



UNIVERSITÀ POLITECNICA DELLE MARCHE

FACOLTA' DI INGEGNERIA
Corso di Laurea in Ingegneria Informatica e dell'Automazione

**SINCRONIZZAZIONE DI SISTEMI
DINAMICI NON LINEARI
SU MANIFOLD IN PRESENZA
DI DISTURBI**

**SYNCHRONIZATION OF
NON LINEAR DYNAMICAL SYSTEMS
ON MANIFOLDS IN THE
PRESENCE OF DISTURBANCES**

Relatore:
Chiar.mo Prof.
SIMONE FIORI

Correlatore:
Dr. HWEE KUAN LEE

Presentata da:
ITALO CERVIGNI
S1079961

Anno Accademico 2018/19

*The most beautiful thing we can experience
is the mysterious. It is the source of
all true art and science
— A. Einstein, Physicist*

Contents

1	Introduction	1
2	Dynamical systems on manifolds and Lie groups	3
2.1	Notation in manifold calculus	3
2.2	Newton-like second-order dynamical systems on manifolds	4
3	Regulation of second-order dynamical systems	7
3.1	Regulation of a second-order system by an M-PID controller	8
3.2	Control efforts	10
3.3	Application of the general M-PID control theory to time-synchronization of second-order systems	10
4	Interesting oscillators	13
4.1	Duffing-type oscillators	13
4.2	Numerical simulation of the hard Duffing oscillator	15
4.3	Numerical simulation of the soft Duffing oscillator	17
4.3.1	Euler method	17
4.3.2	Heun method	19
4.3.3	Runge (RK2) method	19
4.4	Numerical simulation of the double-well Duffing oscillator	21
5	Numerical simulations	24
5.1	The action of the cancelling component of the control field	26
5.2	Evaluation of the action of M-PID controllers on twin oscillators with identical reference points	28
6	Real systems application	39
6.1	Simulations on moderately damped oscillators	41
6.2	Simulations on oscillators affected by disturbances	43
6.3	Simulations on oscillators in the presence of delays	48

6.4	Application of synchronization to signal masking	50
7	Oscillators on the orthogonal group $SO(3)$	51
7.1	Visual rendering via a group action on a manifold	52
7.2	Numerical simulation of a hard Duffing oscillator	52
8	Mathematical model of a quadcopter drone	55
8.1	Mathematical model of quadrotor drone on $SO(3) \times \mathfrak{so}(3) \times \mathbb{R}^3$. .	55
8.2	Numerical simulation methods	59
8.3	Results of the numerical simulation of the quadcopter drone . . .	60
9	Synchronization applied to quadcopter drones	64
9.1	Extension of PID regulation to Lie groups (L-PID)	64
9.1.1	Control efforts	66
9.2	Application of L-PID to the time-synchronization of the attitude of two quadrotor drones	67
9.3	Numerical experiments on the time-synchronization of drones . .	69
9.3.1	Quadrotor drones synchronization – attitude control	69
9.3.2	Quadrotor drones synchronization – mixed attitude and position control	70
9.3.3	Quadrotor drones synchronization in the real cases	78
10	Conclusion	81

Sommario

Questa tesi si concentra sullo studio e il controllo di sistemi non lineari la cui traiettoria giace su alcuni Manifold, in particolare lo studio sarà concentrato sull'ipersfera \mathbb{S}^{n-1} e sullo spazio delle iperrotazioni $SO(3)$, quest'ultimo molto utile per rappresentare l'orientamento di un corpo tridimensionale nello spazio. Inizialmente verrà studiata la dinamica di alcune tipologie di oscillatori su Manifold. Inseguito, dati due oscillatori, un follower e un leader, verrà applicato un controllo PID per permettere la sincronizzazione della traiettoria del follower con quella del leader. Il tutto sarà analizzato sia nel caso ideale che nel caso reale in presenza di disturbi e ritardi. Infine verranno prese in esame le dinamiche di oggetti reali, come i droni e i satelliti, al fine di ottenerne la sincronizzazione, sempre con la possibile presenza di disturbi e ritardi.

Abstract

This thesis focuses on the study and control of non-linear systems whose trajectory lies on some Manifolds, in particular the study will focus on the hypersphere \mathbb{S}^{n-1} and on the space of the hyperrotations $SO3$, the latter very useful for representing the orientation of a three-dimensional body in space. Initially the dynamics of some types of oscillators on Manifold will be studied. Following, given two oscillators, a follower and a leader, a PID control will be applied to allow synchronization of the trajectory of the follower with that of the leader. Everything will be analyzed both in the ideal case and in the real case in the presence of disturbances and delays. Finally, the dynamics of real objects, such as drones and satellites, will be examined in order to obtain synchronization, always with the possible presence of disturbances and delays

1. Introduction

Proportional-integral-derivative control (PID) is a well-known and widely utilized control theory [1]. The general idea to control a dynamical system into approaching a desired state trajectory is to define an error field and to define an input signal as a linear combination of the error signal, of the time-derivative of the error signal and of a time-integral of the error. The vast majority of PID schemes were developed to regulate dynamical systems whose input, state and output signals take values in a hyper-cube \mathbb{R}^n . The present document introduces a novel PID-inspired control theory for second-order dynamical systems whose state space is not a flat field but rather a *curved state manifold*, which will be referred to as M-PID regulation. Control of systems on state manifolds is a relatively new research branch in non-linear control theory which is gaining increasing interest especially in the field of mechanical systems regulation [4, 5, 15, 22].

In particular, in the present document we focus on a specific instance of non-linear control that affords the *synchronization* in time of the dynamics of two (either identical or non-identical) non-linear systems evolving on the same state manifold. Synchronization is a well-studied phenomenon that occurs spontaneously in nature when a large number of individuals are bound in a network [23], such as in a bird flock or within a heart muscle tissue. Synchronization of a leader-follower pair of dynamical systems found widespread applications in sciences and engineering as in secure transmission of information by signal masking [12] as well as in humanoid robotics [18]. Even in this field, the large majority of synchronization techniques were developed taking as systems' state space the flat field \mathbb{R}^n . In the present research work, we shall be assuming that the state of the leader is accessible to the follower and that both states are accessible to the controller, although we assume that the leader state might be received upon being corrupted by noise (this is the case, for example, in brain cells networks [21]). Research works suggested that unavailability of accurate state measurements may be overcome by the help of Kalman filtering [11]. In this research work, we shall cope with synchronization of two second-order dynamical systems whose state spaces are curved state

manifolds. The behavior of the devised M-PID-type time-synchronization theory will be illustrated through numerical experiments on two Duffing-type non-linear oscillators tailored to evolve on the unit hyper-sphere.

In order to simulate the behavior of non-linear dynamical systems on manifolds and of a control strategy on a computing platform, it is necessary to develop adequate numerical methods to approximate the trajectories generated by these dynamical systems and to compute the values of the control fields. Classical numerical methods, such as the Euler method or the Runge-Kutta class, will fail if applied directly, as they are intrinsically designed to work on flat spaces and are not suitable to keep up with the non-flat structure of curved manifolds. These numerical methods may, however, be extended so as to cope with curved state manifolds by means of numerical calculus on manifold [7, 8].

The present document is organized as follows. Section 2 recalls some notation from manifold calculus and describes some kinds of second-order dynamical systems on smooth manifolds, centered on the Duffing-type oscillators, that will be taken as cases of study in the following sections. Section 3 outlines possible extensions of a proportional-integral-derivative (PID) scheme to abstract systems whose state spaces are curved state manifolds, with particular emphasis to application to time-synchronization of second-order dynamical systems. Section 4 describes some interesting oscillators that will be studied in this document and their dynamics visualized through numerical simulations using Euler, Heun and Runge methods. Section 5 illustrates the use of a cancelling component of the control field to get a better control. Section 6 illustrates the application to real cases in the presence of damping, disturbances and delays and also the application of this studies to signal masking. Section 7 describes oscillators on another manifold, the orthogonal group and shows some numerical simulations of oscillators on this manifold. Section 8 describes the mathematical model of a quadcopter drone with some simulation while section 9 illustrates the application of synchronization to drones in the ideal and real cases.

2. Dynamical systems on manifolds and Lie groups

The present section recalls some notation from manifold calculus and describes some kinds of dynamical systems on smooth manifolds and Lie groups.

2.1 Notation in manifold calculus

Let M denote a Riemannian manifold. At a point $x \in M$, the tangent space to the manifold M is denoted as $T_x M$. The symbol TM denotes the tangent bundle defined as $TM := \{(x, v) \mid x \in M, v \in T_x M\}$.

A Riemannian manifold M is endowed with a bilinear, positive-definite form $\langle \cdot, \cdot \rangle_x : T_x M \times T_x M \rightarrow \mathbb{R}$. A local metric $\langle \cdot, \cdot \rangle_x$ also defines a local norm $\|v\|_x := \sqrt{\langle v, v \rangle_x}$, for $v \in T_x M$.

The Riemannian gradient of a function $\psi : M \rightarrow \mathbb{R}$ evaluated at the point $x \in M$ is denoted as $\text{grad}_x \psi$. The covariant derivative of a vector field w in the direction of a vector $v \in T_x M$ is denoted as $\nabla_v w$. We assume M to be endowed with a *metric connection* (namely, that the covariant derivative of the metric tensor is identically zero) and the parallel transport operator $P^{x \rightarrow y}$ transports a tangent vector from $T_x M$ to $T_y M$.

A *manifold exponential* map $\exp : TM \rightarrow M$ applies as $\exp_x(v)$. Its inverse ‘log’ is defined only locally and is termed *manifold logarithm*. Given points $x, y \in M$, a manifold logarithm computes a tangent vector $v = \log_x(y) \in T_x M$ such that $\exp_x(v) = y$.

Given two points $x, y \in M$ connectable by a geodesic arc, their *Riemannian distance* is denoted by $d(x, y)$. On a Riemannian manifold, the distance between two nearby points may be evaluated by $d(x, y) = \|\log_x(y)\|_x$. A fundamental result of the calculus on manifolds states that the Riemannian gradient of a squared distance

function reads

$$\text{grad}_x d^2(x, y) = -2 \log_x(y), \quad (2.1)$$

wherever the logarithm is defined.

The covariant derivative of a vector field $w_x \in T_x M$ in the direction of a vector $v \in T_x M$ is denoted as $\nabla_v w$. We assume M to be endowed with a *metric connection* (namely, that the covariant derivative of the metric tensor is identically zero). The parallel transport operator $P^{x \rightarrow y}$ transports a tangent vector from $T_x M$ to $T_y M$. Parallel transport and covariant derivation are closely related to one another, in particular, covariant derivation may be expressed in terms of parallel transport as follows:

$$\nabla_v w = \lim_{h \rightarrow 0} \frac{P^{\gamma(h) \rightarrow x}[w_{\gamma(h)}] - w_x}{h}, \quad (2.2)$$

where γ denotes any smooth curve such that $\gamma(0) = x \in M$ and $\dot{\gamma}(0) = v \in T_x M$. Such relationship leads to a numerical approximation of the covariant derivative at a point x , namely

$$\nabla_v w \approx \frac{P^{\tilde{x} \rightarrow x}(w_{\tilde{x}}) - w_x}{h}, \quad (2.3)$$

where $\tilde{x} = \tilde{x}(x, h)$ denotes a point in M close to x .

2.2 Newton-like second-order dynamical systems on manifolds

The *kinetic energy function* $\mathcal{K} : TM \rightarrow \mathbb{R}$ for a dynamical system is defined by $\mathcal{K} := \frac{1}{2} \langle v, v \rangle_x$ for $(x, v) \in TM$. On a Riemannian manifold, the metric is positive-definite, hence, on every trajectory, it holds that $\mathcal{K} \geq 0$. A *potential energy function* $\mathcal{V} : M \rightarrow \mathbb{R}$ depends on the coordinate $x \in M$ only. In absence of any external solicitation, the dynamical system generates a trajectory $x = x(t)$ that follows the landscape of the potential energy function. The total energy (or Hamiltonian) \mathcal{H} of the dynamical system (2.4) is defined by $\mathcal{H} := \mathcal{K} + \mathcal{V}$.

A general second-order dynamical system on a Riemannian manifold M reads

$$\nabla_{\dot{x}} \dot{x} = \mathbb{F}(t, x, \dot{x}), \quad (2.4)$$

where $\mathbb{F} : \mathbb{R} \times TM \rightarrow M$ denotes a *forcing term*, hence the formulation (2.4) is termed *Newton-like* (the notion of ‘mass’ does not play any role in the context of Newton-like systems on manifolds and is generally absorbed into the metric). The forcing term may be composed as the sum of several contributions, like:

- *Friction-type damping force field:* This kind of damping is expressed by the forcing term $-\mu \|v\|_x^{2(\varepsilon-1)} v$, with $\mu \geq 0$ denoting a friction coefficient. Since, by definition, $\|v\|_x^2 = 2\mathcal{K}$, this kind of damping takes values $-\mu(2\mathcal{K})^{\varepsilon-1} v$.
- *Active control field:* It represents a vector field that depends on the system variables that brings energy into the system from an external source. In the present context, such vector field represents active control and will be denoted as u .
- *Conservative force field:* It stems from a potential energy function \mathcal{V} as its gradient, namely, it is defined as $-\text{grad}_x \mathcal{V}$.

The second-order dynamical system (2.4) may be split into two coupled differential equations on the tangent bundle TM as:

$$\begin{cases} \dot{x} = v, & \text{(Evolution of the position variable)} \\ \nabla_v v = -\mu \|v\|_x^{2(\varepsilon-1)} v - \text{grad}_x \mathcal{V} + u, & \text{(Evolution of the velocity variable)} \end{cases} \quad (2.5)$$

where $(x, v) \in TM$ at any time.

It is straightforward to prove that, in the absence of any active control, the system (2.5) loses energy at a rate proportional to the kinetic energy, unless $\mu = 0$, in which case the system is conservative.

Theorem 2.2.1. *In the absence of active control, the total energy of the system (2.5) changes over time according to the power law $\dot{\mathcal{H}} = -\mu(2\mathcal{K})^\varepsilon$.*

Proof. By manifold calculus, we get:

$$\frac{d\mathcal{H}}{dt} = \frac{d\mathcal{V}}{dt} + \frac{d\mathcal{K}}{dt} = \langle \text{grad}_x \mathcal{V}, \dot{x} \rangle_x + \langle v, \nabla_v v \rangle_x. \quad (2.6)$$

From the equations (2.5), taking $u = 0$, it follows that

$$\frac{d\mathcal{H}}{dt} = \langle \text{grad}_x \mathcal{V}, v \rangle_x + \langle v, -\mu \|v\|_x^{2(\varepsilon-1)} v - \text{grad}_x \mathcal{V} \rangle_x. \quad (2.7)$$

Therefore, from the bilinearity of the inner product and from the definition of the kinetic energy, it follows that $\dot{\mathcal{H}} = -\mu \|v\|_x^{2\varepsilon} \leq 0$, hence the assertion. \square

Only if $\mu = 0$ the dynamical system may retain its initial energy endlessly.

An instance of the general-type dynamical system (2.5) is The Duffing oscillator, named after Georg Duffing, is a non-linear second-order differential equation used to model the motion of a damped oscillator with a more complex potential than in simple harmonic motion [13]. In physical terms, it models, for instance, a

spring pendulum whose stiffness does not exactly obey Hooke's law. An extension of the Duffing potential to Riemannian manifolds reads:

$$\mathcal{V}^{(\text{duf})} := \pm \frac{1}{2}d^2(x, r) \pm \frac{1}{4}\kappa d^4(x, r), \quad (2.8)$$

where $d(\cdot, \cdot)$ denotes the Riemannian (geodesic) distance on the manifold M , $\kappa > 0$ is a free parameter and $r \in M$ denotes a reference point. The signs \pm were introduced to account for the *soft*, *hard* and the *double-well* Duffing oscillator. Let's distinguish the case that both addenda appear with the same sign or the case that the two addenda appear with opposite sign:

- If the Duffing potential is written as $\mathcal{V}_{++}^{(\text{duf})} = \frac{1}{2}d^2(x, r) + \frac{1}{4}\kappa d^4(x, r)$, then $\mathcal{V}_{++}^{(\text{duf})} \geq 0$, while if $\mathcal{V}_{--}^{(\text{duf})} = -\mathcal{V}_{++}^{(\text{duf})}$, then $\mathcal{V}_{--}^{(\text{duf})} \leq 0$ for any value of $x \in M$. In both cases, the point $x = r$ is the only point where the potential vanishes to zero and the only point of stationarity of the potential.
- If the Duffing potential is written as $\mathcal{V}_{+-}^{(\text{duf})} = \frac{1}{2}d^2(x, r) - \frac{1}{4}\kappa d^4(x, r)$ or as $\mathcal{V}_{-+}^{(\text{duf})} = -\mathcal{V}_{+-}^{(\text{duf})}$, then the Duffing potential may change sign according to the magnitude of the distance $d^2(x, r)$.

According to the calculation rule (2.1), its Riemannian gradient reads:

$$\text{grad}_x \mathcal{V}^{(\text{duf})} = [\mp 1 \mp \kappa d^2(x, r)] \log_x(r). \quad (2.9)$$

Therefore, in the case of a hard Duffing potential, the associated dynamical system reads:

$$\begin{cases} \dot{x} &= v, \\ \nabla_v v &= -\mu \|v\|_x^{2(\varepsilon-1)} v + [1 + \kappa d^2(x, r)] \log_x(r). \end{cases} \quad (2.10)$$

Moreover, given a soft Duffing potential, the associated dynamical system reads:

$$\begin{cases} \dot{x} &= v, \\ \nabla_v v &= -\mu \|v\|_x^{2(\varepsilon-1)} v + [1 - \kappa d^2(x, r)] \log_x(r). \end{cases} \quad (2.11)$$

Instead, given a double-well Duffing potential, the associated dynamical system reads:

$$\begin{cases} \dot{x} &= v, \\ \nabla_v v &= -\mu \|v\|_x^{2(\varepsilon-1)} v + [-1 + \kappa d^2(x, r)] \log_x(r). \end{cases} \quad (2.12)$$

3. Regulation of second-order dynamical systems

Feedback control, as illustrated in the Figure 3.1 is the cornerstone of modern control theory.

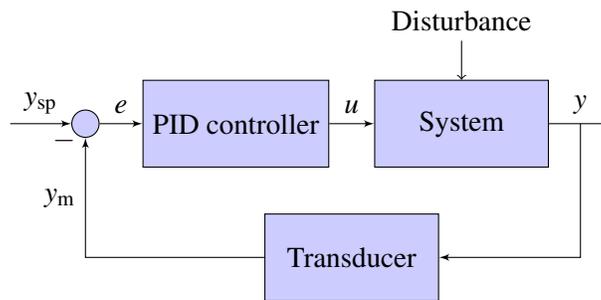


Figure 3.1: Feedback control scheme, where y_{sp} denotes the set point, y_m denotes a measure of the controlled variable y , u denotes a control signal and e denotes an error signal, which quantifies the discrepancy between the measured controlled variable and the set point.

This section outlines possible extensions of a proportional-integral-derivative (PID) scheme to abstract systems whose state-spaces are smooth manifolds (i.e., state-manifolds).

A second-order system on a smooth manifold M is described by

$$\begin{cases} \dot{x}(t) = v(t), & t \geq 0, \\ \nabla_{\dot{x}(t)} v(t) = \mathbb{S}(t, x(t), v(t)) + u(t), \end{cases} \quad (3.1)$$

where $x(t) \in M$ denotes the system state, $v(t) \in T_{x(t)}M$ denotes the system velocity, $\mathbb{S} : \mathbb{R} \times TM \rightarrow TM$ denotes a time-varying state-transition map, and $u(t) \in TM$

denotes the system input vector field. The initial states are $x(0) = x_0 \in M$ and $v(0) = 0$. The system (3.1) may be rewritten compactly as

$$\nabla_{\dot{x}(t)} \dot{x}(t) = \mathbb{S}(t, x(t), \dot{x}(t)) + u(t), \quad t \geq 0, \quad (3.2)$$

which resembles a Newton's equation on the manifold M . (In fact, notice that, if the terms on the right-hand side are set to zero, the system evolves along a geodesically-uniform trajectory.)

A second-order system admits *position control* as well as *velocity control*, which are discussed separately in the following.

3.1 Regulation of a second-order system by an M-PID controller

The purpose of a positional control is to drive the state $x(t)$ toward a reference point, which may be moving over time, denoted by $z(t) \in M$. The first step consists in defining an *error field*, which we take as

$$e(t) := \log_{x(t)} z(t) \in T_{x(t)} M, \quad (3.3)$$

that generalizes the Euclidean error $e(t) = z(t) - x(t)$ for the standard case that $M = \mathbb{R}^n$. Note that $e(t)$ is a tangent vector field in TM .

Note that, for a second-order system, it is often impossible to control the state-velocity directly as it is only possible to control the state-acceleration (as in a mass-spring-damper system, where the spring and the damper produce mechanical forces that affect the acceleration of the mass). A position M-PID controller may then be designed as:

$$u(t) := \kappa_P e(t) + \kappa_I \int_0^t \mathbf{P}^{x(\tau) \rightarrow x(t)}(e(\tau)) d\tau + \kappa_D \nabla_{\dot{x}(t)} e(t) + u_C(t) \in T_{x(t)} M. \quad (3.4)$$

however, a choice like this would lead to a control strategy that is both computationally complicated to implement and theoretically inconclusive.

Therefore, on the basis of the previous discussion, we suggest the following control scheme that generalizes a PD scheme discussed in [10, 17]:

$$\begin{cases} \nabla_{\dot{x}(t)} \dot{x}(t) = \mathbb{S}(t, x(t), \dot{x}(t)) + u(t), \quad t \geq 0, \quad (\text{Second-order dynamical system}) \\ e(t) := \log_{x(t)} z(t), \quad (\text{Control error}) \\ \varepsilon(t) := \mathbf{P}^{z(t) \rightarrow x(t)}(\dot{z}(t)) - \dot{x}(t), \quad (\text{Control error velocity}) \\ w(t) := \int_0^t \mathbf{P}^{x(\tau) \rightarrow x(t)}(e(\tau)) d\tau, \quad (\text{Integrated control error}) \\ u(t) := \kappa_P e(t) + \kappa_I w(t) + \kappa_D \varepsilon(t) + u_C(t) \quad (\text{Control field}). \end{cases} \quad (3.5)$$

with $u_C(t)$ to be defined. Taking the covariant derivative of the error field ε leads to:

$$\nabla_{\dot{x}}\varepsilon = \nabla_{\dot{x}}\mathbf{P}^{z \rightarrow x}(\dot{z}) - \mathbb{S}(t, x, \dot{x}) - u. \quad (3.6)$$

Plugging the equations (9.1) into the equation (9.2) yields

$$\nabla_{\dot{x}}\varepsilon = \nabla_{\dot{x}}\mathbf{P}^{z \rightarrow x}(\dot{z}) - \mathbb{S}(t, x, \dot{x}) - \kappa_P e - \kappa_I \omega - \kappa_D \varepsilon - u_C. \quad (3.7)$$

The term u_C is meant to cancel all terms in the tight-hand side of the above expression that do not depend on the error-type fields [16], namely:

$$u_C := \nabla_{\dot{x}}\mathbf{P}^{z \rightarrow x}(\dot{z}) - \mathbb{S}(t, x, \dot{x}). \quad (3.8)$$

The resulting error dynamics is governed by the equation

$$\nabla_{\dot{x}}\varepsilon + \kappa_P e + \kappa_I \omega + \kappa_D \varepsilon = 0. \quad (3.9)$$

The above second-order error system is characterized by the following result.

Theorem 3.1.1. *The error system (3.9) converges asymptotically to the state zero as long as $\kappa_P > 0$, $\kappa_D \geq 0$, $\kappa_I = \tilde{\kappa}_I \langle \omega, \varepsilon \rangle_x$, with $\tilde{\kappa}_I \geq 0$ (at least one among κ_D and $\tilde{\kappa}_I$ must differ from zero).*

Proof. The convergence of the synchronization error to zero is proven by showing that the following function is Lyapunov:

$$\mathscr{W}(x, z) := \frac{1}{2} \langle \varepsilon, \varepsilon \rangle_x + \frac{\kappa_P}{2} d^2(x, z). \quad (3.10)$$

The function \mathscr{W} is written as the sum of two terms, the first of which is always non-negative and, since the base-manifold M is Riemannian, the term $\langle \varepsilon, \varepsilon \rangle_x$ is zero only if $\varepsilon = 0$. The second term is non-negative because, by hypothesis, $\kappa_P > 0$. Deriving the function \mathscr{W} with respect to the time gives:

$$\begin{aligned} \dot{\mathscr{W}} &= \kappa_P \langle e, \varepsilon \rangle_x + \langle \nabla_{\dot{x}}\varepsilon, \varepsilon \rangle_x \\ &= \langle -\kappa_P e - \kappa_I \omega - \kappa_D \varepsilon, \varepsilon \rangle_x + \kappa_P \langle e, \varepsilon \rangle_x \\ &= -\kappa_D \langle \varepsilon, \varepsilon \rangle_x - \tilde{\kappa}_I \langle \omega, \varepsilon \rangle_x^2, \\ &\leq 0, \end{aligned} \quad (3.11)$$

which proves the assertion. □

Oddly enough, the value of the constant κ_P does not influence the rate of decrease $\dot{\mathscr{W}}$.

Notice that positional control implies velocity control (although the opposite is not true, in general).

3.2 Control efforts

The ‘control effort’ is a scalar index that quantifies the magnitude of the control field u . In physical control systems, the control effort is related to energy consumption of the actuator and its quantification is used to evaluate the suitability of an actuator with respect to a given control goal [14]. Formally, we define a control effort function $\sigma : TM \rightarrow \mathbb{R}$ as:

$$\sigma := \frac{1}{2} \|u\|_x^2. \quad (3.12)$$

Likewise, we define a control effort function $\sigma_C : TM \rightarrow \mathbb{R}$ and a function $\sigma_{\text{PID}} : TM \rightarrow \mathbb{R}$ associated to the ‘cancelling’ component of the control field and to the PID component of the same control field, as:

$$\sigma_C := \frac{1}{2} \|u_C\|_x^2, \quad \sigma_{\text{PID}} := \frac{1}{2} \|u - u_C\|_x^2. \quad (3.13)$$

It is important to evaluate the control efforts associated to the different components of the above-proposed M-PID control schemes. It is clear that the control effort σ_{PID} is directly related to the parameters κ_P , κ_I , κ_D .

3.3 Application of the general M-PID control theory to time-synchronization of second-order systems

Synchronization of systems on manifolds is an instance of non-linear control where the reference trajectory is generated by a dynamical system (which may coincide with, or differ from, the controlled system). The dynamical system that provides a reference is termed *leader*, described by a state-transition operator \mathbb{S}_L , while the controlled system will be referred to as *follower*, described by a state-transition operator \mathbb{S}_F . We suggest the following leader-follower synchronization scheme:

$$\left\{ \begin{array}{l} \dot{z}(t) = w(t), \quad \nabla_{w(t)} w(t) = \mathbb{S}_L(t, z(t), w(t)), \quad t \geq 0, \quad (\text{Leader dynamical system}) \\ \dot{x}(t) = v(t), \quad \nabla_{v(t)} v(t) = \mathbb{S}_F(t, x(t), v(t)) + u(t), \quad (\text{Follower dynamical system}) \\ e(t) := \log_{x(t)} z(t), \quad (\text{Control error}) \\ \varepsilon(t) := \mathbf{P}^{z(t) \rightarrow x(t)}(\dot{z}(t)) - \dot{x}(t), \quad (\text{Control error velocity}) \\ \omega(t) := \int_0^t \mathbf{P}^{x(\tau) \rightarrow x(t)}(e(\tau)) d\tau, \quad (\text{Integrated control error}) \\ u_C(t) := \nabla_{\dot{x}(t)} \mathbf{P}^{r(t) \rightarrow x(t)}(\dot{z}(t)) - \mathbb{S}_F(t, x(t), \dot{x}(t)), \quad (\text{Cancelling component of the control field}) \\ u(t) := \kappa_P e(t) + \kappa_I \omega(t) + \kappa_D \varepsilon(t) + u_C(t), \quad (\text{M-PID control field}) \end{array} \right. \quad (3.14)$$

The term u_C apparently ‘cancels’ the internal dynamics of the follower system and replaces it with a copy of the internal dynamics of the leader system. Notice that

the leader system dynamics and the follower system dynamics have been split in two first-order equations by introducing two velocity fields v, w .

In order to implement the leader system, the controlled follower system and the control scheme on a computing platform, it is necessary to come up with discrete-time versions of the equations (3.14).

The first step consists in replacing the continuous-time variable t with a discrete-time index $k = 0, 1, 2, \dots$ related by $t = hk$, where $h > 0$ is a discretization stepsize that is generally much smaller than 1. Likewise, the continuous-time variables describing the dynamics of the systems as well as the control field will be replaced by their discrete-time counterparts.

The equations (3.14) are then replaced by their numerically-approximated counterparts and, in particular, the approximation (2.3) was repeatedly made use of:

$$\left\{ \begin{array}{l} z_{k+1} := \exp_{z_k}(h w_k), \quad k = 0, 1, 2, \dots, \\ w_{k+1} := \mathbf{P}^{z_k \rightarrow z_{k+1}}(w_k + h \mathbb{S}_L(hk, z_k, w_k)), \\ x_{k+1} := \exp_{x_k}(h v_k), \\ v_{k+1} := \mathbf{P}^{x_k \rightarrow x_{k+1}}(v_k + h \mathbb{S}_F(hk, x_k, v_k) + h u_k), \\ e_k := \log_{x_k} z_k, \\ \varepsilon_k := \mathbf{P}^{z_k \rightarrow x_k}(w_k) - v_k, \\ \omega_k := \begin{cases} h e_0, & \text{if } k = 0, \\ \mathbf{P}^{x_{k-1} \rightarrow x_k}(\omega_{k-1}) + h e_k, & \text{if } k > 0, \end{cases} \\ u_{C,k} := \begin{cases} -\mathbb{S}_F(0, x_0, v_0), & \text{if } k = 0, \\ \frac{1}{h} (\mathbf{P}^{r_k \rightarrow x_k}(w_k) - \mathbf{P}^{x_{k-1} \rightarrow x_k}[\mathbf{P}^{r_{k-1} \rightarrow x_{k-1}}(w_{k-1})]) - \mathbb{S}_F(hk, x_k, v_k), & \text{if } k > 0, \end{cases} \\ u_k := \kappa_P e_k + \kappa_I \omega_k + \kappa_D \varepsilon_k + u_{C,k}, \end{array} \right. \quad (3.15)$$

where $x_0, z_0 \in \mathbf{M}$, $v_0 \in T_{x_0}\mathbf{M}$ and $w_0 \in T_{z_0}\mathbf{M}$ are given initial states. The dynamics of the leader system is described by the variables pair $(z_k, w_k) \in T\mathbf{M}$ and the dynamics of the follower is described by the variables pair $(x_k, v_k) \in T\mathbf{M}$. The flow of the continuous-time dynamics of the systems has been approximated numerically by a forward Euler stepping [8], while special care has been devoted to the numerical approximation of the cumulative error control term ω_k (via an transported-accretion method) and of the cancelling component of the control field $u_{C,k}$ (via an incremental-ratio-type approximation of covariant derivation).

It is worth noticing that, in the equation to compute the cancelling component of the control field, in general $\mathbf{P}^{x_{k-1} \rightarrow x_k} \circ \mathbf{P}^{r_{k-1} \rightarrow x_{k-1}} \neq \mathbf{P}^{r_{k-1} \rightarrow x_k}$. The equality holds only when the three points belong to the same geodesic arc. however, upon synchronization, r_k and x_k are pretty close to one another and so are x_k and x_{k-1}

because of the small value of the stepsize h , therefore, from a numerical point of view, the equation to compute the cancelling component of the control field might be replaced by:

$$u_{C,k} := \frac{1}{h} (\mathbf{P}^{r_k \rightarrow x_k}(w_k) - \mathbf{P}^{r_{k-1} \rightarrow x_k}(w_{k-1})) - \mathbb{S}_F(hk, x_k, v_k), \quad k > 0. \quad (3.16)$$

The above numerical approximation is lighter in computational complexity than the approximation in (3.15).

A better numerical approximation of the actual flows of the system might be obtained by replacing the forward Euler stepping method with higher-order numerical stepping schemes, such as second-order Heun and Runge schemes and a fourth-order Runge-Kutta scheme [8]. In the present context, a forward Euler scheme looks adequate to illustrate the behavior of the control scheme devised in the above sections.

4. Interesting oscillators

The problem of synchronization will be studied starting from the simple case of oscillators and then it will be extended to the more complex dynamics of real objects like drones and satellites. This section introduces the Duffing-type oscillator that will be used as a "toy problem" for the study of synchronization.

4.1 Duffing-type oscillators

The Duffing oscillator, named after Georg Duffing, is a non-linear second-order differential equation used to model the motion of a damped oscillator with a more complex potential than in simple harmonic motion [13]. In physical terms, it models, for example, a spring pendulum whose spring's stiffness does not exactly obey Hooke's law. An extension of the Duffing potential reads:

$$\mathcal{V}^{(\text{duf})} := \pm \frac{1}{2}d^2(x, r) \pm \frac{1}{4}\kappa d^4(x, r), \quad (4.1)$$

where again $d(\cdot, \cdot)$ denotes the Riemannian (geodesic) distance on the manifold M , $\kappa > 0$ is a free parameter and $r \in M$ denotes a reference point. The signs \pm were introduced to account for the *soft* and the *double-well* Duffing oscillator. Let's distinguish the case that both addenda appear with the same sign or the case that the two addenda appear with opposite sign:

- If the Duffing potential is written as $\mathcal{V}_{++}^{(\text{duf})} = \frac{1}{2}d^2(x, r) + \frac{1}{4}\kappa d^4(x, r)$, then $\mathcal{V}_{++}^{(\text{duf})} \geq 0$, while if $\mathcal{V}_{--}^{(\text{duf})} = -\mathcal{V}_{++}^{(\text{duf})}$, then $\mathcal{V}_{--}^{(\text{duf})} \leq 0$ for any value of $x \in M$. In both cases, the point $x = r$ is the only point where the potential vanishes to zero and the only point of stationarity of the potential.
- If the Duffing potential is written as $\mathcal{V}_{+-}^{(\text{duf})} = \frac{1}{2}d^2(x, r) - \frac{1}{4}\kappa d^4(x, r)$ or as $\mathcal{V}_{-+}^{(\text{duf})} = -\mathcal{V}_{+-}^{(\text{duf})}$, then the Duffing potential may change sign according to the magnitude of the distance $d^2(x, r)$. In both cases, the point $x = r$ is not

the only point where the potential vanishes to zero nor the only point of stationarity of the potential. In fact, by defining the critical distance $d_b := \sqrt{2/\kappa}$, it is immediate to see that at every point $x \in M$ such that $d(x, r) = d_b$ the potential changes sign.

Examples of Duffing potentials corresponding to different combinations of signs and different values of κ are illustrated in the Figure 4.1. The Riemannian gradient

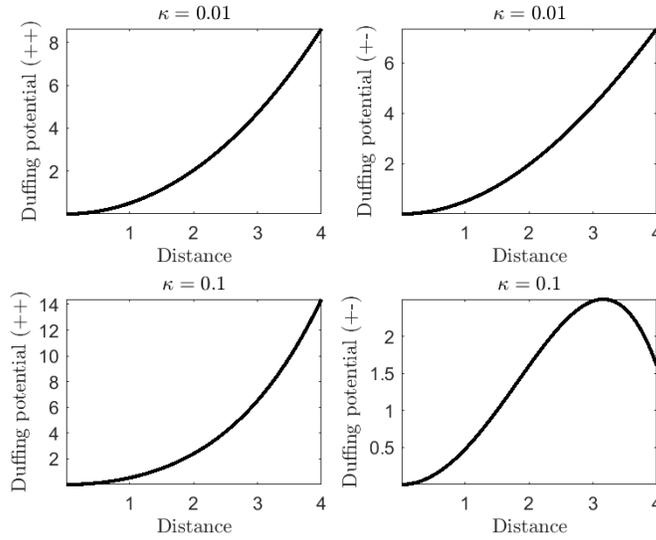


Figure 4.1: *Example of Duffing potential corresponding to different combinations of signs and different values of κ . The $(++)$ combination is referred to as hard Duffing potential while the combination $(+-)$ is referred to as soft Duffing potential. (A combination $(-+)$, not shown in the figure, is referred to as double-well potential.)*

of Duffing potentials corresponding to different combinations of signs reads:

$$\text{grad}_x \mathcal{V}^{(\text{duf})} = [\mp 1 \mp \kappa d^2(x, r)] \log_x(r). \quad (4.2)$$

Apparently, the potential vanishes when $x = r$ but, in case of mixed signs, it also vanishes at every point $x \in M$ such that $d(x, r) = d_b$, therefore this system presents infinitely many critical points. Namely, we may define a set

$$\mathbb{C}^{(\text{duf})} := \left\{ x \in M \mid d(x, r) = \sqrt{2/\kappa} \right\} \quad (4.3)$$

of critical points in M .

Therefore, the dynamical system associated to a Duffing potential reads:

$$\begin{cases} \dot{x} &= v, \\ \nabla_v v &= -\mu \|v\|_x^{2(\varepsilon-1)} v + \begin{cases} [1 + \kappa d^2(x, r)] \log_x(r), & \text{hard Duffing,} \\ [-1 - \kappa d^2(x, r)] \log_x(r), & \text{soft Duffing,} \\ [-1 + \kappa d^2(x, r)] \log_x(r), & \text{double-well Duffing.} \end{cases} \end{cases} \quad (4.4)$$

4.2 Numerical simulation of the hard Duffing oscillator

The time-discretized version of the dynamical system (4.4) associated to the hard Duffing potential reads:

$$\begin{cases} x_{k+1} = x_k \cos(h\|v_k\|) + v_k \sin(h\|v_k\|)/(\|v_k\|), \\ v_{k+1} = \mathbf{P}^{x_k \rightarrow x_{k+1}} [v_k - h\mu (v_k^\top v_k)^{\varepsilon-1} v_k + h(1 + \kappa d^2(x_k, r)) \log_{x_k}(r)]. \end{cases} \quad (4.5)$$

The hard Duffing oscillator on the sphere was simulated numerically in the presence as well as in the absence of a damping term.

The Figure 4.2 illustrates the behavior of a Duffing oscillator on the ordinary sphere \mathbb{S}^2 in the absence of a damping term. As expected, the state of this nonlinear dynamical system oscillates around the reference point r . Moreover, the motion of the system continues endlessly due to the conservation of the total energy. Instead the Figure 4.3 illustrates the behavior of a hard Duffing oscillator on the ordinary sphere \mathbb{S}^2 in the presence of a damping term. In this case, the total energy of the system tends asymptotically to zero due to damping, hence the trajectory of the system spirals into the reference point.

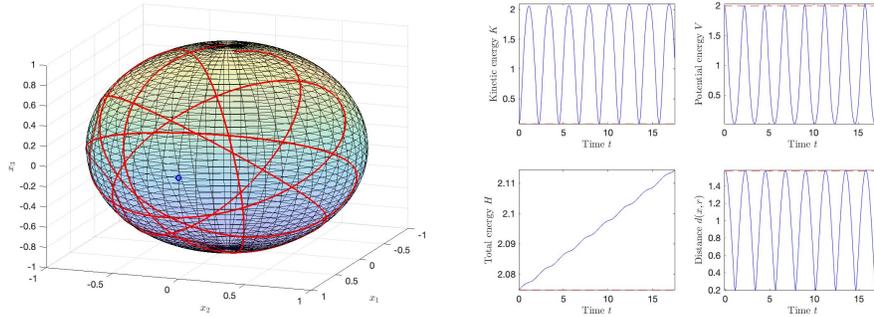


Figure 4.2: Behaviour of the hard Duffing (4.5) in the absence of non-linear damping (namely, $\mu = 0$). The left-hand side panel shows the trajectory in the space \mathbb{S}^2 , when the starting point is $x_0 = [0 \ 0 \ 1]^\top$, the reference point for the oscillator is $r = [1 \ 0 \ 0]^\top$ (denoted by a blue open circle) and the initial speed $v_0 = [0 \ -0.4 \ 0]^\top$. The parameters used in the simulation are $\kappa = 0.5$, $h = 0.0005$.

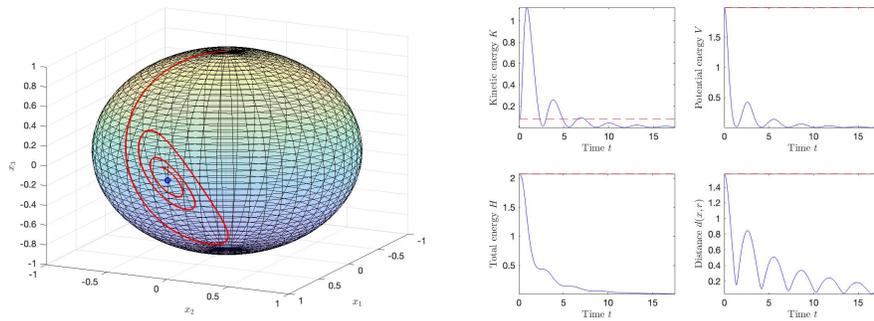


Figure 4.3: Behaviour of the hard Duffing (4.5) in the presence of non-linear damping. The left-hand panel shows the trajectory in the space \mathbb{S}^2 , when the starting point is $x_0 = [0 \ 0 \ 1]^\top$ the reference point for the oscillator is $r = [1 \ 0 \ 0]^\top$ (denoted by a green open circle) and the initial speed $v_0 = [0.5 \ -0.9 \ 0]^\top$. The parameters used in the simulation are $\kappa = 0.5$, $\mu = 0.5$, $\varepsilon = 1.3$ and $h = 0.001$.

4.3 Numerical simulation of the soft Duffing oscillator

In this case, starting from the equation (4.1), the potential corresponding to the soft Duffing oscillator reads:

$$\mathcal{V}^{(\text{duf})} := +\frac{1}{2}d^2(x, r) - \frac{1}{4}\kappa d^4(x, r), \quad (4.6)$$

and according to the calculation rule (4.2):

$$\text{grad}_x \mathcal{V}^{(\text{duf})} = (-1 + \kappa d^2(x, r)) \log_x(r). \quad (4.7)$$

This particular dynamical system has been implemented numerically by three different methods to be able to observe a better oscillation of kinetic and potential energies, a better conservation of total energy and to better evaluate the effects of the critical distance. Furthermore, two other methods called ‘‘Heun-like’’ and ‘‘RK2-like’’ have been implemented, based on the two classical methods used for first-order systems.

4.3.1 Euler method

The time-discretized version of the dynamical system (4.4), implemented through the Euler method and associated to the soft Duffing potential reads:

$$\begin{cases} x_{k+1} = x_k \cos(h\|v_k\|) + v_k \sin(h\|v_k\|)/\|v_k\|, \\ v_{k+1} = \mathbf{P}^{x_k \rightarrow x_{k+1}} [v_k - h\mu(v_k^\top v_k)^{\varepsilon-1} v_k - h(-1 + \kappa d^2(x_k, r)) \log_{x_k}(r)]. \end{cases} \quad (4.8)$$

The Figure 4.4 illustrates the behavior of a soft Duffing oscillator, implemented through the Euler method, on the ordinary sphere \mathbb{S}^2 in the absence of a damping term. As expected, the state of this non-linear dynamical system oscillates around the reference point r . Moreover, the motion of the system continues endlessly due to the conservation of the total energy.

The Figure 4.5 illustrates the behavior of a soft Duffing oscillator, implemented through the Euler method, on the ordinary sphere \mathbb{S}^2 in the presence of a damping term. In this case, the total energy of the system tends asymptotically to zero due to damping, hence the trajectory of the system spirals into the reference point.

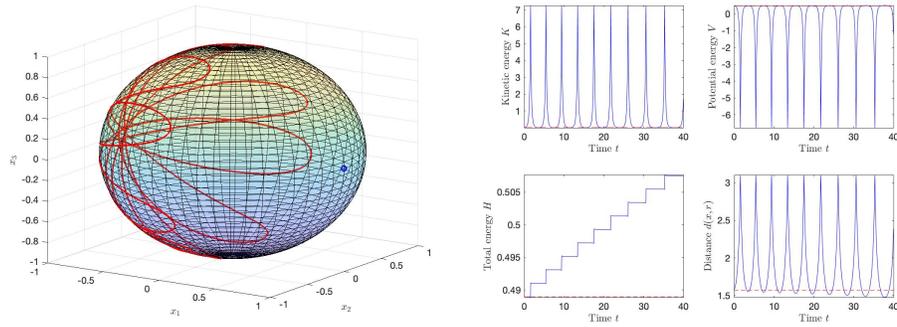


Figure 4.4: Behaviour of the soft Duffing (4.8) in the absence of non-linear damping (namely, $\mu = 0$), implemented through the Euler method. The left-hand side panel shows the trajectory in the space \mathbb{S}^2 , when the starting point is $x_0 = [0 \ 0 \ 1]^\top$, the reference point for the oscillator is $r = [1 \ 0 \ 0]^\top$ (denoted by a blue open circle) and the initial speed is $v_0 = [-1 \ -1.5 \ 0]^\top$. The parameters used in the simulation are $\kappa = 0.5$, $h = 0.0001$.

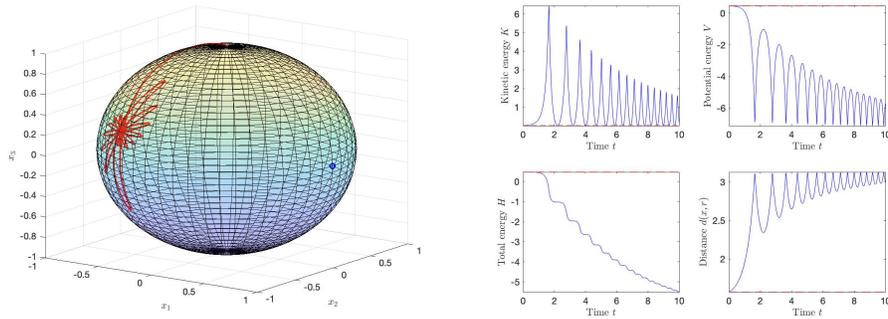


Figure 4.5: Behaviour of the soft Duffing (4.8) in the presence of non-linear damping, implemented through the Euler method. The left-hand panel shows the trajectory in the space \mathbb{S}^2 , when the starting point is $x_0 = [0 \ 0 \ 1]^\top$, the reference point for the oscillator is $r = [1 \ 0 \ 0]^\top$ (denoted by a blue open circle) and the initial speed $v_0 = [-1 \ -1.5 \ 0]^\top$. The parameters used in the simulation are $\kappa = 0.5$, $\mu = 0.2$, $\varepsilon = 1.3$ and $h = 0.0001$.

4.3.2 Heun method

The time-discretized version of the dynamical system (4.4), implemented through the Heun method, associated to the soft Duffing potential reads:

$$\begin{cases} \hat{x}_k := \exp_{x_k}(h v_k), \\ \beta_{1,k}^v := -\mu (v_k^\top v_k)^{\varepsilon-1} v_k - (-1 + \kappa d^2(x_k, r)) \log_{x_k}(r), \\ \hat{v}_k := P^{x_k \rightarrow \hat{x}_k}(v_k + h \beta_{1,k}^v), \\ \beta_{2,k}^v := -\mu (\hat{v}_k^\top \hat{v}_k)^{\varepsilon-1} \hat{v}_k - (-1 + \kappa d^2(\hat{x}_k, r)) \log_{\hat{x}_k}(r), \\ x_{k+1} = \exp_{x_k}\left(h v_k + \frac{h^2}{2} \beta_{1,k}^v\right), \\ v_{k+1} = P^{x_k \rightarrow x_{k+1}}\left[v_k + \frac{h}{2} \left(\beta_{1,k}^v + P^{\hat{x}_k \rightarrow x_k}(\beta_{2,k}^v)\right)\right]. \end{cases} \quad (4.9)$$

The Figure 4.6 illustrates the behavior of a soft Duffing oscillator, implemented through the Heun method, on the ordinary sphere \mathbb{S}^2 in the absence of a damping term. As expected, the state of this non-linear dynamical system keeps oscillating around the reference point r .

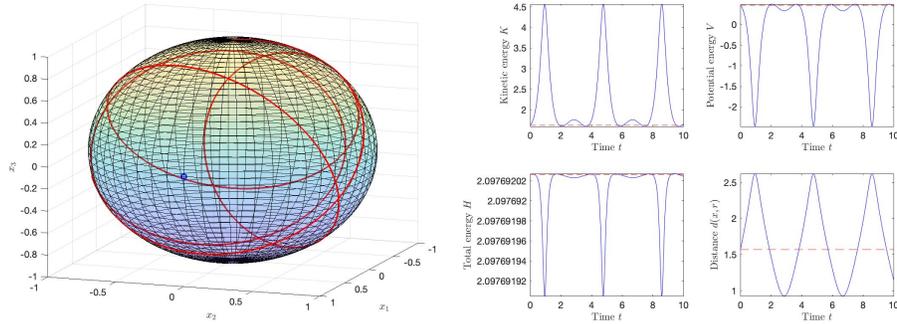


Figure 4.6: *Behaviour of the soft Duffing (4.9) in the absence of non-linear damping (namely, $\mu = 0$), implemented through the Heun method. The left-hand side panel shows the trajectory in the space \mathbb{S}^2 , when the starting point is $x_0 = [0 \ 0 \ 1]^\top$, the reference point for the oscillator is $r = [1 \ 0 \ 0]^\top$ (denoted by a blue open circle) and the initial speed is $v_0 = [-1 \ -1.5 \ 0]^\top$. The parameters used in the simulation are $\kappa = 0.5$, $h = 0.0001$.*

4.3.3 Runge (RK2) method

The time-discretized version of the dynamical system (4.4), implemented through the Runge method, associated to the soft Duffing potential reads:

$$\begin{cases} \hat{x}_k := \exp_{x_k} \left(\frac{h}{2} v_k \right), \\ \beta_{1,k}^v := -\mu (v_k^\top v_k)^{\varepsilon-1} v_k - (-1 + \kappa d^2(x_k, r)) \log_{x_k}(r), \\ \hat{v}_k := \mathbf{P}^{x_k \rightarrow \hat{x}_k} \left(v_k + \frac{h}{2} \beta_{1,k}^v \right), \\ \beta_{2,k}^v := -\mu (\hat{v}_k^\top \hat{v}_k)^{\varepsilon-1} \hat{v}_k - (-1 + \kappa d^2(\hat{x}_k, r)) \log_{\hat{x}_k}(r), \\ x_{k+1} = \exp_{x_k} \left(h v_k + \frac{h^2}{2} \beta_{1,k}^v \right), \\ v_{k+1} = \mathbf{P}^{x_k \rightarrow x_{k+1}} \left(v_k + h \mathbf{P}^{\hat{x}_k \rightarrow x_k}(\beta_{2,k}^v) \right). \end{cases} \quad (4.10)$$

where $k = 0, 1, 2, \dots$. It is worth underlining that, in general, the composition $\mathbf{P}^{x_k \rightarrow x_{k+1}} \circ \mathbf{P}^{\hat{x}_k \rightarrow x_k}$ does not coincide with $\mathbf{P}^{\hat{x}_k \rightarrow x_{k+1}}$, unless the points \hat{x}_k , x_k and x_{k+1} belong to the same geodesic arc. In practice, these points are pretty close to one another, therefore, in the context of a numerical stepping method, it would be feasible to replace the last step with $v_{k+1} = \mathbf{P}^{x_k \rightarrow x_{k+1}}(v_k) + h \mathbf{P}^{\hat{x}_k \rightarrow x_{k+1}}(\beta_{2,k}^v)$. Such approximation would save no computation, though!

The Figure 4.7 illustrates the behavior of a soft Duffing oscillator, implemented through the Runge method, on the ordinary sphere \mathbb{S}^2 in the absence of a damping term. As expected, the state of this non-linear dynamical system oscillates around the reference point r . The total energy is not exactly preserved by the Euler method.

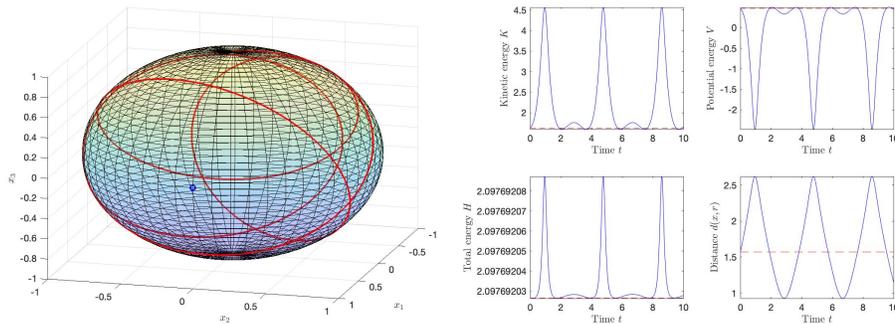


Figure 4.7: Behaviour of the soft Duffing (4.10) in the absence of non-linear damping (namely, $\mu = 0$), implemented through the Runge-Kutta method. The left-hand side panel shows the trajectory in the space \mathbb{S}^2 , when the starting point is $x_0 = [0 \ 0 \ 1]^\top$, the reference point for the oscillator is $r = [1 \ 0 \ 0]^\top$ (denoted by a blue open circle) and the initial speed is $v_0 = [-1 \ -1.5 \ 0]^\top$. The parameters used in the simulation are $\kappa = 0.5$, $h = 0.0001$.

It can be seen that by increasing the order of the method, the error becomes smaller

(in the Heun-based and Runge-based simulations, it is present in the tenth decimal digit). Therefore, it can be concluded that the fact that the total energy does not remain constant in the simulation made with Euler method is due to errors implicit in the method itself, and not to errors of code or formulas.

Moreover the Figures 4.4, 4.5, 4.6 and 4.7 show the relationship between the distance $d(x, r)$ and the potential energy $\mathcal{V}^{(\text{duf})}$, that is also interesting in this case. In fact, one can observe stationary points (local minima and maxima) of the potential energy:

- A local maximum of the distance $d(x, r)$ corresponds to a local minimum of potential energy $\mathcal{V}^{(\text{duf})}$.
- At a local minimum of distance $d(x, r)$ corresponds a local minimum of potential energy $\mathcal{V}^{(\text{duf})}$.

The critical distance value is $d_b \approx 2$. When the distance is smaller than the critical distance the potential energy becomes positive, while it is negative when the distance is larger than the critical distance. It can also be noted that in the soft Duffing case the repulsive action wins over the attractive one and therefore the oscillator tends to stabilize in the sphere at the farthest point from the reference.

4.4 Numerical simulation of the double-well Duffing oscillator

In this case, starting from the equation (4.1), the potential corresponding to the double-well Duffing oscillator reads:

$$\mathcal{V}^{(\text{duf})} := -\frac{1}{2}d^2(x, r) + \frac{1}{4}\kappa d^4(x, r), \quad (4.11)$$

and according to the calculation rule (4.2):

$$\text{grad}_x \mathcal{V}^{(\text{duf})} = (1 - \kappa d^2(x, r)) \log_x(r). \quad (4.12)$$

The time-discretized version of the dynamical system (4.4), implemented through the Euler method and associated to the double-well duffing potential reads:

$$\begin{cases} x_{k+1} = x_k \cos(h\|v_k\|) + v_k \sin(h\|v_k\|)/\|v_k\|, \\ v_{k+1} = \mathbf{P}^{x_k \rightarrow x_{k+1}} [v_k - h\mu(v_k^\top v_k)^{\varepsilon-1} v_k + h(-1 + \kappa d^2(x_k, r)) \log_{x_k}(r)]. \end{cases} \quad (4.13)$$

The Figure 4.8 illustrates the behavior of a double-well Duffing oscillator, implemented through the Euler method, on the ordinary sphere \mathbb{S}^2 in the absence of a damping term. As expected, the state of this non-linear dynamical system oscillates around the reference point r .

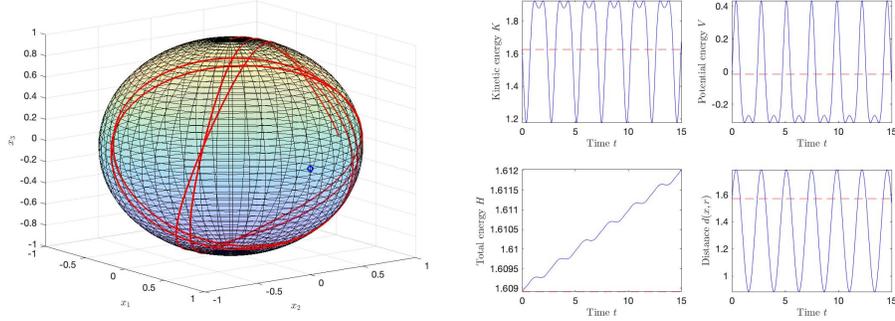


Figure 4.8: *Behaviour of the double-well Duffing (4.13) in the absence of non-linear damping (namely, $\mu = 0$), implemented through the Euler method. The left-hand side panel shows the trajectory in the space \mathbb{S}^2 , when the starting point is $x_0 = [0 \ 0 \ 1]^\top$, the reference point for the oscillator is $r = [1 \ 0 \ 0]^\top$ (denoted by a blue open circle) and the initial speed is $v_0 = [-1 \ -1.5 \ 0]^\top$. The parameters used in the simulation are $\kappa = 0.8$, $h = 0.0001$.*

The Figure 4.9 illustrates the behavior of a double-well Duffing oscillator, implemented through the Euler method, on the ordinary sphere \mathbb{S}^2 in the presence of a damping term. Moreover, the Figures 4.8 and 4.9 show the relationship between the distance $d(x,r)$ and the potential energy $\mathcal{V}^{(\text{duf})}$. In fact, there are stationary points of the potential energy:

- A local maximum of the distance $d(x,r)$ corresponds to a local maximum of potential energy $\mathcal{V}^{(\text{duf})}$.
- At a local minimum of distance $d(x,r)$ corresponds a local maximum of potential energy $\mathcal{V}^{(\text{duf})}$.

In the Figure 4.8, the critical distance is $d_b \approx 1.58$, while in the Figure 4.9, the critical distance is $d_b \approx 1.19$. When the distance is less than the critical distance the potential energy becomes negative, while it is positive when the distance is greater than the critical distance. Therefore we can observe a dual behavior with respect to the case of soft Duffing, due to the fact that they have the same potential and the same gradient, with opposite signs. It can also be noted that in the soft Duffing the repulsive action wins over the attractive one and therefore the oscillator tends to stabilize in the sphere at the point furthest from the reference. While in the double-well case, the attractive action wins, therefore the oscillator tends to stop in the reference point.

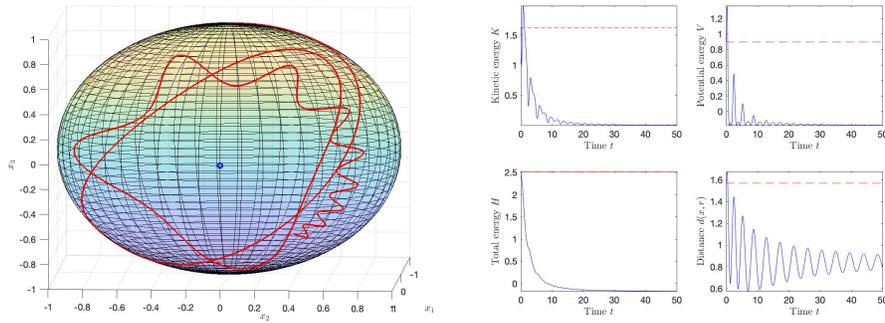


Figure 4.9: Behaviour of the double-well Duffing (4.13) in the presence of nonlinear damping, implemented through the Euler method. The left-hand panel shows the trajectory in the space \mathbb{S}^2 , when the starting point is $x_0 = [0 \ 0 \ 1]^\top$, the reference point for the oscillator is $r = [1 \ 0 \ 0]^\top$ (denoted by a blue open circle) and the initial speed $v_0 = [-1 \ -1.5 \ 0]^\top$. The parameters used in the simulation are $\kappa = 1.4$, $\mu = 0.2$, $\varepsilon = 1.3$ and $h = 0.0001$.

5. Numerical simulations

In this section, synchronization of second-order oscillators by the M-PID control theory will be illustrated through a number of numerical simulations aimed at investigating several interesting aspects, such as speed of synchronization, magnitude of the control efforts, and effectiveness in synchronizing two ‘twin’ systems as well as a leader and a follower that differ structurally to one another. In the present document there are the results of several simulations of the synchronization process of two oscillator with the same dynamics (considering in particular hard Duffing oscillators) or even with different dynamics, namely, two different oscillators (considering in particular the case of a soft Duffing and a hard duffing oscillator). Another studied variation to the problem is the case in which the two oscillators have two different reference points (plotted on the sphere using different colors).

The initial conditions (initial position and initial velocity) of the two systems are taken randomly and so they could be more or less favorable to reach a fast synchronization. The graphics of kinetic, potential and total energy are plotted on the same graphic window with the distance between the states of the two systems on the sphere, the distance of each system from its reference point and a scalar quantification of the control effort over time. A sufficient and necessary condition to the reaching of synchronization between the two systems is that the distance $d(z, x)$ tends to zero, because the two systems have to follow the same trajectory at the end of the simulation (assuming that the duration of the simulation is long enough).

Instead, the fact that the graphic of the distances between each oscillator trajectory and its own reference point follow the same trajectory after a certain value of time is not a necessary and sufficient condition of synchronization, because it depends on whether the two reference points are the same or not. In fact, if they are different, for each system the graphic of distance from the own reference point would oscillate independently, because the follower and the leader achieve synchronization, so the same trajectory, but the distances from the reference points are

different because they are two distinct points. The same considerations apply to potential energies that oscillate independently, being the reference points different.

Synchronization is achieved asymptotically, after an initial transient when the difference between the two trajectories is larger. It is also interesting to notice that the total energy of the follower system will not result to be constant over time and this is easily explicable: the follower is not an isolated system which does not interact with the external environment. In fact it is subject to the M-PID control which is another independent system with an own power supply used to correct the follower's trajectory in order to adapt to the leader's one. So, initially, the follower system will have a surplus of energy, and over time it will reach a constant value, because, once achieved synchronization, the control effort progressively reduces, tending to zero at regime. Instead, the leader is an isolated system ideally and so it is not subject to external forces that change its total energy which keeps constant over time.

Talking about the control effort, at the beginning it has high values, and it tends to zero over time, upon reaching synchronization.

Looking at the kinetic energies of the follower and the leader, it can be noticed that they reach synchronization, that confirms that the synchronization of position implies the synchronization of velocity (because kinetic energy depends only from one variable that is velocity).

Talking about synchronization, it can be considered that the error system (9.2) is linear, so it is possible to extend some considerations from classical control theory on flat spaces to the case of manifolds studied in this document. The control field (3.4) is calculated on the basis of an error term, of its derivative and of its integral with respect to time (without considering the added term u_C to delete the internal dynamics of a system), so it will be composed by a proportional term, a derivative term and an integral term.

How is synchronization reached changing the values of these terms (so changing the value of the corresponding coefficients) ? The results obtained are similar to the known case in a flat space. Increasing the proportional term coefficient, the follower system will be more "ready" and will adapt to the leader system in a short time, which, however, means using more energy, because the control effort would be greater and it could also be useless to increase this term too much because it could cause a long series of overshoots with a consequent slow transient to the desired synchronization. The derivative term has the effect to stabilize the trajectory of the follower controlled by M-PID. It decreases the oscillations around the desired set point, getting a stable dynamics that does not oscillate too much. Finally, the integral term is important to endow a controlled system with some interesting and fundamental properties like an error field with respect to the reference state that tends to zero and to reduce the impact of disturbances on the closed-loop

system. Unfortunately, another effect of this term is to contrast the stabilization property of the derivative term, making the controlled system less stable and the synchronization slower. All these known properties of classical PID controllers can be extended to the case of control of systems whose state belongs to a smooth manifold.

M-PID tuning corresponds to the choice of the best weights to give to proportional, derivative and integral terms in order to get a controlled system that is performing better (and that achieves a faster synchronization, in this particular case). As discussed, the effects of the three terms are multiple and conflicting therefore, to get the best M-PID tuning, it would be necessary to adopt an optimal control strategy [19], which will not be treated in the present document.

In most of the following numerical simulations, the leader's dynamics starting point was set to $z_0 = [0 \ 1 \ 0]^\top$ and its initial velocity was set to $w_0 = [1 \ 0 \ 0.8]^\top$, while the follower's starting point was taken as $x_0 = [0 \ 0 \ 1]^\top$ and its initial velocity was taken as $v_0 = [-1.5 \ -1.5 \ 0]^\top$. The reference points for the leader and the follower oscillators are $r_L = r_F = [0 \ 0 \ -1]^\top$.

5.1 The action of the cancelling component of the control field

The control field (3.4) can be split into two contributions:

- u_C that deletes the internal dynamics of the follower;
- $u - u_C$ that provides the synchronization.

The two contributions can be represented graphically over time by means the total control effort, which is the sum of them, as well as by partial control efforts, as defined in (3.12) and (3.13).

Several simulations carried out lead to the conclusion that the contribution of the term u_C in the M-PID is not indispensable for reaching the synchronization of the oscillators. In fact, u_C is a control term whose purpose is to cancel the internal dynamics of the follower, which anyway would be modified by the control action provided by M-PID control even without the u_C term. Nevertheless, the term u_C gives an improvement in the reaching of synchronization. In particular, we observed an improvement in the "readiness" of the follower system, which will adapt to the leader system faster. In fact, especially in the case (showed in several simulations) where the initial velocity of the follower system is completely opposite to the velocity imposed from the leader system and it is also of large intensity, taking out

the u_C term the M-PID controller would take longer time to modify the trajectory of the follower system. This behavior appears obvious because the M-PID controller imposes a force on the follower system (that possesses its own inertia) to make it follow the leader, but it is readily understood how it is easier to control the a “dead body” without any intrinsic dynamics than a body moving in the space according to its own dynamics. On the other hand, there is the case in which the follower systems initial velocity is in the right direction (according to the leader trajectory), so the u_C term would be useless (because, in this case, the intrinsic dynamics of the follower would take it around the leader and so deleting its dynamics would be a waste of energy), while using only the simple control field provided by the M-PID the follower trajectory would be only modified to synchronize to the leader. Instead with the u_C term, that deletes the internal dynamics of the follower system, the latter will immediately start to follow the leader trajectory independently from its own initial conditions.

These considerations are more evident in the case of different dynamical systems, where the two systems to synchronize have not only different initial conditions, but also different internal dynamics.

A numerical experiment implemented in the present document consists in controlling initially the follower with both the components of the control field and, at a pre-fixed time, in switching off the u_C term. The observed behavior is different if the two oscillators possess the same structure and when they differ from one another. In the following, we present the results of two simulations in which u_C is turned to zero from half of the time-range considered. The two simulations are illustrated in the Figure 5.1.

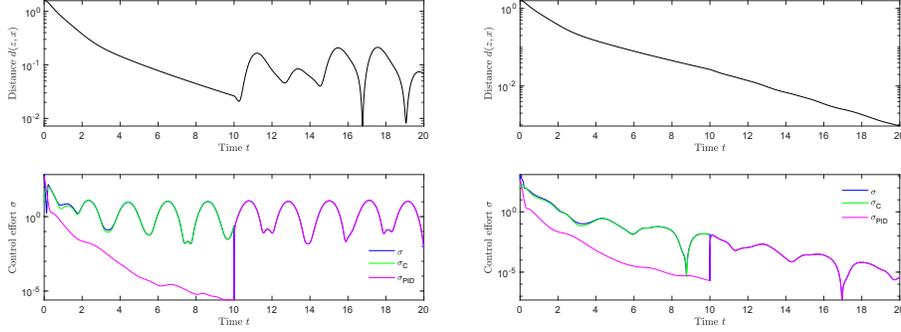


Figure 5.1: *Synchronization of two Duffing-type oscillators by a M-PID controller. The left-hand panel shows the values of the distance $d(z,x)$ on the top and of the control effort σ on the bottom, taken by a hard Duffing oscillator and a soft Duffing oscillator. Instead, the right-hand panel shows the values of the distance $d(z,x)$ on the top and of the control effort σ on the bottom, taken by two hard Duffing oscillators. In both cases, the follower is controlled by an M-PID controller with u_C in the first 10 seconds of the simulation and from $t = 10$ it is set to zero. The proportional control coefficient was set to $\kappa_P = 10$, the integral control coefficient was set to $\kappa_I = 5$ and the derivative control coefficient was set to $\kappa_D = 10$.*

5.2 Evaluation of the action of M-PID controllers on twin oscillators with identical reference points

The following considerations will deal with the case of two oscillators with the same reference point, in order to make synchronization more evident.

Synchronization of two hard Duffing oscillators on the sphere \mathbb{S}^2 has been simulated. If there is only the proportional term, the two oscillators do not achieve synchronization, as shown in the Figure 5.2, because the information about the error's dynamics is not sufficient for calculating an effective control field. This result is coherent with the theoretical findings summarized in the Theorem 3.1.1, according to which, a second-order system needs at least a derivative or an integral term, besides the proportional term, to be effective.

At the same time, the proportional term is essential for synchronization because it is the term that provides the largest contribution in term of energy to the control field in order to reach the leader's trajectory.

It can be noticed from the results shown in the Figure 5.3 that in the simulations with both integral and derivative terms the distance between the states of the two oscillators over time tends to zero more quickly and regularly, without ripples.

Instead, if the control field is composed only by the proportional term and the

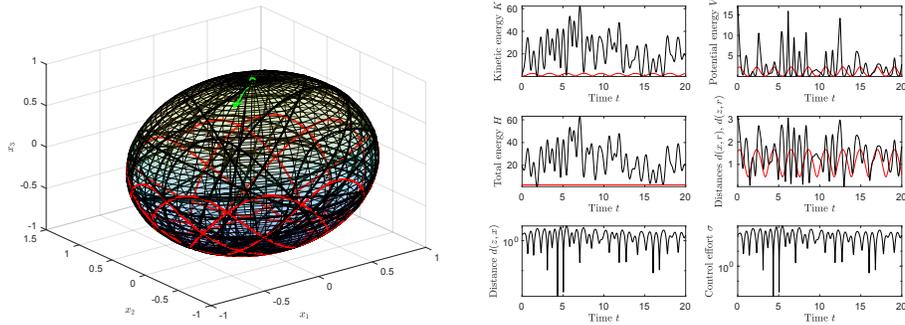


Figure 5.2: Synchronization of two hard Duffing oscillators where the follower is controlled by a P-controller (a M-PID with only the proportional term). In both panels the black lines are about the follower dynamics while the red lines are referred to the leader dynamics. The left-hand panel shows the trajectories on the sphere. The follower's initial velocity is represented by the green arrow. The right-hand panel shows the values taken by the kinetic energy, the potential energy, the total energy and the control efforts and distances over the generated trajectory. The proportional control coefficient was set to $\kappa_P = 0.01$.

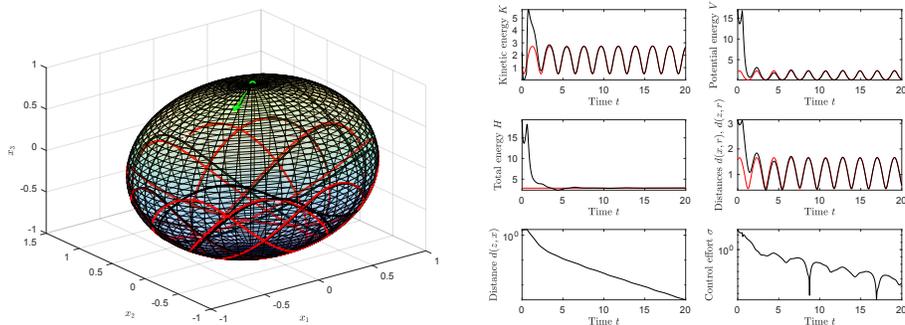


Figure 5.3: Synchronization of two hard Duffing oscillators where the follower is controlled by a full M-PID controller. Values and graphic elements are as in the Figure 5.2. The integral control coefficient is $\kappa_I = 5$, the derivative control coefficient is $\kappa_D = 10$ and the proportional control one is $\kappa_P = 10$.

integral term, synchronization is slower and the distance function tends to zero slowly, as illustrated numerically in the Figure 5.4.

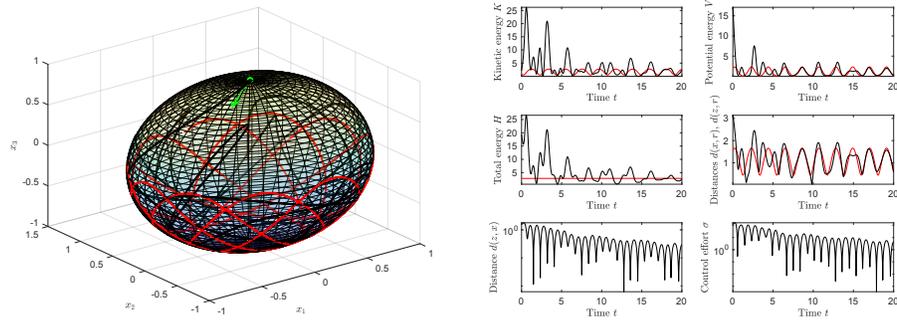


Figure 5.4: Synchronization of two hard Duffing oscillators where the follower is controlled by a PI-controller, with only the proportional and integral terms. Values and graphic elements are as in the Figure 5.2. The integral coefficient was set to $\kappa_I = 2.5$ and the proportional term coefficient was set to $\kappa_P = 10$.

If there are only the proportional and derivative terms, the initial transient is shorter compared to the case with all contributions, because the integral term decreases the system's stability, as it can be readily observed from the results displayed in the Figure 5.5.

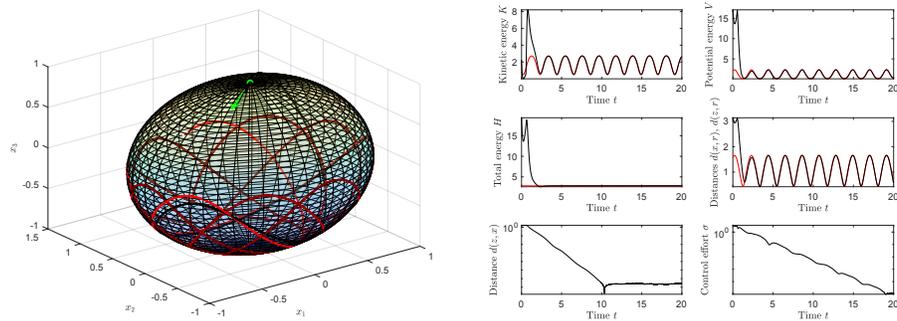


Figure 5.5: Synchronization of two hard Duffing oscillators where the follower is controlled by a PD-controller, with only the proportional and derivative terms. Values and graphic elements are as in the Figure 5.2. The derivative control coefficient is $\kappa_D = 10$ and the proportional control coefficient is $\kappa_P = 10$.

The same consideration can be made in the case of two oscillators with the same dynamics, for example two hard Duffing oscillators, but with two different

reference points.

The theory studied for PID applied to systems on a flat space, from simulations, seem to be valid also for the case of a system on manifold. For example, it is known that the proportional term moves the trajectory of the system controlled near the reference trajectory (in this case the leader's one), but the system will show overshoots for values of this term too large. This is valid (in a limited range of variation of κ_P) also in the case of oscillators on manifold as it can be seen in the simulations illustrated in the Figure 5.6, where it has been changed only the value of κ_P coefficient and the other (integral and derivative coefficients) are kept constant. As it can be noticed, before synchronization is reached there is an initial

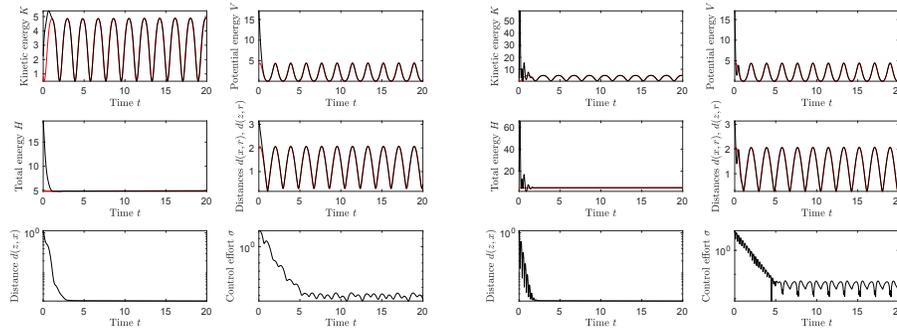


Figure 5.6: Synchronization of two hard Duffing oscillators where the follower is controlled by PID controller: comparison with two different values of proportional term coefficient. In both panels, the black lines denote the follower's dynamic, while the red lines denote the leader's dynamic. The left-hand panel shows the values taken by the kinetic energy, the potential energy, the total energy, control effort and distances over the generated trajectory, when the proportional coefficient $\kappa_P = 5$. The right-hand panel shows the same quantities when the proportional coefficient $\kappa_P = 150$. In both cases, integral coefficient is $\kappa_I = 5$ and derivative coefficient is $\kappa_D = 5$.

transient with overshoots of the follower oscillator near the leader's trajectory.

Instead, the derivative term of the PID control scheme has the property to stabilize the trajectory of the controlled system, decreasing oscillations (in fact, in the continuous-time case, the derivative term corresponds to a zero of the transfer function in the origin of the Laplace space, that stabilizes the system and shortens the transient). It can be verified from the two following simulations that this is valid also in the case studied here (for limited variation range of κ_I). In the results displayed in the Figure 5.7, the coefficients κ_D and κ_P have the same value while κ_D changed. If the derivative coefficient is larger, the follower trajectory is stabilized

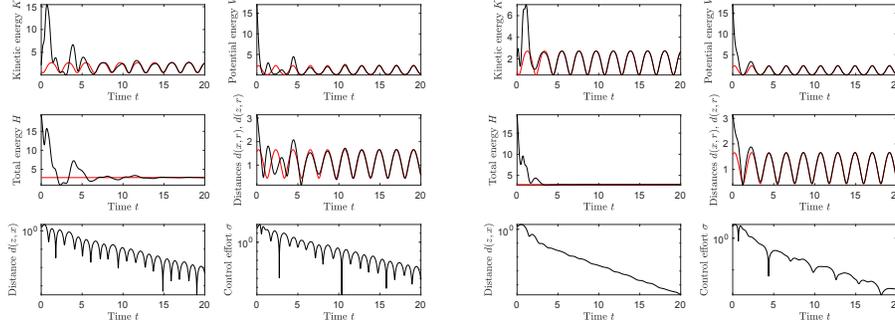


Figure 5.7: Synchronization of two hard Duffing oscillators where the follower is controlled by PID controller: comparison with two different values of derivative term coefficient. Values and graphic elements are as in the Figure 5.6. The left-hand panel shows results when the derivative coefficient is $\kappa_D = 0.5$, while the right-hand panel shows results when the derivative coefficient is $\kappa_D = 3$. In both cases, the integral coefficient is $\kappa_I = 25$ and the proportional coefficient is $\kappa_P = 5$.

on the leader one, while if the derivative coefficient is lower, it is characterized by ample oscillations around leader trajectory before stabilizing.

Finally, the integral term has the property to reduce the permanent regime error and the unwanted effect of disturbances (in fact in the continuous-time case, the integral term corresponds to a pole of the transfer function at the origin in the Laplace space, that introduces an internal model for step inputs (disturbances or reference functions)). From the following two simulations it can be seen that this is valid also for the studied case, in a limited range of variation for κ_I . Similarly to the previous simulations, κ_D and κ_P have been maintained constant, while only κ_I has been changed. The results are displayed in the Figure 5.8. In the case with a lower value of the integrative coefficient the distance between the follower and the leader trajectories tends to zero slower, while if its value is larger the distance function tends to zero more quickly. At the same time, it is known that the integral term decrease the stabilizing effect of derivative term, in fact it can be seen that if the integrative coefficient is larger the follower trajectory has oscillations around leader one, while if it is lower this negative effect is negligible.

These considerations can be extended to the case with a control effort composed not only by proportional, derivative and integrative terms, but also by u_C whose aim is to delete the internal dynamics of the follower system. In fact, as already said, the presence of u_C can make synchronization quicker if the own dynamics of the follower tends to take the system far from the leader, but the PID contribution is the same: the contribution u_C changes the duration of the transient.

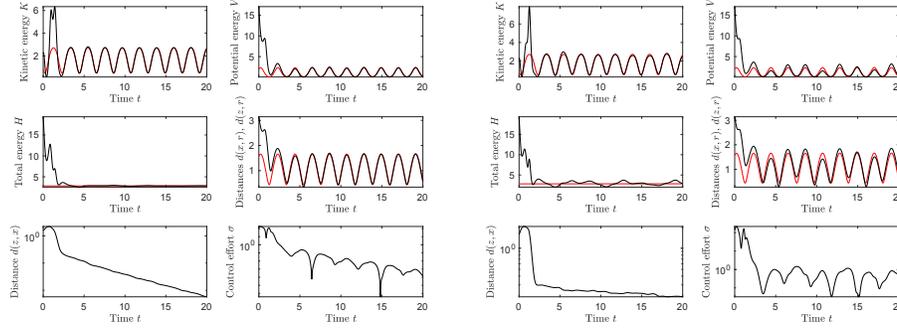


Figure 5.8: Synchronization of two hard Duffing oscillators where the follower is controlled by PID controller: comparison with two different values of integral term coefficient. Values and graphic elements are as in the Figure 5.6. The left-hand panel shows results when the integral coefficient is $\kappa_I = 3$, while the right-hand panel shows results when the value of the integral coefficient is $\kappa_I = 25$. In both cases, the derivative coefficient is $\kappa_D = 5$ and the proportional coefficient is $\kappa_P = 5$.

The results are also the same if the two oscillators considered are different in their reference points, but also in their dynamics (in particular, a hard Duffing and a soft Duffing oscillator have been considered).

The Figures 5.9 and 5.10 show a comparison between the case with u_C and without u_C , respectively, with the follower initial velocity (represented in the figure on the sphere as a tangent vector) with a direction unfavorable for the convergence to the leader trajectory (because it is opposed to it).

It can be noticed from the previous simulations that the presence of the u_C term makes synchronization quicker because the internal dynamics of the follower oscillator would take it far from the leader trajectory. In fact, in the case with u_C the initial transient is shorter than without u_C .

The Figures 5.11 and 5.12 illustrate a comparison between the case with u_C and without u_C , respectively, with the follower initial velocity (represented in the figure on the sphere as a tangent vector) with a direction favorable for the convergence to the leader trajectory (because it is directed towards it). It can be seen that the initial transient is shorter in the case without u_C because, in this case, it is not useful to delete the follower internal dynamics, in fact, this component of the control field tends to take the follower oscillator near the leader trajectory. It would be a useless waste of energy of the PID controller. Therefore, the choice of a M-PID controller with the u_C term or without it is a problem that has not an univocal solution, but it depends on the case considered.

The simulations shown in the Figure 5.13 illustrate the case of two hard Duff-

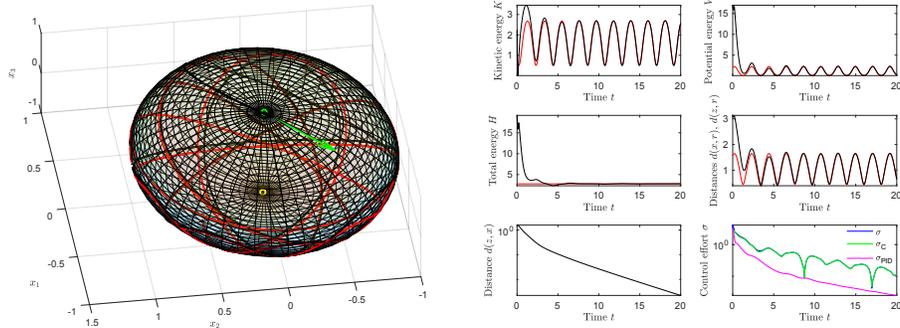


Figure 5.9: Synchronization of two hard Duffing oscillators where the follower is controlled by a M-PID controller with u_C term and the initial velocity direction of the follower is not favorable for synchronization (i.e., it is opposed to the leader trajectory). In both panels the black lines indicate the follower's dynamics, while the red lines indicate the leader's dynamics. The left-hand panel shows the trajectories of the two oscillators on the sphere over time. The follower's initial velocity is represented by a green arrow and his starting point with a green point, instead the leader's starting point is represented by a white point. The right-hand panel shows the values taken by the kinetic energy, the potential energy, the total energy and control effort and distances over the generated trajectory. In particular, in the panel at the right-bottom corner, the total control effort σ is represented in blue, the PID contribution σ_{PID} in magenta and the contribution deriving from u_C in green color. The proportional control coefficient is $\kappa_P = 10$, the integral control coefficient is $\kappa_I = 5$ and the derivative control coefficient is $\kappa_D = 10$.

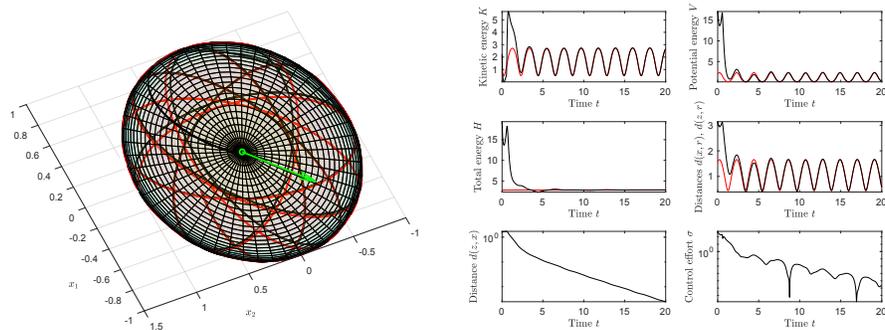


Figure 5.10: Synchronization of two hard Duffing oscillators where the follower is controlled by a M-PID controller without the u_C term and the initial velocity direction of the follower is not favorable for synchronization. Values and graphic elements are as in the Figure 5.9.

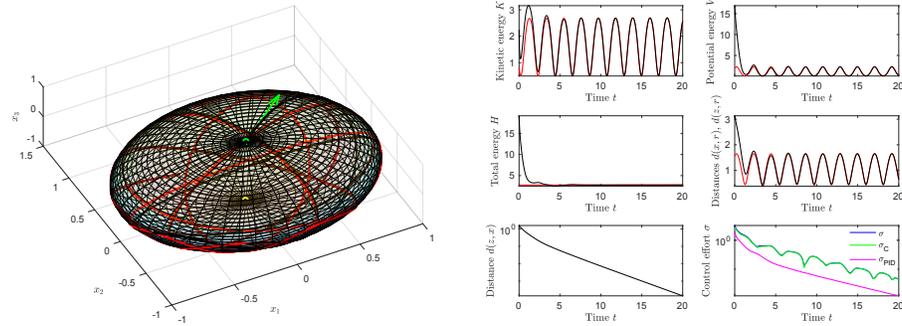


Figure 5.11: Synchronization of two hard Duffing oscillators where the follower is controlled by a M-PID controller with the u_C term when the initial velocity direction of the follower is favorable for synchronization. In both panels the black lines denote the follower's dynamics, while the red lines denote for the leader's dynamics. The left-hand panel shows the trajectories of the two oscillators on the sphere over time. The follower's initial velocity is represented by a green arrow and his starting point with a green point, instead the leader's starting point is represented by a white point. The right-hand panel shows the values taken by the kinetic energy, the potential energy, the total energy and control effort and distances over the generated trajectory. In particular, the right-bottom panel shows the total control effort σ in blue, the PID contribution σ_{PID} in magenta and the contribution deriving from u_C in green color. The proportional control coefficient is $\kappa_P = 10$, the integral control coefficient is $\kappa_I = 5$ and the derivative control coefficient is $\kappa_D = 10$.

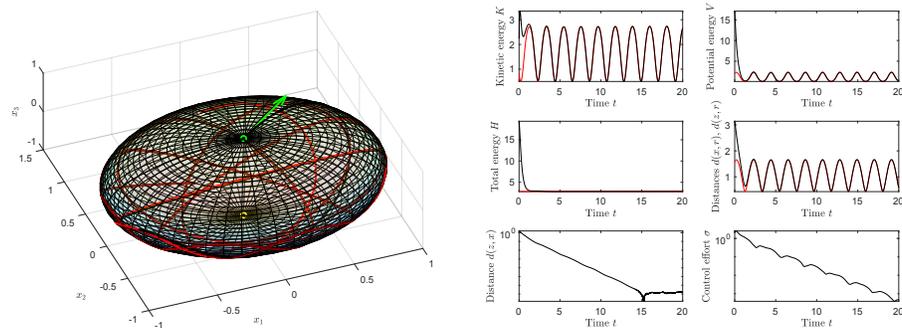


Figure 5.12: Synchronization of two hard Duffing oscillators where the follower is controlled by a M-PID controller without the u_C term when the initial velocity direction of the follower is favorable for synchronization. Values and graphic elements are as in the Figure 5.11.

ing oscillators with two different reference points. Although synchronization is

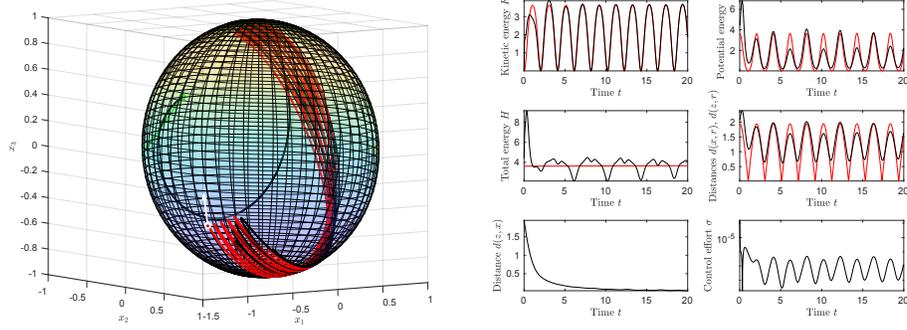


Figure 5.13: Synchronization of two hard Duffing oscillators where the follower is controlled by M-PID controller. Moreover in both panels the black lines denote the follower’s dynamics, while the red lines denote the leader’s dynamics. The left-hand panel shows the trajectories of the two oscillators on the sphere over time. The follower’s initial velocity is represented by green arrow, while the leader’s initial velocity is represented by a white arrow. The two different reference points have been represented as two colored points on the sphere (green for the leader reference point and white for the follower’s one). The right-hand panel shows the values taken by the kinetic energy, the potential energy, the total energy and control effort and distances over the generated trajectory. In this case, the leader’s and follower’s starting points, initial velocities and reference points are taken randomly. Moreover the proportional control coefficient is $\kappa_P = 10$, the coefficient of the integral control term is $\kappa_I = 5$ and the derivative coefficient is $\kappa_D = 10$.

reached, the potential functions oscillates in phase over time but with different trajectories: this is due to the fact that the follower and the leader, after synchronization, follow the same trajectory, but, being the reference points two different points on the sphere, the distances (as it can be noticed from the curve of the distances $d(x, r)$ and $d(z, r)$) will be different like the potential functions. Another interesting observation is that the total energy of the follower will not tend to a constant value but continues to change over time: this is due to the control field which has to act continuously over time, because of the follower reference point’s force that tends to move away the follower from the leader trajectory. Instead, in the case with the same reference points the total energy tends to a constant value, because, after an initial transient, the control effort is very low.

Finally, in the simulations shown in the Figure 5.14 and 5.15, it is illustrated a case-study with two oscillators possessing different dynamics, in particular, the synchronization of a hard Duffing and a soft Duffing oscillator is taken into con-

sideration. The Figure 5.14 shows the synchronization behavior of a M-PID controller endowed with the u_C term, while the results illustrated in Figure 5.15 were obtained without this term, but there are only the proportional, derivative and integrative terms. It can be noticed that the u_C term makes synchronization quicker, in fact the distance function tends quickly to zero.

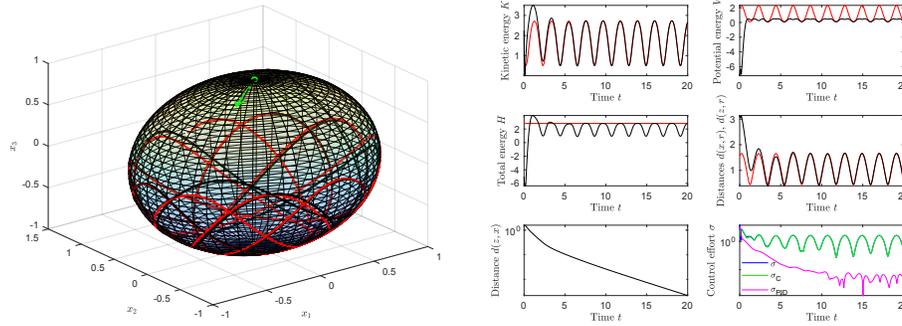


Figure 5.14: *Synchronization of an hard Duffing oscillator and a soft Duffing oscillator, where the follower is controlled by a M-PID controller with the presence of the u_C term. In both panels the black lines stand for the follower’s dynamic, while the red ones stand for the leader’s dynamic. The left-hand panel shows the trajectories of the two oscillators on the sphere over time. The follower’s initial velocity is represented by the green vector in the figure. The right-hand panel shows the values taken by the kinetic energy, the potential energy, the total energy and control effort and distances over the generated trajectory. In particular in the right-bottom panel, the total control effort σ is represented in blue, the PID contribution σ_{PID} in magenta and the contribution deriving from u_C in green color. The proportional control coefficient is $\kappa_P = 10$, the integral control coefficient is $\kappa_I = 5$ and the derivative control coefficient is $\kappa_D = 10$.*

As already mentioned, by changing the values of the PID parameters (proportional, derivative and integrative coefficients) it can be noticed that all the properties studied for the linear systems’ control are valid also for the oscillators that are non-linear systems (in particular they are systems of the second order because in their dynamics there are two differential equations, one for velocity and one for the position).

These simulations, concerning control by M-PID to achieve the synchronization of two oscillators with the same or different dynamics, is a sort of “toy problem” before the study of synchronization of two real-world rigid bodies in the space, like two gyrostat satellites and quadcopter drones, with their inertia.

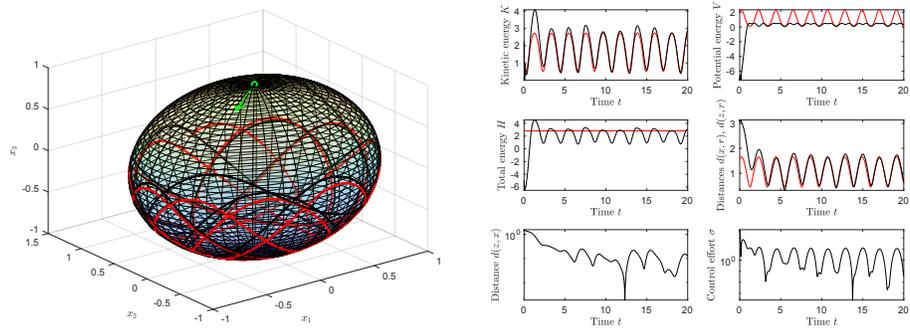


Figure 5.15: Synchronization of an hard Duffing oscillator and a soft Duffing oscillator, where the follower is controlled by a M-PID controller without the u_C term. Values and graphic elements are as in the Figure 5.14.

6. Real systems application

In an ideal control system the value of a system's state is perfectly known in each instant. Unfortunately, in the real cases there are lots of external and internal factors that make this simplification not possible, as shown in the Figure 3.1.

In fact, all the systems studied have to be embedded into a real environment that implies friction forces and damping that obstruct the motion of the system, the introduction of external disturbances, like in a flying drone the variable wind thrust or also internal disturbances, like the noise caused by the electronic part of a system, and the presence of time delays, due, for example, to the distance between a receiver and a transmitter, like two drones that have to reach the same trajectory or rotational attitude, but they are at a certain distance (so the exchanges of information between the two drones cannot be instantaneous but characterized by a variable time delay). All these factors must be considered remembering that whatever system studied is not isolated but interacting with a dynamical environment.

Here follows a list of possible elements of non-idealism.

- *Time delay*: In this paper two systems are considered, with the same or different dynamics. Firstly abstract systems are studied, oscillators of different nature modelling real phenomena (weather turbulence, spring pendulum dynamics, ...), then models of real devices as quadcopter drones or gyrostatt satellites will be considered. One of the two systems, the follower, has to synchronize to the other, the leader. So the follower has to take the information about leader state (velocity and position) through particular communication channels, in this way it can be considered a receiver and the leader the transmitter of its own state signals. The communication in the real cases is never perfect because of disturbances, attenuation of the signals and time delay in particular, that can be considered proportional to the distance between the two bodies. Therefore, the follower will know the state of the leader, to reach by way of synchronization, at a previous instant and the error calculated for the digital PID implementation will be the difference of two states

evaluated at different time-stamps.

- *Damping*: The ideal oscillators have the same total energy over time, for example the classic pendulum has a total energy that is the sum of potential and kinetic energies, and in the ideal case it remains constant over time. Instead, in the real case the oscillator will not have the same potential energy over time, because friction forces oppose to its motion causing a loss of total energy that will bring the oscillator's total energy to zero. So, the maximal values of kinetic and potential energies decrease over time until collapsing to zero, as the total energy.
- *Disturbances/noise*: Obviously, the follower system will not know the exact state of the leader system in the real case, in fact, the follower will receive the information about the leader through communications channels subject to disturbances and, in particular, additive noise. Noise is a signal with an infinite frequency amplitude, therefore it exhibits rapid variations which correspond to a multitude of discontinuity points. The information about the leader affected by noise (when received by the follower) are position and velocity. So according to the follower systems it will see the leader system velocity with sudden variations over time, as if it would be carried by a discontinuous force. In this way, the control system acting upon the follower will elaborate \hat{z} and \hat{w} instead of z and w (the real information about the leader system's state). The problem is to verify if synchronization will be achieved even in the presence of noise. From the simulations it results that synchronization is achieved, because the M-PID controller acts as a low-pass filter, filtering out the largest frequencies and isolating the low frequencies, including, in particular, the average value corresponding to null frequency. As it is well-known, the input value of the M-PID, as in the linear controllers, is the error, that is the difference between the follower and the leader states (affected by noise) in this case. So, the noise present in the leader state will also affect the error field. Thanks the PID's low-pass filtering property, the controller will consider only the average value of the error affected by noise, which corresponds to the effective error (because noise is a casual disturbance which preserves the average value of the original noiseless signal). Consequently, synchronization is possible even in a realistic case with the presence of noise and without a perfect communication between the follower and the leader. Since the output of the M-PID is calculated from the error affected by noise, which is a highly variable signal, also the control field will have the same characteristic. A conclusion is that the presence of noise implies a great consumption of energy by the M-PID controller. So, in

order to reduce this problem, a solution could be to use a filter to limit the effect of that noise.

In the real-world cases, the studied oscillators are a simplification of real systems such as quadcopter drones and gyrostat satellites, so another way could be to provide the real follower and leader systems with better ways of communication shielded from any external disturbance, so that the signal received by the follower system about the trajectory of the leader could be as close as possible to the original one.

6.1 Simulations on moderately damped oscillators

In this subsection are presented some simulations about synchronization in the presence of noise, time delay and damping, in order to consider situations as similar as possible to the real cases. This is a preliminary step necessary to introduce more complex systems moving and reaching synchronization in the real case, like quadcopter drones and gyrostat satellites. Initially, considering both follower and leader affected by damping, it is important to notice that damping cannot be too high (so the constant μ cannot be too large) in order to make synchronization evident. In fact, if the damping would be too high the leader, because of its own high damping, would stop quickly and the follower would follow it, without showing any evident synchronization of the trajectories, but only two different trajectories reaching the same stationary point.

In the following experiments, synchronization of a soft Duffing oscillator (follower) to a hard Duffing system (leader) with the same damping coefficient μ (so they are considered affected by the same friction forces) has been simulated. In particular, the Figure 6.1 is concerned with the case $\mu = 0.1$, while the Figure 6.2 is concerned with the case $\mu = 0.9$.

It can be noticed from these simulations that if the damping coefficient takes too a high value, in particular in the second example where $\mu = 0.9$, synchronization is not evident because the distances tend to zero but the trajectories do not exhibit the same shape. Instead, if the damping coefficient is not too high, in particular in the first example where $\mu = 0.1$, synchronization is more evident and the two systems oscillates for more time reaching the same trajectory, before collapsing to the leader's reference point. In signal masking, for example, it is very important to make sure that damping is not high, because the received signal contains the information encrypted to send to the receiver, so it has to be preserved in its trajectory as much as possible, or the information received could be not correct. So this kind of encrypted communication processes has to use communication channels with a low damping to preserve the information carried by the signal.

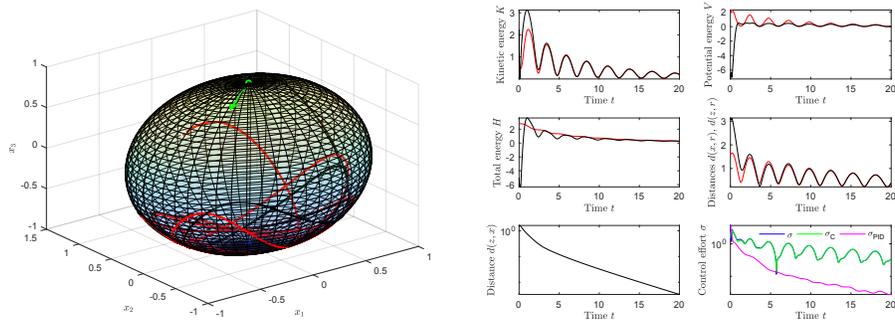


Figure 6.1: Synchronization of a hard Duffing oscillator and a soft Duffing oscillator, where the follower is controlled by a M-PID controller and the leader is damped. In both panels, the black lines correspond to the follower's dynamics, while the red lines correspond to the leader's dynamics. The left-hand panel shows the trajectories of the two oscillators on the sphere over time. The follower's initial velocity is represented by the green arrow. The right-hand panel shows the values taken by the kinetic energy, the potential energy, the total energy and control effort and distances over the generated trajectory. The proportional control coefficient was set to $\kappa_P = 10$, the integral control coefficient was set to $\kappa_I = 5$ and the derivative coefficient was set to $\kappa_D = 10$. In this case, the damping coefficient is $\mu = 0.1$.

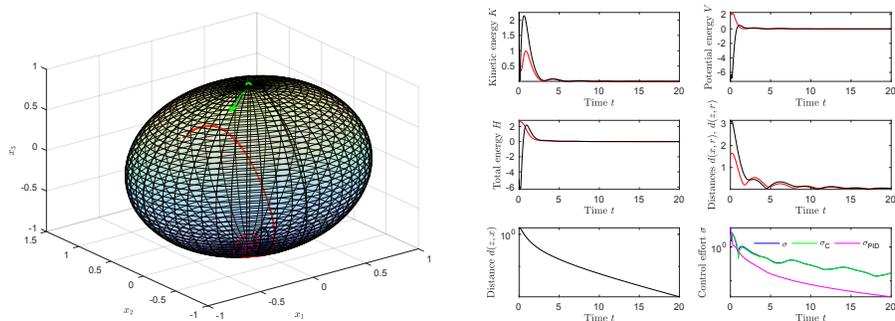


Figure 6.2: Synchronization of a hard Duffing oscillator and a soft Duffing oscillator, where the follower is controlled by a M-PID controller and the leader is damped. Values and graphic elements are as in the Figure 6.1. In this case the damping coefficient is $\mu = 0.9$.

6.2 Simulations on oscillators affected by disturbances

Noise is another potentially disturbing factor when dealing with the time-synchronization of two systems. Noise is a disturbance signal due to several internal or external causes which adds up to the state variables (in this case, position and velocity of the leader oscillator). Because of noise, the control system will elaborate imprecise information about the leader as it cannot rely on a perfect knowledge of the leader's state. Noise is a detrimental factor for reaching synchronization of the two systems. It is interesting to investigate if synchronization is possible or not for such oscillators. In the following simulations it will be illustrated the case of the synchronization of a soft Duffing oscillator (follower) to a hard Duffing one (leader), where the follower does not receive a perfect information about leader's state (velocity and position) but signals corrupted by noise. Since both oscillators evolve on a curved manifold, the noise signal has not to be considered as summed to a function as in a flat phase-space, but affecting a trajectory on a sphere's tangent bundle, namely, on the sphere for position and on each tangent space for velocity.

This means that the leader's trajectory received by the follower will not be $z(t)$ but it would be $\hat{z}(t)$, a distortion of the first one, and it has been implemented using the exponential map by applying it to each point $z(t)$ in a random tangent direction to $z(t)$ and with a random amplitude of the distortion. With reference to the synchronization scheme (3.15) about notation, a random tangent direction to z_k and with a random amplitude has been implemented for each discrete-time step k as follows:

$$\mathbf{v}_{z,k} := b_z(I_3 - z_k z_k^\top) \alpha_k, \quad (6.1)$$

where α_k is a random 3×1 array (each entry is drawn from a normal distribution). Moreover b_z is a constant controlling the amplitude of the noise affecting the variable z . The result is, at each instant, a random tangent vector to z_k with an amplitude given by a constant b_z . A noisy version of the state z_k is then computed as

$$\hat{z}_k = \exp_{z_k}(h \mathbf{v}_{z,k}), \quad (6.2)$$

where 'exp' stands again for the exponential map on \mathbb{S}^2 and h denotes the same step-size considered in the synchronization scheme (3.15). As known from manifold calculus, the result of the exponentiation is a point on the sphere reached moving away from the initial point z_k with initial velocity $\mathbf{v}_{z,k}$.

The other information received by the follower about leader's state is its velocity which will not be $w(t)$ but it would be $\hat{w}(t)$, a distortion of the actual velocity of the leader oscillator, and it has been calculated using parallel transport from z_k to \hat{z}_k (the point of the sphere where \hat{w}_k is tangent) of the actual velocity vector w_k , corrupting it with a random tangent vector to \hat{z}_k with a random amplitude. This

tangent vector with a random tangent direction to \hat{z}_k and with a random amplitude has been implemented for each k as follows:

$$\mathbf{v}_{w,k} = b_w(I_3 - \hat{z}_k \hat{z}_k^\top) \beta_k, \quad (6.3)$$

where β_k denotes a random 3×1 array. The result is at each instant a random tangent vector to \hat{z}_k with an amplitude denoted as b_w . A noisy version of the velocity w_k is then computed as

$$\hat{w}_k := \mathbf{P}^{\hat{z}_k \rightarrow \hat{z}_k}(w_k) + \mathbf{v}_{w,k}, \quad (6.4)$$

where ‘P’ stands again for parallel transport on \mathbb{S}^2 .

By changing the values of the constants b_z and b_w , it is possible to modify the noise signal that corrupts the leader’s trajectory seen by the controller/follower complex. As it has been already said, because of the rapid changes the slave will follow the average trajectory of the leader.

The Figures 6.3 and 6.4 show results of numerical simulations pertaining to two different values for the constant b_z while keeping the constant b_w fixed. It

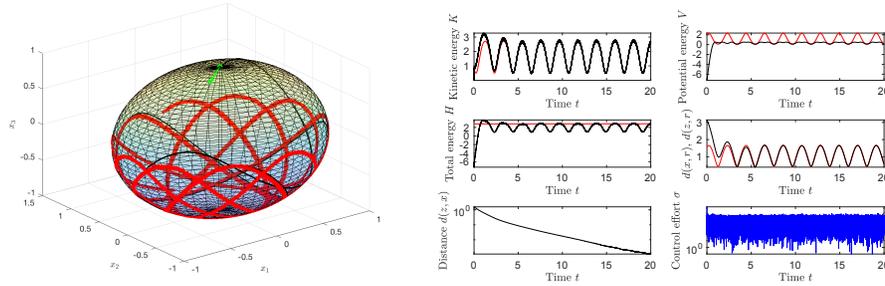


Figure 6.3: *Synchronization of an hard Duffing oscillator and a soft Duffing oscillator, where the follower is controlled by a M-PID controller, in the presence of additive noise. In both panels the black lines denote the follower’s dynamics, while the red lines denote the leader’s dynamics. The left-hand panel shows the trajectories of the two oscillators on the sphere over time. The follower’s initial velocity is represented by the green arrow. The right-hand panel shows the values taken by the kinetic energy, the potential energy, the total energy and control effort and distances over the generated trajectory. The proportional control coefficient was set to $\kappa_P = 10$, the integral control coefficient was set to $\kappa_I = 5$ and the derivative coefficient was set to $\kappa_D = 10$. In this case, the coefficients b_z was set to 50 and the coefficient b_w was set 0.01.*

can be seen that the red trajectory on the sphere in these cases do not represent the real leader trajectory, but the leader’s position received by the follower over time,

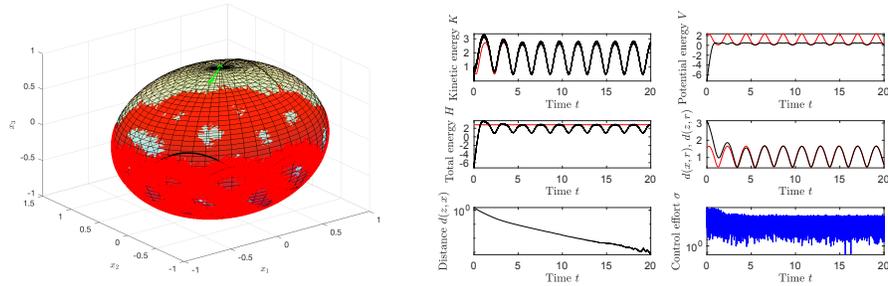


Figure 6.4: *Synchronization of an hard Duffing oscillator and a soft Duffing oscillator, where the follower is controlled by a M-PID controller. Values and graphic elements are as in the Figure 6.1 except that, in this case, the coefficients b_z was set to 300.*

affected by noise, namely, \hat{z} . In the second case, the red trajectory on the sphere is more distorted because of the larger value of the constant b_z chosen. It can be seen that in both cases the follower synchronizes with the leader system. However, in the second case, with a larger noise on the position sensing, synchronization is slower as it can be noticed from the distance figure which tends to zero more slowly. In addition, the control effort is more irregular in its trajectory over time because of the larger noise on position reading.

The Figure 6.5 shows the results obtained by raising the value of the constant b_w , so that the noise on leader's velocity has been increased in its amplitude. It can

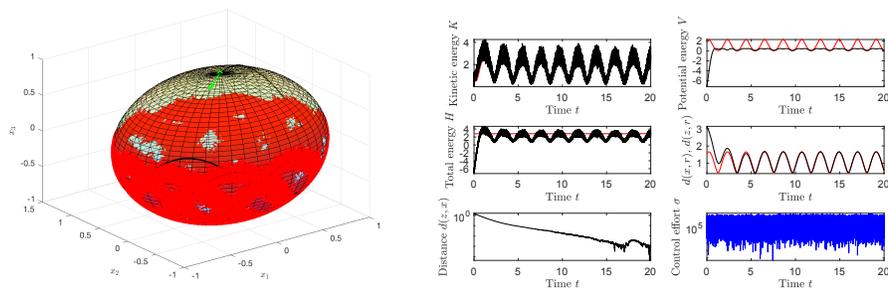


Figure 6.5: *Synchronization of an hard Duffing oscillator and a soft Duffing oscillator, where the follower is controlled by a M-PID controller. Values and graphic elements are as in the Figure 6.1 except that, in this case, the coefficients b_z was set to 300 and the coefficient b_w was set to 0.1.*

be seen that synchronization is reached but the distance function tends to zero more

slowly and with some irregularity. Also, the control effort looks more irregular over time because of the larger noise on velocity.

It is also interesting to consider a sinusoidal disturbance signal with a particular frequency and amplitude. It could represent a periodical disturbance that corrupts the information received by the follower about leader's state. This particular kind of disturbance is a signal that can be summed both to the leader's position and velocity, finding, like in the previous case, the leader's position and velocity as received by the follower oscillator. The sinusoidal disturbance can be implemented modifying the previous formulas:

$$v_{z,k} := b_z(I_3 - z_k z_k^\top) [1 \quad 1 \quad 1]^\top \cos(\omega t), \quad (6.5)$$

where ω denotes an angular frequency (corresponding to a proper frequency $f = \omega/2\pi$). The noisy position-state observation is then computed as in (6.2). Similarly, a velocity disturbance may be defined as

$$v_{w,k} := b_w(I_3 - \hat{z}_k \hat{z}_k^\top) [1 \quad 1 \quad 1]^\top \cos(\omega t), \quad (6.6)$$

and the noisy velocity-state observation is then computed as in (6.4).

It is interesting to change the frequency of the sinusoidal noises and to study the behavior of the follower and its synchronization. As in the previous cases synchronization is reached, thanks to the M-PID control, but there are some considerations to note. In fact, if the frequency of the sinusoidal signal is high, the follower system controlled by a M-PID will behave as a low-pass filter (as in the previous case with a noise that has an high frequency amplitude of its Fourier transform and will not react to fast oscillations, achieving synchronization: the problem of synchronization in this case is successfully solved. Instead, if the frequency of the sinusoidal signal is not high enough, the follower system will not behave as a low-pass filter and it will follow a wrong trajectory, affected by the sinusoidal disturbance: in this situation, synchronization is achieved although to a wrong trajectory. This can be a problem if the amplitude of the sinusoidal disturbance is large, because the trajectory of the follower would be too different compared to the trajectory of the leader. The results of numerical simulations shown in the Figure 6.6 and 6.7 illustrate two cases in which the frequency of the sinusoidal disturbance takes two very different values. In the second simulation, the frequency of the sinusoidal disturbance has been chosen to be 200 times smaller than in the first case. It can be noticed that the distance $d(x, z)$ in the first case tends to zero without oscillation, because the follower reaches synchronization with $z(t)$, the real trajectory of the leader. Instead, the distance $d(x, z)$ in the second case, after an initial transient, starts to oscillate, because the follower reaches synchronization thanks to the action of the M-PID controller, although to $\hat{z}(t)$.

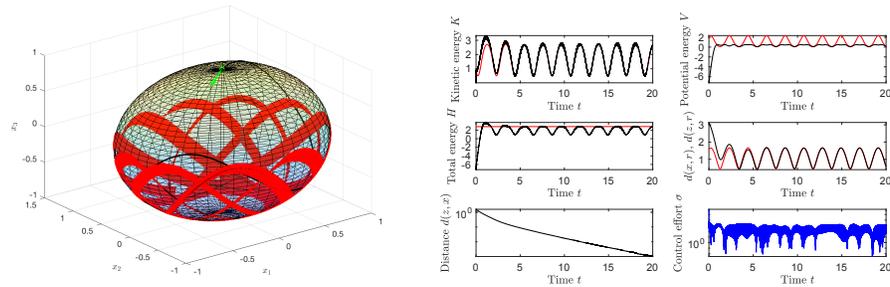


Figure 6.6: Synchronization of an hard Duffing oscillator and a soft Duffing oscillator, where the follower is controlled by a M-PID controller. In both panels the black lines denote the follower's dynamics, while the red lines denote the leader's dynamics. The left-hand panel shows the trajectories of the two oscillators on the sphere over time. The follower's initial velocity is represented by a green arrow. The right-hand panel shows the values taken by the kinetic energy, the potential energy, the total energy and control effort and distances over the generated trajectory. The proportional control coefficient was set to $\kappa_P = 10$, the integral control coefficient was set to $\kappa_I = 5$ and the derivative coefficient was set to $\kappa_D = 10$. In this simulation, the noise coefficients were set to $b_z = 300$ and $b_w = 0.01$ and the sinusoidal disturbance's angular frequency was set to $\omega = 1,000$ rad/s.

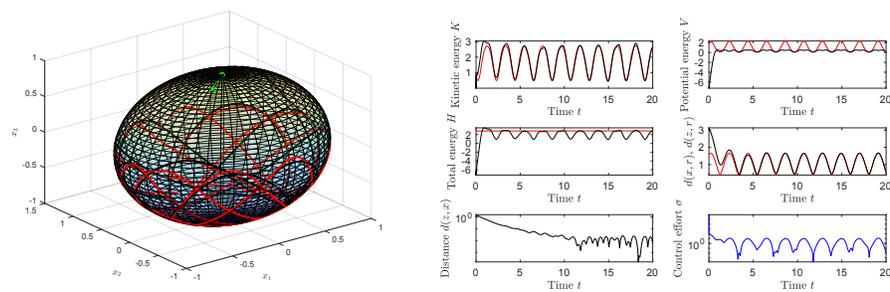


Figure 6.7: Synchronization of an hard Duffing oscillator and a soft Duffing oscillator, where the follower is controlled by a M-PID controller. Values and graphic elements are as in the Figure 6.6 except that the sinusoidal disturbance's angular frequency was set to $\omega = 5$ rad/s.

6.3 Simulations on oscillators in the presence of delays

Another problem that can negatively affect the synchronization of the follower oscillator to the leader trajectory is time delay, deriving from many causes, like, for example, the physical distance between the two systems to synchronize. In an ideal situation, the follower system receives the information about leader's position and velocity instantly, without any time delay. In this ideal case, the error between the position states was defined as $e(t) := \log_{z(t)} x(t)$, and the M-PID will act to reduce this error between follower and leader trajectories to zero through its control action in order to achieve the synchronization of the two systems.

Instead, in the real this does not happen, in fact, there is the presence of a time delay that we could consider constant, denoted by l : the follower state at the instant t will be described by the values $x(t)$ and $v(t)$ and the follower will receive the information about leader's state evaluated at $t - l$, namely $z(t - l)$ and $w(t - l)$. So also the error will be defined in a different way, because the information received by the follower is different from the previous cases without time delay:

$$e(t) = \log_{z(t-l)} x(t), \quad (6.7)$$

It can be noticed that if $l = 0$ the error formula is the same of the previous case without time delay. The same considerations are valid for the calculation of the other error terms to evaluate the M-PID control field.

The M-PID will act to reduce the error between follower and leader trajectories to zero through its control field, but this error is calculated using two values of position at different instants $x(t)$ and $z(t - l)$. Therefore, the follower will reach the same trajectory of the leader but it will be delayed with respect to the leader's trajectory, as shown in the numerical simulation displayed in the Figure 6.8. It can be seen from the potential and kinetic energies and from the distance $d(x, r)$ that there is a time delay in synchronization, in fact, the shapes of the trajectories are the same but they are translated over time (as a matter of fact, the trajectory of the follower is delayed over time compared to the trajectory of the leader). For larger values of l the translation over time of follower's trajectory with respect to leader's one is more evident. Therefore, time delay has to be as low as possible because it does not permit a real synchronization of the two trajectories over time. Looking at the figure showing the trajectories of the two oscillators on the sphere it can be seen that, after a certain transient, the follower's trajectory is superimposed to the past leader's one, but the last one is gone on. It can be noticed also that in the ideal case, without time delay between the two systems, the distance between $z(t)$ and $x(t)$, so between leader and follower tends to zero over time asymptotically. Instead in the real case, with time delay, after a certain transient, the distance between $z(t)$ and $x(t)$, so between leader and follower, start to oscillate between a maximum and a

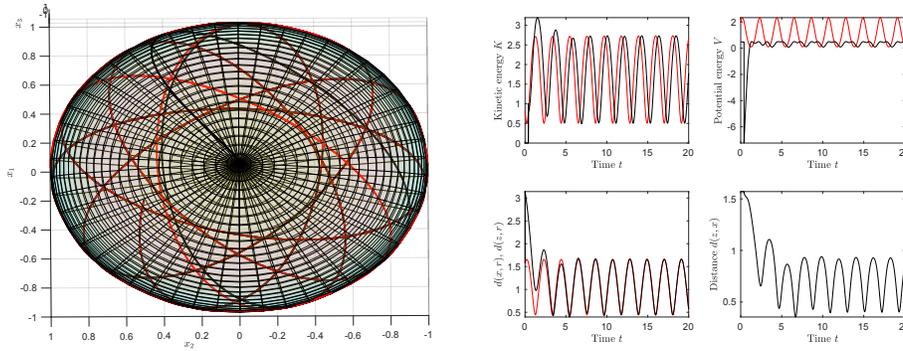


Figure 6.8: *Synchronization of an hard Duffing oscillator and a soft Duffing oscillator, where the follower is controlled by a M-PID controller. In both panels the black lines denote the follower’s dynamics, while the red lines denote the leader’s dynamics. The left-hand panel shows the trajectories of the two oscillators on the sphere over time. The right-hand panel shows the values taken by the kinetic energy and the potential energy. The proportional control coefficient was set to $\kappa_P = 10$, the integral control coefficient was set to $\kappa_I = 5$ and the derivative coefficient was set to $\kappa_D = 10$. In this case, the time delay is $l = 0.4$ seconds and $h = 0.0002$ is the value of the stepsize.*

minimum over time. It means that the follower is assuming the same trajectory of the leader but this trajectory is delayed respect to leader’s one.

Time delay and noise can be considered altogether because they are two common problems for synchronization. The simulation illustrated in the Figure 6.9 shows a case-study where they are both present in the communication between follower and leader. It can be seen that the behavior of synchronization is a mix of the characteristics of the previous cases.

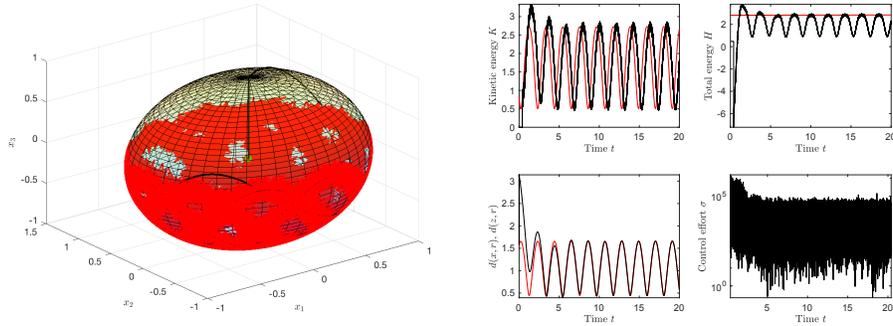


Figure 6.9: *Synchronization of an hard Duffing oscillator and a soft Duffing oscillator, where the follower is controlled by a M-PID controller in the presence of both time-delay and additive noise. Values and graphic elements are as in the Figure 6.8 and $b_z = 300$ and $b_w = 0.01$. As it can be noticed the total energy of the leader oscillator is constant over time, because it is supposed not subjected to external forces. In fact leader's total energy is preserved over time, while follower's total energy is not constant being subjected to the control effort of the PID system.*

6.4 Application of synchronization to signal masking

In the real applications this disturbance (noise) can be considered as the information to transmit from a transmitter to a receiver. In this case, synchronization of oscillators can be used to communicate securely by encrypting the carried information, so that only the receiver and the transmitter know the real signal transmitted. In fact, the receiver will get an oscillating signal (the carrier) corrupted by noise (information) and, with another or also the same kind of oscillator, will be able to synchronize it to the carrier (because noise, so the information carried, is a signal with a great amplitude of frequency changing rapidly over time, so the follower will not consider it because of the PID controller's inertia and will be able to isolate the part of the received signal generated by the leader). Therefore, to get the information dispatched by the transmitter, the receiver will have to calculate the difference between the signal obtained after synchronization and the signal received (leader signal plus information content). In the considered simulations the term standing for noise has been implemented using a random function to represent the unpredictability of noise. The amplitude of noise is controlled by a constant (for example b_z) multiplied to a function. In the real applications, such constant has to take a value neither too small nor too large in order to make it possible the extraction of information from the receiver input. This kind of problem deals with the so-called signal masking [8].

7. Oscillators on the orthogonal group $\text{SO}(3)$

For the special orthogonal group $\text{SO}(3)$ endowed with the canonical metric $\langle W, V \rangle_x := \text{tr}(W^T V)$, it holds that:

$$\begin{cases} d^2(X, Y) = -\text{tr}[\text{Log}^2(X^T Y)], \\ \exp_X(V) = X \text{Exp}(X^T V), \\ \log_X(Y) = X \text{Log}(X^T Y), \end{cases} \quad (7.1)$$

where $X, Y \in \text{SO}(3), V \in T_X \text{SO}(3) := \{X\Omega \mid \Omega \in \mathfrak{so}(3)\}$, with $\mathfrak{so}(3) := \{\Omega \in \mathbb{R}^{3 \times 3} \mid \Omega^T + \Omega = 0\}$. Moreover, the symbols Exp and Log denote the matrix exponential and the principal matrix logarithm, respectively. The matrix logarithm was implemented using a special numerical recipe based on a Schur decomposition, explained in [20]. The kinetic energy for the dynamical systems in this section takes the expression $\mathcal{K} = -\frac{1}{2} \text{tr}[\Omega^2]$, where $\Omega \in \mathfrak{so}(3)$ denotes the system's angular speed matrix. According to the expressions of the geometric quantities of interest recalled above, the discrete-time dynamical system reads:

$$\begin{cases} X_{k+1} = X_k \text{Exp}(h\Omega_k), \\ \Omega_{k+1} = (1 - h\mu \|\Omega_k\|^{2(\varepsilon-1)})\Omega_k - hX_k^T \text{grad}_{X_k} \mathcal{V}, \end{cases} \quad (7.2)$$

with $h > 0$ being a discretization stepsize for the dynamical system and $\varepsilon > 0$ being the exponential for nonlinear damping. The system state is represented by the pair $(X_k, \Omega_k) \in \text{SO}(3) \times \mathfrak{so}(3)$ for $k \in \mathbb{N}$. The first equation of the numerical method (7.2) represents a geodesic-based Euler-like step-forward numerical approximation of the flow associated with the first differential equation on the tangent bundle $T\text{SO}(3)$. The second equation represents a direct Euler-like step-forward method that takes place on a linear space, namely, the Lie algebra $\mathfrak{so}(3)$.

7.1 Visual rendering via a group action on a manifold

In order to achieve visual rendering of the $\text{SO}(3)$ -trajectories generated by the non-linear oscillators, we shall be exploiting the notion of group action on a manifold. Given a group \mathbb{G} and a smooth manifold M , a *group action* $\Lambda : \mathbb{G} \times M \rightarrow M$ is denoted as $y = \Lambda_g(x)$, where $g \in \mathbb{G}$ and $x, y \in M$ [6]. It is worth recalling that an *orbit* of a point $x \in M$ under a group action Λ is defined as $\mathbb{G}^\Lambda \cdot x := \{\Lambda_g(x) \mid g \in \mathbb{G}\}$ [6]. In the case of interest in the present report, a suitable group action is invoked in order to provide a 3-dimensional graphical rendering of trajectories generated in the Lie group of 3-dimensional rotations, which are represented by 3×3 matrices. Therefore, we take $\mathbb{G} = \text{SO}(3)$ and $M = \mathbb{S}^2$ and define $\Lambda_X(p) := Xp$, with $X \in \text{SO}(3)$ and $p \in \mathbb{S}^2$. Clearly, a trajectory will be represented as a subset of $\text{SO}(3)^\Lambda \cdot p \cong \mathbb{S}^2$.

In other words, in the numerical simulations, the trajectories generated by the oscillators have been represented in the hyper-sphere \mathbb{S}^2 . In particular, taken an arbitrary ‘seed’ on the hyper-sphere, it is multiplied by the initial rotation matrix, belonging to $\text{SO}(3)$. Also the reference point is obtained multiplying the starting point by the reference rotation matrix. Therefore, in this way, the starting vector assumes an initial inclination and it will oscillate around a reference inclination. Only in the non-damped case (when $\mu = 0$) the reference inclination matrix is taken randomly, because in this way, better trajectories are obtained during the simulations. As the inclination matrix evolves over time, it will change its value according to the laws of the considered oscillator. So, multiplying this matrix by the initial vector belonging to \mathbb{S}^2 , we will obtain, instant by instant, a vector that will belong to the sphere with a new inclination.

7.2 Numerical simulation of a hard Duffing oscillator

In this instance of a dynamical system on the Lie group $\text{SO}(3)$, the potential energy function takes the expression $\mathcal{V}^{(\text{duf})} := \mp \frac{1}{2} \text{tr} [\text{Log}^2(X^\top R)] \pm \frac{1}{4} \kappa \text{tr} [\text{Log}^2(X^\top R)]^2$, where $X \in \text{SO}(3)$ denotes the system’s orientation state matrix and $R \in \text{SO}(3)$ denotes a reference point for the system’s oscillations. Since the potential is a quartic form in the Riemannian distance (but not in the system’s state), the potential might be referred to as *quartic*.

The time-discretized version of the dynamical system (4.4) on the manifold of special orthogonal matrices, associated to the hard Duffing oscillators, reads:

$$\begin{cases} X_{k+1} = X_k \text{Exp}(h\Omega_k), \\ \Omega_{k+1} = (1 + h\mu \text{tr}[\Omega_k^2]^{\varepsilon-1})\Omega_k + h(1 + \kappa d^2(X_k, R)) \text{Log}(X_k^\top R). \end{cases} \quad (7.3)$$

The Figure 7.1 illustrates the behavior of a hard Duffing oscillator on the special orthogonal group $SO(3)$ embedded in \mathbb{R}^3 in the absence of a damping term via a Lie-group action-based rendering technique.

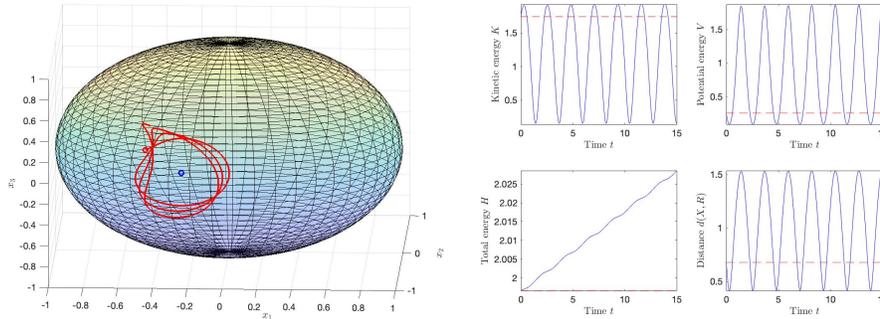


Figure 7.1: Behaviour of a hard Duffing oscillator (7.3) on the special orthogonal group $SO(3)$, in the absence of non-linear damping (namely, $\mu = 0$). The left-hand side panel shows the trajectory on the special orthogonal group $SO(3)$, when the starting point and also the reference point are taken randomly, (denoted by a blue open circle) and the initial speed will be random, because related to the initial state. The parameters used in the simulation are $\kappa = 0.5$ and $h = 0.0005$.

The Figure 7.2 illustrates the behavior of a hard Duffing oscillator on the special orthogonal group $SO(3)$ embedded in \mathbb{R}^3 in the presence of a damping term.

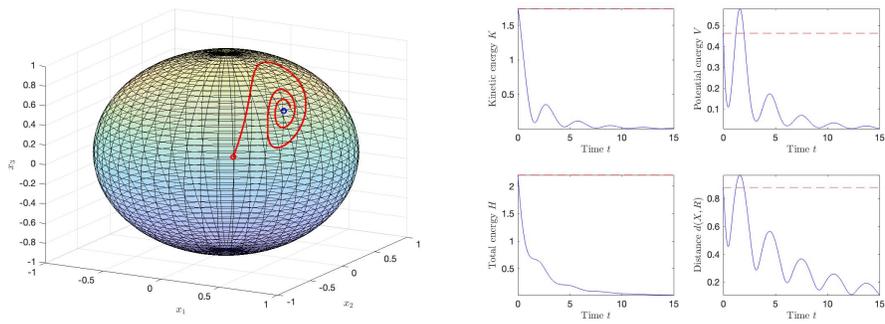


Figure 7.2: Behaviour of a hard Duffing oscillator (7.3) in the presence of non-linear damping. The left-hand panel shows the trajectory in the special orthogonal group $SO(3)$, when the starting point and also the reference point are taken randomly (denoted by a blue open circle) the initial speed will be random, because related to the initial state. The parameters used in the simulation are $\kappa = 0.5$, $\mu = 0.5$, $\varepsilon = 1.3$ and $h = 0.0008$.

8. Mathematical model of a quadcopter drone

The present section summarizes a mathematical models of a *quadrotor drone* within the framework of Lie-group system theory.

8.1 Mathematical model of quadrotor drone on $\text{SO}(3) \times \mathfrak{so}(3) \times \mathbb{R}^3$

In this part of the thesis the document [9] can be recalled. A quadrotor is made of a body \mathcal{B} and of four rotors \mathcal{R}_a , with $a = 1, 2, 3, 4$, also termed *propellers*. We assume that one pair of propellers ($a = 2$ and $a = 4$) is rotating clockwise, while the other pair of propellers ($a = 1$ and $a = 3$) is rotating counterclockwise, as shown in the Figure 8.1. The rotor \mathcal{R}_1 is located along the $+x$ axis (hence the rotor \mathcal{R}_3 is located along the $-x$ axis), while the rotor \mathcal{R}_4 is located along the $+y$ axis (hence the rotor \mathcal{R}_2 is located along the $-y$ axis). The coordinates of each volume element of the drone are referred to an *earth reference frame* denoted by \mathcal{F}_E . A reference frame, denoted as \mathcal{F}_B , is attached to the body of the quadrotor, with origin in its center of mass, as shown in the Figure 8.1. By adjusting the rotors speeds, a control system may drive a quadcopter into a number of maneuvers, such as:

- **Vertical motion:** A drone can be driven into three maneuvers in the vertical plane: hovering, climbing, or descending. To hover, the net thrust of the four rotors pushing the drone up must be equal to the gravitational force pulling it down. Increasing the thrust of the four rotors so that there is a non-zero upward force that is greater than the weight causes the drone to climb. Decrease the thrust causes the drone to descend vertically.
- **Turning (yawing):** Assume a drone is hovering and one wants to rotate it along its vertical axis (which is referred to as *yawing*). According to the

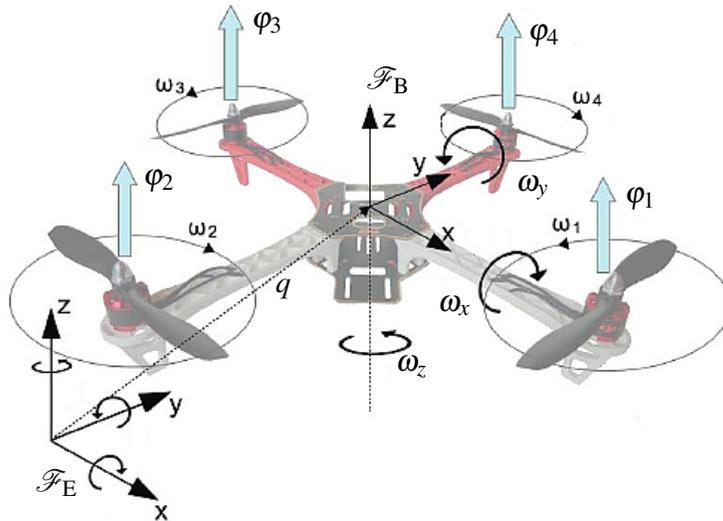


Figure 8.1: *Quadcopter reference frames.*

Figure 8.1, two rotors are spinning counterclockwise and two are spinning clockwise. With the two sets of rotors rotating in opposite directions, the total angular momentum is zero. Since there is no torque on the drone, then the total angular momentum must remain constant (zero in this case). Decreasing the angular velocity of the rotors spinning clockwise and increasing the angular velocity of the rotors spinning counterclockwise, the drone rotates clockwise so that the body of the drone has an angular momentum that compensates for the total angular momentum of the rotors. Namely, the angular momentum of the rotors does not add up to zero, so the drone body must rotate. However, the total force remains equal to the gravitational force and the drone continues to hover. Since the lower thrust rotors are diagonally opposite from each other, the drone can still stay balanced and does not tip.

- **Moving forwards and sideways (pitching and rolling):** In most cases, there is no difference between moving forward or backward (that is referred to as *pitching*), because the drone is symmetrical. The same holds true for side-to-side motion (which is referred to as *rolling* or *banking*). In order to fly forward, it is necessary to generate a forward component of thrust via the rotors. With reference to the Figure 8.1, increase the rotation rate of rotors 3 and 4 (the rear ones) and decrease the rate of rotors 1 and 2. The total thrust force will remain equal to the weight, so the drone will stay at the same vertical level. Also, since one of the rear rotors is spinning counterclockwise and

the other clockwise, the increased rotation of those rotors will still produce zero angular momentum. The same holds true for the front rotors, and so the drone does not rotate. However, the greater force in the back of the drone means it will tilt forward. Now a slight increase in thrust for all rotors will produce a net thrust force that has a component to balance the weight along with a forward motion component.

The model includes a number of physical constants, some of which appear in the following matrices:

$$\begin{cases} D := \text{diag} \left(\sqrt{\frac{J_y J_z}{J_x}}, \sqrt{\frac{J_x J_z}{J_y}}, \sqrt{\frac{J_x J_y}{J_z}} \right), \\ \hat{J}_q = \frac{1}{2} \begin{bmatrix} J_y - J_x + J_z & 0 & 0 \\ 0 & J_x - J_y + J_z & 0 \\ 0 & 0 & J_x + J_y - J_z \end{bmatrix}. \end{cases} \quad (8.1)$$

The model also includes four input variables, namely the spinning velocities of the four rotors $\omega_a(t) > 0$, $a = 1, 2, 3, 4$, two state variables, $\Omega(t) \in \mathfrak{so}(3)$ and $v(t) \in \mathbb{R}^3$, and two output variables, $R(t) \in \text{SO}(3)$ and $q(t) \in \mathbb{R}^3$.

The equations of a quadrotor drone may be cast in a Lie-group setting as:

$$\begin{cases} T := br(\omega_4^2 - \omega_2^2)\Omega_x + br(\omega_3^2 - \omega_1^2)\Omega_y + \gamma(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2)\Omega_z, \\ B := J_{\mathcal{R}}(-\omega_1 + \omega_2 - \omega_3 + \omega_4)\Omega_z, \\ \dot{\Omega} = D^{-1}([\hat{J}_q, \Omega^2] + [B, \Omega] - \dot{B} + T)D^{-1}, \\ \dot{R} = R\Omega, \\ \dot{v} = \frac{1}{2} \frac{b}{M_q} (\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) R e_z - p e_z - \frac{1}{M_q} \Gamma v, \\ \dot{q} = v, \end{cases} \quad (8.2)$$

for $t \in [0, 3]$ (seconds). It is customary to make sure that, during motion, a zero residual rotor angular velocity, namely $-\omega_1 + \omega_2 - \omega_3 + \omega_4 = 0$, holds. In this case, $B = 0$ and $\dot{B} = 0$. The parameter $J_{\mathcal{R}}$ then disappears from the equations. The first four equations concern the orientation/attitude of the drone, while the last two equations (where $e_z := [0 \ 0 \ 1]^T$ denotes a vertical axis) concern the translation.

This differential system contains three equations on vector spaces (the equation for Ω has support $\mathfrak{so}(3)$ and the equations for q and v have support \mathbb{R}^3) and an equation on a Lie group. The numerical values of the parameters are taken from

[2], namely:

$$\begin{cases} M_q = 0.650 \text{ kg}, \\ J_x = J_y = 7.5 \times 10^{-3} \text{ kg} \cdot \text{m}^2, \\ J_z = 1.3 \times 10^{-2} \text{ kg} \cdot \text{m}^2, \\ b = 3.13 \times 10^{-5}, \text{ N} \cdot \text{s}^2 \\ r = 0.23 \text{ m}, \\ \gamma = 7.5 \times 10^{-7} \text{ N} \cdot \text{m} \cdot \text{s}^2, \\ p = 9.81 \text{ m} \cdot \text{s}^{-2}, \\ \Gamma = 0.25 I_3 \text{ kg} \cdot \text{s}^{-2}. \end{cases} \quad (8.3)$$

Even though the mathematical model deals with angular velocities ω_a as input variables, the input values are chosen to be the RPM n_a of the rotors, because this is the more natural way of setting the propeller rotation velocity. The relation between angular velocity and rotation per minute is $\omega_a = \frac{\pi}{30} n_a$. We recall four test experiments suggested in [3, Table 1] adapted to the present model. The rotors speed was adapted to the present case by means of a hovering design. Namely, we considered the case that the thrust produced by four rotors spinning at equal speed ω_h (the hovering speed) balances perfectly the gravitational force, formally:

$$\frac{1}{2} \frac{b}{M_q} (4\omega_h^2) = p \Rightarrow \omega_h^2 := \frac{pM_q}{2b}. \quad (8.4)$$

The resulting steady-state rotor speed in RPM (rounded to the nearest integer) reads

$$n_h := \left\lceil \frac{30}{\pi} \sqrt{\frac{pM_q}{2b}} \right\rceil = 3048, \quad (8.5)$$

that corresponds to a hovering velocity of about 319 rad/s.

Four typical maneuvers, and the corresponding rotors RPM (adapted from [3]), are outlined in Table 8.1. Notice that, in all four maneuvers, the residual angular velocity is null, namely $\Omega_r = 0$.

Table 8.1: Summary of four numerical experiments design, adapted from [3].

Maneuver	Rotors RPM			
	n_1	n_2	n_3	n_4
Hover	3048	3048	3048	3048
Yaw & Roll (-y)	3048	3047	3048	3049
Yaw & Pitch (+x)	3047	3048	3049	3048
Pitch (+x) & Roll (+y)	3047	3049	3049	3047

8.2 Numerical simulation methods

A discrete-time system to simulate numerically the dynamics of a gyrostat satellite, implemented by Euler-like stepping methods, is as follows:

$$\left\{ \begin{array}{l} C_k := [\kappa_1(\omega_r - \omega_{x,k}) + \kappa_2(\omega_r^3 - \omega_{x,k}^3)]\Omega_x + [\kappa_3(\omega_r - \omega_{y,k}) + \kappa_4(\omega_r^3 - \omega_{y,k}^3)]\Omega_y + \\ \quad [\kappa_5(\omega_r - \omega_{z,k}) + \kappa_6(\omega_r^3 - \omega_{z,k}^3)]\Omega_z, \\ B_k := J_{11}\omega_1\Omega_x + J_{22}\omega_2\Omega_y + J_{33}\omega_3(1 + b\cos(vhk))\Omega_z, \\ \dot{B}_k := -bJ_{33}\omega_3\sin(vhk)\Omega_z, \\ \Omega_{k+1} = \Omega_k + hD^{-1}([\hat{J}_g, \Omega_k^2] + [B_k, \Omega_k] - \dot{B}_k - \{P, \Omega_k\} + C_k)D^{-1}, \\ R_{k+1} = R_k \text{Exp}(h\Omega_k), \end{array} \right. \quad (8.6)$$

where $\Omega_k = \omega_{x,k}\Omega_x + \omega_{y,k}\Omega_y + \omega_{z,k}\Omega_z$, $k = 0, 1, 2, \dots$ and the initial conditions are $\Omega_0 = 0$ and $R_0 = I_3$. All the physical constants have the values specified in (??).

A discrete-time system to implement the model of the quadrotor drone, implemented by a Euler-like stepping method, is:

$$\left\{ \begin{array}{l} T_k := br(\omega_{4,k}^2 - \omega_{2,k}^2)\Omega_x + br(\omega_{3,k}^2 - \omega_{1,k}^2)\Omega_y + \gamma(-\omega_{1,k}^2 + \omega_{2,k}^2 - \omega_{3,k}^2 + \omega_{4,k}^2)\Omega_z, \\ \Omega_{r,k} := -\omega_{1,k} + \omega_{2,k} - \omega_{3,k} + \omega_{4,k}, \\ B_k := J_{\mathcal{R}}\Omega_{r,k}\Omega_z, \\ \Omega_{k+1} = \Omega_k + hD^{-1}([\hat{J}_q, \Omega_k^2] + [B_k, \Omega_k] - \frac{1}{h}(B_k - B_{k-1}) + T_k)D^{-1}, \\ R_{k+1} = R_k \text{Exp}(h\Omega_k), \end{array} \right. \quad (8.7)$$

to what concern the rotational component of motion, where $k = 0, 1, 2, \dots$ and the initial conditions are $\Omega_0 = 0$, $R_0 = I_3$ and $B_{-1} = 0$, where all the constants take the values specified in (8.3), plus

$$\left\{ \begin{array}{l} v_{k+1} = v_k + h\left(\frac{1}{2}\frac{b}{M_q}(\omega_{1,k}^2 + \omega_{2,k}^2 + \omega_{3,k}^2 + \omega_{4,k}^2)R_k e_z - p e_z - \frac{1}{M_q}\Gamma v_k\right), \\ q_{k+1} = q_k + h v_k, \end{array} \right. \quad (8.8)$$

to what concerns the translational component of motion, where $v_0 = 0$ and $q_0 = 0$.

8.3 Results of the numerical simulation of the quadcopter drone

The quadcopter drone exhibits a complex dynamics, characterized by two pairs of IVPs, one for translational component of motion on the 3-dimensional real space (affecting the position of its center of mass) and one for the rotational component of motion on the $SO(3)$ manifold (affecting its attitude). In the whole lot of the following numerical simulations, the value $h = 10^{-3}$ has been chosen.

The first simulation concerns only the rotational component of motion, whose computation involves only one of the two pairs of differential equations. In fact, it can be seen from the model (8.2) of the drone that the rotational movement is completely independent of the translational one.

In the the Figure 8.2 it is possible to observe that the drone oscillates around the center of mass, which however stay still in space. Instead it can not independently achieve any translation, since, as it appears from the mathematical model, the translational component of motion is related to the orientation of a drone.

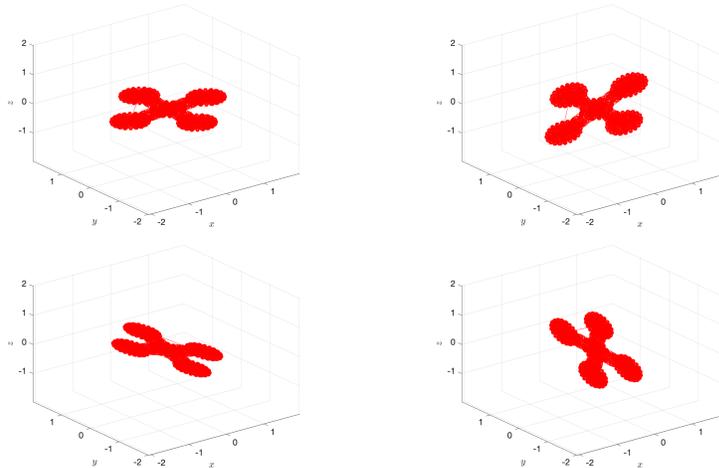


Figure 8.2: *Behaviour of the quadrotor drone (8.7) that performs only rotational movements, in which it oscillates around the center of mass, that is in the origin of the reference system.*

Complete simulations were obtained to show a drone which moves along a trajectory in \mathbb{R}^3 , changing its attitude according a chosen maneuver. In particular the simulations concern four different maneuvers:

- **Hover**, that describes the static behavior of the drone on the space remain-

ing on its position parallel to x - y plane. In this situation, the rotors of the drone have only to overcome gravity (in absence of other external forces which are not described by our model). The obtained results are shown in the Figure 8.3.

- **Yaw and roll**, with the drone turning and moving along the negative y direction. The obtained results are shown in the Figure 8.4.
- **Yaw and pitch**, with the drone turning and moving along the positive x direction. The obtained results are shown in the Figure 8.5.
- **Pitch and roll**, with the drone turning and moving along the positive direction of both x and y . The obtained results are shown in the Figure 8.6.

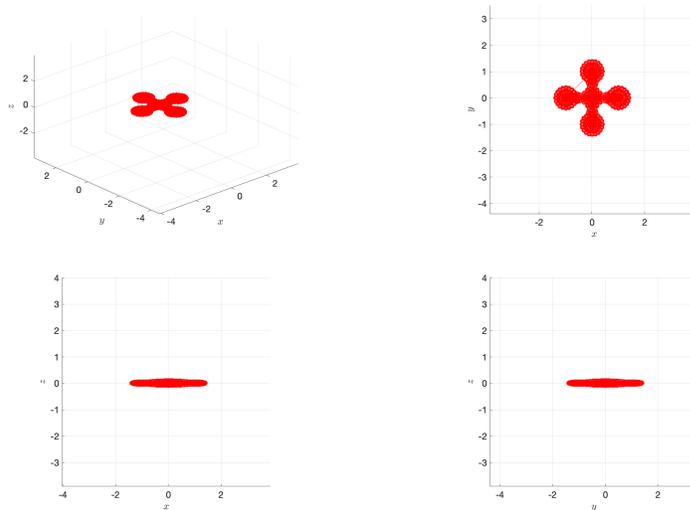


Figure 8.3: *Behaviour of a drone performing a hover maneuver. It is possible to see that the object remains parallel to the x - y plane thus maintaining the same height. In the top-left panel it can be seen a 3D view, while in the top-right panel there is a x - y view, in the bottom-left panel there is a x - z view and in the bottom-right panel there is a y - z view.*

These maneuvers are implemented by changing the speed of the four rotors of the drone. When their speed is the same the drone will move vertically upwards or downwards or it will remain on its position.

There exists a particular value of rotors speed, as given in (8.5), that approximately balances the gravitational acceleration, so that the resulting force acting on

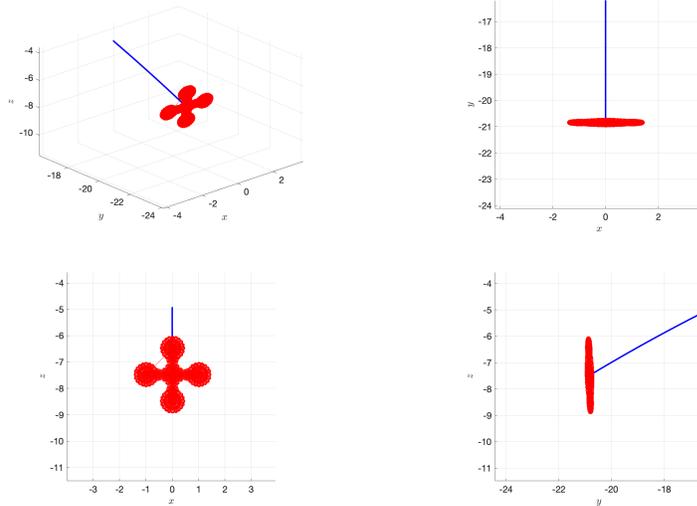


Figure 8.4: *Behaviour of a drone performing a yaw and roll maneuver. It is possible to see that the object is turning and moving along the negative y direction.*

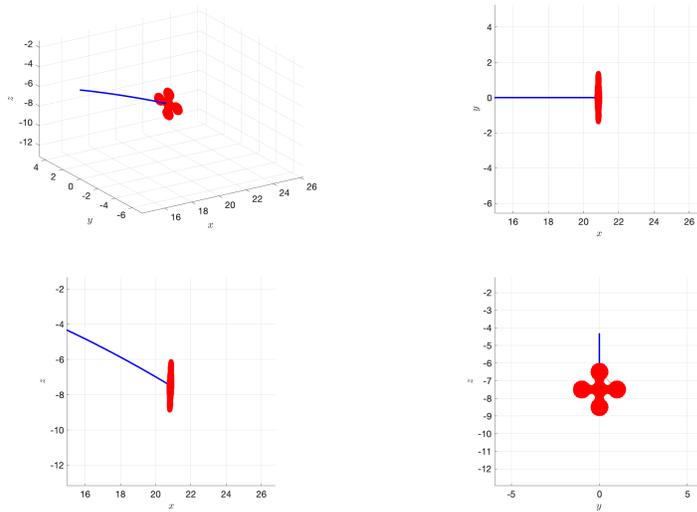


Figure 8.5: *Behaviour of a drone performing a yaw and pitch maneuver. It is possible to see that the object is turning and moving along the positive x direction.*

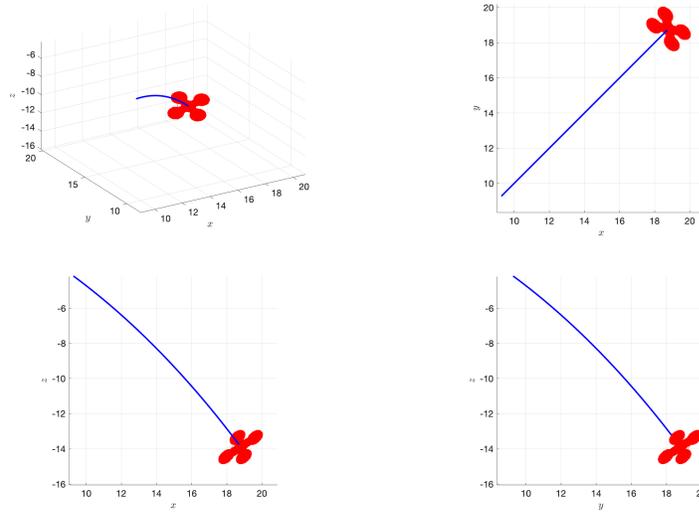


Figure 8.6: *Behaviour of a drone performing a pitch and roll maneuver. It is possible to see that the object is turning and moving along the positive direction of both x and y axes.*

the system is null and the drone does not move. For larger values of rotors speed, the drone will be subjected to a vertical lift and for smaller values it will move downwards vertically.

As already mentioned, the differential equation involving the translational component of motion depends from rotational parameters, while the rotational component of motion is not influenced by the translational one. For this reason it has firstly been possible to simulate only the rotational dynamics of the drone, because it can be considered separately being independent from translation.

9. Synchronization applied to quadcopter drones

9.1 Extension of PID regulation to Lie groups (L-PID)

The purpose of feedback control is to drive the state $X(t)$ of a Lie-group dynamical system towards a reference point, which may be moving over time, denoted by $Z(t) \in \mathbb{G}$. The first step consists in defining an *error field* that generalizes the Euclidean error $e(t) = z(t) - x(t)$ for the standard case that $\mathbb{G} = \mathbb{R}^n$.

We propose that a second-order, L-PID controlled dynamical system be designed as:

$$\left\{ \begin{array}{l} \frac{d}{dt}(X^{-1}(t) \cdot \dot{X}(t)) = X^{-1}(t) \cdot \mathbb{S}(t, X(t), V(t)) + U(t), \in \mathfrak{g} \text{ (Second-order dynamical system)} \\ E(t) := X^{-1}(t) \cdot \log_{X(t)} Z(t) \in \mathfrak{g}, \text{ (Control error)} \\ \Psi(t) := \int_0^t E(\tau) d\tau \in \mathfrak{g}, \text{ (Integrated control error)} \\ \mathcal{E}(t) := Z^{-1}(t) \cdot \dot{Z}(t) - X^{-1}(t) \cdot \dot{X}(t) \in \mathfrak{g}, \text{ (Control error velocity)} \\ U(t) := \kappa_P E(t) + \kappa_I \Psi(t) + \kappa_D \mathcal{E}(t) + U_C(t) \in \mathfrak{g}, \text{ (Control field).} \end{array} \right. \quad (9.1)$$

with $U_C(t)$ to be defined.

Taking the derivative of the error velocity field $\mathcal{E}(t)$ leads to:

$$\dot{\mathcal{E}} = \frac{d}{dt}(Z^{-1} \cdot \dot{Z}) - X^{-1} \cdot \mathbb{S}(t, X, V) - U. \quad (9.2)$$

Plugging the definition of the control field given in (9.1) into the equation (9.2) yields

$$\dot{\mathcal{E}} = \frac{d}{dt}(Z^{-1} \cdot \dot{Z}) - X^{-1} \cdot \mathbb{S}(t, X, V) - \kappa_P E - \kappa_I \Psi - \kappa_D \mathcal{E} - U_C. \quad (9.3)$$

The purpose of the term U_C is to cancel those terms in the tight-hand side of the above expression that do not depend on the error-type fields (E, \mathcal{E}, Ψ) [16], namely, we set:

$$U_C := \frac{d}{dt}(Z^{-1} \cdot \dot{Z}) - X^{-1} \cdot \mathbb{S}(t, X, V). \quad (9.4)$$

The resulting error dynamics (9.2) is governed by the equation:

$$\dot{\mathcal{E}} + \kappa_P E + \kappa_I \Psi + \kappa_D \mathcal{E} = 0. \quad (9.5)$$

The above second-order error system is characterized by the following important result.

Theorem 9.1.1. *The error system (9.5) converges asymptotically to the state zero as long as the compatibility condition $X^{-1} \cdot \log_X Z = -Z^{-1} \cdot \log_Z X$ holds true, $\kappa_P > 0$, $\kappa_D \geq 0$, $\kappa_I = \tilde{\kappa}_I \langle \Omega, \mathcal{E} \rangle_{\mathfrak{g}}$, with $\tilde{\kappa}_I \geq 0$ (at least one among κ_D and $\tilde{\kappa}_I$ must differ from zero).*

Proof. The convergence of the synchronization error to zero is proven by showing that the following function is Lyapunov:

$$\mathcal{W}(X, Z) := \frac{1}{2} \langle \mathcal{E}, \mathcal{E} \rangle_{\mathfrak{g}} + \frac{\kappa_D}{2} d^2(X, Z). \quad (9.6)$$

Since $\kappa_P > 0$, it holds that $\mathcal{W}(X, Z) \geq 0$.

Let us consider the first term $\langle \mathcal{E}, \mathcal{E} \rangle_{\mathfrak{g}}$, whose time-derivative reads:

$$\frac{d}{dt} \langle \mathcal{E}, \mathcal{E} \rangle_{\mathfrak{g}} = \langle \dot{\mathcal{E}}, \mathcal{E} \rangle_{\mathfrak{g}} + \langle \mathcal{E}, \dot{\mathcal{E}} \rangle_{\mathfrak{g}} = 2 \langle \dot{\mathcal{E}}, \mathcal{E} \rangle_{\mathfrak{g}}. \quad (9.7)$$

Let us now consider the second term $d^2(X, Z)$. According to known equation and the definition of control error field given in (9.1), it turns out that

$$\begin{cases} \text{grad}_X d^2(X, Z) = -2 \log_X Z = -2X \cdot E, \\ \text{grad}_Z d^2(X, Z) = -2 \log_Z X = 2Z \cdot E \text{ (by the compatibility condition),} \\ \frac{d}{dt} d^2(X, Z) = \langle \text{grad}_X d^2(X, Z), \dot{X} \rangle_X + \langle \text{grad}_Z d^2(X, Z), \dot{Z} \rangle_Z. \end{cases} \quad (9.8)$$

Therefore, it can be seen that the derivative $\frac{d}{dt} d^2(X, Z)$ takes the expression:

$$\begin{aligned} \frac{d}{dt} d^2(X, Z) &= \langle -2X \cdot E, \dot{X} \rangle_X + \langle 2Z \cdot E, \dot{Z} \rangle_Z \\ &= -2 \langle X^{-1} \cdot X \cdot E, X^{-1} \cdot \dot{X} \rangle_{\mathfrak{g}} + 2 \langle Z^{-1} \cdot Z \cdot E, Z^{-1} \cdot \dot{Z} \rangle_{\mathfrak{g}} \\ &= 2 \langle E, Z^{-1} \cdot \dot{Z} - X^{-1} \cdot \dot{X} \rangle_{\mathfrak{g}} \\ &= 2 \langle E, \mathcal{E} \rangle_{\mathfrak{g}}. \end{aligned} \quad (9.9)$$

In conclusion, deriving the function \mathcal{W} with respect to the time gives:

$$\begin{aligned}
\mathcal{W} &= \kappa_P \langle E, \mathcal{E} \rangle + \langle \dot{\mathcal{E}}, \mathcal{E} \rangle_{\mathfrak{g}} \\
&= \langle -\kappa_P E - \kappa_I \Psi - \kappa_D \mathcal{E}, \mathcal{E} \rangle_{\mathfrak{g}} + \kappa_P \langle E, \mathcal{E} \rangle_{\mathfrak{g}} \\
&= -\kappa_D \langle \mathcal{E}, \mathcal{E} \rangle_{\mathfrak{g}} - \tilde{\kappa}_I \langle \Psi, \mathcal{E} \rangle_{\mathfrak{g}}^2, \\
&\leq 0,
\end{aligned} \tag{9.10}$$

since the coefficients are non-negative and at least one of them is non-zero, which proves the assertion. \square

The Theorem 9.1.1 is based on the assumption that $X^{-1} \cdot \log_X Z = -Z^{-1} \cdot \log_Z X$. It is important to remark that the property $\log_Z X = -Z \cdot X^{-1} \cdot \log_X Z$ is a special feature of a Lie group and of its logarithmic map and it does not extend to other tangent fields (namely, if $P^{X \rightarrow Z}$ denotes parallel transport, in general $P^{X \rightarrow Z} \neq Z \cdot X^{-1}$.)!

Let us consider two examples where this property can be easily verified:

- **Special orthogonal group:** In this group, inversion coincides to matrix transposition and $\log_X Z := X \text{Log}(X^\top Z)$, therefore, it can be readily verified that $X^\top \log_X Z = -Z^\top \log_Z X$ is implied by $\text{Log}(X^\top Z) = -\text{Log}(Z^\top X)$. However, parallel transport in $\text{SO}(3)$ reads $P^{X \rightarrow Z}(V) := X \sqrt{X^\top Y} X^\top V \sqrt{X^\top Y}$, which is apparently very different from $ZX^\top V$ (which is, in fact, a possible instance of *vector transport*).
- **Group of symmetric, positive-definite matrices:** In this group, each tangent space coincides directly to the Lie algebra, therefore left translation is an identity. Moreover, $\log_X Z := \text{Log}(Z) - \text{Log}(X)$, which verifies the property.

Notice that positional control implies velocity control (although the converse is not true, in general).

9.1.1 Control efforts

The notion of *control effort* refers to a quantification of the magnitude of a control field, which is related to the energy consumption of the actuators and is used to evaluate the suitability of the actuators with respect to a given control goal [14]. Formally, we define a global control effort function $\sigma : TM \rightarrow \mathbb{R}$ as:

$$\sigma := \frac{1}{2} \|U\|_{\mathfrak{g}}^2. \tag{9.11}$$

Likewise, we define a control effort function $\sigma_C : TM \rightarrow \mathbb{R}$ and a function $\sigma_{PID} : TM \rightarrow \mathbb{R}$ associated to the ‘cancelling’ component of the control field and to the PID component of the same control field, as:

$$\sigma_C := \frac{1}{2} \|U_C\|_{\mathfrak{g}}^2, \quad \sigma_{PID} := \frac{1}{2} \|U - U_C\|_{\mathfrak{g}}^2. \quad (9.12)$$

It is important to evaluate the control efforts associated to the different components of the above-proposed M-PID control schemes. It is clear that the control effort σ_{PID} is directly related to the parameters $\kappa_P, \kappa_I, \kappa_D$.

9.2 Application of L-PID to the time-synchronization of the attitude of two quadrotor drones

We shall assume that the two drones were described by the same mathematical model and with the same constants. The main difference between the follower and the leader is how their dynamics is determined. In fact, the dynamics of the leader is governed by the mechanical torque T generated by the propellers, that are assumed to be constant, as described in the system (8.8). The control field for the follower is produced by the L-PID controller.

The drone can perform rotational and translational movements. The rotational movements are independent of the translational ones, but the converse is not true. Therefore it is possible to implement two different types of synchronization: the first one controls only the attitude of the drone, without considering the translation movements, and the second one allows to modify at the same time the attitude and position of the drone.

The leader drone is described by the continuous time system:

$$\begin{cases} T := br(\omega_{L,4}^2 - \omega_{L,2}^2)\Omega_x + br(\omega_{L,3}^2 - \omega_{L,1}^2)\Omega_y + \\ \quad \gamma(-\omega_{L,1}^2 + \omega_{L,2}^2 - \omega_{L,3}^2 + \omega_{L,4}^2)\Omega_z, \\ \Omega_{L,r} := -\omega_{L,1} + \omega_{L,2} - \omega_{L,3} + \omega_{L,4}, \\ B_L := J_{\mathcal{R}}\Omega_{L,r}\Omega_z, \\ \dot{W} = D^{-1}([\hat{f}_q, W^2] + [B_L, W] - \dot{B}_L L + T)D^{-1}, \\ \dot{Z} = ZW, \end{cases} \quad (9.13)$$

where all the constants used have the values specified in (8.3). The follower drone

is described by the continuous time system:

$$\begin{cases} \Omega_{F,r} := -\omega_{F,1} + \omega_{F,2} - \omega_{F,3} + \omega_{F,4}, \\ B_F := J_{\mathcal{R}} \Omega_{F,r} \Omega_z, \\ \dot{V} = D^{-1} ([\hat{J}_q, V^2] + [B_F, V] - \dot{B}_F) D^{-1} + U, \\ \dot{X} = X V. \end{cases} \quad (9.14)$$

The leader system is described by the variables $(Z, W) \in \text{SO}(3) \times \mathfrak{so}(3)$ and the follower system by $(X, V) \in \text{SO}(3) \times \mathfrak{so}(3)$, while the control signal generated by the PID controller is denoted as U , and it is defined as follows:

$$\begin{cases} E(t) := \text{Log}(X(t)^{-1}Z(t)), \\ \mathcal{E}(t) := W(t) - V(t), \\ \Psi(t) := \int_0^t E(\tau) d\tau, \\ \mathbb{S}_F(t, X, V) = D^{-1} ([\hat{J}_q, V^2] + [B_F, V] - \dot{B}_F) D^{-1}, \\ U_C(t) := \dot{W}(t) - \mathbb{S}_F(t, X, V), \\ U(t) := \kappa_P E(t) + \kappa_I \Psi(t) + \kappa_D \mathcal{E}(t) + U_C(t) \end{cases} \quad (9.15)$$

A discrete time systems of the leader, realized with a forward Euler method and used in the simulations, is the following:

$$\begin{cases} T_k := br(\omega_{L,4,k}^2 - \omega_{L,2,k}^2) \Omega_x + br(\omega_{L,3,k}^2 - \omega_{L,1,k}^2) \Omega_y + \\ \quad \gamma(-\omega_{L,1,k}^2 + \omega_{L,2,k}^2 - \omega_{L,3,k}^2 + \omega_{L,4,k}^2) \Omega_z, \\ \Omega_{L,r,k} := -\omega_{L,1,k} + \omega_{L,2,k} - \omega_{L,3,k} + \omega_{L,4,k}, \\ B_{L,k} := J_{\mathcal{R}} \Omega_{L,r,k} \Omega_z, \\ W_{k+1} = W_k + hD^{-1} ([\hat{J}_q, W_k^2] + [B_{L,k}, W_k] - \frac{1}{h} (B_{L,k} - B_{L,k-1}) + T_k) D^{-1}, \\ Z_{k+1} = Z_k \text{Exp}(hW_k), \end{cases} \quad (9.16)$$

where $B_{L,-1} = 0$. Likewise, a discrete time systems of the follower is the following:

$$\begin{cases} \Omega_{F,r,k} := -\omega_{F,1,k} + \omega_{F,2,k} - \omega_{F,3,k} + \omega_{F,4,k}, \\ B_{F,k} := J_{\mathcal{R}} \Omega_{F,r,k} \Omega_z, \\ V_{k+1} = V_k + hD^{-1} ([\hat{J}_q, V_k^2] + [B_{F,k}, V_k] - \frac{1}{h} (B_{F,k} - B_{F,k-1})) D^{-1} + hU_k, \\ X_{k+1} = X_k \text{Exp}(hV_k), \end{cases} \quad (9.17)$$

where $B_{F,-1} = 0$.

In the discrete time version, the leader system is described by the variables $(Z_k, W_k) \in \text{SO}(3) \times \mathfrak{so}(3)$ and the follower system by $(X_k, V_k) \in \text{SO}(3) \times \mathfrak{so}(3)$,

while the discrete time version of the control signal generated by the PID controller is denoted as U_k , and it is defined as follows:

$$\begin{cases} E_k := \text{Log}(X_k^{-1}Z_k), \\ \mathcal{E}_k := W_k - V_k, \\ \Psi_k := h \sum_{i=0}^k E_i, \\ S_{F,k} := D^{-1}([\hat{f}_q, V_k^2] + [B_{F,k}, V_k] - \frac{1}{h}(B_{F,k} - B_{F,k-1}))D^{-1}, \\ U_{C,k} := \begin{cases} -S_F(0, X_0, V_0), & \text{if } k = 0, \\ \frac{1}{h}(W_k - W_{k-1}) - S_{F,k}, & \text{if } k > 0, \end{cases} \\ U_k := \kappa_P E_k + \kappa_I \Psi_k + \kappa_D \mathcal{E}_k + U_{C,k}, \end{cases} \quad (9.18)$$

where $B_{F,-1} = 0$.

9.3 Numerical experiments on the time-synchronization of drones

The present section illustrates L-PID-based synchronization of second-order dynamical systems taking as cases of study two identical quadrotor drones. For the sake of clarity, the synchronization of the motion of two quadcopters is treated in two different ways: first, pure rotation synchronization is illustrated; second, mixed rotation/translation synchronization is discussed from a theoretical point of view and successively illustrated numerically, since this specific instance of synchronization requires to modify the control equations non-trivially.

In all experiments presented in this section, the stepsize was set to $h = 0.001$.

9.3.1 Quadrotor drones synchronization – attitude control

During the simulations, no differences were observed whether the control signal U_C is present or absent. Indeed, in this case, since the internal dynamics of the two drones are the same, once that synchronization is reached, therefore the leader and the follower systems reached the same attitude, they will begin to follow the same trajectory, making sure that the control field tends to zero. The purpose of term U_C is to cancel the internal dynamics of the follower, replacing it with that given by the the leader, and compensate for the acceleration difference of the two systems. But since the two internal dynamics are the same, once the two drones have reached the same attitude, the contribution of the term U_C becomes dispensable. In particular, in the simulation where U_C is not applied, shown in Figure 9.1, synchronization is achieved. Indeed, it is possible to see that the U_C signal has a great value only at

the beginning (if it is applied), when the two drones are not synchronized, but once they have reached the same attitude it vanishes to zero.

In the simulations, several tests have been carried out, both to demonstrate the effective validity of the theorem and to oversee the contribution of each of the three terms that make up the PID control field.

First, a PID controller comprising only a proportional control term was tested. The result of this test is that the distance between the leader's and the follower's states does not converge to zero, as can be seen in Figure 9.2. The error behaves in a similar manner if a control action is applied which does not contain the proportional part, but only integrative and derivative, as can be seen in the Figure 9.3. This happens because in these two cases the conditions of the Theorem 9.1.1 are not respected. Otherwise, synchronization is successful if Theorem 9.1.1 is respected, and therefore when the control action is of type PI (proportional-integral), PD (proportional-derivative) or PID (proportional-integral-derivative). It can be noted that in the simulation in which the controller is of the PI type (so the derivative action is not applied, and $\kappa_D = 0$), the follower state oscillates much around the leader system state, even if it achieves synchronization, as confirmed by the Figure 9.4. Indeed, from the distance graph it is possible to see that it oscillates, but with very small orders of magnitude. Moreover the controller has been implemented in its complete form (proportional, derivative and integral parts with the contribution of U_C are applied), achieving the synchronization, as shown in the Figure 9.5.

In all numerical simulations where synchronization happens successfully, the values taken by the control efforts span several magnitude orders.

9.3.2 Quadrotor drones synchronization – mixed attitude and position control

In the first numerical experiment, the synchronization/control goal is to drive the follower drone to take the same attitude as the leader drone, though the follower drone must sync to the position of the leader drone up to a fixed distance of 5 units along the z axis of the earth reference system.

In this case, the second part of the attitude controller (that depends on the error field M) has a greater effect at the beginning of the synchronization, when the two drones must approach one another, then its contribution will become very small, since the direction perpendicular to the plane of the propellers and the direction of α will almost coincide and it will no longer be necessary to apply any rotation to the follower drone to approach it to the leader. While the first part (that depends to the distance between the attitude states of the leader and the follower) has a smaller effect at the beginning, allowing the two drones to approach one another,

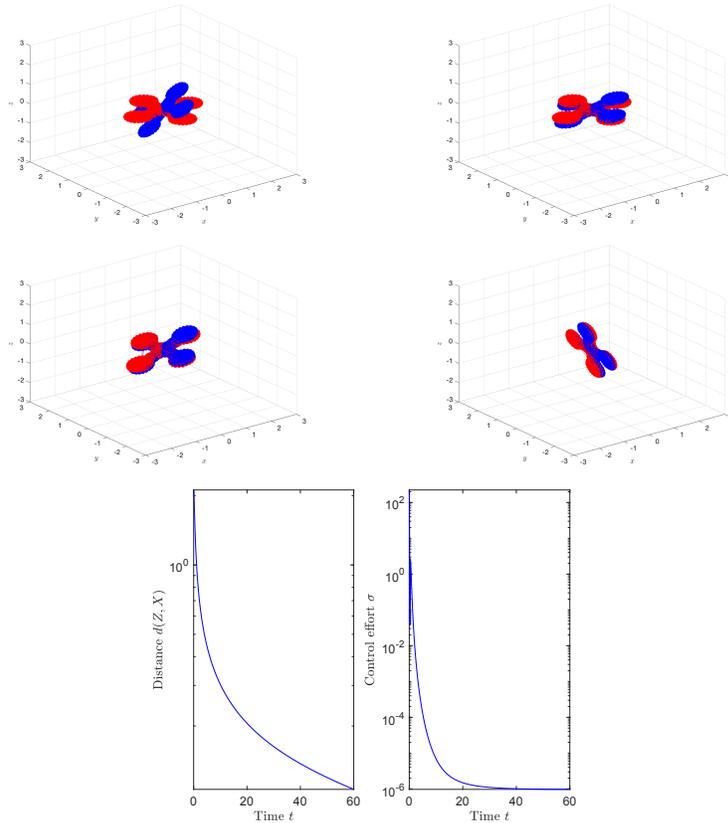


Figure 9.1: Synchronization of two drones, where the follower is controlled by a L -PID controller, without the contribution of U_C . The first four figures represent four successive instants during the process of synchronization of the two bodies. In particular the blue object denotes the follower, while the red one denotes the leader. The last two figures show the behavior of the distance between the leader and follower attitude and the control effort σ . The proportional control coefficient was set to $\kappa_P = 10$, the integral control coefficient was set to $\kappa_I = 5$ and the derivative coefficient was set to $\kappa_D = 10$.

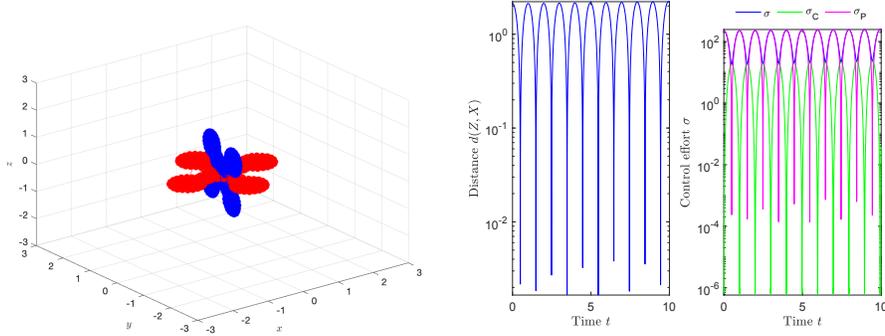


Figure 9.2: Synchronization of two drones, where the follower is controlled by a L-PID controller and the PID is made by only the proportional part (P). The first figure represents the last instant of the process of synchronization of the two bodies. In particular the blue object denotes the follower, while the red one denotes the leader. The last two figures show the behavior of the distance between the leader and follower attitude and the three control efforts: σ , σ_C and σ_P . The proportional control coefficient was set to $\kappa_P = 10$, the integral control coefficient was set to $\kappa_I = 0$ and the derivative coefficient was set to $\kappa_D = 0$.

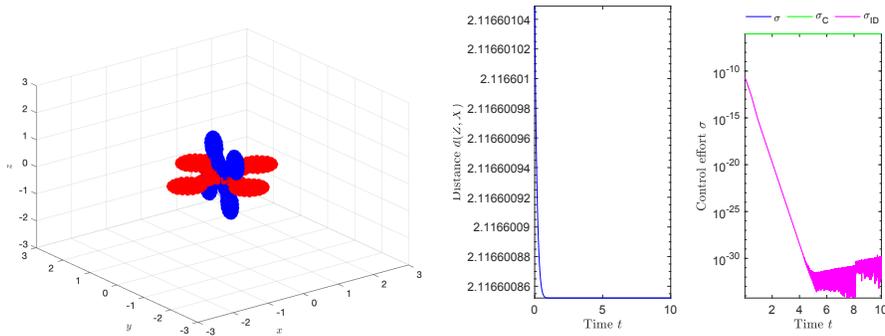


Figure 9.3: Synchronization of two drones, where the follower is controlled by a L-PID controller and the PID is made by only the integral and derivative parts (ID). The first figure represents the last instant of the process of synchronization of the two bodies. In particular the blue object denotes the follower, while the red one denotes the leader. The last two figures show the behavior of the distance between the leader and follower attitude and the three control efforts: σ , σ_C and σ_{ID} . The proportional control coefficient was set to $\kappa_P = 0$, the integral control coefficient was set to $\kappa_I = 5$ and the derivative coefficient was set to $\kappa_D = 10$.

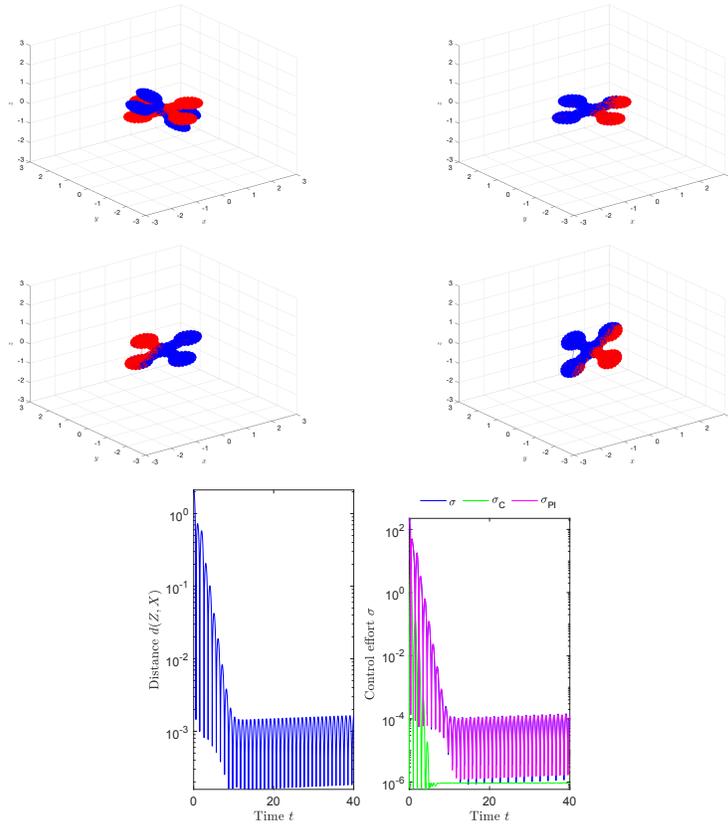


Figure 9.4: Synchronization of two drones, where the follower is controlled by a L-PID controller and the PID is made by only proportional and integral parts (PI). The first four figures represent four successive instants during the process of synchronization of the two bodies. In particular the blue object denotes the follower, while the red one denotes the leader. The last two figures show the behavior of the distance between the leader and follower attitude and the three control efforts: σ , σ_C and σ_{PI} . The proportional control coefficient was set to $\kappa_P = 10$, the integral control coefficient was set to $\kappa_I = 5$ and the derivative coefficient was set to $\kappa_D = 0$.

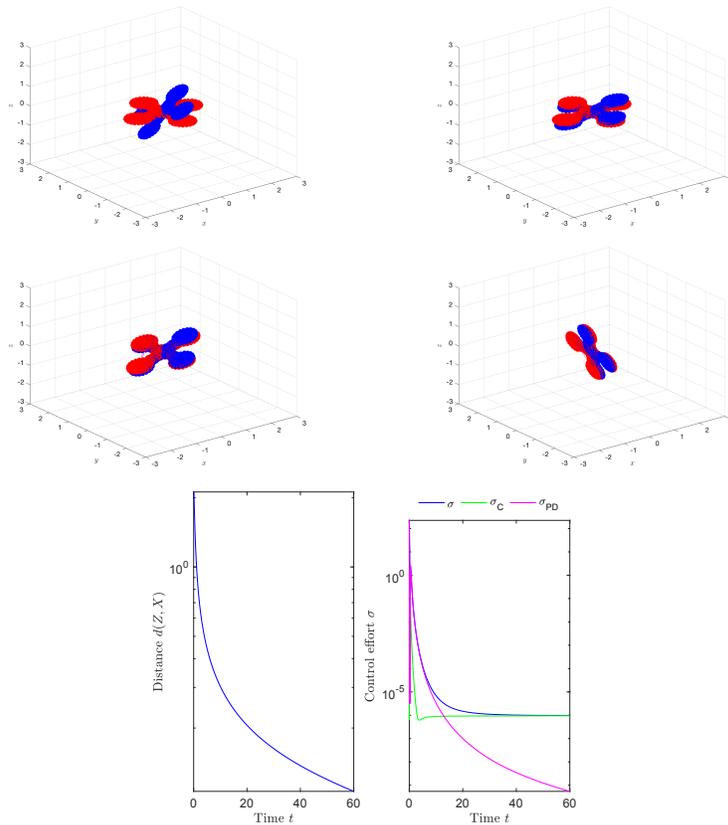


Figure 9.5: Synchronization of two drones, where the follower is controlled by a L -PID controller. The first four figures represent four successive instants during the process of synchronization of the two bodies. In particular the blue object denotes the follower, while the red one denotes the leader. The last two figures show the behavior of the distance between the leader and follower attitude and the three control efforts: σ , σ_C and σ_{PID} . The proportional control coefficient was set to $\kappa_P = 10$, the integral control coefficient was set to $\kappa_I = 5$ and the derivative coefficient was set to $\kappa_D = 10$.

then when they come sufficiently near, so the error field M will be very small, the rotation of the follower drone is determined almost exclusively by synchronization with the leader attitude.

In the first simulation, the leader makes only the maneuver “Yaw & Pitch (+x)” and synchronization is successful, as it can be observed from the results displayed in the Figure 9.7. It is interesting to note that by increasing the weight of M in the attitude controller, the range, where the distance between the two drones oscillates, decreases. However, at the same time the distance of the attitude between the two drones increases. So it is necessary to find a compromise between the two situations, finding the right values for the PID coefficients and weight of M , making sure that both synchronizations take place.

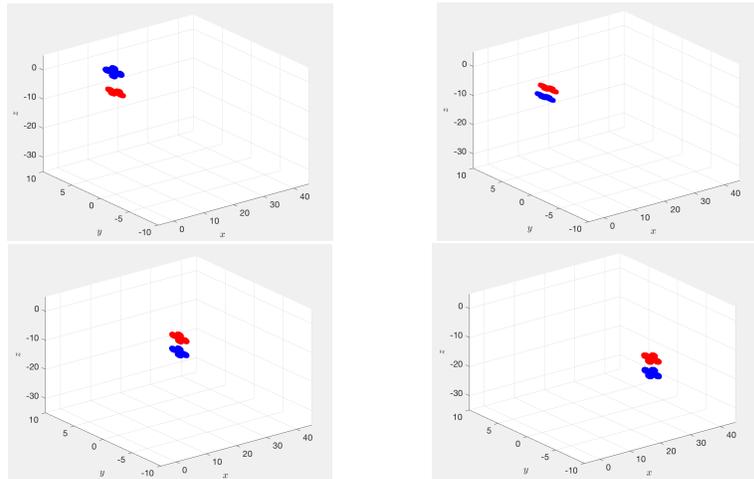


Figure 9.6: *Synchronization of two drones, where the follower is controlled by a mixed attitude and position controller. The figures illustrates the process of synchronization of the two bodies. In particular the blue object denotes the follower, while the red one denotes the leader.*

In the second numerical experiment, the leader makes three different maneuvers: “Pitch (+x) & Roll (+y)”, “Yaw & Pitch (+x)” and “Yaw & Roll (-y)”, creating a more complex maneuver in the space, and therefore more difficult to synchronize. From the Figure 9.9, it is possible to see that each time it changes maneuver, the follower moves away. Therefore the distance between the two attitudes increases because the rotational effect to align the drone to the α direction increases, losing synchronization with the leader. Thus, it is important that a sufficiently long time has elapsed since the last maneuver changes to properly achieve synchronization. In fact, from the simulations, it emerged that if the previous con-

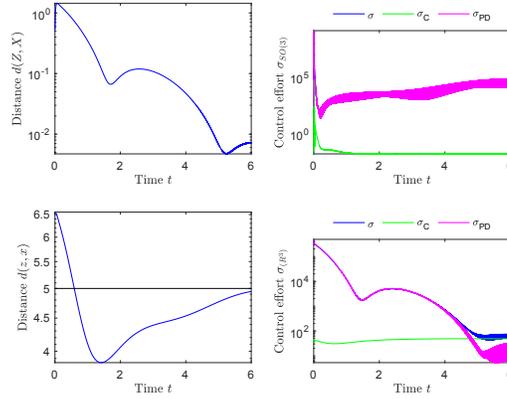


Figure 9.7: These four figures show the behavior of the distance between the leader and follower attitude, the distance between the leader and follower position, and the three control efforts of both control signal (position and attitude): σ , σ_C and σ_{PID} . In the position controller the coefficients are $\kappa_P = 100$, $\kappa_I = 10$, $\kappa_D = 80$, while in the attitude controller the coefficients are $\kappa_P = 50$, $\kappa_I = 10$, $\kappa_D = 50$.

sideration is respected, the synchronization takes place in all cases.

From the simulations are emerged different considerations on the parameters of the two PID controllers. As already seen in the previous cases, by varying the coefficients of the PID improves the control obtained. In the case of mixed maneuvers, it is necessary to have the system very reactive, since the follower must be able to quickly change his trajectory when the leader changes maneuver. This thing, however, means that the system loses stability, not by synchronizing and seeing the follower “flying” around the leader. Finally, a higher value of the last coefficient, that is κ_I , guarantees to have an ever smaller error at a permanent regime, but worsening the stability of the system.

The orders of magnitude of the coefficients of two PID are related because the PID implemented on a Lie group uses the control signal of the PID implemented in \mathbb{R}^3 . If the weight of the latter is too large, the rotation control will cause the α signal is applied with its maximum value, rotating the drone towards the alpha direction, losing synchronization of the attitude with the leader. Therefore it is important that the follower is reactive to be able to follow all the movements and maneuvers that the leader performs, but it is still necessary to find a compromise in order not to make the system unstable.

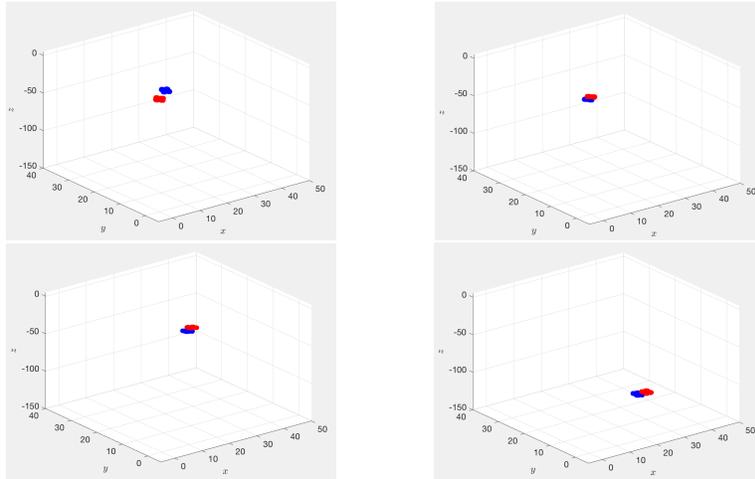


Figure 9.8: Synchronization of two drones, where the follower is controlled by mixed attitude and position controller and the leader performs three different maneuvers over the time. The figures illustrates the process of synchronization of the two bodies. In particular the blue object denotes the follower, while the red one denotes the leader.

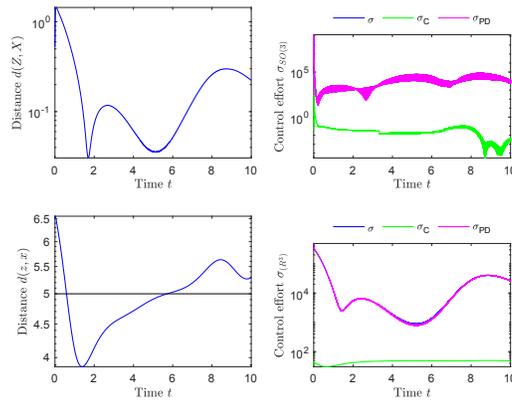


Figure 9.9: These four figures show the behavior of the distance between the leader and follower attitude, the distance between the leader and follower position, and the three control efforts of both control signal (position and attitude): σ , σ_C and σ_{PID} . In the position controller the coefficients are $\kappa_P = 100$, $\kappa_I = 10$, $\kappa_D = 80$, while in the attitude controller the coefficients are $\kappa_P = 50$, $\kappa_I = 10$, $\kappa_D = 50$.

9.3.3 Quadrotor drones synchronization in the real cases

The follower drone receives not only the information about the leader rotational position and velocity, but also of its position and velocity on the 3-dimensional space. These signals can be affected by noise and delay.

A problem can be time delay between the real state of the leader and the signal received by the follower. The Figure 9.10 displays synchronization of a leader drone and a follower one with a time delay.

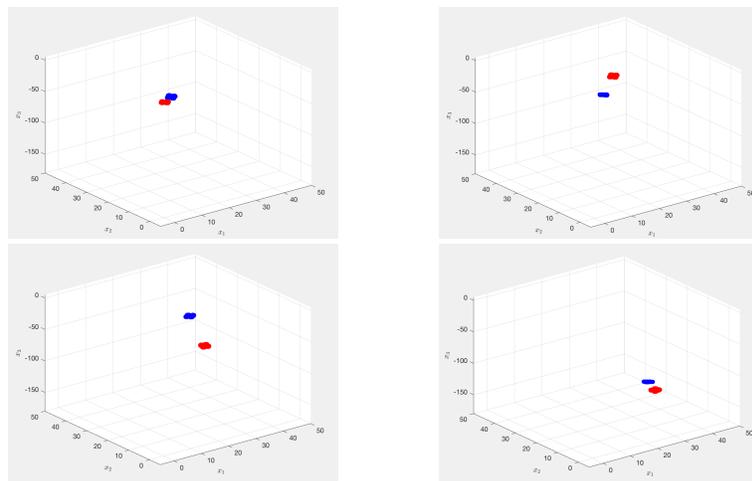


Figure 9.10: Synchronization of two quadcopter drones, where the follower is controlled to achieve the synchronization of its rotational and translational attitudes to the leader ones. The figures show the process of synchronization of the two bodies. In particular the blue object denotes the follower, while the red one denotes the leader. The proportional control coefficient was set to $\kappa_P = 10$, the integral control coefficient was set to $\kappa_I = 5$ and the derivative coefficient was set to $\kappa_D = 10$. In this case the time delay is $l = 2$ seconds, with step size $h = 0.001$.

As it can be noticed, the follower receives the information about leader's state with a certain time delay, so the two bodies exhibit the same trajectory, but not the same rotational attitude at same instants. However, in this case time delay also affects the velocity and position on the 3-dimensional space, because drones are bodies with a more complex dynamics. This problem affects synchronization of the two bodies and it has to be reduced as much as possible.

The signals with the information about the leader's state that the follower receives can be affected by different kinds of disturbances, like noise. The Figure 9.11 shows a comparison between the synchronization reached with noise (on the

left-hand figure) and without noise (on the right-hand figure). It can be seen that

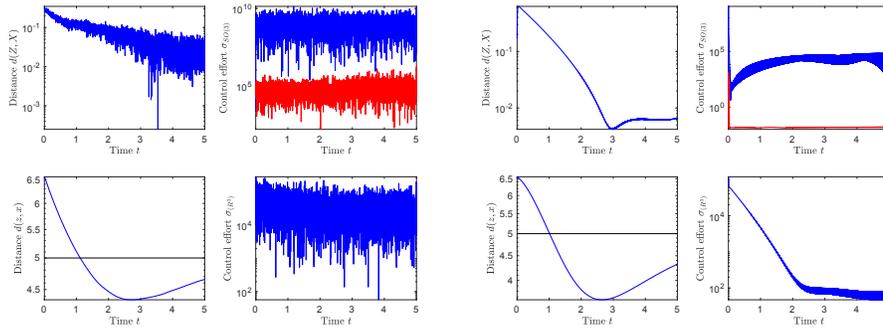


Figure 9.11: Synchronization of two quadcopter drones, where the follower is controlled to achieve the synchronization of its rotational and translational attitudes to the leader ones. The left-hand panel represents the case with additive noise, instead the right-hand one represents the case without it. In each panel are represented the distances of the states of the leader and follower and the control efforts generated over time. In both cases for both rotational attitude and position control the proportional control coefficient was set to $\kappa_P = 10$, the integral control coefficient was set to $\kappa_I = 5$ and the derivative coefficient was set to $\kappa_D = 10$.

synchronization is reached in both cases, but the control effort is larger in the case with noise and it is also irregular over time. The distance on R^3 tends to a constant value (equal to 5) in both the simulations, but in the case without noise it tends to this value faster, while the distance on $SO(3)$ is very irregular over time. So, in order to get a faster synchronization and to spend less energy in terms of control effort, it is important to reduce as much as possible the effects of these unwanted signals shielding the system from disturbances.

Finally the case of sinusoidal disturbances is shown in the simulation 9.12. The difference in the two simulations is that in the first case the sinusoidal disturbance has a frequency 10^2 times more than in the second case, so the high frequencies of it are filtered by the PID controller and synchronization is reached. Instead in the second case the frequency of the sinusoidal disturbance is lower, so the low frequency are not filtered and the PID controller will calculate the error to reduce to zero on the basis of the noisy measurements.

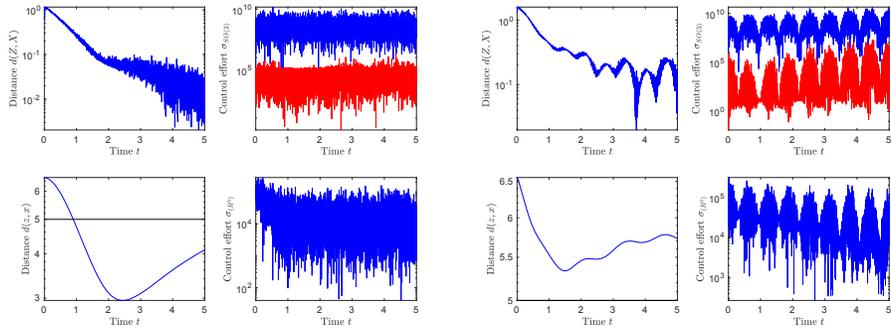


Figure 9.12: Synchronization of two quadcopter drones, where the follower is controlled to achieve the synchronization of its rotational and translational attitude to the leader ones and in both cases there is the presence of additive sinusoidal disturbance. The left panel shows the case with a sinusoidal disturbance with $\omega = 500$ rad/sec, instead the right one shows the case with $\omega = 5$ rad/sec. In each panel there are the graphics of the distances of the states of the leader and follower and of the control efforts generated over time. In both cases for both rotational attitude and position control the proportional control coefficient was set to $\kappa_P = 10$, the integral control coefficient was set to $\kappa_I = 5$ and the derivative coefficient was set to $\kappa_D = 10$.

10. Conclusion

The present thesis outlined the mathematical model and dynamics of a general second-order dynamical system on manifolds and on Lie groups expressed through the language of manifold calculus and Lie-group theory. Regulation of these second order dynamical systems has been studied, using M-PID controller applied on manifolds. Some interesting oscillators, the Duffing oscillators, has been presented in the several numerical simulations to achieve synchronization between two different oscillators. This simple control problem has been seen as a sort of 'toy problem' before analysing more complex real dynamics (quadcopter drones for example has been studied in this thesis). The several control simulations to oscillators have been also extended to real cases, with the presence of damping, disturbances and time delay and also with the introduction to the problem of the signal masking using this kind of synchronization between oscillators. In the second part of this thesis the mathematical model of quadcopter drone has been introduced, it is a composed dynamics because it is described by its rotational attitude (expressed using the orthogonal group) and its position in the 3-dimensional space. The second step has been to extend synchronization from simple oscillators to the more complex dynamics of drones with several simulations. The synchronization problem of a follower drone to a leader one has been extended to the real case with time delay for example, as for the oscillator case.

Bibliography

- [1] D. P. ATHERTON, *Almost six decades in control engineering*, IEEE Control Systems Magazine, 34 (2014), pp. 103–110.
- [2] M. BECKER, R. SAMPAIO, S. BOUABDALLAH, V. DE PERROT, AND R. SIEGWART, *In-flight collision avoidance controller based only on os4 embedded sensors*, Journal of the Brazilian Society of Mechanical Sciences and Engineering, 34 (2012), pp. 294–307.
- [3] Z. BENIĆ, P. PILJEK, AND D. KOTARSKI, *Mathematical modelling of unmanned aerial vehicles with four rotors*, Interdisciplinary Description of Complex Systems, 14 (2016), pp. 88–100.
- [4] A. M. BLOCH, *An introduction to aspects of geometric control theory*, in Nonholonomic Mechanics and Control, P. Krishnaprasad and R. Murray, eds., vol. 24 of Interdisciplinary Applied Mathematics, Springer, New York, NY, 2015.
- [5] F. BULLO AND A. D. LEWIS, *Geometric Control of Mechanical Systems*, vol. 49 of Texts in Applied Mathematics, Springer Verlag, New York-Heidelberg-Berlin, 2004.
- [6] M. EIE AND S.-T. MINKING, *A Course on Abstract Algebra*, World Scientific, 2010.
- [7] S. FIORI, *Nonlinear damped oscillators on riemannian manifolds: Numerical simulation*, Communications in Nonlinear Science and Numerical Simulation, 47 (2017), pp. 207 – 222.
- [8] ———, *Non-delayed synchronization of non-autonomous dynamical systems on Riemannian manifolds and its applications*, Nonlinear Dynamics, 94 (2018), pp. 3077–3100.

- [9] ———, *Model formulation over lie groups and numerical methods to simulate the motion of gyrostats and quadrotors*, Mathematics (MDPI), 7 (2019), p. Article No.935.
- [10] R. FUENTES, G. P. HICKS, AND J. M. OSBORNE, *The spring paradigm in tracking control of simple mechanical systems*, Automatica, 47 (2011), pp. 993–1000.
- [11] V. GHAFARI AND F. SHABANINIA, *Synchronization of nonlinear dynamical systems using extended Kalman filter and its application in some well-known chaotic systems*, Nonlinear Studies, 25 (2018).
- [12] O. GOLEVYCH, O. PYVOVAR, AND P. DUMENKO, *Synchronization of nonlinear dynamic systems under the conditions of noise action in the channel*, Latvian Journal of Physics and Technical Sciences, 55 (2018), pp. 70–76.
- [13] I. KOVACIC AND M. BRENNAN, *The Duffing Equation: Nonlinear Oscillators and their Behaviour*, Wiley, 2011.
- [14] M. A. MAGDY AND T. S. NG, *Regulation and control effort in self-tuning controllers*, IEE Proceedings D – Control Theory and Applications, 133 (1986), pp. 289–292.
- [15] A. NÁVRAT AND P. VAŠÍK, *On geometric control models of a robotic snake*, Note di Matematica, 37 (2017), pp. 120–129.
- [16] K. OJO, S. OGUNJO, AND A. OLAGUNDOYE, *Projective synchronization via active control of identical chaotic oscillators with parametric and external excitation*, International Journal of Nonlinear Science, 24 (2017), pp. 76–83.
- [17] J. M. OSBORNE AND G. P. HICKS, *The geodesic spring on the Euclidean sphere with parallel-transport-based damping*, Notices of the AMS, 60 (2013), pp. 544–556.
- [18] L. RIGHETTI, *Control and Synchronization with Nonlinear Dynamical Systems for an application to Humanoid Robotics*, Ecole Polytechnique Fédérale de Lausanne, 3 2004.
- [19] R. W. H. SARGENT, *Optimal control*, Computational and Applied Mathematics, 124 (2000), pp. 361–371.
- [20] N. SHERIF AND E. MORSY, *Computing real logarithm of a real matrix*, International Journal of Algebra, 2 (2008), pp. 131–142.

-
- [21] M. SHIINO AND K. OKUMURA, *Control of attractors in nonlinear dynamical systems using external noise: Effects of noise on synchronization phenomena*, Discrete and Continuous Dynamical Systems - Series S, (2013), pp. 685–694.
- [22] K. SREENATH, T. LEE, AND V. KUMAR, *Geometric control and differential flatness of a quadrotor UAV with a cable-suspended load*, in 52nd IEEE Conference on Decision and Control, Dec. 2013, pp. 2269–2274.
- [23] C. W. WU, *Synchronization in Complex Networks of Nonlinear Dynamical Systems*, World Scientific Publishing Co Pte Ltd, Singapore, October 2007.

Acknowledgments

The present research work was completed while the author IC was taking an internship at the Agency for Science, Technology and Research (A*STAR, Republic of Singapore) during April-June 2019. The author wishes to gratefully thank Dr. Hwee Kuan Lee who made this internship possible.