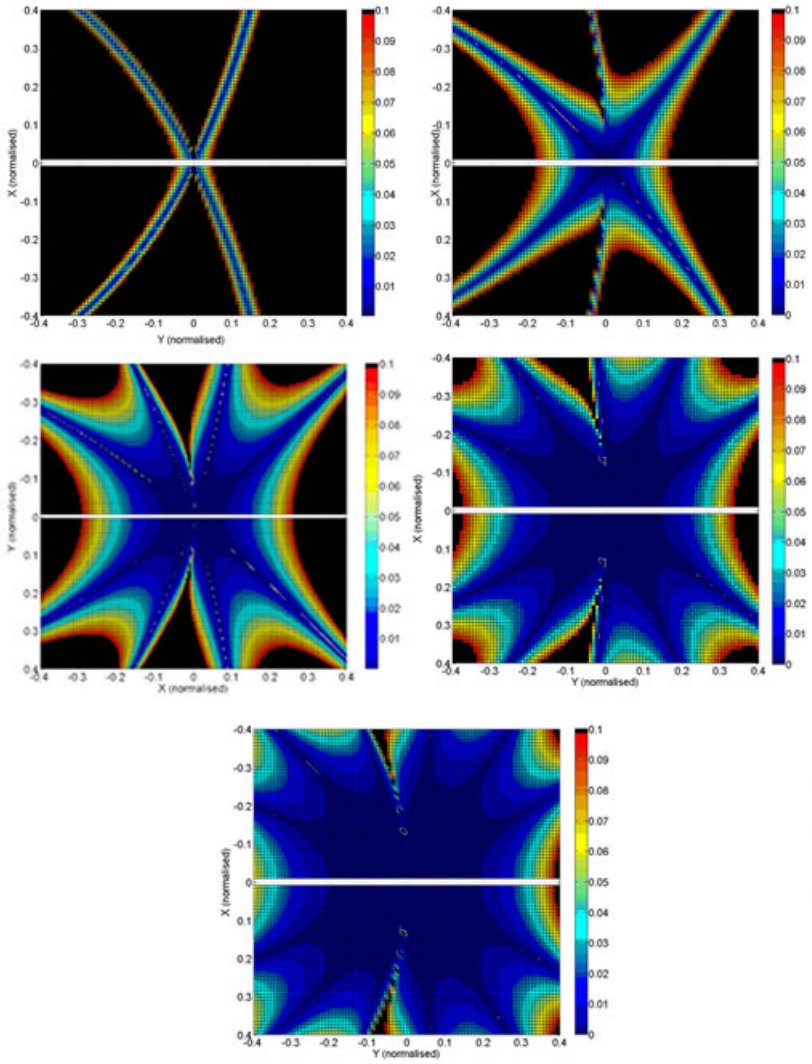


Figure 22: Influence of the order on ay on XY

It appears that the order 2 is quite inaccurate. The order 3 can be considered since the near 0 area is still sufficiently large before to reach more than 10%, but

an order 4 will be preferred since the area lower than few percent is far away larger. As expected, a higher order is giving a better accuracy. On the YZ plan, results are totally similar, but with an odd dependency: only odd orders are different.

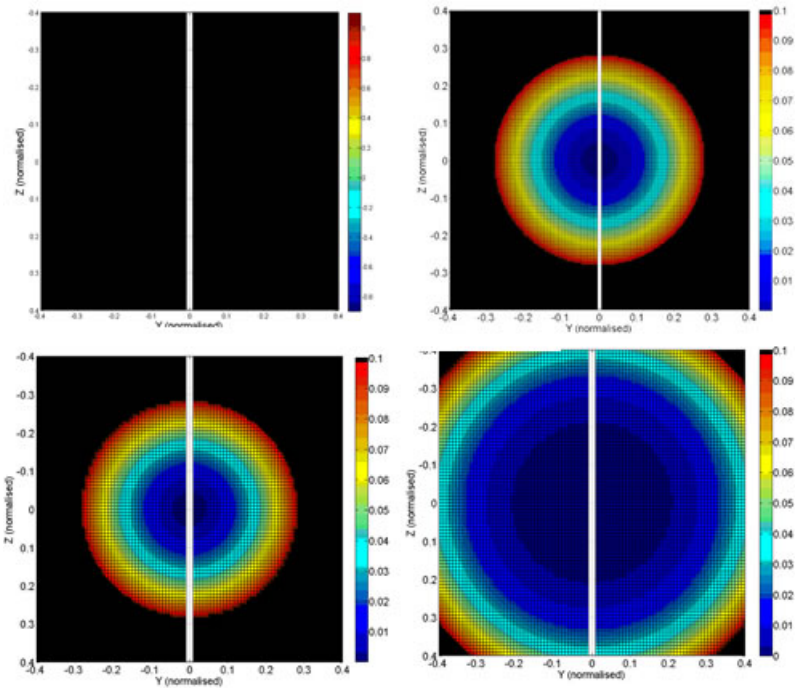


Figure 23: Influence of the order on ay on YZ

6.3.5 Conclusion on the accuracy

The accuracy analysis is giving good results, showing that the Taylor expansion at a low order, 3 or 4; might be sufficient, as well as for the relative dynamics equations. It reaches a low relative error (few %) for distances to the target up to 0.1 times the distance with the main attractor, which is a really far in general. Besides it has been shown that the accuracy is really depending on the number of poles and their distances. A large number of poles require a higher order, but if they are near, the area having a low accuracy is quite restricted.

The methods developed to test the accuracy showed their relevancy since the obtained results can be correlated with the analytical form of the equations.

6.3.6 Controllability

Now that the Taylor part is checked, the parameterization choice can be discussed. As said, the chosen criterion is an adapted form of the controllability matrix determinant: $\sqrt{\det(M_c'.M_c)}$ since M_c is not a squared matrix. As explained before, the number of factorization sequences is equal to $(n!)^n$. According to the relative dynamics equations, this number would be really high, round 10^{17}. But it is obvious that the factorization order for the triplet (u,v,w) has no impact since it is a fully linear part, having a unique factorization form. Then the sequences can be reduced to (x,y,z), so only 216 the possibilities.

The computation of the controllability is performed on the domain $\rho \leq 0.25$ (spherical coordinates) with speeds ≤ 0.8. The same Sobol set is used to compute the values on the whole considered window.

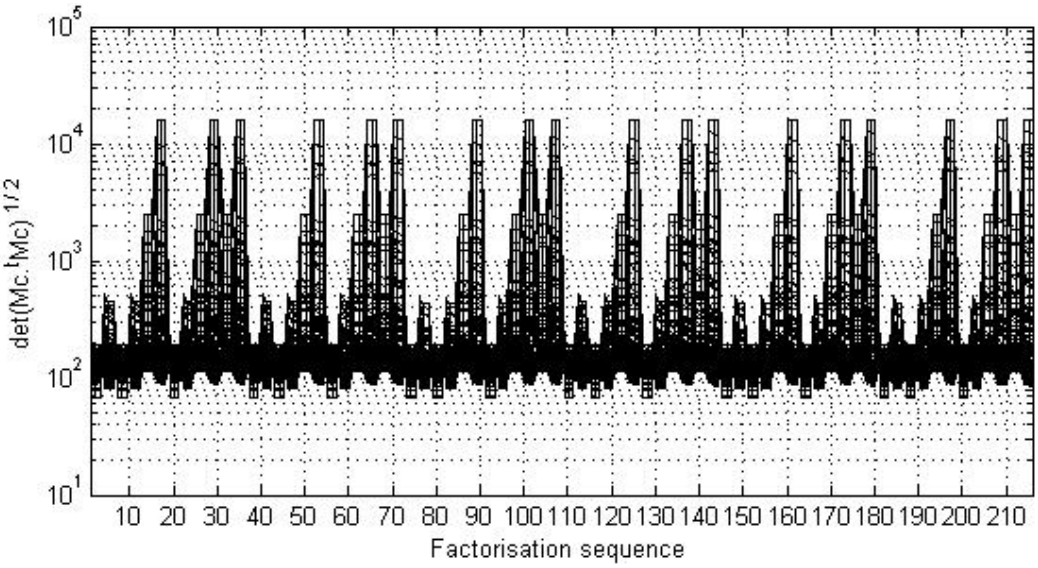


Figure 24: Controllability comparison

This diagram is presenting the controllability performances for each of the 216 factorization sequences for the triplet (x,y,z). It is the XZ view of a 3D plot defined according to:

- The x axis is referencing the identification number of the factorization sequences (here 1 is ((x,y,z), (x,y,z), (x,y,z)).
- The y axis (depth) is the list of random points from the Sobol set generated for the Monte-carlo simulation, with 250 points.
- The z axis is the controllability value for a given sequence at a given point.

Then this diagram represents a map of points, where the results for all the random points are projected on the same plan. This means that, for a given sequence, the upper point is the random point where the controllability is the highest, the lowest point being the random point with the lowest controllability. Being pretty hard to analyze, this diagram is more made to have a first idea about how the controllability is changing according to the selected factorization sequence. Firstly, it can be seen that the controllability is high, never going to less than 80, so far from 0: the controllability is not a matter in itself.
It can also be seen that the diagram is periodic, with a step of 36. This means that the factorization order for the first acceleration is not influencing, since the 36 first are corresponding to the sequence [x,y,z] for ax, the 36 following [x;z;y]. This can be explained by the presence of the factor $(1 + x)$ which implies no evident factorization, so a term present for all the variables. At the opposites, having y (resp. z) as a factor in ay (resp. az), the fact to start the factorization by this variable or not will implies zero or non zero coefficients for the two others variables. To illustrate this, a small example can be considered:

$$Horner_{y;x;z}\big(y * (x + y + z)\big) = [0\ x + y + z\ 0]\begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$Horner_{x;z;y}\big(y * (x + y + z)\big) = [y \quad y \quad y]\begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Then the controllability is governed by the number of zeros in the matrix.
So, the diagram can be zoomed on only 36 sequences (180-216).

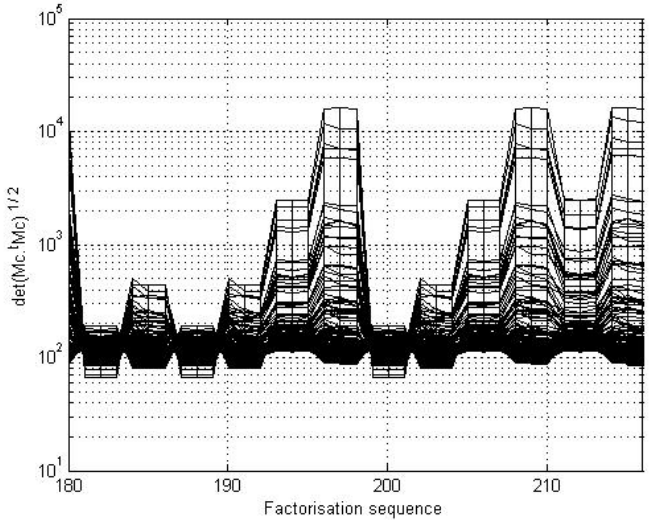


Figure 25: Controllability comparison: zoom

It appears that the properties are different, making difficult the choice of the 'best' sequence. Indeed, as shown in Fig.26 the ones showing the maximum controllability on the domain shows also the minimum one. The mean value/variance properties could be another selection criterion, but in this would lead to take sequences that are reaching a minimum in term of controllability, in a part of the domain.

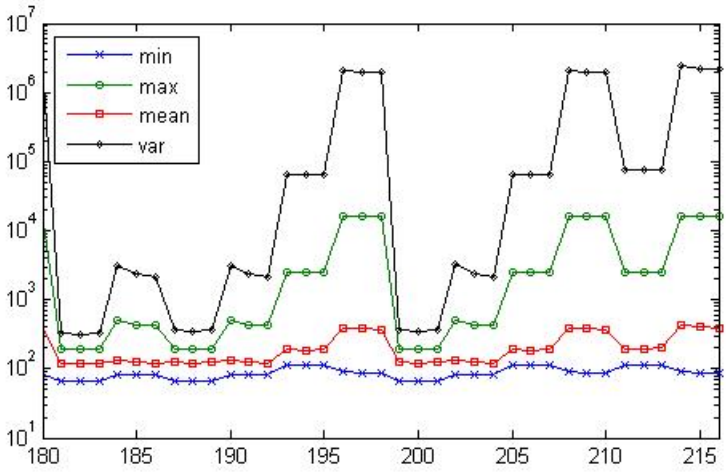


Figure 26: Controllability evaluation

Then the criterion used in this application was to have the highest minimum controllability, and among theses the one with the highest maximum controllability. This leads to the selection of the factorization 211 or 213.
For a better understanding, an analysis according to all the variables can be done.

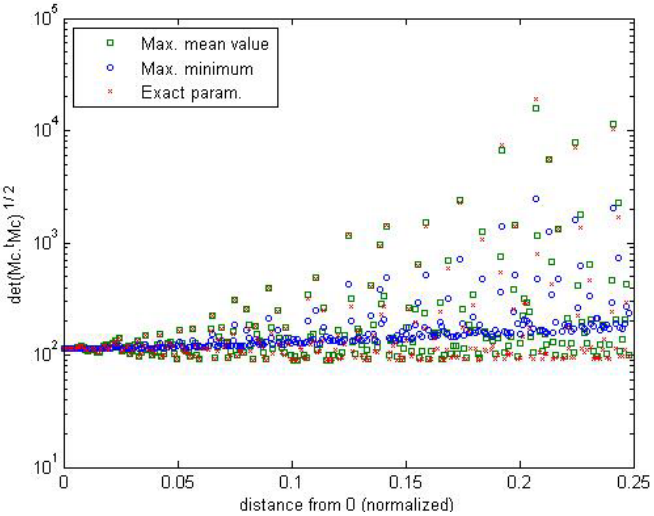


Figure 27: Controllability according to the radius

The plot is comparing three solutions: The maximum mean value (sequence 216, green squares), the maximum minimum controllability (sequence 213, blue dots), and the real parameterization presented previously (red crosses). Thus, the distance from zero is acting a lot on the mean/variance: whatever the distance, the sequence 216 has a line of minimums higher, and also a lower variance.
The two others are really similar. For small distances (the rendezvous procedures are really small), sequence 216 offers a more constant behavior.
According to the trajectories usually designed, and the one considered for this application, the approach angle is round 270°. In this situation, the variance is quite low, but the blue dots keep a small advantage.

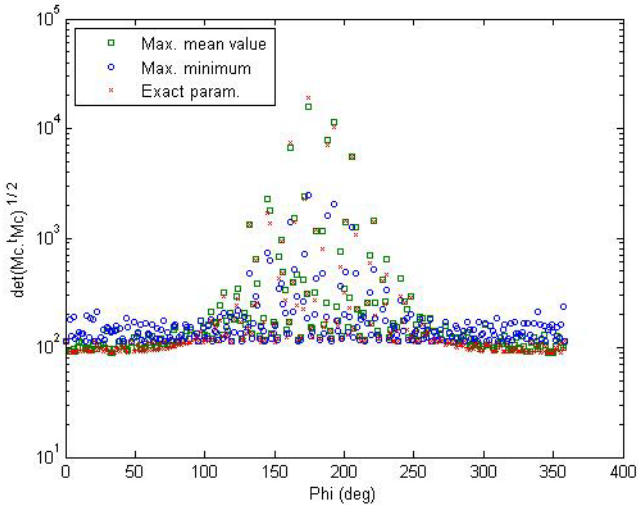


Figure 28: Controllability according to phi

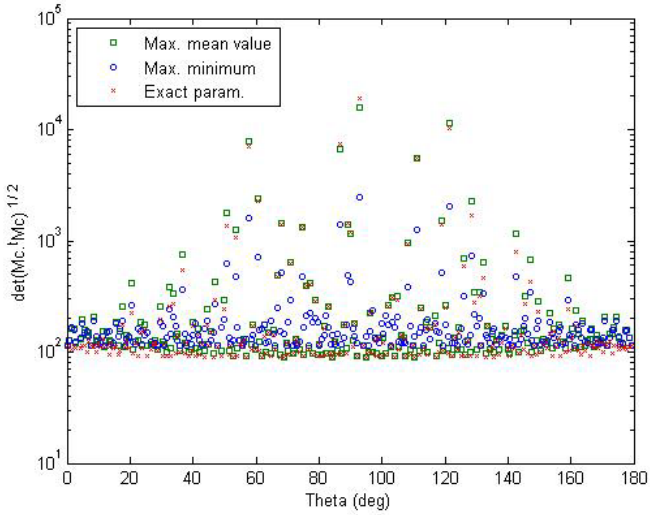


Figure 29: Controllability according to theta

Concerning theta, the main angle is 90° since most of the operation is performed in the XY plan. The differences between the different sequences are not so important. It appears those theses parameterizations are more or less equivalent, the performances are slightly different but in any case the performances are relevant.

6.3.7 Derivatives

The derivatives are computed and compared for the selected parameterization and the exact parameterization. The following graph is presenting the relative difference between the derivative and the matrix, according to the distance from 0.

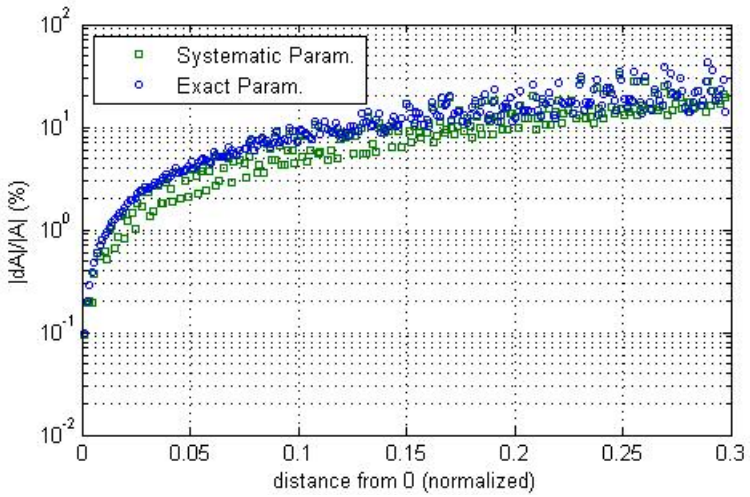


Figure 30: Derivatives evaluation according to ρ

Thus, near to zero, the derivatives are round 0.1%, going to 10% near to 0.1. The accuracy of the Taylor expansion on this domain is then far away lower than the fact to neglect the derivatives. This shows that the systematic method is convenient for this kind of problem. The analysis of the derivatives repartition according to theta or phi is far away less evident, it seems to be no correlation between these variables.

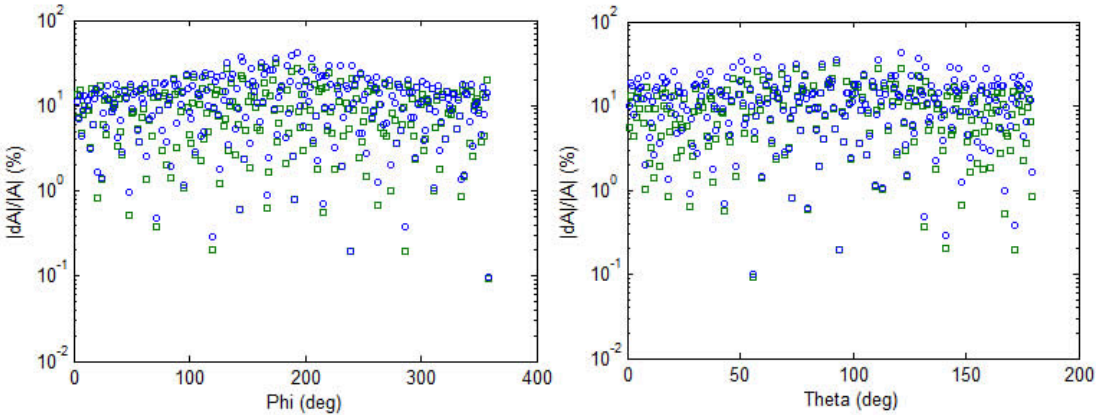


Figure 31: Derivatives evaluation according to phi and theta

6.4 Conclusion

The application of the systematic method for this problem seems totally relevant, since the derivative assumption is creating a loss of accuracy higher than the Taylor expansion, but with values still correct. Thus, the SDC parameterization obtained is relevant and can be inserted in the SDRE simulator for the rendezvous control problem defined previously. The controllability is never a problem, even for really large distance from the center, showing a controllability rising. The rank of the control matrix is always at its maximum. The selection of the factorization sequence is not of main importance since the performances are globally equivalent. The main things to retain is that the SDC matrices where obtained automatically from the relative dynamics equation, producing a parameterization that is pretty accurate and which should bring to the same sub-optimality.

Chapter 7

Simulations

The matrix A(x,y,z,u,v,w) obtained thanks to the systematic method can be inserted in the SDRE solver since its accuracy have been proved. For that, a patch to the solver was needed, updating the function computing the value of the matrices at each step (ABScheeresDimM.m). For that, the variables are adimensionalized to compute the matrix coefficients, and then the matrix is multiplied by ω^2, the target's orbit angular speed, for the coefficient factorized by x, y or z, and by ω for the coefficients factorized by u, v and w. Like that, the product $A(\boldsymbol{X})\boldsymbol{X}$ is homogenous to acceleration for the dynamics part.
On the other hand, the computation of the cost function has been added to the main code in order to measure the differences between the models, thanks to a discrete Riemann summation.
The solver is run with the weight matrices considered during the development of the solver last year, that is to say:

$$Q = diag_{6x6}(1e-4)$$
$$R = diag_{3x3}(1.5)$$

The following table is summarizing the different steps of the trajectory to follow:

	x [m]	y [m]	z [m]
Initial position	-1.02	-3500	0.1
Position at the end of hop1	-1.02	-1500	0
Position at the end of hop2	-1.02	-800	0
Position at the end of hop3	-1.02	-250	0
Final position	-1.02	-1	0

Table 1: Planned trajectory

In addition to these steps, some attitude control maneuvers are performed between each Hohmann transfer, implying a need in station keeping as well, that is why the overall trajectory is in fact composed by seven phases, four corresponding to displacement, three to station keeping operations.

7.1 Preliminary results

The projected trajectory is followed easily, the computation is converging. The cost function reaches $J = 4.5639e + 14$.
Thus, with few impulses, the control manages to match the trajectory:

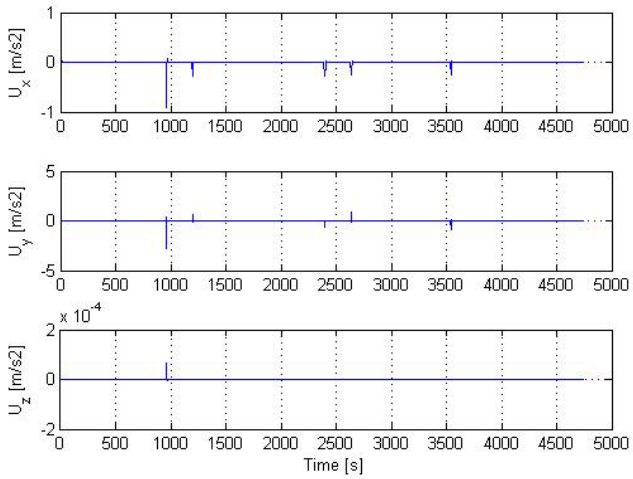


Figure 32: Control profile

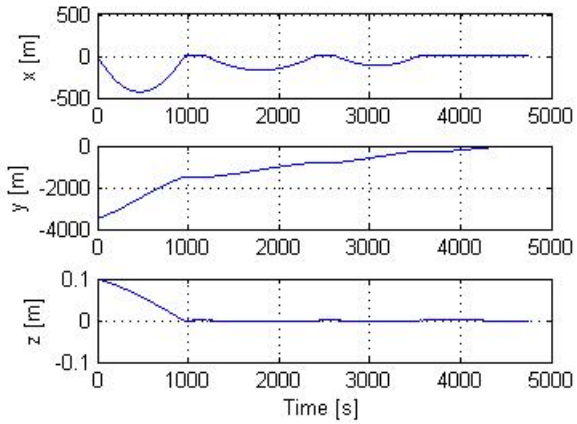


Figure 33: Relative position

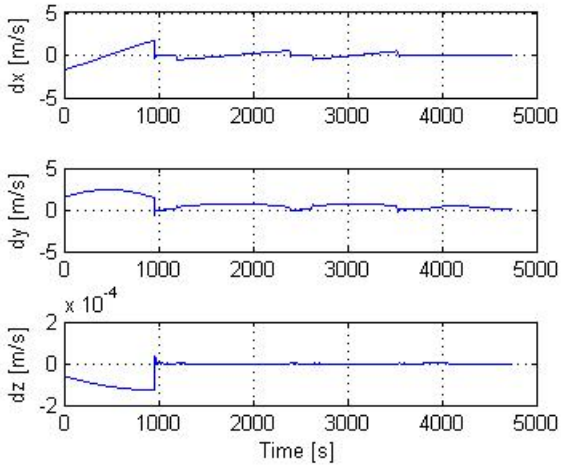


Figure 34: Relative position

Indeed the final vector, $[1.02, -1, 0]$, is reached with a small. To have an idea of the trajectory in space:

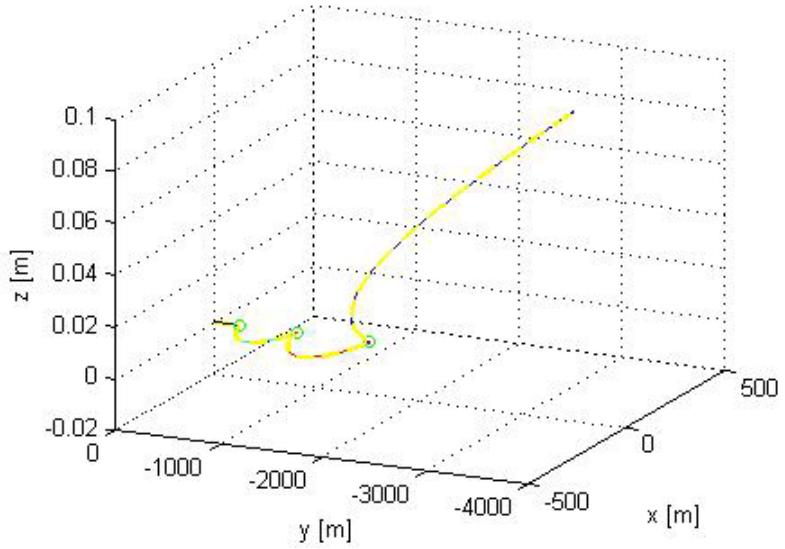


Figure 35: Controlled trajectory

But unfortunately, this could be expected, since a thing is hidden behind this problem: the extreme low values of the dynamics. Indeed, the pursuer is following a trajectory really near to the origin, since the target and the pursuer are in phase and near, implying acceleration with an order of magnitude from 10^{-6} to 10^{-10}. In fact, the effect of the relative dynamics is totally negligible compare to the required control accelerations for the different maneuvers.

7.2 Cost function evaluation

This situation being known, a work on the weight matrices was performed in order to try to obtain a difference between the different parameterization. For that a comparison between three SDC matrices where tried: the exact one, the systematic one and one with all the dynamics parameters equals to zero. The following table is giving some of these results.

Q	R	Jexact	Jsys.	J0
~10^{-4}	1.5	4.5639e+05	4.5639e+05	4.5639e+05
~10^{-4}	150	4.5718e+05	4.5718e+05	4.5718e+05
~10^{-4}	1.5 10^{5}	4.6309e+05	4.6309e+05	4.6309e+05
~10^{-4}	1.5 10^{6}	5.6634e+05	5.6634e+05	5.6634e+05

Table 2: Cost function evaluation

The last value is not convenient since the error on the trajectory at the end is far away too high. To illustrate that, the cost function has been computed for each phase of the maneuver, for the first case and for $R \propto 5.75e5$, with the corresponding position and velocity errors.

	Parameterization	$e_x\ [m]$	$e_v\ [m.s^{-1}]$	Cost function (J)
HOP1	Syst. SDC	1,1298	2,2594	323016,1839
	Exact SDC	1,1298	2,2594	323016,1839
	Linear	1,1298	2,2594	323016,1839
	Zero-dyn	1,1298	2,2594	323014,9166
SK1	Syst. SDC	0,0006	0,0000	26917,6089
	Exact SDC	0,0006	0,0000	26917,6089
	Linear	0,0006	0,0000	26917,6089
	Zero_dyn	0,0006	0,0000	26917,0207
HOP2	Syst. SDC	0,0000	0,0000	68937,6793
	Exact SDC	0,0000	0,0000	68937,6793
	Linear	0,0000	0,0000	68937,6793
	Zero-dyn	0,0000	0,0000	68937,5000
SK2	Syst. SDC	0,0005	0,0000	21859,8623
	Exact SDC	0,0005	0,0000	21859,8623
	Linear	0,0005	0,0000	21859,8623
	Zero_dyn	0,0005	0,0000	21859,7654
HOP3	Syst. SDC	0,3309	0,6617	13969,8065
	Exact SDC	0,3309	0,6617	13969,8065
	Linear	0,3309	0,6617	13969,8065
	Zero-dyn	0,3309	0,6617	13969,7191
SK3	Syst. SDC	0,0005	0,0000	747,1369
	Exact SDC	0,0005	0,0000	747,1369
	Linear	0,0005	0,0000	747,1369
	Zero-dyn	0,0005	0,0000	747,1090
Final	Syst. SDC	0,1060	0,0002	987,2564
	Exact SDC	0,1060	0,0002	987,2564
	Linear	0,1060	0,0002	987,2564
	Zero-dyn	0,1060	0,0002	987,2424

Table 3: errors and cost of each phase

As expected, the differences are not visible, except with a null dynamics, even if negligible. It can be seen that the final position error is near to 10 cm which is totally correct for a safe approach. To enlarge the difference, simulations with a stronger constraint on the command by raising R were performed. The results are summarized in the following table:

	Parameterization	$e_x\ [m]$	$e_v\ [m.s^{-1}]$	Cost function (J)
HOP1	Syst. SDC	8,6386	2,2539	339233,15
	Exact SDC	8,6386	2,2539	339233,15
	Linear	8,6386	2,2539	339233,20
	Zero	15,0590	2,2060	336119,15
SK1	Syst. SDC	162,0809	0,3487	25852,8745
	Exact SDC	162,0809	0,3487	25852,8745
	Linear	162,0802	0,3487	25852,8791
	Zero	166,8999	0,2888	25534,0439
HOP2	Syst. SDC	5,7009	0,0180	69579,1854
	Exact SDC	5,7009	0,0180	69579,1854
	Linear	5,7009	0,0180	69579,1669
	Zero	7,9894	0,0250	67184,5712
SK2	Syst. SDC	44,6089	0,0940	21616,1249
	Exact SDC	44,6089	0,0940	21616,1249
	Linear	44,6088	0,0940	21616,1260
	Zero	45,4527	0,0663	21626,9881
HOP3	Syst. SDC	3,2321	0,6649	14966,0864
	Exact SDC	3,2321	0,6649	14966,0864
	Linear	3,2321	0,6649	14966,0865
	Zero	4,9495	0,6451	14524,5470
SK3	Syst. SDC	47,8235	0,1039	740,2068
	Exact SDC	47,8235	0,1039	740,2068
	Linear	47,8235	0,1039	740,2068
	Zero	48,6322	0,0858	722,9051
Final	Syst. SDC	25,7948	0,0445	1189,9722
	Exact SDC	25,7948	0,0445	1189,9722
	Linear	25,7948	0,0445	1189,9720
	Zero	30,7779	0,0632	1115,1981

Table 4: errors and cost of each phase, a second case

This time the differences are easier to see, at least between the linear and the systematic/exact parameterization, which takes a small advantage on the cost function. But the gain is negligible, and for this case, the final error is to high (25m) to be considered as a plausible trajectory.

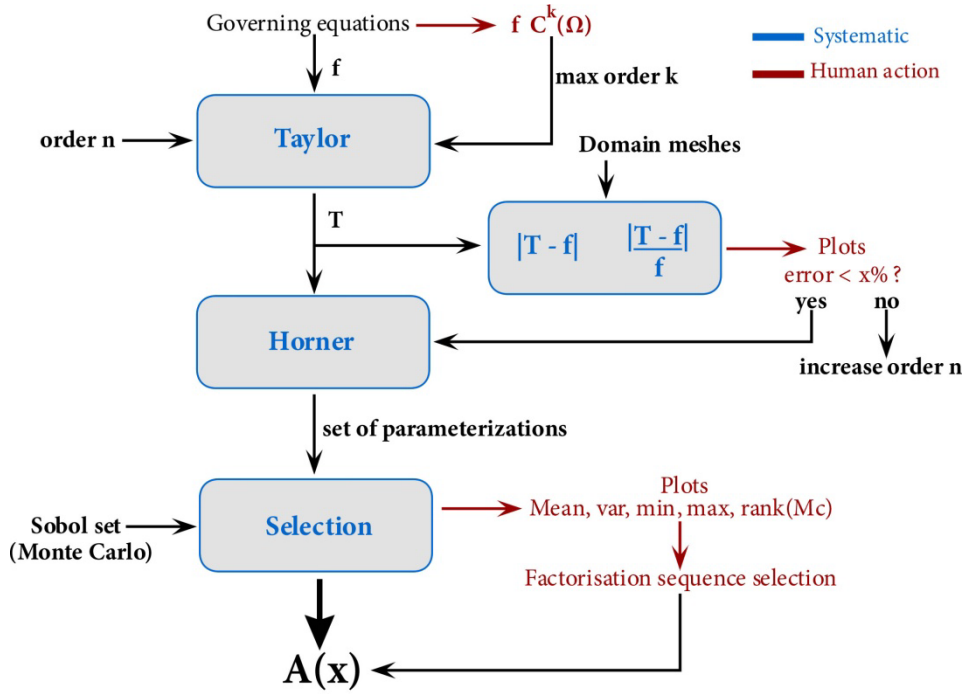
Other values where tested by playing also on the ratio of the coefficients of Q (part corresponding to position, part corresponding to the velocities). But the most important thing to notice is that there are no differences between the different models for this application, no matter the SDC are, making the simulation quite useless. A simple linear controller would give the same performances, but with a constant control law.

7.3 Simulation conclusion

This section can't be really more detailed since the effectiveness of the control resides on the fact that $a_{aimed} = U_{control}$, since the dynamics can totally be neglected. Even if the computed parameterization is well inserted and the pursuer following the good trajectory, the computing cost added for the same performances than a linear controller makes the implementation of an SDRE controller for this trajectory a non sense.

Conclusions

This study aimed to develop a systematic method able to propose a SDC parameterization in the framework of a SDRE control problem for any kind of governing equations, under some specific conditions that have been exposed. The relevance of this method is assured by the sub-optimality of the SDRE control method. The theory assuring the existence of such a method has been set, to be then implemented in a practical case, a rendezvous procedure. Also some semi-systematic codes were developed to measure the accuracy, but needing to make some choices on the variables of interest, and on the criteria to assess the good accuracy. The method is summarized in this figure already presented before:



Unfortunately, the implementation part, in order to have a first example of the power of this method, revealed that the SDRE implementation for the wanted trajectory is a non sense since a 0 dynamics model is as efficient. The SDRE solver developed in a previous study is not in question and can be adapted, but the trajectory under test is not relevant to prove the efficiency of SDRE control compare to basic linear control.

Future developments

The code developed for the production of the SDC matrices being systematic; it can be applied to another control problem. To keep the accuracy study made before, the rendezvous problem can be extended by adding the attitude control, and even the structural problem, as described by Ming Xin and Hejia Pan in [12]. But in this case an update of the code will be required to add the creation of the matrix B that will not be anymore linear, so needing to be factorized as well.

Bibliography

[1] ***Analytical mechanics of aerospace systems, second Edition***
AIAA Education Series; 2009
H.Schaub, J.L.Junkins

[2] ***Solving optimal control problems with generating functions***
Department of Aerospace Engineering, University of Michigan
D.J. Scheeres, C.Park, V.Guibout

[3] ***Commande des systèmes linéaires***
Lectures notes, ISAE-ENSICA 2010
J.Bordeneuve-Guibé

[4] ***Nonlinear regulation and nonlinear H∞ control via the state-dependent Riccati equation technique***
Proceedings from the First International Conference on Nonlinear Problems in Aviation and Aerospace, 1996
J.R.Cloutier, C.N.D'Souza, C.P.Mracek

[5] ***Systematic and effective design of nonlinear feedback controllers via the state-dependent Riccati equation (SDRE) method***
Annual Reviews in Control 34, 2010
T.Cimen

[6] ***State-Dependent Riccati equation (SDRE) regulation of systems with state and control nonlinearities***
National Institute of Aerospace, Hampton, Virginia, 2004
S.C.Beeler

[7] ***Suboptimal control for the nonlinear quadratic regulator problem***
Automatica volume 11, Elsevier Ltd, 1975
A.Wernli, G.Cook

[8] **A *'rhobustness' margin for unstructured nonlinear time-varying deviations***
Proceedings of the 30th IEEE Conference on Decision and Control, 1991
R.W.Bass

[9] ***Mathématiques Tout-en-un, 2^e année MP***
Editions Dunod, 2006
Directed by: C.Deschamps, A.Warusfel

[10] ***Calcul différentiel***
Lectures notes, 2005
Christian Squarcini

[11] ***On the multivariate Horner scheme***
Society for Industrial and Applied Mathematics, 2000
J.M.Peña, T.Sauer

[12] ***Integrated nonlinear optimal control of spacecraft in proximity operations***
Department of Aerospace Engineering, Mississippi State University, 2009
Ming Xin, Hejia Pan

[13] ***Greedy Algorithms for Optimizing Multivariate Horner Schemes***
ACM SIGSAM Bulletin, Vol 38, No. 1, March 2004
M.Ceberio, V.ladik Kreinovich