

JET-SCHEDULING CONTROL FOR FLAT SYSTEMS

THÈSE N° 3996 (2008)

PRÉSENTÉE LE 18 JANVIER 2008

À LA FACULTÉ DES SCIENCES ET TECHNIQUES DE L'INGÉNIEUR

LABORATOIRE D'AUTOMATIQUE

PROGRAMME DOCTORAL EN INFORMATIQUE, COMMUNICATIONS ET INFORMATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Davide BUCCIERI

ingénieur mécanicien diplômé EPF
de nationalité italienne

acceptée sur proposition du jury:

Prof. R. Longchamp, président du jury
Prof. D. Bonvin, Dr Ph. Müllhaupt, directeurs de thèse
Prof. H. Bleuler, rapporteur
Prof. Z.-P. Jiang, rapporteur
Prof. J. Lévine, rapporteur



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2008

A Ivo

Remerciements

Le parcours conduisant à ce travail a été tantôt une aventure enthousiasmante et tantôt un chemin de croix. Je voudrais exprimer toute ma gratitude aux personnes qui m'ont soutenu. Cette thèse n'aurait pas pu être possible sans leur aide.

Je tiens à remercier vivement le Professeur Dominique Bonvin, de m'avoir accueilli au sein du laboratoire d'automatique, pour son soutien et pour la liberté qu'il m' a accordée dans mes travaux de recherche. Je lui suis aussi particulièrement reconnaissant de m'avoir permis et encouragé à poursuivre mes études à New York dans le laboratoire du Professeur Zhong-Ping Jiang, que je remercie pour ses précieux conseils. Cette expérience s'est avérée extrêmement enrichissante et a contribué à l'aboutissement de ce travail.

Un grand merci à Denis Gillet d'avoir joué un rôle prédominant dans mon entrée au laboratoire d'automatique.

Je voudrais également adresser ici une pensée chaleureuse et particulièrement reconnaissante à Philippe Müllhaupt, d'avoir été un mentor dévoué durant tout le cheminement de cette thèse. Il m'a fait partager avec enthousiasme son ample culture scientifique, son expérience et son amour pour la belle rhétorique. Ce travail est le fruit de nos interminables et passionnantes discussions sur l'automatique et sur le monde en général. Au cours de ces années, Philippe est également devenu un ami sincère, avec qui je partage (peu modérément) et encore pour longtemps, les grandes tables et les bonnes bouteilles.

Je voudrais évoquer Christophe Salzman, informaticien de génie, sans qui les applications temps réel présentées dans cette thèse n'auraient pas pu voir le jour. Un grand merci.

Je voudrais exprimer toute ma gratitude au Professeur Hannes Bleuler, au Professeur Jean Lévine, au Professeur Zhong-Ping Jiang et au Professeur Roland Longchamp d'avoir accepté d'évaluer ce travail en tant que rapporteurs et pour leurs remarques pointues.

Je tiens à remercier chaleureusement mes collègues et amis de bureau, Yvan Michellod et Damien Perritaz pour les bons moments que nous avons passés ensemble et pour la solidarité qui s'est créée entre nous. Leur soutien m'a permis de surmonter les doutes et les difficultés.

Je souhaite remercier de tout coeur l'équipe féminine du secrétariat, Francine Eglese, Ruth Benassi et Sol Torres. Leur présence quasi maternelle m'a toujours réconforté. Un grand merci également à Philippe Cuanillon pour son aide et son support technique toujours efficace. Je tiens à exprimer ma gratitude envers tous les collègues du laboratoire d'avoir constitué une grande famille et un cadre de travail idéal.

Un grand merci à tous mes amis et spécialement à mon ami d'enfance Ismaël pour son intérêt et son soutien. Vorrei ringraziare di cuore i miei fratelli silvaroli, Vittorio, Mario, Stefano e Filippo, per essermi sempre stati vicini, malgrado la distanza, per avermi fatto sentire un silvarolo emigrato, piuttosto che uno svizzero con origini silvarole. Silvi è la nostra California.

Je tiens à remercier ma famille, Francesco, Loredana, Sabina et Debora, pour leur soutien inconditionnel et leur aide omniprésente. Je voudrais également exprimer toute ma gratitude à Sabrina qui, jour après jour, partage mes joies et mes peines, me supporte, m'encourage, me conseille,...., m'aime.

Infine, vorrei dedicare questo lavoro di tesi a mio nonno Ivo per essere esattamente com'è. Visionario.

Abstract

The control of cranes and nonholonomic robots has gained increased interest mainly because of the civilian and military industrial need to achieve fast and accurate transport of goods and equipment. Old and new harbors are now venturing into fully automated systems combining automated trolleys and classical cranes.

From a theoretical viewpoint these systems are challenging because they are strongly dynamically coupled and offer interesting and useful control problems. Therefore, to take full advantage of their potential, the control design must take into account as much structural information as possible.

The structural property that is exploited in the control design proposed in this thesis is the differential flatness property of these systems, that is the existence of particular functions of the states (called flat outputs), the time parametrization of which implies parametrization of all the individual states and inputs. This property is extremely useful for motion planning problems where the system should move quickly from one configuration to another, without inducing too much overshoot or residual oscillations.

However, the flatness property is not sufficient to guarantee the design of an efficient controller in the presence of uncertain and unmodeled dynamics. This is especially the case for cranes where the winching mechanism, expressed in terms of the engine and pulleys, has a large amount of unmodeled dry friction.

This robustness issue is normally addressed by splitting the control task into a feedforward-like part that handles the dynamical couplings and a feedback term that enforces the tracking of the reference values stemming from the feedforward motion planning algorithm. In contrast, this thesis proposes to combine these two mechanisms, resulting in what will be called the jet-scheduling controller.

Classically, the flatness property guarantees the construction of a feed-

forward input based on a planned motion of the flat outputs by simply combining values of the flat outputs and their time derivatives, i.e. without having to integrate differential equations. Therefore, in the absence of perturbation, this mechanism is sufficient to move the system from one state to another, once a trajectory compatible with the initial and final positions has been designed. However, when the system has some unmodeled dynamics, an additional mechanism must be provided to make sure that the planned trajectory is indeed tracked accurately.

The point of view adopted in this thesis is that, instead of specifying a trajectory to be tracked explicitly, a dynamical system called “the jet scheduler” provides the derivatives (the jets) of an ideal stabilizing trajectory. These jets are updated regularly according to measurements so as to react to unknown perturbations.

The flat correspondence is used to provide the values of the jets, and a subsidiary controller is designed to ensure that these jets are really matched asymptotically by the true system. Unfortunately, each of these mechanisms could possibly break the equivalence between the original nonlinear system and the linear extended system (contrary to the classical feedback linearization approach for which this correspondence is guaranteed at every time instant).

The design of the jet-scheduling controllers and the implication of the possible loss of correspondence are detailed in this work. In addition, stability issues are addressed. Applications to two classes of systems are shown, namely, nonholonomic robots and cranes. The specific properties of these systems are used to achieve a rigorous stability proof. The controller for both the nonholonomic robot and a new crane design labeled SpiderCrane that fully takes advantage of the jet-scheduling mechanism are tested on real setups.

Keywords: Nonlinear Control, Flatness-based Control, Trajectory Tracking, Stabilization, Nonholonomic Robot, Crane.

Résumé

Le contrôle des grues et des robots non-holonomes a connu un intérêt croissant ces dernières décennies. Ceci notamment afin d'améliorer la vitesse et la précision du transport de marchandises et d'équipements, à des fins tant civiles que militaires. Des anciens ports et d'autres plus modernes sont en train de prospecter l'utilisation de systèmes complètement automatisés qui combindraient l'utilisation de camions et de grues autonomes.

D'un point de vue théorique, ces systèmes sont difficiles à contrôler, car ils présentent de forts couplages dynamiques. Ils suscitent par là même des problèmes de contrôle intéressants et utiles. Afin d'utiliser pleinement les propriétés dynamiques de ces systèmes, le développement du contrôleur doit tenir compte au tant que possible de leur structure fondamentale.

La propriété structurelle qui est exploitée dans la méthodologie proposée dans cette thèse est la platitude différentielle. Cette propriété garantit l'existence de fonctions particulières des états (appelées sortie plate), dont la paramétrisation temporelle permet de déterminer chaque état individuellement ainsi que les entrées. Cette propriété est extrêmement utile dans les problèmes de planification de mouvement, où le système doit être déplacé rapidement d'une configuration vers une autre sans induire d'oscillations résiduelles (par exemple dans le cas des grues).

Cependant, la propriété de platitude n'est pas suffisante pour synthétiser un contrôleur performant lorsque sont présentes des dynamiques mal ou non modélisées. Ceci est particulièrement vrai dans le cas des grues, où le mécanisme de treuillage composé d'un moteur et de poulies, possède un fort frottement sec non modélisé.

Ce problème de robustesse est généralement adressé en séparant la tâche de contrôle en une commande a priori qui se charge des couplages dynamiques et en un terme de bouclage garantissant le suivi de la trajectoire de référence provenant d'un algorithme de planification de mouvement. Cepen-

dant, l'idée matresse de cette thèse est de fusionner ces deux mécanismes conduisant ainsi, à ce qui sera appelé le contrôleur planificateur de jets.

Classiquement, la propriété de platitude garantit la construction d'une entrée a priori basée sur un mouvement planifié de la sortie plate. Ceci se fait en combinant simplement la trajectoire désirée de la sortie plate et ses dérivées temporelles, c'est-à-dire sans devoir intégrer d'équations différentielles. Par conséquent, en absence de perturbation et une fois qu'une trajectoire compatible avec les conditions initiales et finales est définie, ce mécanisme est suffisant pour déplacer un système d'un état vers un autre.

Le point de vue adopté dans cette thèse est, au lieu de définir une trajectoire à suivre explicitement, qu'un système dynamique, baptisé "le planificateur de jet", fournit les dérivées (les jets) d'une trajectoire stabilisante idéale. Ces jets sont remis à jour régulièrement en tenant compte des mesures afin de réagir à des perturbations inconnues.

La correspondance basée sur la platitude est utilisée pour fournir la valeur de ces jets, et un contrôleur subsidiaire est synthétisé pour garantir que ces jets soient effectivement suivis de façon asymptotique par le système réel. Malheureusement, ces deux mécanismes peuvent briser l'équivalence entre le système non linéaire d'origine et le système linéaire étendu (contrairement à la linéarisation exacte par bouclage dynamique qui garantit la correspondance à chaque instant du temps).

La synthèse du contrôleur et les implications d'une perte possible de correspondance, ainsi que la description détaillée de comment le mécanisme de planification se connecte avec la stabilité, sont décrits dans ce travail. De plus, des applications à deux classes de systèmes, les robots non-holonomes et le contrôle des grues, sont présentées. Les propriétés spécifiques de ces deux systèmes sont utilisées pour réaliser une preuve de stabilité rigoureuse. Le contrôle d'un robot mobile et d'un nouveau type de grue, baptisé SpiderCrane, tirant pleinement partie du mécanisme de planification des jets sont testés sur des systèmes réels.

Mots-clés: Commande Non Linéaire, Commande basé sur la platitude, Suivi de Trajectoire, Stabilisation, Robot Non-Holonyme, Grue.

*En atteignant le but,
on a manqué tout le reste.*

Proverbe japonais

Contents

1	Introduction	1
1.1	Motivation	1
1.2	State of the Art	2
1.3	Objectives and Organization of the Thesis	6
2	Preliminaries	9
2.1	Lie-Bäcklund Equivalence and Flatness	10
2.1.1	System definition	10
2.1.2	Lie-Bäcklund equivalence	11
2.1.3	Flatness	12
2.1.4	Application to open-loop control	14
2.1.5	Dynamical feedback linearization	15
2.2	Lyapunov Stability	18
2.2.1	Stability	18
2.2.2	Lyapunov theorem	20
2.2.3	Vanishing perturbed system	23
3	Jet-Scheduling Control	27
3.1	Introduction	27
3.2	The Concept of Jet-Scheduling Control	28
3.3	Construction of Jet-Scheduling Controllers	30
3.3.1	Introductory example	30
3.3.2	Jet-scheduling control	34
3.3.3	Dynamical feedback linearization based on jet scheduling	35
3.3.4	Reduced dynamical extension	37
3.4	Stability Analysis	47

3.5	Comparison with Dynamic Linearization	49
3.6	Example: Linear System	49
3.7	Conclusions	52
4	Application to Mobile Robots	55
4.1	Introduction	55
4.2	Preliminaries	57
4.3	Jet-Scheduling Control for a Mobile Robot	62
4.3.1	Point stabilization	62
4.3.2	Trajectory tracking	69
4.4	Simulation Study	70
4.4.1	Stabilization at the origin	70
4.4.2	Modification of the smooth controller	71
4.4.3	Discontinuous controller	73
4.4.4	Circular trajectory tracking	75
4.5	Stability Analysis	79
4.5.1	Convergence of the S_ξ -system	84
4.5.2	Transition and exceptional sets	88
4.5.3	Convergence of x_3	94
4.5.4	Partition of the initial conditions	97
4.5.5	Proof of the main propositions	99
4.5.6	A modification of Proposition 4.1	102
4.5.7	Proof of Proposition 4.3	103
4.6	Experimental Verification	105
4.6.1	Model mismatch and hardware limitations	105
4.6.2	Control implementation	106
4.6.3	Experimental results	107
4.6.4	Discussion	112
4.7	Conclusions	113
5	Application to Cranes	115
5.1	Introduction	115
5.2	Preliminaries	116
5.2.1	Crane description	117
5.2.2	Crane modeling	120
5.2.3	Flatness	125
5.2.4	Dynamical feedback linearization for the overhead crane	128
5.3	Jet-Scheduling Control for Cranes	131
5.3.1	Jet-scheduling control for the overhead crane	131

5.3.2	Jet-scheduling control for cranes	136
5.3.3	Jet-scheduling control for cranes: extension to trajectory tracking	144
5.4	Simulation Study	145
5.4.1	Stabilization	145
5.4.2	Circular trajectory tracking	146
5.5	Stability Analysis	149
5.5.1	Discussion about singularities	150
5.5.2	Stability analysis	153
5.6	Experimental Verification	161
5.6.1	Setup description	162
5.6.2	Dynamic model	163
5.6.3	Model discrepancies	164
5.6.4	Flatness of SpiderCrane	165
5.6.5	Jet-scheduling control	168
5.6.6	SpiderCrane implementation	170
5.6.7	Experimental results	171
5.7	Conclusion	174
6	Conclusions	177
	Curriculum vitæ	187

Chapter 1

Introduction

1.1 Motivation

Automatic control systems are present everywhere, in cars, in homes, in computers and in basically every modern device. There seems to be no reason for this trend not to continue in the future. Indeed, the benefits in terms of efficiency and safety make automatic control inevitable for almost all technical projects. Moreover, the improvement in computational capability and formal calculus allow implementation of more and more sophisticated control laws. Thanks to these improvements, complex systems that were once too demanding are now starting to be satisfactorily addressed. One can mention for instance underactuated mechanical systems, nonholonomic systems and systems with strong dynamic couplings.

In particular, the control of cranes and nonholonomic robots has gained considerable interest mainly because of the civilian and military industrial needs to achieve fast and accurate transport of goods and equipment. Old and new harbors are now venturing into fully automated systems combining automated trolleys and classical cranes.

The development of efficient automatic controllers can lead to economic benefits and improved performance. For instance, the stabilization of loads that are carried by cranes is tedious, and the lack of truly efficient strategies implies a large economic loss due to the additional time involved in the process. Indeed, most crane operators move the load with the cable almost vertical; only very few of them, probably skilled through many hours of practice, venture to shift the upper trolley in anticipation of the swing and the desired final load position. To a certain extent, they avail themselves of

the crane model based on their observation and experience. Improvement of the work rate requires abandoning the quasi-static approach and introducing a control law that can cope with the dynamic couplings. Hence, the development of control techniques capable of dealing with complex dynamics is of considerable interest.

Only a good knowledge of the physical system can lead to an efficient controller. The control design must take into account as much structural information as possible, in the same way as an experienced crane operator knows his crane perfectly after hours of practice. Such a model-based controller can take advantage of the dynamic capabilities of the system. Moreover, a model-based controller respects the system dynamics, meaning that the control input remains compatible with the structure of the system.

The basic principles raised while addressing specific control aspects of the applications (nonholonomic mobile robots and cranes), have led to a general methodology that can be applied to a wide class of systems.

1.2 State of the Art

Automatic control deals mainly with the design of control laws that impose a desired behavior on a dynamic system. This desired behavior is often associated with the notion of stability, meaning that the system state should be bounded and eventually converge to a desired point. The control paradigm used in this thesis is somewhat different from the classical one, as will be described below.

Classical control strategies [56, 13, 24] enforce a stable behavior while constructing a suitable control input based on the error between the actual state and the final desired point. Considering a free mass with force as input, classical control can be seen as the generalization of the introduction of both a spring and a dashpot so as to stabilize the mass. The spring is responsible for the proportional gain and the dashpot for the derivative term (e.g. PD control).

Another control paradigm exploits the subtle difference between the concepts of stabilizability and controllability. On the one hand, the classical control paradigm is linked to stabilizability as it proposes a map between the state and inputs of the system, so that the state converges to the desired point. Indeed, the spring uses the position, and the dashpot the velocity, to generate a force, the input to the mass. On the other hand, the new paradigm uses the fact that, due to the controllability property, there exists a trajectory that brings the system back to the desired point. Stabilizability

is then achieved by both regenerating the trajectory, which exists thanks to controllability, and tracking locally this reference.

Before defining explicitly this paradigm, an introductory example is useful. Consider a driver trying to park his car. He mentally creates a trajectory that should lead the car to its final position. He then makes sure that the car approaches the trajectory. It is quite unnatural to focus on the parked position, especially when the car is initially far away. Furthermore, if the driver loses his attention due to an unpredictable event, he has to recreate a new mental picture. He has to regenerate a new trajectory compatible with his current position. He will achieve his goal as long as he keeps updating his mental pictures (trajectories) in a meaningful way and steers the car accordingly.

The paradigm proposed in this thesis is a generalization of this idea. It has two components: (i) A trajectory-regeneration process that frequently updates the feasible trajectory ending up at the desired equilibrium point, and (ii) a local controller that ensures appropriate tracking of the trajectory.

Suppose that a sudden perturbation shifts the current state to a quite different value. The current reference trajectory is no longer appropriate for this new state value. Therefore, by guaranteeing a systematic redefinition of trajectories, the domain of attraction can be enlarged. As a result, the tracking controller sees its error strongly reduced by the regeneration mechanism, thus leading to improved stability.

This trajectory regeneration mechanism has been abundantly used in model-based predictive control (MBPC) [2, 5, 14, 15, 29, 30, 67, 74]. Indeed, the MBPC constructs both an optimal trajectory and a control input that leads the system to the desired point. This trajectory construction is done through optimization and is repeated at regular time intervals. Knowledge of the current state together with this regeneration mechanism guarantees a closed-loop operation. However, the main drawback of MBPC is that the optimization step is often time consuming, thus making the control of fast systems difficult. In order to circumvent this challenge, some alternatives have been developed that simplify or avoid the optimization stage.

For instance, in [72], it is proposed to track a path instead of a trajectory, the difference being that a path is parameterized by a curvilinear coordinate, while a trajectory is parametrized by time. Therefore, a path loses the explicit notion of time. The time constraint being absent leads to an easier formulation of the path-regeneration process and to easier disturbance rejection. Indeed, it is not required that the system reaches a given point at a specific time instant; any point on the path is acceptable. This

fact simplifies both the formulation and the solution to path tracking compared with trajectory tracking. As a consequence, the methodology labeled predictive path planning (PPP) [4, 26, 61, 64] has been developed.

It is important to underline that the regeneration of trajectories must compute feasible trajectories. Here, “feasible” should be understood as compatible with the dynamics of the systems and additionally with other external constraints such as speed limitation. In MBPC and PPP, these constraints are taken into account by resolving an optimization problem under constraints. As discussed above, this optimization stage could be difficult and very time consuming. In order to eliminate this optimization stage, it would be very useful to have a way of parametrizing and constructing feasible trajectories. Of course, such a trajectory parametrization for nonlinear systems is neither generic nor straightforward.

However, there exists a class of systems for which such a parametrization is possible and systematic. These systems are called flat systems [33, 34, 32, 58, 54]. The flatness property guarantees the existence of particular functions of the states, (called flat outputs), with the following property: Parametrizing the flat outputs as time functions implies parametrizing all states and inputs as well. This gives a correspondence between the system trajectories and the flat-output trajectories. Therefore, upon specifying a planned trajectory for the flat outputs, the flatness property guarantees the construction of a feedforward input based on the trajectory by simply combining values of the flat outputs and their time derivatives. This property is extremely useful for trajectory planning since the system dynamics are taken into account in a static way, i.e. without having to integrate differential equations.

In order to exploit the powerful parametrization of the flat outputs, the MBPC technique has been tailored to flat systems. This new approach is called flatness-based predictive control FBPC [36, 18, 17, 57]. This way the optimization stage is simplified.

It is important to underline that the flatness property can be seen from a different point of view than simply using a special parametrization to compute the feedforward input. Indeed, it has been established in [35, 31, 12, 58, 54] that every flat system is exactly linearizable by dynamical feedback. This leads to dynamical feedback linearization DFL, where a nonlinear system is transformed into a trivial linear system through dynamical feedback. The resulting linear system can then be stabilized with classical state feedback (e.g. pole-placement, LQR). DFL is an extension of exact feedback linearization [60, 42] that introduces a dedicated dynamical

extension. Contrary to the extension used in flatness-based linearization, there is no structure in the dynamical extension. Flatness induces endogenous dynamical extension. However, it has been recently shown that DFL and linearization using an endogenous transformation are equivalent [55].

The DFL methodology can impose the cancellation of nonlinearities in order to linearize the system. This cancellation is done using the system model. However, this model is often inaccurate, thus inducing an inexact cancellation of nonlinearities. This imperfect cancellation can introduce some unstable dynamics leading to a loss of stability of the closed-loop system.

In order to deal with this drawback, the exact feedforward linearization (FFL) based on differential flatness has been proposed in [39]. The key idea of this methodology is to compensate the nonlinearities through feedforward instead of feedback. In this way, the imperfect knowledge of the system does not introduce unstable dynamics, but just an exogenous perturbation. This leads to a more robust behavior of the controlled system. However, FFL is only capable of linearizing perfectly the nonlinear system when the system state is on the reference trajectory. Therefore, FFL guarantees the stability only in the neighborhood of the reference trajectory (local stability), while DFL guarantees global stability.

Most of the generic techniques presented above have been applied to nonholonomic mobile robots and cranes.

MBPC [23, 52, 79] has been extensively used for mobile robot control with good results, in particular in obstacle avoidance. Even if the kinematic model of the mobile robot is quite simple, the optimization stage is still not easy and can be very time consuming. This limitation could be a problem when fast mobile-robot displacements are required. However, the main advantage of MBPC is that it provides a framework in which a variety of control objectives and operation constraints can be accommodated. The PPP methodology has also been frequently explored in the mobile-robot context ([72, 4, 26, 61, 64]). This alternative to MBPC can be used as long as the robot does not need to be in a specific location at a specific time as in the interception problem.

Another property of the nonholonomic mobile robot that has interested the community is differential flatness. Indeed, it can be shown that the nonholonomic mobile robot is a flat system. This has led to the construction of feedforward control based on flatness ([10, 33, 68]). This property allows computing the feedforward input rapidly since the relation between robot

trajectory and input is static. However, the main drawback is that this technique relies essentially on an open-loop controller that does not handle external perturbations and model mismatch.

As mentioned above, the flatness property guarantees that a DFL controller can be designed, and such a controller has been developed for the nonholonomic mobile robot in [65]. As will be shown in Chapter 3, this controller works well in general but can encounter difficulties in the presence of a constant perturbation on the rotational axis.

Regarding crane control, the research work using MBPC ([7, 48, 59]) is not so extensive as that for mobile robot control. The reason is that the crane model consists of highly nonlinear equations that lead to a complex optimization stage as illustrated in [59]. The papers proposing an MBPC approach for cranes deal essentially with the minimization of the load sway and do not consider the case of trajectory tracking.

Meanwhile, some analytical methods that do not require optimization have been developed. In particular, [49] defines a class of systems called “the class of cranes”. This definition gives a general framework that includes all the classical cranes like the overhead crane, the cantilever, and the US-Navy crane. Both the dynamic equations and the flatness property of “the class of cranes” are presented and, thanks to the flatness property, an open-loop controller able to compute the feedforward input compatible with the desired trajectory is given.

In [50], a closed-loop flatness-based control scheme is proposed. This controller consists of a PD controller using only motor position measurements and guarantees global stability at an equilibrium point. However, only local stability in trajectory tracking is achieved.

A comprehensive state of the art for nonholonomic mobile robots and cranes will be given in Chapter 4 and 5, respectively.

1.3 Objectives and Organization of the Thesis

The main issue regarding trajectory regeneration is how to regenerate the trajectories in order to reach the desired objective. The available solutions are essentially based on optimization (MBPC), the essential drawback of which is the computation time.

An interesting alternative is a flatness-based controller as proposed in [50]. However, [50] does not address how the trajectories should be regenerated since it is proposed to track a given trajectory locally.

The main objective of this thesis is to develop a general methodology based on regenerating trajectories without any optimization stage. In order to reach this objective, the flatness property is used, as in [50].

The point of view adopted in this thesis is that, instead of specifying a trajectory and tracking it explicitly, a dynamical system called “jet scheduler” provides the derivatives (the jets) of an ideal stabilizing trajectory. These jets are updated regularly according to state measurements so as to react to unknown disturbances. The proposed controller can be seen as an extension of [50] that achieves a wider domain of attraction at the cost of requiring full-state measurement.

The flat correspondence is used to provide the values of the jets, and a subsidiary controller is designed to ensure that these jets are really matched asymptotically by the true system. Unfortunately, both of these mechanisms could possibly break the equivalence between the original nonlinear system and the linear extended system (contrary to the DFL approach for which this correspondence is guaranteed at every time instant).

Chapter 2 will present preliminary material concerning stability issues and flatness. The implication of the possible loss of correspondence, together with the full details on how the scheduling mechanism is achieved, will be covered in the methodological chapter of the thesis (Chapter 3). Chapter 4 applies the general formalism to a mobile robots. The specific properties of these systems are used to achieve a rigorous convergence proof. Chapter 5 addresses the application of the general formalism to cranes, in particular to SpiderCrane, a new design that takes full advantage of the jet-scheduling mechanism. Both the nonholonomic and crane controllers are tested on real setups. Finally, conclusions and perspectives are given in Chapter 6.

Chapter 2

Preliminaries

This thesis addresses a new methodology for the control of flat systems [58, 54, 32]. It is based on some fundamental concepts that will be recalled briefly in this chapter.

First, the general definition of flat systems will be given. The specific notations used throughout the thesis will also be set. The definition uses the language of differential geometry in infinite dimension. However, both the systems to be controlled and the dynamical extension needed to transform the system into a linear equivalent system are finite dimensional. Therefore, care will be taken to convert the infinite dimensional setting into a finite dimensional one. Moreover, the exposition will underline the specific structural properties associated with flatness.

Second, particular stability properties and definitions pertaining to the stability analysis to be attempted will be recalled. This is important since, in contrast to classical-flatness based approaches, the jet-scheduling control breaks the exact equivalence with a linear system. This means that normal stability issues that are trivial in the classical usage of flatness become more cumbersome using the jet-scheduling methodology. Extra dedicated mathematical machinery is needed to overcome the obstacles linked with convergence and stability properties. Classical stability concepts such as Lyapunov stability will be recalled and specific formulations required in the forthcoming chapters presented.

2.1 Lie-Bäcklund Equivalence and Flatness

2.1.1 System definition

The classical definition of a dynamical system is generalized so as to give a natural framework for the notion of equivalence between systems and flatness.

In traditional control theory, a dynamical system is represented by the set of differential equations

$$\dot{x} = f(x, u) \quad (2.1)$$

where f is smooth on the open subset $X \times U \subset \mathbb{R}^n \times \mathbb{R}^m$. The collection of variables x is called the state of the system and the collection of inputs is given by u . Although u stands for a collection of m different inputs u_i , $i = 1, \dots, m$, it will be called the input u .

From a geometrical point of view, $f(x, u)$ is a vector field for a fixed value of the input u . Therefore, if the input is not specified, f can be considered as an infinite collection of vector fields on X parametrized by the input u .

The solution of (2.1) is a mapping $t \rightarrow (x(t), u(t))$ with values in $X \times U$ such that

$$\dot{x}(t) = f(x(t), u(t)) \quad \forall t \geq 0 \quad (2.2)$$

Instead of this description [58, 54, 32], we can consider the infinite mapping

$$t \rightarrow \xi(t) = (x(t), u(t), \dot{u}(t), \ddot{u}(t), \dots)$$

taking values in $X \times U \times \mathbb{R}_m^\infty$, where $\mathbb{R}_m^\infty = \mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}^m \times \dots$ is a countable infinite product of copies of \mathbb{R}^m . Naturally, this mapping satisfies

$$\dot{\xi}(t) = (f(x(t), u(t)), \dot{u}(t), \ddot{u}(t), \dots) \quad \forall t \geq 0. \quad (2.3)$$

Equation (2.3) can be seen as a system of infinite dimension

$$\dot{\xi}(t) = F(\xi(t)) = (f(x(t), u(t)), \dot{u}(t), \ddot{u}(t), \dots) \quad (2.4)$$

where F is an infinite vector field on $X \times U \times \mathbb{R}_m^\infty$. All the components of F (i.e. F_i , $i = 1, \dots, \infty$) depend only on a finite number of coordinates.

The validity of this new interpretation of the differential equations (2.2) can be seen through the preservation of its solutions. Indeed, the solutions $(x(t), u(t))$ of (2.2) and those $(x(t), u(t), \dot{u}(t), \ddot{u}(t), \dots)$ of (2.4) are the same in the sense that the first two components are equal $\forall t \geq 0$.

The above considerations lead to the following definition of a system:

Definition 2.1 A dynamical system is a pair (Ω, F) where Ω is a smooth manifold, possibly of infinite dimension, and F is a smooth vector field on Ω , which is a prolongation of a finite dimensional vector field.

It is important to underline that, in Definition 2.1, the notion of state dimension is lost. Indeed, using Definition 2.1, both systems

$$\dot{x} = f(x, u) \quad (2.5)$$

and

$$\dot{x} = f(x, u) \quad (2.6)$$

$$\dot{u} = v \quad (2.7)$$

have the same description $(X \times U \times \mathbb{R}_m^\infty, F)$.

2.1.2 Lie-Bäcklund equivalence

The notion of equivalence between two systems is now addressed [58, 54, 32]. This can be understood in the sense that the trajectories of both systems can be exchanged by an invertible transformation.

This notion is explained in a formal way considering two systems (Ω, F) and (Π, G) and a smooth mapping $\psi : \Omega \rightarrow \Pi$. If $\xi(t)$ is a trajectory of (Ω, F) , i.e.

$$\dot{\xi}(t) = F(\xi(t)),$$

the composed mapping $v = \psi(\xi(t))$ satisfies the chain rule

$$\dot{v} = \frac{\partial \psi}{\partial \xi}(\xi(t))F(\xi(t))$$

The vector fields F and G are called ψ -related if

$$G(\psi(\xi)) = \frac{\partial \psi}{\partial \xi}F(\xi).$$

Assuming that F and G are ψ -related, it follows that

$$\dot{v} = G(\psi(\xi(t))) = G(v(t))$$

which means $v(t) = \psi(\xi(t))$ is a trajectory of (Π, G) . Moreover, if ψ has a smooth inverse φ , F and G are called φ -related and there is a one-to-one correspondence between the trajectories of both systems. An invertible ψ relation between F and G is called an endogeneous transformation.

Definition 2.2 *Two systems (Ω, F) and (Π, G) are Lie-Bäcklund equivalent if there exists an endogeneous transformation from Ω to Π .*

2.1.3 Flatness

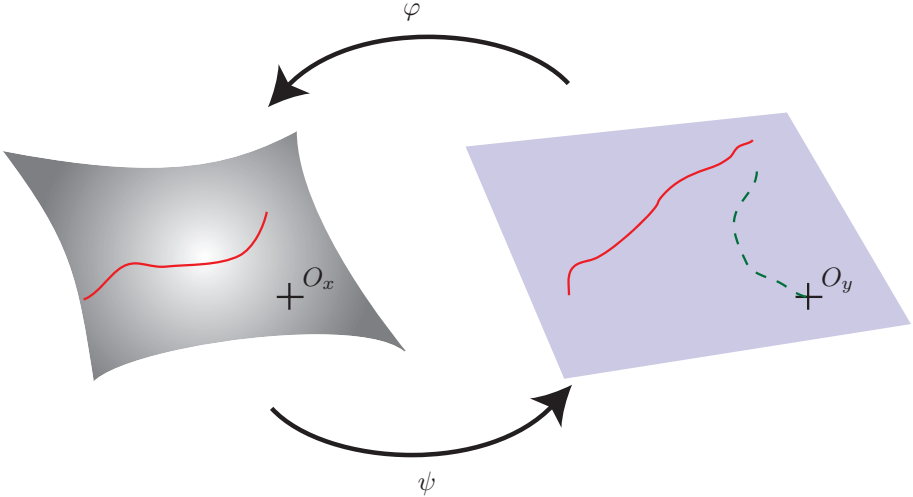


Figure 2.1: Lie-Bäcklund equivalence between two representations of the same flat system. The left-hand figure represents the system using the original coordinates, while the right-hand figure represents the same system using special coordinates. The mappings φ and ψ give the correspondences between these representations. A trajectory is represented as a solid line. It depends on the representation chosen as illustrated in the figure. The dashed line illustrates a convergent trajectory. It does not necessarily start from a point on the solid line.

In this section, the notion of a flat system is addressed. A flat system is a system that is equivalent to the trivial system composed of m chains of integrators. This trivial system is defined as follows:

Definition 2.3 *The trivial system is the pair $(\mathbb{R}_m^\infty, F_m)$ with the coordinates $(y, \dot{y}, \ddot{y}, \dots)$ and the vector field*

$$F_m(y, \dot{y}, \ddot{y}, \dots) = (\dot{y}, \ddot{y}, y^{(3)}, \dots)$$

The definition of flatness follows directly:

Definition 2.4 *The system (Ω, F) is flat if it is equivalent to the trivial system $(\mathbb{R}_m^\infty, F_m)$; y is called flat output.*

In particular, this definition can be applied to a system $(X \times U \times \mathbb{R}_m^\infty, F)$. Moreover, it is proved in [58, 54] that the dimension of y is equal to the dimension of u .

Remark 2.1 *Since a flat system is equivalent to the trivial system that is linear, one can consider a flat system as a generalization of linear systems even though the system may be quite nonlinear in almost any natural representation.*

Since a flat system $(X \times U \times \mathbb{R}_m^\infty, F)$ is equivalent to the trivial system $(\mathbb{R}_m^\infty, F_m)$, it follows that the endogeneous transformation ψ takes the form

$$\psi(x, u, \dot{u}, \ddot{u}, \dots) = (y, \dot{y}, \ddot{y}, \dots)$$

and the transformation φ the form

$$\varphi(y, \dot{y}, \ddot{y}, \dots) = (x, u, \dot{u}, \ddot{u}, \dots),$$

where φ is the inverse of ψ .

Such an equivalence allows one to analyze and synthesize control laws for the system $(X \times U \times \mathbb{R}_\infty^m, F)$ using the system $(\mathbb{R}_\infty^m, F_m)$. In fact, $(\mathbb{R}_\infty^m, F_m)$ is usually more tractable than $(X \times U \times \mathbb{R}_\infty^m, F)$ thanks to its structure. Some aspects can be defined for $(\mathbb{R}_\infty^m, F_m)$ and then extended to $(X \times U \times \mathbb{R}_\infty^m, F)$ using the Lie-Bäcklund equivalence. In particular:

Definition 2.5 *Origin. The origin O_y of the manifold (\mathbb{R}_∞^m) is the point $O_y = (0, 0, 0, \dots)$. Moreover, the origin O_x of the manifold $(X \times U \times \mathbb{R}_\infty^m)$ is $O_x = \varphi(O_y)$.*

Definition 2.6 *Convergent Trajectory. A trajectory $\eta(t)$ on (\mathbb{R}_∞^m) is called convergent if $\lim_{t \rightarrow \infty} \eta(t) = O_y$.*

In Definition 2.6 the limit is taken in the usual Euclidian finite dimensional sense once we admit that there exists a sufficiently large finite integer k' such that $\eta^{(k)} = 0 \quad \forall k > k'$.

Figure 2.1 illustrates the correspondences φ and ψ . The origin O_x and O_y are also represented together with a convergent trajectory (dashed line).

Without loss of generality, the flatness property can be expressed also in finite dimension.

Definition 2.7 *The system*

$$\dot{x} = f(x, u), \quad x \in X = (x_1, \dots, x_n), \quad u \in U = (u_1, \dots, u_m) \quad (2.8)$$

is flat if and only if there exist (i) a flat output $y \in \mathbb{R}^m$, (ii) two integers r and s , and (iii) mappings ψ from $(X \times (\mathbb{R}^m)^{s+1})$ on \mathbb{R}^m and (φ_x, φ_u) from $\mathbb{R}^{m(r+2)}$ on $\mathbb{R}^n \times \mathbb{R}^m$ such that

$$y = \psi(x, u, \dot{u}, \dots, u^{(s)}) \quad (2.9)$$

$$x = \varphi_x(y, \dot{y}, \dots, y^{(r)}) \quad (2.10)$$

$$u = \varphi_u(y, \dot{y}, \dots, y^{(r+1)}) \quad (2.11)$$

Remark 2.2 *The flatness property in infinite dimension establishes a diffeomorphic correspondence between two systems of infinite dimension. However, the flatness expressed in finite dimension loses this property of diffeomorphic correspondence, meaning that expressions (2.9)-(2.11) are not invertible.*

Notice that, in order to be more precise, the expressions (2.10) and (2.11) should be written with the following multi-index notation

$$x = \varphi_x(y_1, \dot{y}_1, \dots, y_1^{(r_1)}, \dots, y_m, \dot{y}_m, \dots, y_m^{(r_m)}) \quad (2.12)$$

$$u = \varphi_u(y_1, \dot{y}_1, \dots, y_1^{(r_1+1)}, \dots, y_m, \dot{y}_m, \dots, y_m^{(r_m+1)}), \quad (2.13)$$

which means that the state x and the input u can be represented by chains of integrators, which are not necessarily of the same size (i.e. r_1, \dots, r_m are not necessarily equal). In the expressions (2.10) and (2.11), the parameter r is equal to $\max(r_1, \dots, r_m)$

2.1.4 Application to open-loop control

Controlling system (2.1) in its natural coordinates may be very difficult because it is multivariable, nonlinear, and potentially contains strong coupling terms. However, as shown in Section 2.1.3, whenever the system (2.1) is flat, it can be represented as a collection of chains of integrators. This means that, through the Lie-Bäcklund equivalence, all the nonlinearities and the coupling terms are compensated. It is shown next how the flatness property can be used for solving the open-loop control problem.

The problem is addressed by the manner in which a feedforward input is computed so as to follow a reference trajectory. A pre-specified sufficiently smooth trajectory y_{ref} must be followed through the computation of a suitable open-loop input. Therefore, in order to follow this trajectory, Equation (2.11) is used. Indeed, since the input u is a function of the flat output y and its derivatives $\dot{y}, \ddot{y}, \dots, y^{(r+1)}$, the reference y_{ref} and its derivatives $\dot{y}_{ref}, \ddot{y}_{ref}, \dots, y_{ref}^{(r+1)}$ are inserted in Equation (2.11) to generate the feedforward input

$$u_f = \varphi_u(y_{ref}, \dot{y}_{ref}, \dots, y_{ref}^{(r+1)}).$$

Consequently, specifying the flat output trajectories and using the Lie-Bäcklund equivalence, the system (2.1) can be controlled while compensating all the nonlinearities and coupling terms.

2.1.5 Dynamical feedback linearization

The control strategy exposed in Section 2.1.4 uses an open-loop point of view and, consequently, the methodology does not lead to a controller that can reject perturbations satisfactorily. For this reason, a closed-loop control strategy is needed. A very popular idea, called dynamic feedback linearization, consists in interpreting equivalence in terms of feedback [12, 32, 58, 54].

First, the notion of dynamical feedback is recalled. Considering the system

$$\dot{x} = f(x, u), \tag{2.14}$$

a dynamical feedback for this system is defined as

$$\dot{\gamma} = \beta(x, \gamma, w) \tag{2.15}$$

$$u = \alpha(x, \gamma, w), \tag{2.16}$$

where w is a new input. This way, the closed-loop system becomes

$$\dot{x} = f(x, \alpha(x, \gamma, w)) \tag{2.17}$$

$$\dot{\gamma} = \beta(z, \gamma, w). \tag{2.18}$$

As shown in Section 2.1.1, the system (2.14) can be represented in infinite-dimension as $(X \times U \times \mathbb{R}_m^\infty, F)$. In the same way, the closed-loop system (2.17) allows the infinite dimensional representation $(\tilde{X} \times W \times \mathbb{R}_m^\infty, \tilde{F})$.

Normally, the systems $(X \times U \times \mathbb{R}_m^\infty, F)$ and $(\tilde{X} \times W \times \mathbb{R}_m^\infty, \tilde{F})$ are not Lie-Bäcklund equivalent. However, there exists a particular dynamical feedback called endogeneous dynamical feedback, which does not modify the infinite-dimensional representation of (2.14). Such a feedback is defined hereafter:

Definition 2.8 *A dynamical feedback*

$$\dot{\gamma} = \beta(x, \gamma, w) \quad (2.19)$$

$$u = \alpha(x, \gamma, w) \quad (2.20)$$

is called endogeneous if the closed-loop system

$$\begin{aligned} \dot{x} &= f(x, \alpha(x, \gamma, w)) \\ \dot{\gamma} &= \beta(x, \gamma, w) \end{aligned} \quad (2.21)$$

is Lie-Bäcklund equivalent to the system $(X \times U \times \mathbb{R}_m^\infty, F)$.

This notion is particularly important because, through an endogeneous dynamical feedback, the infinite representation of the resulting closed-loop system always remains Lie-Bäcklund equivalent to the *same system* $(X \times U \times \mathbb{R}_m^\infty, F)$.

In Section 2.1.3, it has been shown that the system $\dot{x} = f(x, u)$ is flat if its infinite-dimension representation $(X \times U \times \mathbb{R}_m^\infty, F)$ is Lie-Bäcklund equivalent to the trivial system $(\mathbb{R}_m^\infty, F_m)$. It is interesting to notice that, by definition, the flat system $\dot{x} = f(x, u)$ together with an endogeneous dynamical feedback is also represented in infinite dimension by $(X \times U \times \mathbb{R}_m^\infty, F)$ that is equivalent to the trivial system $(\mathbb{R}_m^\infty, F_m)$. This means that the property of flatness is preserved by endogeneous dynamical feedback.

As stated in Definition 2.7, the flat equivalence can be expressed also in finite dimension but, in this case, the expression of equivalence (2.10) is not necessarily diffeomorphic. However, we will assume that the closed-loop system (2.21) is flat and moreover in diffeomorphic correspondence with the finite-dimensional trivial system

$$y^{(r+1)} = v,$$

where v is the input of the trivial system. This means that there exist two invertible functions $\varphi_{x\gamma}$ and φ_w such that

$$(x, \gamma) = \varphi_{x\gamma}(y, \dot{y}, \dots, y^{(r)}) \quad (2.22)$$

$$w = \varphi_w(y, \dot{y}, \dots, y^{(r+1)}) \quad (2.23)$$

This guarantees the existence of a one-to-one correspondence between the extended state (x, γ) and $(y, \dot{y}, \dots, y^{(r)})$. This means that, knowing (x, γ) , the state of the trivial system is perfectly known as well. The stabilization of the trivial system

$$y^{(r+1)} = v \tag{2.24}$$

is considered first. Since the trivial system (2.24) is linear and composed of pure integrator chains, the input v is chosen as

$$v = K(y, \dot{y}, \dots, y^{(r)}),$$

with a suitable matrix gain K that guarantees a stable closed-loop behavior.

Expressing the above differential equation in the equivalent representation associated with (2.21), w can be rewritten using (2.22), (2.23) and (2.24) as

$$w = \varphi_w(\varphi_{x\gamma}^{-1}(x, \gamma), K\varphi_{x\gamma}^{-1}(x, \gamma)).$$

This input makes the closed-loop system

$$\begin{aligned} \dot{x} &= f(x, \alpha(x, \gamma, w)) \\ \dot{\gamma} &= \beta(x, \gamma, w) \end{aligned}$$

equivalent to the stable linear system

$$y^{(r+1)} = K_1 y + K_2 \dot{y} + \dots + K_{r-1} y^{(r)}.$$

Here, equivalence is understood as the existence of a finite-dimensional diffeomorphism between $(y, \dot{y}, \dots, y^{(r)})$ and (x, γ) .

Remark 2.3 *Linearization of the system $\dot{x} = f(x, u)$ by feedback without dynamical extension is usually not possible because the correspondence between the state x and $(y, \dot{y}, \dots, y^{(r)})$ is not necessarily a one-to-one correspondence. On the other hand, thanks to the dynamical feedback, the new state γ enforces a one-to-one correspondence between the extended state (x, γ) and $(y, \dot{y}, \dots, y^{(r)})$. This is the reason for using a dynamical extension.*

Remark 2.4 *Naturally, this methodology can be used for tracking a sufficiently smooth pre-specified trajectory y_{ref} . In this case, the input v of the trivial system becomes*

$$v = K_1(y - y_{ref}) + K_2(\dot{y} - \dot{y}_{ref}) + \dots + K_{r-1}(y^{(r)} - y_{ref}^{(r)}) + y_{ref}^{(r+1)},$$

imposing thereby stable closed-loop error dynamics.

It is shown in [58, 54] that the dynamics are flat if and only if they are linearizable by endogeneous feedback and coordinate change. This means that this technique can be applied to every flat system.

2.2 Lyapunov Stability

As described in the introduction, the purpose of jet-scheduling control is to achieve stability through instantaneous generation of feasible reference trajectories. Therefore, it is important to recall the concept of stability [47, 77, 42]. We first recall the notion of stability of an equilibrium point in the sense of Lyapunov. Then, a technique not involving the explicit solution of the differential equations is given, which depends on the definition of a Lyapunov function.

2.2.1 Stability

We consider the dynamical system

$$\dot{x} = f(x) \quad x(0) = x_0, \quad (2.25)$$

where f is smooth on the open subset $X \subset \mathbb{R}^n$. Here, contrary to (2.1) there is no input to the system. Normally, the input is set using $u = k(x)$ (i.e. using a feedback) so that $\dot{x} = f(x, u)$ reads $\dot{x} = f(x, k(x))$, that is $\dot{x} = \tilde{f}(x)$. In this section, we simply use $f(x)$ instead of $\tilde{f}(x)$ so as to simplify the notation. It is also assumed (without loss of generality) that the origin is an equilibrium point, i.e. $0 = f(0)$.

The notion of stability of an equilibrium point is characterized by the behavior of the solution $x(t)$ of system (2.25) away from its equilibrium point.

Definition 2.9 *The system given by the ordinary differential equations (2.25) is stable if $\forall R > 0, \exists r > 0$ such that $\|x_0\| < r$ implies $\|x(t)\| < R$.*

As illustrated in Figure 2.2, this definition means that, whatever the size R of a chosen ball, it is always possible to find another ball of size r such that, for all the initial conditions comprised in the ball of size r , the resulting trajectory $x(t)$ remains within the ball of size R .

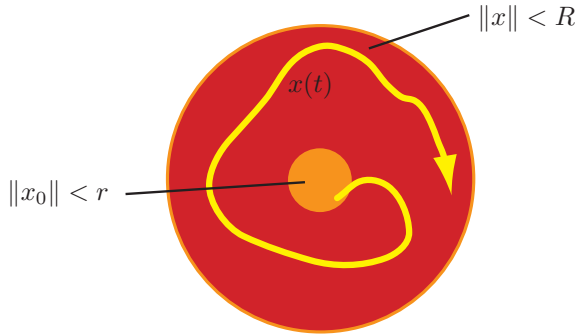


Figure 2.2: Illustration of the stability definition. For every chosen ball of size R , it is possible to find a ball of initial conditions $\|x_0\| < r$ such that, for all trajectories $x(t)$ that start in this ball, $x(t)$ remains in the ball of size $\|x\| < R$.

The stability of a system is often not sufficient, since it precludes the possibility of knowing exactly where the solutions of the system will converge. It would be more convenient if the definition could guarantee that the trajectory $x(t)$ of the system goes back to its equilibrium point $x = 0$.

Definition 2.10 *The equilibrium point $x = 0$ of the system (2.25) is asymptotically stable if*

1. *the system is stable*
2. *there exists a ball of initial conditions $\|x_0\| < r_0$ such that the resulting trajectories $x(t) \rightarrow 0$ when $t \rightarrow \infty$.*

Moreover, an asymptotically stable system is called exponentially stable if $\exists \alpha > 0$ and $\exists \lambda > 0$ such that

$$\|x(t)\| \leq \alpha \|x(0)\| e^{-\lambda t}, \quad \forall t > 0$$

It is important to notice that a system may be convergent, i.e. $x(t) \rightarrow 0$ when $t \rightarrow \infty$, without being stable. This means that it is impossible to dominate the transient behavior of the resulting trajectory, even if the initial conditions are chosen arbitrarily close to the equilibrium point. This is illustrated by the following example [77].

Example 2.1

$$\begin{aligned}\dot{x}_1 &= \frac{x_1^2(x_2 - x_1) + x_2^5}{(x_1^2 + x_2^2)[1 + (x_1^2 + x_2^2)^2]} \\ \dot{x}_2 &= \frac{x_2^2(x_2 - 2x_1)}{(x_1^2 + x_2^2)[1 + (x_1^2 + x_2^2)^2]}.\end{aligned}$$

In Figure 2.3, the trajectories begin close to the origin, move away and then return to the origin following the horizontal axis. This system is unstable since the trajectories cannot be constrained to stay within a given ball (of sufficiently small radius R) even when the radius r of the ball of initial conditions is reduced to an infinitesimal but nonzero value. Indeed, the more the initial condition is close to 0 (i.e. $x_1(0) > 0$ and $x_2(0) > 0$ with both $x_1(0) \rightarrow 0^+$ and $x_2(0) \rightarrow 0^+$) the larger $\sup_{t \in (0; \infty)} x(t)$ becomes, which shows that there exists a given ball of radius R for which no matter how small the ball of initial condition is chosen (i.e. r) there will always exist at least one initial condition for which the solution leaves the ball of radius R .

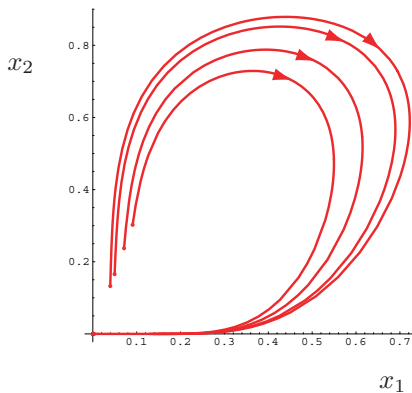


Figure 2.3: Convergent but unstable system.

2.2.2 Lyapunov theorem

The notion of stability defined in the previous section is based on the solution $x(t)$ of the system of differential equations (2.25). However, it is

not easy to get a closed-form solution, since the integration of a nonlinear system may be very difficult or impossible to obtain analytically. It is also possible to compute this solution by numerical integration, but in this case, the solution must be evaluated for every initial condition. For this reason, some results leading to stability assessment without integrating the system (2.25) are presented hereafter.

Lyapunov stability rests on a procedure that replaces the condition appearing in the stability definition by a simple statement concerning a scalar-valued function. In a sense, it is a systematic description of the balls using the scalar-valued function which subsumes the ideas involved in the definition. Indeed, a ball is simply replaced by a suitable closed-level set of the Lyapunov function.

The formal definition of a Lyapunov function candidate is given next.

Definition 2.11 (*Lyapunov candidate*) *A function $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ is called a Lyapunov candidate if*

1. $V(x)$ is positive definite
2. $V(x)$ is of Class- C^1 .

Establishing stability consists in simply checking the decrease or conservation of this Lyapunov function. This guarantees that the trajectories do not escape the initial level set given by the initial value of the Lyapunov function, i.e. the initial ball of radius R .

Theorem 2.1 *A system is stable if there exists a Lyapunov candidate $V(x)$ such that*

$$\dot{V}(x) = \left(\frac{\partial V}{\partial x} \right)^T f(x) = L_f V \leq 0.$$

Moreover, a system is asymptotically stable if

$$\dot{V}(x) = \left(\frac{\partial V}{\partial x} \right)^T f(x) = L_f V < 0, \forall x \neq 0$$

.

Additionally, another theorem enables one to establish the exponential stability thanks to more restrictive conditions on $V(x)$. It is given below.

Theorem 2.2 *A system is exponentially stable if there exists a Lyapunov candidate $V(x)$ such that*

$$\begin{aligned}\alpha_1 \|x\|^c &\leq V(x) \leq \alpha_2 \|x\|^c \\ \dot{V}(x) &\leq -\alpha_3 \|x\|^c\end{aligned}$$

for some positive constants $\alpha_1, \alpha_2, \alpha_3, c \geq 2$.

A particular application of Lyapunov theorem concerns linear systems of the form

$$\dot{x} = Ax, \quad x(0) = x_0.$$

In this case, a Lyapunov candidate $V(x)$ is chosen as

$$V(x) = x^T P x$$

where P is a symmetric strictly positive definite matrix. This way, $V(x)$ is naturally positive definite and continuous. Then, differentiating $V(x)$ gives

$$\dot{V}(x) = x^T (A^T P + P A)x.$$

Using Theorem 2.1 to establish asymptotic stability of the linear system imposes $\dot{V}(x)$ be strictly negative definite. This means that

$$\dot{V}(x) = x^T (A^T P + P A)x = -x^T Q x,$$

where Q is strictly positive definite. The condition of existence of such a matrix P is given in the following theorem.

Theorem 2.3 *Given a stable linear system $\dot{x} = Ax$ (i.e. $\forall \lambda, \operatorname{Re}(\lambda) < 0$) and for any positive definite matrix $Q > 0$, there exists a positive definite matrix $P > 0$ such that*

$$A^T P + P A = -Q \tag{2.26}$$

From Theorems 2.1 and 2.2, it is possible to establish stability by constructing a Lyapunov function.

It is often desirable to take advantage of the knowledge of the existence of a Lyapunov function without explicitly computing it. It relies on the fact that the system is known to be stable from the outset. This sounds a bit strange but it is very useful in the analysis of connected systems,

mainly because the Lyapunov functions of the subparts can be used to assess the stability of the whole system without necessarily knowing exactly the structure of the respective Lyapunov functions.

Therefore, one can use what is called the converse Lyapunov theorem [47], which guarantees the existence of a Lyapunov function associated with a stable system.

Theorem 2.4 *If a system is exponentially stable, there exists a positive definite and proper function V such that*

$$\begin{aligned} \alpha_1 \|x\|^2 &\leq V(x) \leq \alpha_2 \|x\|^2, \\ \frac{\partial V}{\partial x} f(x) &\leq -\alpha_3 \|x\|^2, \\ \left\| \frac{\partial V}{\partial x}(x) \right\| &\leq \alpha_4 \|x\| \end{aligned} \tag{2.27}$$

for some positive constants $\alpha_1, \alpha_2, \alpha_3, \alpha_4$.

2.2.3 Vanishing perturbed system

As described in the introduction, the purpose of jet-scheduling control is to achieve stability through the instantaneous generation of feasible reference trajectories in a continuous way. This idea leads to an asymptotic linearization meaning that, through the use of the jet-scheduling controller, the nonlinear system becomes equivalent to a linear one, once intermediate variables have converged. These intermediate variables lead to a split in the initial state space.

We will see throughout the thesis how this split is done in practice and how the controllers are designed to achieve such a separation. Nevertheless, the necessary preliminaries, which involve established techniques available for naturally separated dynamical systems, are recalled here.

The state has two components, one labeled Υ whose evolution depends on its current value and also on the second set of variables labeled Ξ , and the second component Ξ which behaves according to its own dynamics without being influenced by the first set Υ :

$$\begin{aligned} \dot{\Upsilon} &= F(\Upsilon, \Xi) \\ \dot{\Xi} &= G(\Xi). \end{aligned} \tag{2.28}$$

Two assumptions are introduced

- $\dot{\Xi} = G(\Xi)$ is asymptotically stable
- $F(\Upsilon, 0) = A\Upsilon$, where A is Hurwitz.

The stability of system (2.28) is studied by Lyapunov Theorem. Since Ξ goes to 0 asymptotically, $F(\Upsilon, \Xi)$ becomes $F(\Upsilon, 0) = A\Upsilon$ asymptotically. Based on this observation, the linear part is factorized as

$$\dot{\Upsilon} = F(\Upsilon, 0) + (F(\Upsilon, \Xi) - F(\Upsilon, 0)) = A\Upsilon + (F(\Upsilon, \Xi) - F(\Upsilon, 0))$$

Now, choosing a symmetric positive definite matrix Q , together with the fact that A is Hurwitz by assumption, the Lyapunov equation

$$PA + A^T P = -Q$$

has a solution (cf. Theorem 2.3). Because $V(\Upsilon) = \Upsilon^T P \Upsilon$ is a quadratic function stemming from a positive definite matrix P , it follows that

$$\begin{aligned} \lambda_{\min}(P)\|\Upsilon\|^2 &\leq V(\Upsilon) \leq \lambda_{\max}(P)\|\Upsilon\|^2 \\ \frac{\partial V}{\partial \Upsilon} A\Upsilon &= -\Upsilon^T Q \Upsilon \leq -\lambda_{\min}(Q)\|\Upsilon\|^2 \\ \left\| \frac{\partial V}{\partial \Upsilon} \right\| &= \|2\Upsilon^T P\| \leq 2\|P\|\|\Upsilon\| = 2\lambda_{\max}(P)\|\Upsilon\|. \end{aligned}$$

The derivative of $V(\Upsilon)$ along the solutions is

$$\begin{aligned} \dot{V}(\Upsilon) &= \frac{\partial V}{\partial \Upsilon} A\Upsilon + \frac{\partial V}{\partial \Upsilon} (F(\Upsilon, \Xi) - F(\Upsilon, 0)) \\ &\leq -\lambda_{\min}(Q)\|\Upsilon\|^2 + 2\lambda_{\max}(P)\|\Upsilon\|\|F(\Upsilon, \Xi) - F(\Upsilon, 0)\| \end{aligned} \quad (2.29)$$

At this point, we notice that $\dot{V}(\Upsilon)$ is bounded by an expression containing a negative part $-\lambda_{\min}(Q)\|\Upsilon\|^2$, which corresponds to the linear behavior of the system, and by the positive term $2\lambda_{\max}(P)\|\Upsilon\|\|F(\Upsilon, \Xi) - F(\Upsilon, 0)\|$, which is the influence of the nonlinear part. If

$$\|F(\Upsilon, \Xi) - F(\Upsilon, 0)\| < \frac{\lambda_{\min}(Q)\|\Upsilon\|}{2\lambda_{\max}(P)\|\Upsilon\|}, \quad (2.30)$$

then $\dot{V}(\Upsilon)$ is strictly negative and the stability of (2.28) can be established.

When the condition (2.30) is not met, it is possible to develop a milder condition than (2.30), since Ξ is asymptotically stable by definition. We will consider that $\|F(\Upsilon, \Xi) - F(\Upsilon, 0)\|$ can be bounded by $l(\|\Xi\|)\|\Upsilon\|$, where l is of class \mathcal{L} , i.e.

$$\|F(\Upsilon, \Xi) - F(\Upsilon, 0)\| < l(\|\Xi\|)\|\Upsilon\|. \quad (2.31)$$

This additional condition represents the idea that the difference between $F(\Upsilon, \Xi)$ and $F(\Upsilon, 0)$ increases strictly with Ξ and increases in a linear way with Υ . Now introducing (2.31) into (2.29) gives

$$\dot{V}(\Upsilon) \leq (-\lambda_{\min}(Q) + 2\lambda_{\max}(P)l(\|\Xi\|))\|\Upsilon\|^2 \quad (2.32)$$

Consequently, $\dot{V}(\Upsilon) \leq 0$, as long as

$$l(\|\Xi\|) < \frac{\lambda_{\min}(Q)}{2\lambda_{\max}(P)}.$$

Notice that, since Ξ is asymptotically stable, choosing a constant c arbitrarily small, there will always exist a finite time T for which

$$l(\|\Xi\|) < c \quad \forall t \geq T. \quad (2.33)$$

In particular, choosing $c = \frac{\lambda_{\min}(Q)}{2\lambda_{\max}(P)}$ gives

$$l(\|\Xi\|) < \frac{\lambda_{\min}(Q)}{2\lambda_{\max}(P)} \quad \forall t \geq T_c. \quad (2.34)$$

It follows that

$$\dot{V}(\Upsilon) < 0 \quad \forall t \geq T_c. \quad (2.35)$$

Thus (2.28) is stable. Nevertheless, we have to pay attention to the fact that the behavior of (2.28) is not known for $t < T_c$. Thus, it is important to check that there is no finite escape during this initial time interval. For this purpose, the next lemma based on a growth rate condition specifies that the difference between the linear system $F(\Upsilon, 0)$ and the nonlinear system $F(\Upsilon, \Xi)$ must grow in a linear way to ensure no finite escape time in Υ .

Lemma 2.1 *Consider a cascade-connected system of the form*

$$\begin{aligned}\dot{\Upsilon} &= F(\Upsilon, \Xi) \\ \dot{\Xi} &= G(\Xi)\end{aligned}\tag{2.36}$$

with $\Upsilon \in \mathbf{R}^n$ and $\Xi \in \mathbf{R}^m$. Assume that

$$\dot{\Upsilon} = F(\Upsilon, 0)\tag{2.37}$$

is globally exponentially stable at $\Upsilon = 0$ and

$$\dot{\Xi} = G(\Xi)\tag{2.38}$$

is globally stable at $\Xi = 0$. If the growth rate condition

$$\|F(\Upsilon, \Xi) - F(\Upsilon, 0)\| \leq c_1 \|\Upsilon\| + \psi(\|\Xi\|) + c_2\tag{2.39}$$

is satisfied for some constants $c_1, c_2 \geq 0$ and a class \mathcal{K} function ψ , then the cascade system does not exhibit any finite escape time.

Proof: Since the system $\dot{\Upsilon} = F(\Upsilon, 0)$ is globally exponentially stable at $\Upsilon = 0$, from the converse Lyapunov theorem [47], there exists a positive definite and proper function V such that

$$\begin{aligned}\alpha_1 \|\Upsilon\|^2 &\leq V(\Upsilon) \leq \alpha_2 \|\Upsilon\|^2, \\ \frac{\partial V}{\partial \Upsilon} F(\Upsilon, 0) &\leq -\alpha_3 \|\Upsilon\|^2, \\ \left\| \frac{\partial V}{\partial \Upsilon}(\Upsilon) \right\| &\leq \alpha_4 \|\Upsilon\|\end{aligned}\tag{2.40}$$

for some positive constants $\alpha_1, \alpha_2, \alpha_3, \alpha_4$. Then, by hypothesis and by completing the squares,

$$\begin{aligned}\dot{V} &= \frac{d}{dt} V(\Upsilon, (t)) = \frac{\partial V}{\partial \Upsilon} F(\Upsilon, 0) + \\ &\quad \frac{\partial V}{\partial \Upsilon} (F(\Upsilon, \Xi) - F(\Upsilon, 0)) \\ &\leq -\alpha_3 \|\Upsilon\|^2 + \alpha_4 \|\Upsilon\| (c_1 \|\Upsilon\| + \psi(\|\Xi\|) + c_2) \\ &\leq c_4 V + \psi(\|\Xi\|)^2 + c_5\end{aligned}$$

with appropriate nonnegative constants c_4 and c_5 . Now, using the hypothesis that the system $\dot{\Xi} = G(\Xi)$ is globally stable, it follows that $V(\Upsilon, (t))$ does not exhibit any finite escape time. This, in turn, completes the proof of the lemma. \blacksquare

Chapter 3

Jet-Scheduling Control

3.1 Introduction

The previous chapter introduced the necessary concepts (flatness and stability) that will be used to develop the jet-scheduling methodology. This control method relies on a particular usage of the flatness property together with stabilization arguments.

The idea is to keep generating and tracking stabilizing trajectories that lead the system to the origin. The controller is split in two parts to reach this objective.

The first part, called the jet scheduler, provides reference values for the jets (time derivatives) associated with a convergent trajectory. Based on the current measured values of the flat output, the jet scheduler provides reference values for the derivatives of the flat output (the scheduled jets). Using this information, the feedforward part of the input is set to the value corresponding to these jets through the flatness equivalence expression, much in the same way as for the motion planning technique using flatness-based system inversion.

The second part is a low-order controller that ensures that these scheduled jets are eventually reached by the true system.

The combination of these two entities (jet scheduler and low-order controller) guarantees asymptotic convergence of the state for point stabilization and tracking.

This chapter is organized as follows. Section 3.2 describes the general idea of the jet-scheduling controller. The construction of the jet-scheduling controller is detailed in Section 3.3. Section 3.4 gives a sketch of the proof

for the stability of the jet-scheduling controller. Section 3.5 discusses the difference between the jet-scheduling controller and dynamic feedback linearization. A linear system controlled by the jet-scheduling controller illustrates the method in Section 3.6.

3.2 The Concept of Jet-Scheduling Control

In a general way, a dynamical system is described by

$$\dot{x} = f(x, u), \quad x \in X = (x_1, \dots, x_n), u \in U = (u_1, \dots, u_m) \quad (3.1)$$

where x is the state of the system, u_1, \dots, u_m the inputs and f is a parametrized vector field that represents the dynamics.

However, for reasons detailed in Chapter 2, it is useful to describe a flat system in the infinite dimension framework with the coordinates

$$(x_1, \dots, x_n, u_1^{(0)}, \dots, u_m^{(0)}, u_1^{(1)}, \dots, u_m^{(1)}, \dots, u_1^{(k)}, \dots, u_m^{(k)}, \dots, u_1^{(\infty)}, \dots, u_m^{(\infty)})$$

In this case, the system (3.1) is described by

$$F = \begin{pmatrix} f(x, u) \\ \dot{u} \\ \ddot{u} \\ \vdots \\ u^{(\infty)} \end{pmatrix}$$

According to Definition 2.4, $(\dot{x}, \dot{u}, \dots, u^{(\infty)}) = F(x, u, \dots, u^{(\infty)})$ is equivalent to m chains of integrators

$$y^{(\infty)} = v$$

with the coordinates $(y, \dot{y}, \ddot{y}, \dots, y^{(\infty)})$ and $v = (v_1, \dots, v_m)$ as input.

For flat systems, there exists a $r \in \mathbb{N}$ (recall Definition 2.7) such that the equivalence between $(\dot{x}, \dot{u}, \dots, u^{(\infty)}) = F(x, u, \dots, u^{(\infty)})$ and $y^{(\infty)} = v$ can be reduced to finite dimensional relations expressed as:

$$y = \psi(x, u, \dot{u}, \dots, u^{(s)}) \quad (3.2)$$

$$x = \varphi_x(y, \dot{y}, \dots, y^{(r)}) \quad (3.3)$$

$$u = \varphi_u(y, \dot{y}, \dots, y^{(r+1)}) \quad (3.4)$$

The control objective is to bring the system $(\dot{x}, \dot{u}, \dots, u^{(\infty)}) = F(x, u, \dots, u^{(\infty)})$ from given initial condition to O_x . Thanks to the equivalence with the

trivial system $y^{(\infty)} = v$, this problem is equivalent to bringing $y^{(\infty)} = v$, from a compatible initial condition to O_y .

Remark 3.1 *This convergence occurs in the usual Euclidian finite dimensional sense once we admit that there exists a sufficiently large finite integer k' for which the convergence is equivalent to bringing $y^{(k')} = v$ to $(y = 0, \dot{y} = 0, \dots, y^{(k')} = 0)$.*

The usual way to achieve this goal in finite dimension, is to allow the system $y^{(r+1)} = v$ to follow a convergent trajectory. A well-known solution for achieving this goal is a feedforward controller based on flatness depicted in Section 2.1.4. Indeed, for a sufficiently smooth convergent trajectory $\eta(t)$ of y , the feedforward input of the system (3.1) can be computed using (2.11).

Of course, this method requires that the initial conditions $x(0)$ of (3.1) be compatible with the initial conditions $\eta(0)$ of the convergent trajectory. Since this methodology is open-loop, it does not tolerate either perturbations or initial condition mismatches. In order to fill this gap, one possibility

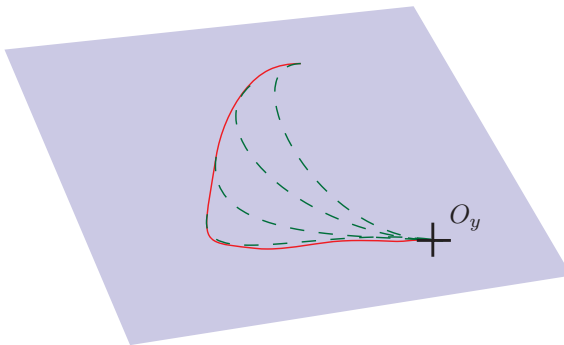


Figure 3.1: Convergent trajectory regeneration. The dashed lines represent different reference trajectories generated to reach O_y , each one starting from a different initial condition. The solid line is the effective trajectory completed by the system. Every converging trajectory has initial conditions compatible with the states of the system. This figure illustrates the effect of a perturbation on the system.

(similar to model predictive control) is to regenerate regularly a new con-

vergent trajectory compatible with the states of the system as illustrated in Figure 3.1.

This solution has some unresolved points: i) How should the convergent trajectory be computed? ii) How should often the convergent trajectory be regenerated? In this chapter, one possible answer to these questions is given by the jet-scheduling controller. In our case, the new trajectory is recomputed at each time instant.

3.3 Construction of Jet-Scheduling Controllers

The jet-scheduling controller consists of two parts. The first part is the jet scheduler that gives the jets to be followed instead of a complete convergent trajectory. This jet is updated continuously based on the measurements of the flat output and its derivatives up to the p th order. The second part is a controller that enforces the actual jet to converge to the desired jet.

Before presenting the general application of the methodology, let us illustrate the idea on a single chain of integrators.

3.3.1 Introductory example

Consider the system composed of three integrators:

$$y^{(3)} = v \tag{3.5}$$

This system can be stabilized using state feedback. In this case, the stabi-

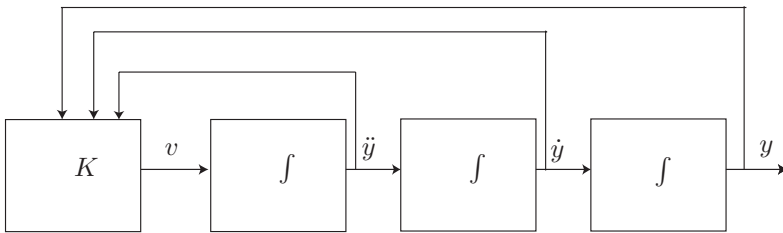


Figure 3.2: Control scheme. Chain of integrators stabilized by a classical state feedback.

lizing input is given by

$$v = k_2 \ddot{y} + k_1 \dot{y} + k_0 y, \tag{3.6}$$

where k_i , $i = 0, 1, 2$ are suitable gains (Figure 3.2). However, the idea developed in this chapter suggests solving the problem of stabilization in a different way. The basic idea is to follow a stabilizing jet instead of computing a stabilizing input based on state feedback. Concretely for the system (3.5), the given jet to be followed, labeled χ , represents the desired velocity that \dot{y} should have to achieve stabilization. In the same way, $\dot{\chi}$ represents the acceleration that \ddot{y} should have. This is represented in the schematic diagram given in Figure 3.3. The role of the input v is to enforce

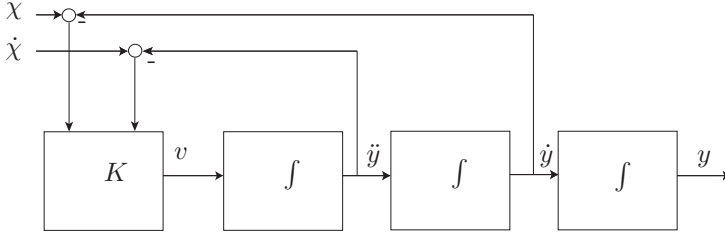


Figure 3.3: Control scheme that guarantees convergence of the jets \dot{y} and \ddot{y} to the scheduled (desired) jets χ and $\dot{\chi}$.

the convergence of \dot{y} to χ and \ddot{y} to $\dot{\chi}$. For this purpose, the input v given as

$$v = q_1(\chi - \dot{y}) + q_2(\dot{\chi} - \ddot{y}) + \ddot{\chi}, \quad (3.7)$$

where q_1 and q_2 are suitable gains, which guarantees a stable behavior of the error $(\chi - \dot{y})$.

An important issue concerns the specification of the desired velocity χ . Naturally, χ needs to be a function of y in order to point towards the origin. Moreover, the update of χ is chosen to be continuous to guarantee a smooth behavior. The idea for solving this problem is to mimic the classical state feedback technique (3.6). Indeed, (3.6) guarantees that the system reaches the origin. Consequently, the mechanism for updating χ is based on a filter with the same structure as (3.6). Hence,

$$\ddot{\chi} = k_2\dot{\chi} + k_1\chi + k_0y. \quad (3.8)$$

In summary, the controller based on the mechanism of regenerating the jet, which is called “jet-scheduling controller” consists of (3.7) and (3.8):

$$\begin{aligned} \ddot{\chi} &= k_2\dot{\chi} + k_1\chi + k_0y \\ v &= q_1(\chi - \dot{y}) + q_2(\dot{\chi} - \ddot{y}) + \ddot{\chi}. \end{aligned} \quad (3.9)$$

The major differences and similarities between the classical feedback controller (3.6) and the jet-scheduling controller (3.9) are the following:

- The state feedback controller does not entail any dynamical extension, i.e. the controller is a map of the state of the system (3.5) to the input v .
- The dynamical extension $\ddot{\chi}$ of the jet-scheduling controller has the same dynamics as those imposed by the state feedback controller. This means that χ given by the jet-scheduling controller leads to the same rate of convergence as with the state feedback controller.
- In the jet-scheduling controller, χ can be understood as a “tutor” that is tracked using the input (3.7).

To show the advantage of using the jet-scheduling controller over the classical state feedback controller, let us assume that the system (3.5) contains an unmodeled constant drift c on \ddot{y} (Figure 3.4). With state feedback,

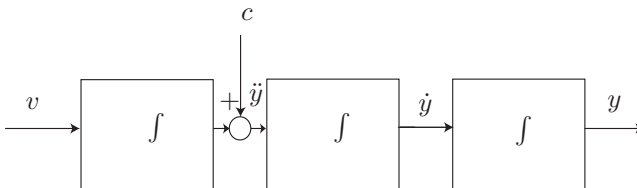


Figure 3.4: An additional constant perturbation is applied to the chain of integrators.

the only way to improve the rejection of the perturbation c is to increase the gains k_0 , k_1 and k_2 , thus modifying the closed-loop dynamics. On the other hand, using the jet-scheduling controller (3.9), it is possible to increase q_1 and q_2 without modifying k_0 , k_1 and k_2 , which means that the dominant closed-loop dynamics are not modified.

The fact that the jet-scheduling controller is able to reject significantly the perturbation c without any modification of the closed-loop dynamics, is a major advantage of this methodology.

Notice that the construction of the jet-scheduling controller is not unique. Indeed, another possibility would be to give “a tutor” (i.e. a desired jet) only

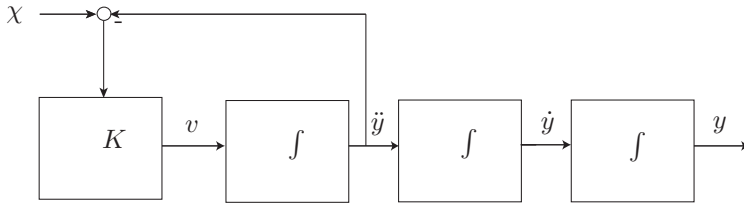


Figure 3.5: Instead of scheduling both \dot{y} and \ddot{y} as in Figure 3.3 only \ddot{y} is scheduled.

for \ddot{y} instead of for both \dot{y} and \ddot{y} as illustrated in Figure 3.5. Consequently, in order to impose the convergence of \ddot{y} to χ , the input v becomes

$$v = q_1(\chi - \ddot{y}) + \dot{\chi}. \tag{3.10}$$

The law for updating χ is now based on y and \dot{y} :

$$\dot{\chi} = k_2\chi + k_1\dot{y} + k_0y \tag{3.11}$$

The difference between the controllers (3.9) and (3.10)-(3.11) lies in the size of the dynamical extension χ . Indeed, for the first controller, the differential equation associated with χ is of order 2, while it is 1 for the second controller. Meanwhile, this difference of degree has a consequence for the robustness. On the one hand, the first controller has two degrees of freedom q_1 and q_2 to impose the convergence of y to χ (see equation 3.9). These parameters have a direct influence on any perturbation affecting \dot{y} and \ddot{y} . On the other hand, the controller (3.10)-(3.11) has the single degree of freedom q_1 , which acts directly only following a perturbation appearing on \ddot{y} . These considerations underline the tradeoff between a low controller order and a large number of degrees of freedom available for rejecting perturbations.

As a summary, a simple chain of integrators can be controlled in two stages.

- (i) A filter (jet scheduler) provides the reference values (χ variables) for derivatives of the output of the integrator chain. For example, such a filter is given by (3.11).
- (ii) A low-level controller that enforces the true values of these derivatives to converge to the references (for example (3.10)).

We will see that in the nonlinear case, it is not trivial to express the low-level errors (i.e. $\ddot{y} - \chi$) using the original coordinate x . A new notation is introduced, namely a set of variable ξ that are in one-to-one correspondence with the low-level errors. In the case of the integrator chain, we have simply $\xi = \ddot{y} - \chi$. For a nonlinear system, a dedicated manifold \mathcal{S} will help in defining the ξ variables.

3.3.2 Jet-scheduling control

The jet-scheduling method has been applied to a somewhat trivial system consisting of three integrators and a single input. However, the Lie-Bäcklund equivalence guarantees that any flat system is equivalent to a set of chains of integrators, possibly of different sizes. This means that the aforementioned technique can be readily adapted to this situation, simply by changing the size of the feedback terms and the amount of gain and applying the technique straightforwardly to each chain of integrators. We will now provide the appropriate details for such a generalization.

The flatness property guarantees the correspondence between the flat output y and their jets $\dot{y}, \ddot{y}, \dots, y^{(r+1)}$ and the state x and the inputs u of the system (3.1). Consequently, a flat system is equivalent to m chains of integrators. As underlined in Section 2.1.3, these chains are not necessarily of the same size. For this reason, they can be represented by the multi-index notation described in Section 2.1.3:

$$\begin{aligned} y_1^{(r_1+1)} &= v_1 \\ &\vdots \\ y_m^{(r_m+1)} &= v_m \end{aligned} \tag{3.12}$$

where v is the input.

As explained in the previous section, the option of scheduling a part of each chain of integrators will be kept open. Therefore, for each chain, the variable p_i is introduced to denote the level at which we stop using the flat outputs and their derivatives directly and start scheduling the higher derivatives through the corresponding jets labeled $\chi_i, \dots, \chi_i^{(r_i-p_i-1)}$. This leads to the following dynamical system

$$\begin{aligned} \chi_i^{(r_i-p_i)} &= k_{0i}y_i + \dots + k_{p_i i}y_i^{(p_i)} + k_{(p_i+1)i}\chi_i + \\ &k_{(p_i+2)i}\dot{\chi}_i + \dots + k_{r_i i}\chi_i^{(r_i-p_i-1)} \quad i = 1, \dots, m. \end{aligned} \tag{3.13}$$

where the k_{ji} are suitable gains.

As described in the introductory example, it is necessary to impose the convergence of the jets $y_i^{(p_i+1)}$ to χ_i . For this, the input v_i is given by

$$v_i = q_{(p_i+1)i}(\chi_i - y_i^{(p_i+1)}) + q_{(p_i+2)i}(\dot{\chi}_i - y_i^{(p_i+2)}) + \dots + k_{ri}(\chi_i^{(r_i-p_i-1)} - y_i^{(r_i)}) + \chi_i^{(r_i-p_i)} \quad i = 1, \dots, m. \quad (3.14)$$

where the q_{ji} are suitable gains.

The input v_i guarantees a stable behavior of the chains of integrators

$$y_i^{(r_i+1)} = v_i.$$

Since the objective is to stabilize the original system (3.1) with the coordinates x and input u , the problem is to compute u based on the equivalence between the original system and the chains of integrators stabilized by the jet-scheduling controller. This problem is discussed in the next section.

3.3.3 Dynamical feedback linearization based on jet scheduling

As shown in the preliminary chapter (Section 2.1.5), the flatness property can be used to construct a dynamical feedback using an endogeneous transformation that transforms the nonlinear system into an equivalent set of integrator chains. In other words, a flat system

$$\dot{x} = f(x, u)$$

can be extended by a dynamical feedback

$$\dot{\gamma} = \beta(x, \gamma, w) \quad (3.15)$$

$$u = \alpha(x, \gamma, w) \quad (3.16)$$

in such way that it becomes equivalent to

$$y_i^{(r_i+1)} = v_i, \quad i = 1, \dots, m.$$

This equivalence means that there exist two invertible functions $\varphi_{x\gamma}$ and φ_w such that

$$(x, \gamma) = \varphi_{x\gamma}(y, \dot{y}, \dots, y^{(r)}) \quad (3.17)$$

$$w = \varphi_w(y, \dot{y}, \dots, v). \quad (3.18)$$

These chains of integrators are stabilized using a closed-loop formulation, as opposed to motion planning done open-loop through the same equivalence (see Section 2.1.4).

Although this process gives a generally satisfactory way of handling the control task, the controller must still be applied to the real system, which means controlling the system in its original coordinates. In other words, the feedback obtained for the chain of integrators must be expressed in the original coordinates x and possibly a dynamical state extension γ .

The previous section has shown how to generalize the idea of controlling a chain of integrators from the classical state-feedback approach to the jet-scheduling approach. This means that, if jet scheduling is adopted for controlling the chain of integrators, there is no reason why it could not be re-expressed in the original coordinates x and γ as this was the case with the classical way of controlling the chain of integrators explained in the previous paragraph. This process is detailed next.

Let us introduce the following selecting map

$$\lambda_{i,j}[y, \dot{y}, \dots, y^{(j)}, \dots, y^{(r)}] = y_i^{(j)} \quad i = 1, \dots, m \quad j = 0, \dots, r.$$

Consequently, the jet scheduler (3.13) can be rewritten as

$$\begin{aligned} \chi_i^{(r_i-p_i)} &= k_{0i} \lambda_{i,0} [\varphi_{x\gamma}^{-1}(x, \gamma)] + \dots + k_{p_i i} \lambda_{i,p_i} [\varphi_{x\gamma}^{-1}(x, \gamma)] + \\ &\quad k_{(p_i+1)i} \chi_i + k_{(p_i+2)i} \dot{\chi}_i + \dots + k_{r_i i} \chi_i^{(r_i-p_i-1)} \\ &= \Pi_{\chi_i}(x, \gamma, \chi_i^{(0)}, \dots, \chi_i^{(r_i-p_i-1)}) \quad i = 1, \dots, m, \end{aligned} \quad (3.19)$$

and the input (3.14) as

$$\begin{aligned} v_i &= q_{(p_i+1)i}(\lambda_{i,p_i+1}[\varphi_{x\gamma}^{-1}(x, \gamma)] - \chi_i) + q_{(p_i+2)i}(\lambda_{i,p_i+2}[\varphi_{x\gamma}^{-1}(x, \gamma)] - \dot{\chi}_i) \\ &\quad + \dots + q_{r_i i}(\lambda_{i,r_i}[\varphi_{x\gamma}^{-1}(x, \gamma)] - \chi_i^{(r_i-p_i-1)}) + \chi_i^{(r_i-p_i)} \\ &= \Pi_{v_i}(x, \gamma, \chi_i, \dots, \chi_i^{(r_i-p_i)}) \quad i = 1, \dots, m. \end{aligned} \quad (3.20)$$

At this point, considering (3.18)-(3.20), the dynamical feedback linearization based on jet scheduling becomes

$$\begin{aligned} \chi_i^{(r_i-p_i)} &= \Pi_{\chi_i}(x, \gamma, \chi_i^{(0)}, \dots, \chi_i^{(r_i-p_i-1)}) \quad i = 1, \dots, m. \\ w &= \varphi_w(\varphi_{x\gamma}^{-1}(x, \gamma), \Pi_w(x, \gamma, \chi_1^{(0)}, \dots, \chi_1^{(r_1-p_1)}, \dots, \chi_m, \dots, \chi_m^{(r_m-p_m)})) \end{aligned} \quad (3.21)$$

Notice that the order of the state extension is slightly larger than the one of the feedback linearization through endogeneous transformations. Indeed, not only must the state be extended using γ , but also it is required to add the χ 's so that the control scheme can benefit from the jet-scheduling process.

The stability of the extended system controlled using (3.21) is guaranteed through the equivalence with the integrator chains controlled by the jet-scheduling controller.

3.3.4 Reduced dynamical extension

Section 3.3.3 gives a general approach for constructing a jet-scheduling controller for a flat system in the original coordinates. The methodology puts strong emphasis on the chain of integrators. It also needs a dedicated dynamical extension (3.15),

$$\dot{\gamma} = \beta(x, \gamma, w)$$

The controller is designed in the coordinates corresponding to the chains of integrators and then is expressed in the original coordinates x and γ through the flat equivalence. This equivalence leads to a good behavior for the closed-loop system since the controller design is done for simple chains of integrators, for which both stability and robustness are guaranteed.

However, since the original system $\dot{x} = f(x, u)$ could be fairly nonlinear, the diffeomorphism $\varphi_{x\gamma}$ used to transform the system into chains of integrators can be complicated. It follows that the controller (3.20), designed in Section 3.3.3, could be awkward and not really intuitive from an engineering point of view. Such a controller could be difficult to implement on a real plant (see Chapter 4.6 and 5.8). For this reason, it would be useful to be able to design a jet-scheduling controller in a more direct way.

The basic idea of jet-scheduling control (see Section 3.3.2) is to impose the convergence of the jets

$$y_1^{(p_1+1)}, \dots, y_m^{(p_m+1)}, \dots, y_1^{(r_1)}, \dots, y_m^{(r_m)}$$

to

$$\chi_1^{(0)}, \dots, \chi_m^{(0)}, \dots, \chi_1^{(r_1-p_1-1)}, \dots, \chi_m^{(r_m-p_m-1)}.$$

The controller (3.20) implements this convergence in the original coordinates x and γ . Indeed, (3.20) provides an input u that has been shown to reach this objective based on the analysis of an equivalent linear system. This

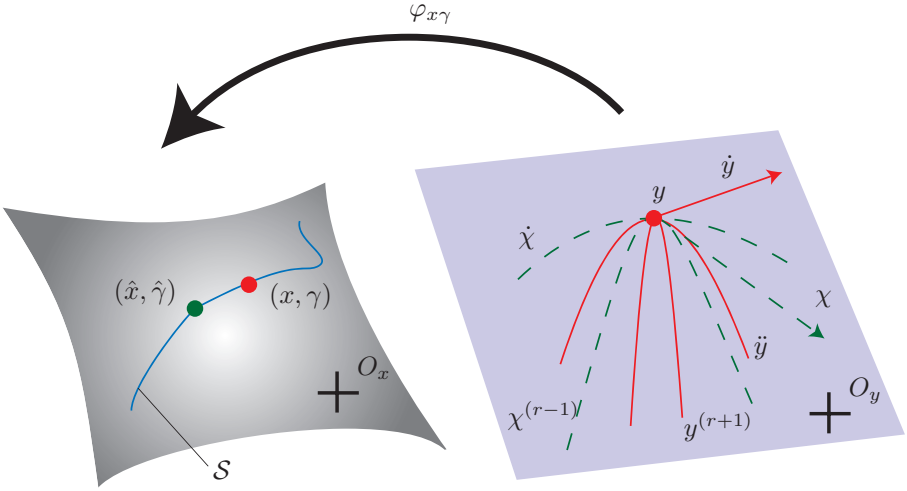


Figure 3.6: Jet construction. The solid lines indicate the position and all the jets $(y, \dot{y}, \dots, y^{(r+1)})$ of the flat system $\dot{x} = f(x, u)$. The dashed lines represent the jet given by the jet scheduler. The point (x, γ) represents the state of system that is naturally compatible with $(y, \dot{y}, \dots, y^{(r+1)})$. The point $(\hat{x}, \hat{\gamma})$ is the desired state compatible with the jets given by the jet scheduler i.e. $(\hat{x}, \hat{\gamma}) = \varphi_{x\gamma}(y, \chi, \dot{\chi}, \dots, \chi^{(r)})$. $(\hat{x}, \hat{\gamma})$ is the origin of \mathcal{S} .

convergence can also be seen from a slightly different point of view. The extended state $(x_1, \dots, x_n, \gamma_1, \dots, \gamma_z)$ given by

$$\begin{aligned}
 (x_1, \dots, x_n, \gamma_1, \dots, \gamma_z) = & \\
 \varphi_{x\gamma}(y_1, \dots, y_m, \dots, y_1^{(p_1)}, \dots, y_m^{(p_m)}, y_1^{(p_1+1)}, \dots, y_m^{(p_m+1)}, \dots, y_1^{(r_1)}, \dots, y_m^{(r_m)}). & \quad (3.22)
 \end{aligned}$$

converges to

$$\varphi_{x\gamma}(y_1, \dots, y_m, \dots, y_1^{(p_1)}, \dots, y_m^{(p_m)}, \chi_1, \dots, \chi_m, \dots, \chi_1^{(r_1-p_1-1)}, \dots, \chi_m^{(r_m-p_m-1)}).$$

Therefore, let us define the quantities $(\hat{x}_1, \dots, \hat{x}_n, \hat{\gamma}_1, \dots, \hat{\gamma}_z)$ that furnish the reference values associated with the scheduled jets $\chi_1^{(0)}, \dots, \chi_m^{(0)}, \dots, \chi_1^{(r_1-p_1)}$,

$$\begin{aligned}
 & \dots, \chi_m^{(r_m - p_m)} \text{ as} \\
 & (\hat{x}_1, \dots, \hat{x}_n, \hat{\gamma}_1, \dots, \hat{\gamma}_z) = \\
 & \varphi_{x\gamma}(y_1, \dots, y_m, \dots, y_1^{(p_1)}, \dots, y_m^{(p_m)}, \chi_1, \dots, \chi_m, \dots, \chi_1^{(r_1 - p_1 - 1)}, \dots, \chi_m^{(r_m - p_m - 1)}).
 \end{aligned} \tag{3.23}$$

Consequently, the controller (3.20) enforces the convergence of

$$(x_1, \dots, x_n, \gamma_1, \dots, \gamma_z)$$

to

$$(\hat{x}_1, \dots, \hat{x}_n, \hat{\gamma}_1, \dots, \hat{\gamma}_z).$$

Notice that $y_1, \dots, y_m, \dots, y_1^{(p_1)}, \dots, y_m^{(p_m)}$ are the same in (3.22) and (3.23). This suggests the following geometrical interpretation (see Figure 3.6). Since $y_1, \dots, y_m, \dots, y_1^{(r_1)}, \dots, y_m^{(r_m)}$ are coordinates of a cartesian product $\mathbb{R}^{r_1+1} \times \dots \times \mathbb{R}^{r_m+1}$, fixing $y_1, \dots, y_m, \dots, y_1^{(p_1)}, \dots, y_m^{(p_m)}$ drops the number of degrees of freedom from $a = \sum_{i=1}^m r_i + 1$ to $b = \sum_{i=1}^m r_i - p_i$. This simply means that the map $\bar{\varphi} : \mathbb{R}^a \rightarrow \mathbb{R}^b$ defined by

$$\begin{aligned}
 & \bar{\varphi}(\alpha_{0,1}, \alpha_{0,2}, \dots, \alpha_{0,m}, \alpha_{1,1}, \alpha_{1,2}, \dots, \alpha_{0,m}, \\
 & \quad \dots, \alpha_{r-p,1}, \alpha_{r-p,2}, \dots, \alpha_{r-p,m}) = \\
 & \varphi(y_1, y_2, \dots, y_m, \dot{y}_1, \dot{y}_2, \dots, \dot{y}_m, \dots, y_1^{(p_1)}, y_2^{(p_2)}, \dots, y_m^{(p_m)}, \\
 & \quad \chi_1 + \alpha_{0,1}, \chi_2 + \alpha_{0,2}, \dots, \chi_m + \alpha_{0,m}, \\
 & \quad \dot{\chi}_1 + \alpha_{1,1}, \dot{\chi}_2 + \alpha_{1,2}, \dots, \dot{\chi}_m + \alpha_{1,m}, \dots, \\
 & \quad \chi_1^{(r-p)} + \alpha_{r-p,1}, \chi_2^{(r-p)} + \alpha_{r-p,2}, \dots, \chi_m^{(r-p)} + \alpha_{r-p,m})
 \end{aligned}$$

is a coordinate chart defining a submanifold of dimension b within the manifold with coordinates $(x_1, \dots, x_n, \gamma_1, \dots, \gamma_z)$. Therefore

$$\begin{aligned}
 \mathcal{S} = & \{ \bar{\varphi}(\alpha_{0,1}, \dots, \alpha_{0,m}, \dots, \alpha_{r-p,1}, \dots, \alpha_{r-p,m}) \mid \\
 & \alpha_{i,j} \in \mathbb{R}, i = 0, \dots, r-p, j = 1, \dots, m \}
 \end{aligned} \tag{3.24}$$

is the aforementioned submanifold in the $(x_1, \dots, x_n, \gamma_1, \dots, \gamma_z)$ space. Unfortunately, the origin and the submanifold itself are not fixed but evolve with time since $y, \dot{y}, \dots, y^{(p)}$ and $\dot{\chi}, \ddot{\chi}, \dots, \chi^{(r-p)}$ change with time. Nevertheless, it is useful to think momentarily as if the submanifold is fixed in time. The purpose of the tracking controller is to guarantee that the true values of $y^{(p+1)}, y^{(p+2)}, \dots, y^{(r+1)}$ do indeed converge to $\chi, \dot{\chi}, \dots,$

$\chi^{(r-p)}$, meaning that the origin of the submanifold is eventually reached. The excursion towards the origin is parametrized by a suitable choice of $\alpha_{i,j} \in \mathbb{R}, i = 0, \dots, r-p, j = 1, \dots, m$

The jets involved in \mathcal{S} are mostly in correspondence with the part of the original state space associated with the extended state γ . This is backed up by physical insights that will be illustrated in Chapters 4 and 5.

Since the objective is to eliminate the dynamical extension, we will construct a new coordinate system Ξ for \mathcal{S} that makes the above reduction straightforward. The key idea is to ensure that some of these coordinates only affect γ and some only affect x . These coordinates will be

$$\begin{aligned} \Xi = & (\xi_{x,1}, \xi_{x,2}, \dots, \xi_{x,c}, \\ & \xi_{\gamma,1}, \xi_{\gamma,2}, \dots, \xi_{\gamma,d}) \end{aligned} \quad (3.25)$$

where the $\xi_{x,i}$'s are the coordinates that affect x and the $\xi_{\gamma,i}$'s the coordinates that affect γ .

The $\xi_{x,i}$'s are used by the controller so as to reach the origin of the submanifold \mathcal{S} . Let T denote the tangent of the manifold whose coordinates are x and γ . T splits as $T_x \oplus T_\gamma$ where T_x spans the subspace associated with the coordinates x .

Among the possible choices for the coordinate patches $\sigma : \mathbb{R}^{\sum_i^m r_i - p_i} \rightarrow \mathcal{S}$ for which the α 's become diffeomorphic to the charts ξ 's and $\sigma(0) = \bar{\varphi}(0)$ (i.e. preserving the origin of \mathcal{S}), there exists at least one with the following condition:

$$\left(\frac{\partial \sigma}{\partial \xi_\gamma} \right)^T v = 0 \quad \forall v \in T_x \quad (3.26)$$

This condition provides the loss of sensitivity that allows one to reduce the order by simply discarding the subpart of \mathcal{S} corresponding to ξ_γ . This is normally not possible with the coordinates α 's. Notice however that the reduction is done at the expense of losing the correspondence between the space parameterized by $y, \dot{y}, \dots, y^{(r+1)}$ and that parameterized by x and γ . The following particular choice of ξ 's in the σ map can be carried out.

Lemma 3.1 *The σ map can be chosen such that*

$$\Xi \rightarrow \begin{pmatrix} \sigma_{x_1}(\xi_{x,1}, \dots, \xi_{x,c}) \\ \sigma_{x_2}(\xi_{x,1}, \dots, \xi_{x,c}) \\ \vdots \\ \sigma_{x_n}(\xi_{x,1}, \dots, \xi_{x,c}) \\ \sigma_{\gamma_1}(\xi_{x,1}, \dots, \xi_{x,c}, \xi_{\gamma,1}, \dots, \xi_{\gamma,d}) \\ \sigma_{u_2}(\xi_{x,1}, \dots, \xi_{x,c}, \xi_{\gamma,1}, \dots, \xi_{\gamma,d}) \\ \vdots \\ \sigma_{\gamma_z}(\xi_{x,1}, \dots, \xi_{x,c}, \xi_{\gamma,1}, \dots, \xi_{\gamma,d}) \end{pmatrix}$$

Proof: The fact that x_1 to x_n do not depend on the $\xi_{\gamma,i}$, follows directly from the condition (3.26) because these ξ_{γ} cannot influence x , since the tangent space associated with x cannot be modified. Therefore only the $\xi_{x,1}, \dots, \xi_{x,c}$ appear on the $\sigma_{x_1}, \dots, \sigma_{x_n}$ ■

This will lead to some difficulties in proving the stability of the closed-loop system.

Example 3.1 *Consider the flat system to be*

$$\begin{aligned} \dot{x}_1 &= u_1 \cos(x_3) \\ \dot{x}_2 &= u_1 \sin(x_3) \\ \dot{x}_3 &= u_2 \end{aligned} \tag{3.27}$$

which is equivalent to the trivial system

$$\begin{aligned} \ddot{y}_1 &= v_1 \\ \ddot{y}_2 &= v_2, \end{aligned}$$

due to the correspondence

$$x_1 = \varphi_{x_1}(y) = y_1 \tag{3.28}$$

$$x_2 = \varphi_{x_2}(y) = y_2 \tag{3.29}$$

$$x_3 = \varphi_{x_3}(\dot{y}) = \text{atan}(\dot{y}_1, \dot{y}_2) \tag{3.30}$$

$$\gamma = u_1 = \varphi_{u_1}(\dot{y}) = \sqrt{\dot{y}_1^2 + \dot{y}_2^2} \tag{3.31}$$

$$\tag{3.32}$$

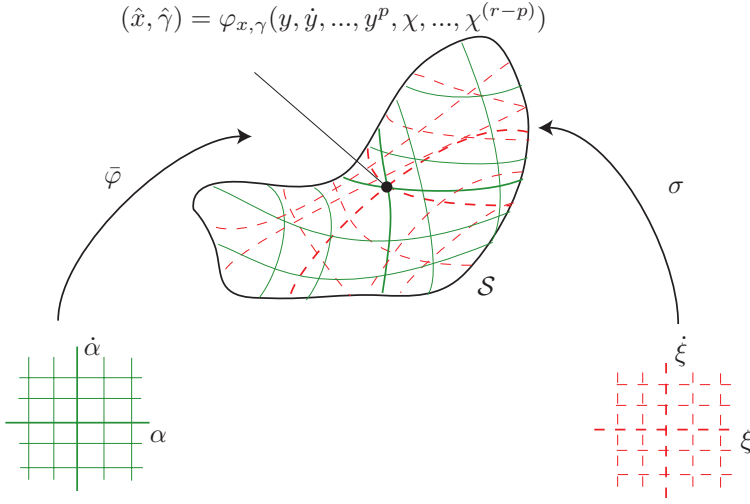


Figure 3.7: Two different coordinate charts for parametrizing the \mathcal{S} manifold. The mapping $\bar{\varphi}$ is based on the flatness equivalence. The mapping σ is used by the jet-scheduling methodology. The point $(\hat{x}, \hat{\gamma})$ is the origin of the \mathcal{S} manifold.

Notice that, in this case, u_1 is chosen as an extended state γ . This means that there is a one-to-one correspondence between

$$(x_1, x_2, x_3, u_1)$$

and

$$(y_1, y_2, \dot{y}_1, \dot{y}_2)$$

Now, setting the parameter p of the jet scheduler as $p = 0$, it follows that the submanifold \mathcal{S} defined by (3.24) becomes

$$\mathcal{S} = \{\bar{\varphi}(\alpha_{0,1}, \alpha_{0,2}, \mid \alpha_{0,j} \in \mathbb{R}, j = 1, 2\} \quad (3.33)$$

where $\bar{\varphi} = (\bar{\varphi}_{x_1}, \bar{\varphi}_{x_2}, \bar{\varphi}_{x_3}, \bar{\varphi}_{u_1})$ with

$$\bar{\varphi}_{x_1} = y_1 \quad (3.34)$$

$$\bar{\varphi}_{x_2} = y_2 \quad (3.35)$$

$$\bar{\varphi}_{x_3} = \arctan(\chi_1 + \alpha_{0,1}, \chi_2 + \alpha_{0,2}) \quad (3.36)$$

$$\bar{\varphi}_{u_1} = \sqrt{(\chi_1 + \alpha_{0,1})^2 + (\chi_2 + \alpha_{0,2})^2} \quad (3.37)$$

The structure of the coordinates x_1, x_2, x_3, u_1 implies a split among a newly defined set of coordinates $\xi_{x,1}, \xi_{\gamma,1}$ such that $\xi_{\gamma,1}$ only influences u_1 without affecting $x_1, x_2,$ and x_3 . This is indeed the case with the map $\sigma : \mathbb{R}^2 \rightarrow \mathbb{R}^4$ given by the following six components

$$\begin{aligned}
 \sigma_{x_1} &= y_1 \\
 \sigma_{x_2} &= y_2 \\
 \sigma_{x_3} &= \arctan(\chi_1, \chi_2) + \xi_{x,1} \\
 \sigma_{u_1} &= \sqrt{\chi_1^2 + \chi_2^2} + \xi_{\gamma,1}
 \end{aligned} \tag{3.38}$$

The tangent space splits as $T = T_x \oplus T_{u_1}$ given as

$$T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \oplus \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

Thus, the choice of coordinates ξ satisfies the condition (3.26)

$$\left(\frac{\partial \sigma}{\partial \xi_{\gamma,1}} \right)^T T_x = (0 \quad 0 \quad 0 \quad 1) \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} = 0.$$

Reduced-order controller design

Now, a control law must be designed to reach the origin of \mathcal{S} . The key idea to reduce the order is to take advantage of the prolongation structure appearing in the Lie-Bäcklund equivalence. Such a prolongation structure allows shortcutting the higher derivatives of u by imposing only the u , without considering $(\dot{u}, \ddot{u}, \dots, u^\infty)$. Ideally, only the direct input should be set, and the upper derivatives will follow suit based on the fact that the natural differentiability of the applied input is respected. We have the following Lemma:

Lemma 3.2 *Let $\hat{x}, \hat{u}, \hat{\dot{u}}, \dots, \hat{u}^\infty$ be the variables which are φ -related to $y, \dot{y}, \dots, y^{(p)}, \chi, \dot{\chi}, \dots, \chi^\infty$. If $x = \hat{x}$ and $u = \hat{u}, \quad \forall t$, then $\gamma = \hat{\gamma}$*

Proof: $y, \dot{y}, \dots, y^{(\infty)}$ can be set freely. In particular, they can be set to $y, \dot{y}, \dots, y^{(p)}, \chi, \dot{\chi}, \dots, \chi^\infty$. No matter how they are chosen, the prolongation

structure along the original is maintained. This means that

$$\hat{x}, \hat{u}, \dot{\hat{u}}, \dots, \hat{u}^\infty$$

is equal to

$$\hat{x}, \hat{u}, \dot{\hat{u}}, \dots, \hat{u}^\infty.$$

Now, by hypothesis $u = \hat{u} \quad \forall t$. Differentiating and taking the above relations gives

$$\dot{u} = \dot{\hat{u}} = \dot{\hat{u}}, \ddot{u} = \ddot{\hat{u}} = \ddot{\hat{u}}, \dots, u^{(\infty)} = \hat{u}^\infty = u^{\hat{\infty}}$$

Therefore,

$$y = y, \dot{y} = \dot{x}, \dots, y^{(p)} = y^{(p)}, y^{(p+1)} = \chi, y^{(p+1)} = \dot{\chi}, \dots, y^{(\infty)} = \chi^{(\infty)}.$$

Finally, considering only the finite number of these quantities that appear in (3.23) gives $\gamma = \hat{\gamma}$. \blacksquare

Remark 3.2 *Lemma 3.2 makes it possible to reduce the number of states by discarding the dynamical extension (3.15).*

Thus the controller objective is to simply ensure that x converges to \hat{x} and u converges to \hat{u} , so that the origin of \mathcal{S} is eventually reached thanks to Lemma 3.2 and the Ξ chart. Using Lemma 3.1, it follows that

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} \sigma_{x_1}(\xi_{x,1}, \dots, \xi_{x,c}) \\ \vdots \\ \sigma_{x_n}(\xi_{x,1}, \dots, \xi_{x,c}) \end{pmatrix} = \sigma_x(\xi_{x,1}, \dots, \xi_{x,c}).$$

This means that, by definition of σ , x converges to the origin of \mathcal{S} when $(\xi_{x,1}, \dots, \xi_{x,c})$ converges to $(0, \dots, 0)$. For this reason, the dynamics of $(\xi_{x,1}, \dots, \xi_{x,c})$ are studied. Notice that

$$\dot{x} = f(x, u) = \frac{\partial \sigma_x}{\partial (\xi_{x,1}, \dots, \xi_{x,c})} \frac{d}{dt} \begin{pmatrix} \xi_{x,1} \\ \vdots \\ \xi_{x,c} \end{pmatrix}. \quad (3.39)$$

It is important to underline that the expression (3.39) is true as long as the variables $y, \dot{y}, \dots, y^{(p)}$ and $\dot{\chi}, \ddot{\chi}, \dots, \chi^{(r-p)}$ are considered fixed in

time. However, since these variables evolve in time, the expression (3.39) becomes

$$\dot{x} = f(x, u) = \frac{\partial \sigma_x}{\partial (\xi_{x,1}, \dots, \xi_{x,c})} \frac{d}{dt} \begin{pmatrix} \xi_{x,1} \\ \vdots \\ \xi_{x,c} \end{pmatrix} + \frac{\partial \sigma_x}{\partial \Upsilon} \frac{d\Upsilon}{dt} \quad (3.40)$$

where $\Upsilon = (y, \dot{y}, \dots, y^{(p)}, \chi, \dot{\chi}, \dots, \chi^{(r-p)})$.

Remark 3.3 Notice that Υ corresponds to the origin of \mathcal{S} . It follows that the term $\frac{\partial \sigma_x}{\partial \Upsilon} \frac{d\Upsilon}{dt}$ can be interpreted as the contribution due to the displacement of the origin of \mathcal{S} .

At this point, it is possible to calculate the dynamics of $(\xi_{x,1}, \dots, \xi_{x,c})$ inverting the relation (3.40). This is done using the pseudo-inverse

$$\begin{aligned} \frac{d}{dt} \begin{pmatrix} \xi_{x,1} \\ \vdots \\ \xi_{x,c} \end{pmatrix} &= \left[\left(\frac{\partial \sigma_x}{\partial (\xi_{x,1}, \dots, \xi_{x,c})} \right)^T \left(\frac{\partial \sigma_x}{\partial (\xi_{x,1}, \dots, \xi_{x,c})} \right) \right]^{-1} \\ &\quad \left(\frac{\partial \sigma_x}{\partial (\xi_{x,1}, \dots, \xi_{x,c})} \right)^T \left(f(x, u) - \frac{\partial \sigma_x}{\partial \Upsilon} \frac{d\Upsilon}{dt} \right) \end{aligned} \quad (3.41)$$

We focus next on the control of the reduced system (3.41). We suppose that (3.41) can be enforced through $u = (u_1, \dots, u_m)$ to be exponentially stable. This way, the $\xi_{x,1}, \dots, \xi_{x,c}$ are guaranteed to reach the origin of \mathcal{S} . Notice that the choice of u in order to achieve this goal may be non-unique.

Assumption 3.1 There exists a $u = K(\xi_x, \Upsilon)$, such that

- $u = K(\xi_x, \Upsilon)$ stabilizes exponentially (3.41)
- $K(0, \Upsilon) = \hat{u}$

The first condition imposes that x converges to \hat{x} . The second one, thanks to Lemma (3.2), imposes that γ converges to $\hat{\gamma}$. This means that the origin of \mathcal{S} is reached.

Remark 3.4 Assumption 3.1 relies on the choice of p . In practice, even when p is small, the dynamics of ξ_x are usually of small size. In Chapters 4 and 5, K is a P and a PD controller, respectively.

At this point, the following conjecture is proposed.

Conjecture 1 *The flat system (3.1), controlled by Controller (3.13) and verifying Assumption 3.1, converges to the origin O_x .*

Example 3.2 *Considering again Example 3.1, we will construct a controller that stabilizes exponentially the origin of \mathcal{S} . The first step is to calculate the reduced dynamics of ξ 's given by (3.41). Since in our case*

$$\sigma_x = \begin{pmatrix} \sigma_{x_1} \\ \sigma_{x_2} \\ \sigma_{x_3} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \arctan(\chi_1, \chi_2) + \xi_{x,1} \end{pmatrix}, \quad (3.42)$$

(3.41) becomes

$$\begin{aligned} \dot{\xi}_{x,1} &= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}^{-1} (0 \ 0 \ 1) \begin{pmatrix} u_1 \cos x_3 - \dot{y}_1 \\ u_1 \sin x_3 - \dot{y}_2 \\ u_2 - \frac{\chi_1 \dot{\chi}_2 - \chi_2 \dot{\chi}_1}{\chi_1^2 + \chi_2^2} \end{pmatrix} \\ &= u_2 - \frac{\chi_1 \dot{\chi}_2 - \chi_2 \dot{\chi}_1}{\chi_1^2 + \chi_2^2} \end{aligned} \quad (3.43)$$

Setting

$$u_2 = \frac{\chi_1 \dot{\chi}_2 - \chi_2 \dot{\chi}_1}{\chi_1^2 + \chi_2^2} - k_p \xi_{x,1},$$

makes $\xi_{x,1}$ exponentially stable and $u_2 = \hat{u}$ when $\xi_{x,1} = 0$. Since the second input u_1 is not used to stabilize $\xi_{x,1}$, the input u_1 is simply set to \hat{u}_1 . Hence, the inputs u_1 and u_2 become

$$u_1 = \sqrt{\chi_1^2 + \chi_2^2} \quad (3.44)$$

$$u_2 = \frac{\chi_1 \dot{\chi}_2 - \chi_2 \dot{\chi}_1}{\chi_1^2 + \chi_2^2} - k_p \xi_1 \quad (3.45)$$

It is interesting to underline that in order to construct the above controller, the dynamical extension γ has never been taken into account explicitly.

Nevertheless, one should still check the stability issue by considering the complete dynamics consisting of the original dynamics together with the jet scheduler and the above controller enforcing the origin of \mathcal{S} to be reached. This will be undertaken in Chapter 4.

3.4 Stability Analysis

In order to study the stability of a flat system controlled through the jet-scheduling controller, we consider the behavior of the flat outputs, their derivatives $y, \dot{y}, \dots, y^{(p)}$ and the scheduled jets $\chi, \dot{\chi}, \dots, \chi^{(r-p)}$. We will see that the analysis can be expressed as the stability problem of a vanishing perturbed system, which has been developed in Section 2.2.3.

The first aspect is to set the problem in the connected form (2.28), namely

$$\begin{aligned}\dot{\Upsilon} &= F(\Upsilon, \Xi) \\ \dot{\Xi} &= G(\Xi).\end{aligned}\tag{3.46}$$

Since our study focuses on the behavior of $y, \dot{y}, \dots, y^{(p)}$ and the scheduled jets $\chi, \dot{\chi}, \dots, \chi^{(r-p)}$, Υ is chosen as

$$\Upsilon = (y, \dot{y}, \dots, y^{(p)}, \chi, \dots, \chi^{(r-p)})$$

and $\Xi = (\xi_{x,1}, \dots, \xi_{x,c}, \xi_{\gamma,1}, \dots, \xi_{\gamma,d})$, where the ξ 's are the variables appearing in the condition (3.26) that describes \mathcal{S} . Thanks to the α 's used in (5.63), it follows that $y^{(p+1)}$ can be expressed as

$$y^{(p+1)} = \chi + \alpha_0.\tag{3.47}$$

Moreover, since the condition (3.26) guarantees the existence of a diffeomorphism

$$(\xi_{x,1}, \dots, \xi_{x,c}, \xi_{\gamma,1}, \dots, \xi_{\gamma,d}) = \Xi = \Phi(\alpha, \dots, \alpha_{(r-p)}),$$

$$y^{(p+1)} = \chi + \Phi_1^{-1}(\Xi, \Upsilon).\tag{3.48}$$

Therefore, since the dynamics of $\chi, \dot{\chi}, \dots, \chi^{(r-p)}$ are given by (3.13), the system to study is

$$\frac{d}{dt} \begin{pmatrix} y \\ \vdots \\ y^{p-1} \\ y^{(p)} \\ \chi \\ \vdots \\ \chi^{(r-p)} \end{pmatrix} = \begin{pmatrix} \dot{y} \\ \vdots \\ y^{(p)} \\ \chi + \Phi_1^{-1}(\Xi, \Upsilon) \\ \dot{\chi} \\ \vdots \\ k_0 y + \dots + k_p y^{(p)} + k_{(p+1)} \chi + k_{(p+2)} \dot{\chi} + \dots + k_r \chi^{(r-p)} \end{pmatrix}\tag{3.49}$$

or, in a more compact notation,

$$\dot{\Upsilon} = F(\Upsilon, \Xi). \quad (3.50)$$

Thanks to Assumption 3.1 and Lemma 3.2, Ξ converges exponentially to 0. Hence, we can turn our attention to the asymptotic system

$$\dot{\Upsilon} = F(\Upsilon, 0).$$

Since $(\xi_{x,1}, \dots, \xi_{x,c}, \xi_{\gamma,1}, \dots, \xi_{\gamma,d}) = 0$ implies that $(\alpha, \dots, \alpha_{(r-p)}) = 0$, (by the choice of the ξ 's). Therefore, $\dot{\Upsilon} = F(\Upsilon, 0)$ reads

$$\frac{d}{dt} \begin{pmatrix} y \\ \vdots \\ y^{p-1} \\ y^{(p)} \\ \chi \\ \vdots \\ \chi^{(r-p)} \end{pmatrix} = \begin{pmatrix} \dot{y} \\ \vdots \\ y^{(p)} \\ \chi \\ \dot{\chi} \\ \vdots \\ k_0 y + \dots + k_p y^{(p)} + k_{(p+1)} \chi + k_{(p+2)} \dot{\chi} + \dots + k_r \chi^{(r-p)} \end{pmatrix} \quad (3.51)$$

which is clearly a linear system of the form

$$\dot{\Upsilon} = F(\Upsilon, 0) = A\Upsilon,$$

with

$$A = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ & & \vdots & & \\ 0 & 0 & 0 & \cdots & 1 \\ k_0 & k_1 & k_2 & \cdots & k_r \end{pmatrix}$$

Since the closed-loop system is now expressed as the vanishing perturbed system (3.50), all the tools developed in Section 2.2.3 can be used to study its stability. For the examples given in Chapters 4 and 5, a specific stability analysis based on this development is carried out.

3.5 Comparison with Dynamic Linearization

In this section, we will consider the jet-scheduling control for the special case $p_i = r_i$, $i = 1, \dots, m$ meaning that there are no χ 's involved. In this case, the jet scheduler (3.13) becomes

$$y_i^{(r_i+1)} = k_{0i}y_i + k_{1i}\dot{y}_i + \dots + k_{r_i i}y_i^{(r_i)}, \quad i = 1, \dots, m$$

and the submanifold \mathcal{S} degenerates to the point

$$(y_i, \dots, y_i^{(r_i)}), \quad i = 1, \dots, m$$

Because the Assumption 3.1 is trivially satisfied, then the gain K can be set to 0. Furthermore, since Ξ is always 0, (3.50) reads

$$\dot{\Upsilon} = F(\Upsilon, 0) = A\Upsilon, \quad (3.52)$$

where $\Upsilon = (y, \dot{y}, \dots, y^{(r)})$. Consequently, (3.52) is a stable linear system $\forall t$. Notice that in this case the closed-loop system is linear $\forall t$ and not only asymptotically as in the general case when $p < r$.

As described in Chapter 2, dynamic feedback linearization forces the system $\dot{x} = f(x, u)$ to be equivalent to

$$\dot{\Upsilon} = A\Upsilon, \quad (3.53)$$

at every time instant. For this reason, we can conclude that dynamic feedback linearization is a particular case of the jet-scheduling control where $p = r$.

3.6 Example: Linear System

Consider the unstable system \mathcal{A} given as

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u \quad (3.54)$$

Construction of jet-scheduling controller

The first step for stabilizing \mathcal{A} using the jet-scheduling controller is to check whether \mathcal{A} is flat. Since \mathcal{A} is linear and controllable, it follows directly that \mathcal{A} is flat, and the flat output y is given by

$$y = \begin{pmatrix} 0 & 1 \end{pmatrix} C^{-1} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = x_1,$$

where \mathcal{C} is the controllability matrix:

$$\mathcal{C} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}.$$

The second step is to find the functions φ_x and φ_u for which $x = \varphi_x(y, \dots, y^{(r)})$ and $u = \varphi_u(y, \dots, y^{(r+1)})$, with $r \in \mathcal{Z}$. For this purpose, the flat output y is differentiated:

$$\begin{aligned} y &= x_1 \\ \dot{y} &= \dot{x}_1 = x_1 + x_2 \\ \ddot{y} &= \dot{x}_1 + \dot{x}_2 = 3x_1 + 2x_2 + u \end{aligned} \tag{3.55}$$

The three equations (3.55) are easily solved for x_1 , x_2 and u , which determines φ_x and φ_u . Notice that $r = 1$ in this example:

$$\begin{aligned} x_1 &= \varphi_{x_1}(y, \dot{y}) = y \\ x_2 &= \varphi_{x_2}(y, \dot{y}) = \dot{y} - y \\ u &= \varphi_u(y, \dot{y}, \ddot{y}) = \ddot{y} - 2\dot{y} - y \end{aligned}$$

The third step consists in fixing the jet scheduler. For $r = 1$, the jet scheduler is

$$\dot{\chi} = -k^2 y - 2k\chi \tag{3.56}$$

The last step is the synthesis of the input u . As described in Section 3.3.4, the state x needs to converge to $\varphi_x(y, \chi, \dots, \chi^{(r-1)})$ that corresponds to the ideal state specified by the jet scheduler. In our case, the states x_1 and x_2 should converge to $\varphi_{x_1}(y, \chi)$ and $\varphi_{x_2}(y, \chi)$, respectively. Since $\varphi_{x_1}(y, \chi) = y = x_1$, the convergence of x_1 to $\varphi_{x_1}(y, \dot{y})$ is always guaranteed. This is no longer the case for x_2 . The convergence of x_2 to $\varphi_{x_2}(y, \chi)$ must be enforced by u . For this reason, u contains two terms. The first one, $\varphi_u(y, \chi, \dot{\chi})$, is imposed directly by the jet scheduler, while the second one, $-k_p(x_2 - \varphi_{x_2}(y, \chi))$, leads to the convergence of x_2 to $\varphi_{x_2}(y, \chi)$. Notice that the second term is a proportional term. Consequently, u is given by

$$\begin{aligned} u &= \varphi_u(y, \chi, \dot{\chi}) - k_p(x_2 - \varphi_{x_2}(y, \chi)) \\ &= \dot{\chi} - 2\chi - y - k_p(x_2 - (\chi - y)) \end{aligned} \tag{3.57}$$

Finally, the system \mathcal{A} (3.54) controlled by the jet scheduler (3.56)-(3.57), can be rewritten as

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{\chi} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \\ -k^2 - k_p + 1 & 1 - k_p & -2k + k_p - 2 \\ -k^2 & 0 & -2k \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \chi \end{pmatrix} \quad (3.58)$$

The poles of the closed-loop system (3.58) are $\lambda = \{-k, -k, 2 - k_p\}$, i.e. it is globally exponentially stable for $k_p > 2$.

Comparison with a static state-feedback controller

Next, the jet-scheduling controller is compared with a static state-feedback controller designed using a pole-placement technique. For the jet-scheduling controller and the static state-feedback controller to share the same dominant poles $-k$, the static state-feedback controller is chosen as

$$u = (3 + 2k + k^2 \quad 2 + 2k) \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad (3.59)$$

Of course, both the jet-scheduling and the static state-feedback controller stabilize \mathcal{A} . However, considering a parametric uncertainty on \mathcal{A} , the robustness of both controllers may be different. Let us consider the perturbed system \mathcal{A}_p :

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 2 & c \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u, \quad (3.60)$$

where $c \in \mathcal{R}$ is a parameter subject to uncertainty.

Both the jet-scheduling and the static state-feedback controllers designed for the nominal system \mathcal{A} are applied to \mathcal{A}_p . In order to compare robustness, the poles of the closed-loop system are computed for both controllers. In Figure 3.8, the largest pole of the closed-loop system is given as a function of c . The parameter k , which imposes the desired dynamics, is fixed at $k = 1$ for both controllers. The dashed line represents the static state-feedback controller, while the solid lines represent the jet-scheduling controller with $k_p = 5$ and $k_p = 10$. For the same dominant dynamics ($k = 1$), the jet-scheduling controller uses its extra degree of freedom k_p to deal with the uncertainty. Increasing k_p , leads to a larger uncertainty being tolerated by the controller, without loss of stability. However, in practice, using a k_p that is too large can amplify the measured noise.

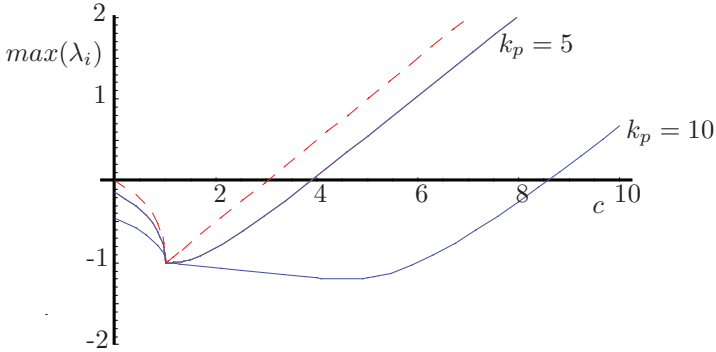


Figure 3.8: Robustness comparison: Jet-scheduling (solid lines) vs a static state-feedback controller (dashed line). In this case $k = 1$. The λ_i are the poles of the closed-loop system. Using a static state-feedback controller, the stability is guaranteed as long as the perturbation term c is smaller than 3. Thanks to its extra degree of freedom k_p , the jet-scheduling controller leads to a larger tolerance.

3.7 Conclusions

The flatness property implies the existence of correspondences between the state and original input and the flat output. These correspondences are given as functions of the flat output and a finite number of its time derivatives. These functions are used to define the jet-scheduling mechanism.

Instead of using fixed polynomials to express the input and the state, a dynamical system (called the jet scheduler) expresses the desired values of some of the higher derivatives of the flat outputs in terms of the measured values in terms of the measured values of the lower derivatives of the flat outputs. This allows appropriate reaction to possible perturbation.

Once the jets are scheduled, they need to be matched by the true system. This is achieved using a reduced-order controller. This controller was designed through a dimensional reduction induced by the prolonged structure appearing in the \mathcal{S} submanifold defined in Section 3.3.4. In more colloquial language, the appearance of the derivatives of the input (in the general dynamical extension needed to convert the nonlinear system to a linear and controllable form) could be by-passed simply by directly using the input to force the true jets to match the ones generated by the jet sched-

uler. This is meaningful from a physical point of view as will be illustrated in forthcoming chapters. Roughly stated, the input is simply used to force parts of the nonlinear system to parts corresponding to the jets, and this does not require the use of a dynamical extension. Nevertheless, the jet scheduler is still mandatory since it acts as a dynamical extension in its own way.

One of the main advantages of this method is that the feedback is separated between (i) the way the jets are rescheduled (normally according to a given dominant rate that is rather small so as to generate a smooth reaction), and (ii) the way they are tracked (using the reduced-order controller). The tracking can be vigorous, achieved with relative high gain, without disturbing the dominant convergence rate. The possibility of using large gains will be seen to be very useful to take into account unknown dynamics that appear in practice (for instance certain aspects of the steering particularities for nonholonomic robotic systems or the winching mechanism in cranes).

This methodology left two open questions, namely the choice of p and the precise design of the low-order controller. Both these issues will be addressed through two specific examples, the first one concerned with the class of nonholonomic robots and the second dealing with the entire class of cranes.

Chapter 4

Application to Mobile Robots

4.1 Introduction

The behavior of the jet-scheduling controller, developed in Chapter 3, will be illustrated on a nonholonomic mobile robot. In fact, we already used — without explicit mention — the model of a nonholonomic robot. It is now detailed and expanded so as to apply the complete methodology in simulation and in real time on a laboratory-scale setup.

Over the last 25 years, the nonholonomic mobile robot has been used as an illustrative example for the control and stabilization of nonholonomic dynamic systems [6, 1, 11, 19, 45, 51, 68]. As is now well known from Brockett's theorem on necessary conditions for stability [8], the main difficulty in controlling nonholonomic systems is that there exists no continuous time-invariant state-feedback control law that asymptotically stabilizes the system at any equilibrium point of interest. Due to the existence of various necessary conditions for local smooth stabilization (Brockett's condition [8]; Zabczyk's condition [81], see also [75] and [69]), the tracking problem is much easier than the stabilization one. Moreover, it is even sometimes advantageous to consider path following instead of trajectory tracking ([72]). The difference lies in the parametrization of the point locus. In the case of a trajectory, points are parametrized as functions of time, whereas for a path, points are parametrized by the choice of a curvilinear coordinate. Thanks to the jet-scheduling control method developed in the previous chapter, both the trajectory-tracking and point-stabilization problems are addressed using a unified methodology. Note that the methodologies proposing a unique controller for both asymptotic stabilization and tracking are in fact rare

[65, 22, 21, 76].

In the literature, the trajectory-tracking problem is dealt with in many different ways, which fall into three main categories: (i) Linearization-based tracking as in [27, 46, 78]; (ii) input-output linearization [62]; (iii) Dynamic feedback linearization [65], [12] and differential flatness [10, 33, 68]. Regarding the globality issue, control laws for global tracking have been proposed in [20, 45, 65, 73].

In the case of the linearization-based tracking approaches in the first category, discontinuous controllers can be extremely sensitive to actuator noise, and their robust redesign is a technically difficult issue that needs further investigation [3]. Some preliminary results are available for a class of nonholonomic systems with uncertainties [44, 6].

Regarding the input-output linearization schemes in the second category, a major shortcoming of the smooth time-varying feedback controllers is that the closed-loop system is only asymptotically stable at the origin with no guarantee of exponential stability [66, 72]. Nevertheless, it should be mentioned that exponential convergence can be achieved for a class of nonholonomic systems by means of Lipschitz continuous homogeneous time-varying feedback. However, as was shown in [44], homogeneous feedback laws often do not guarantee stability in the presence of even small disturbances.

As far as the third category of feedback laws is concerned, the techniques belong to two subclasses: a) dynamic feedback linearization [65], and b) feedforward control based on flatness [34, 35, 68].

This chapter is organized as follows. Section 4.2 presents some preliminary material about mobile robot modeling, flatness and dynamic feedback linearization. The jet-scheduling controller is developed in Section 4.3; asymptotic convergence to a point will be treated first, followed by its extension to trajectory tracking. Section 4.4 illustrates, through simulation, the effectiveness of the proposed control scheme. In Section 4.5, the proofs of stability are detailed. In Section 4.6, the autonomous mobile robot *Fouzy III* is described, and some experimental results for exponential convergence and tracking are given. Section 4.7 concludes the chapter.

4.2 Preliminaries

Four-quadrant arc tangent function

In this chapter, the well-known, four-quadrant arc tangent function \arctan is used. \arctan is defined as a function from $\mathbb{R} \times \mathbb{R}$ to $(-\pi, \pi)$

$$\arctan(\chi_1, \chi_2) \doteq \begin{cases} -i \ln\left(\frac{\chi_1 + i\chi_2}{\sqrt{\chi_1^2 + \chi_2^2}}\right) & (\chi_1, \chi_2) \neq (0, 0) \\ 0 & (\chi_1, \chi_2) = (0, 0) \end{cases} \quad (4.1)$$

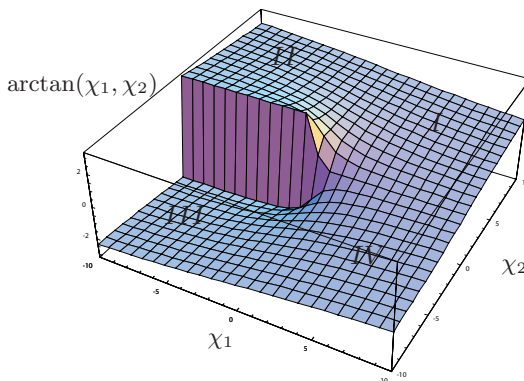


Figure 4.1: Illustration of the four-quadrant arc tangent function. The roman numerals indicate the corresponding quadrant.

As illustrated in Figure 4.1, \arctan exhibits a strong discontinuity on the set $\{\chi_2 = 0, \chi_1 < 0\}$ as a 2π “jump” occurs when the set is reached.

Nonholonomic mobile robot

Consider a mobile robot moving on a planar surface as illustrated in Figure 4.2. Its kinematic equations of motion, which have already been used in Example 3.1 are given by:

$$\dot{x}_1 = u_1 \cos(x_3) \quad (4.2)$$

$$\dot{x}_2 = u_1 \sin(x_3) \quad (4.3)$$

$$\dot{x}_3 = u_2, \quad (4.4)$$

where u_1 and u_2 are the inputs representing the linear and angular veloci-

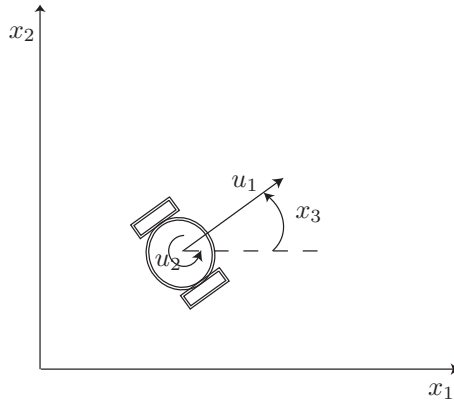


Figure 4.2: Illustration of the mobile robot with the positions x_1, x_2 and heading angle x_3 .

ties, respectively. The coordinates x_1 and x_2 denote the robot position on the plane, while x_3 indicates its orientation, i.e. the angle measured from the x_1 -axis.

Structural difficulty in controlling a nonholonomic mobile robot

The stabilization of a nonholonomic mobile robot exhibits intrinsic difficulties due to the nonholonomy. These difficulties are presented below.

The first difficulty is that the control of a nonholonomic mobile robot is an intrinsic nonlinear problem, in the sense that it is not possible to use the linearized model in order to synthesize a stabilizing controller. Indeed, the linearized approximation $\dot{x} = Ax + Bu$ to the robot (4.2)-(4.4) is given by

$$A = \left. \frac{\partial f}{\partial x} \right|_{x=0, u=0} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad B = \left. \frac{\partial f}{\partial u} \right|_{x=0, u=0} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix},$$

with $\dot{x} = f(x, u) = f(x_1, x_2, x_3, u_1, u_2) = (u_1 \cos x_3, u_1 \sin x_3, u_2)^T$. It follows that the rank of controllability matrix

$$C = \begin{pmatrix} B & AB & A^2B \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

is 2, thus indicating that the linearized model is uncontrollable. This illustrates that a controller based on the linearized approximation cannot be found.

Notice that the uncontrollability of the linearized model does not mean that the mobile robot (4.2)-(4.4) is uncontrollable. Indeed, rewriting the system (4.2)-(4.4) as

$$\dot{x} = g_1(x)u_1 + g_2(x)u_2$$

with $g_1(x) = (\cos(x_3), \sin(x_3), 0)^T$ and $g_2(x) = (0, 0, 1)^T$, it is possible to compute the rank of the accessibility matrix as

$$\text{rank}(g_1, g_2, [g_1, g_2]).$$

It is easy to verify that the accessibility rank condition is globally satisfied. As the system is driftless, this guarantees its controllability although in a nonlinear sense.

The second difficulty is that there does not exist a time-invariant C^1 feedback that stabilizes the nonholonomic robot. Hence, a stabilizing controller needs to be discontinuous or time varying, which means that the construction of such a control law is awkward.

In order to discuss this assertion, Brockett's and Zabczyk's necessary conditions are recalled. The system $\dot{x} = f(x, u)$, where $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ is continuous, is considered. The system is said C^1 stabilizable if there exists a time-invariant C^1 feedback law that makes the equilibrium point $x = 0$ an attractor around which the system is Lyapunov stable.

Theorem 4.1 (Brockett's Necessary Condition [8])

If $\dot{x} = f(x, u)$ is C^1 stabilizable, then the image of f contains an open neighborhood of 0.

The contrapositive is usually invoked to show the impossibility of using a smooth time-invariant feedback law to stabilize the system. However, this might not be sufficient at times. The following condition gives a stronger contrapositive.

Theorem 4.2 (Zabczyk's Necessary Condition [81])

If $\dot{x} = f(x, u)$ is C^1 stabilizable, then f transforms an arbitrary neighborhood of $(0, 0)$ in $\mathbb{R}^n \times \mathbb{R}^m$ onto a neighborhood of 0 in \mathbb{R}^n .

Lemma 4.1 *The nonholonomic robot (4.2)-(4.4), although satisfying Brockett's condition, does not satisfy Zabczyk's condition. This means that there does not exist a time-invariant C^1 feedback that stabilizes the nonholonomic robot.*

Proof: No matter how $\epsilon_i \in \mathbb{R}$, $i = 1, 2, 3$, are chosen, it is always possible to find \bar{u}_1 , \bar{u}_2 and \bar{x}_3 such that $\bar{u}_1 \cos \bar{x}_3 = \epsilon_1$, $\bar{u}_1 \sin \bar{x}_3 = \epsilon_2$, $\bar{u}_2 = \epsilon_3$. Therefore, the condition of Theorem 4.1 is satisfied (with $f(x, u) = f(x_1, x_2, x_3, u_1, u_2) = (u_1 \cos x_3, u_1 \sin x_3, u_2)^T$). Now pick an $\epsilon \in \mathbb{R}$, $0 < \epsilon < \frac{\pi}{2}$, and consider the following neighborhood of $(0, 0)$:

$$\mathcal{O} = (-1, 1) \times (-1, 1) \times (-\epsilon; \epsilon) \times (-1, 1) \times (-1, 1). \quad (4.5)$$

The image of this neighborhood results in two opposite "slices of cake":

$$\begin{aligned} f(\mathcal{O}) &= \{x_1, x_2, x_3 \mid x_1 = \delta \cos \mu, \\ &\quad x_2 = \delta \sin \mu, x_3 \in (-1, 1)\} \\ &\quad \forall \mu \in (-\epsilon, \epsilon) \quad \forall \delta \in (-1, 1), \end{aligned}$$

which do not constitute a neighborhood of 0 , i.e. no matter how small $\epsilon > 0$ is chosen, $(0, \epsilon, 0)^T$ is not in the image of the map $f(x, u)$. Hence, using the contraposition of Theorem 4.2, the conclusion follows. ■

Flatness

It is shown next that the mobile robot (4.2)-(4.4) is a flat system. Choosing x_1 and x_2 as the flat outputs, i.e. $y = (x_1, x_2)^T$, and using (4.2) and (4.3), x_3 can be expressed as:

$$x_3 = \arctan(\dot{x}_1, \dot{x}_2) + \beta\pi \quad \beta = 0, 1 \quad (4.6)$$

Furthermore, the inputs u_1 and u_2 become:

$$u_1 = (-1)^\beta \sqrt{\dot{x}_1^2 + \dot{x}_2^2} \quad \beta = 0, 1 \quad (4.7)$$

$$u_2 = \frac{\dot{x}_1 \ddot{x}_2 - \dot{x}_2 \ddot{x}_1}{\dot{x}_1^2 + \dot{x}_2^2}. \quad (4.8)$$

In view of (4.6)-(4.8), the mobile robot model is differentially flat according to Definition 2.7.

Remark 4.1 Notice that the correspondence becomes singular when $\dot{x}_1 = \dot{x}_2 = 0$. Formally, the robot is not flat around such a point or trajectory. As will be seen in Section 4.3.1, this is a consequence of the contraposition of Zabczyk's condition being satisfied. Nevertheless, it is still possible to apply the flatness arguments outside the singularity set, as will be seen in Section 4.3. A similar singularity occurs for dynamic feedback linearization.

Dynamic Feedback Linearization

Consider the system

$$\dot{x} = f(x, u) \quad x \in \mathbb{R}^n, u \in \mathbb{R}^m. \quad (4.9)$$

As explained in Section 2.1.5, dynamic feedback linearization consists in finding a feedback compensator of the form:

$$\begin{aligned} \dot{\gamma} &= \phi(x, \gamma, w) \\ u &= \varphi(x, \gamma, w) \end{aligned} \quad (4.10)$$

with a dynamic extension $\gamma \in \mathbb{R}^p$ and an input $w \in \mathbb{R}^m$ such that the closed-loop system obtained from (4.9) and (4.10) is equivalent to a linear system under the state transformation $y = T(x, \gamma)$, with w acting as the new input.

In [65], a controller based on dynamic feedback linearization is proposed for the mobile robot (4.2)-(4.4). The compensator reads

$$\dot{\gamma} = w_1 \cos(x_3) + w_2 \sin(x_3) \quad (4.11)$$

$$u_1 = \gamma \quad (4.12)$$

$$u_2 = \frac{w_2 \cos(x_3) - w_1 \sin(x_3)}{\gamma}. \quad (4.13)$$

The closed-loop system formed by (4.2)-(4.4) and (4.11)-(4.13) is equivalent to two chains each of two integrators, for which the inputs are w_1 and w_2 , respectively. A classical linear controller can then be designed to control this equivalent linear system. It is shown that this method gives good results in stabilization and trajectory tracking.

Remark 4.2 When $\gamma = 0$, (4.13) becomes singular. To avoid singularity throughout the convergence process, it is proposed to reset the initial condition $\gamma(0)$ whenever γ becomes small [65]. This is done in real-time.

4.3 Jet-Scheduling Control for a Mobile Robot

In this section, the application of the jet scheduling controller methodology (see Section 3.3.4) will be exposed.

Two types of point-stabilization controllers are presented in Section 4.3.1 (together with relevant convergence properties), namely a smooth feedback and a discontinuous feedback. Both controllers are based on the aforementioned asymptotic linear-equivalence idea. The trajectory-tracking controller is then a straightforward adaptation of this point-stabilization controller, however with an additional difficulty in the convergence proof, which is presented in Section 4.3.2.

4.3.1 Point stabilization

As stated in Lemma 4.1, it is not possible to achieve point stabilization using a smooth feedback law.

Nonetheless, two options are available to overcome this difficulty (these options are also given in [65]). One is to slightly modify the stabilization problem by requiring convergence of all states to zero except for the robot heading angle. By appropriate choice of the controller for the linear equivalent system, it is possible to compel the heading angle to only a few asymptotic values that will depend on the initial conditions (mainly 0 or π , see below). This can be satisfactory for most practical circumstances.

The second option is to use a discontinuous controller. An intricate analysis of the linear equivalent system (see Lemma 4.11 below) shows that the initial conditions can be split into *two connected symmetrical regions* (one inducing convergence to 0 and the other to π) separated by a thin boundary made up of two connected regions (one associated with $-\frac{\pi}{2}$ and the other with $\frac{\pi}{2}$).

To construct a convenient discontinuous controller (leading to the convergence of the heading angle to zero, irrespective of the initial conditions), the symmetry of the initial conditions on each side of the partition is used to switch between two symmetrical controllers of the type described earlier. The switching depends on which side of the partition the state belongs to. This introduces the needed discontinuity. Additionally, by properly adjusting the initial conditions of the dynamic extension — this represents one of the advantages of using a two-dimensional state extension — it is possible to avoid the undesirable states that induce convergence to $-\frac{\pi}{2}$ or $\frac{\pi}{2}$.

The jet-scheduling controller, which corresponds to a smooth controller,

is presented first, together with a proof of convergence. Then, a discontinuous controller will be addressed.

Jet-scheduling controller

This section deals with the construction of a jet-scheduling controller for the mobile robot using the results of Chapter 3. Except for a singularity set called “the exceptional set”, the jet-scheduling controller is capable of achieving the type of convergence described in the previous section. This is a slight modification of classical asymptotic stabilization, where all states are required to converge asymptotically to zero. Here, all states except the heading angle converge to zero. The heading angle is restricted to four different values, mainly 0 and π with the added possibilities of the asymptotic values $-\frac{\pi}{2}$, $\frac{\pi}{2}$ occurring for a small set of initial conditions. The study of the exceptional set is described in Section 4.5.

As shown in Section 4.2, the mobile robot is flat with the flat output $y = (x_1, x_2)^T$, and the state x of the mobile robot is given by

$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \varphi_x(y, \dot{y}) = \begin{pmatrix} y_1 \\ y_2 \\ \arctan(\dot{y}_1, \dot{y}_2) \end{pmatrix}$$

Since $r = 1$ and setting $p = 0$, the jet scheduler (3.13) becomes

$$\dot{\chi}_1 = k_1 x_1 + k_2 \chi_1 \tag{4.14}$$

$$\dot{\chi}_2 = k_3 x_2 + k_4 \chi_2. \tag{4.15}$$

Then, a control law capable of tracking the scheduled jets χ_1 and χ_2 is synthesized using the methodology proposed in Section 3.3.4. Since the mobile robot (4.2)-(4.4) corresponds to the system studied in Example 3.1 and 3.2, the control law developed in these examples can be used. Therefore, using (3.44) and (3.45), the jet-scheduling control for the mobile robot is

$$u_1 = \sqrt{\chi_1^2 + \chi_2^2} \tag{4.16}$$

$$u_2 = \frac{\dot{\chi}_2 \chi_1 - \chi_2 \dot{\chi}_1}{\chi_1^2 + \chi_2^2} - k_p (x_3 - \arctan(\chi_1, \chi_2)) \tag{4.17}$$

$$\dot{\chi}_1 = k_1 x_1 + k_2 \chi_1 \tag{4.18}$$

$$\dot{\chi}_2 = k_3 x_2 + k_4 \chi_2, \tag{4.19}$$

Jet scheduling can be interpreted as a velocity scheduling where χ_1 and χ_2 are the ideal translational velocities (i.e. the ideal values for \dot{x}_1 and \dot{x}_2) that the robot should have (see Figure 4.3). To achieve asymptotic convergence to the desired final position, it is necessary to both schedule adequately these ideal velocities and steer the robot angle x_3 accordingly. The scheduler updates the ideal velocities based on the current robot position (the scheduler is given in (4.14) and (4.15)). The input u_2 is set as the output of a proportional controller that makes x_3 converge to $\bar{x}_3 = \arctan(\chi_1, \chi_2)$.

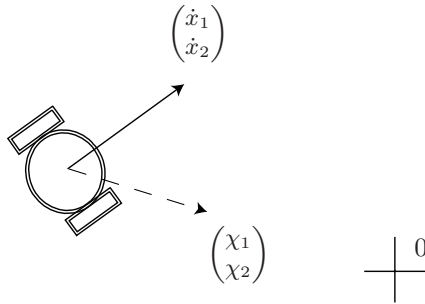


Figure 4.3: The scheduler provides a reference velocity $(\chi_1, \chi_2)^T$, which is the “ideal” velocity to lead the robot to the origin. Thanks to the input u_2 , the real velocity $(\dot{x}_1, \dot{x}_2)^T$ of the mobile robot converges to $(\chi_1, \chi_2)^T$.

The interplay between the controller and the mobile robot ensures asymptotic convergence as stated in the following proposition.

Proposition 4.1 *Choose the controller gains k_i , $1 \leq i \leq 4$, such that the poles $r_1 = \frac{k_2 + \sqrt{4k_1 + k_2^2}}{2}$, $r_2 = \frac{k_2 - \sqrt{4k_1 + k_2^2}}{2}$, $r_3 = \frac{k_4 + \sqrt{4k_3 + k_4^2}}{2}$, $r_4 = \frac{k_4 - \sqrt{4k_3 + k_4^2}}{2}$ are negative real, and consider the system (4.2)-(4.4) together with the following controller:*

$$u_1 = \sqrt{\chi_1^2 + \chi_2^2} \quad (4.20)$$

$$u_2 = \frac{\dot{\chi}_2 \chi_1 - \chi_2 \dot{\chi}_1}{\chi_1^2 + \chi_2^2} - k_p(\bar{x}_3 - \arctan(\chi_1, \chi_2)), \quad (4.21)$$

$$\dot{\chi}_1 = k_1 x_1 + k_2 \chi_1 \quad (4.22)$$

$$\dot{\chi}_2 = k_3 x_2 + k_4 \chi_2, \quad (4.23)$$

where \bar{x}_3 is given as $\bar{x}_3 = x_3 + 2k\pi$ with k a discrete state determined as follows: k is initialized at zero and updated at specific time instants. Let t_i denote any time instant for which the system states are such that $\chi_1(t_i) < 0$ and $\chi_2(t_i) = 0$. Then, update k according to the following rule:

$$\begin{aligned} k &:= k - 1 && \text{if } \dot{\chi}_2(t_i) < 0 \\ k &:= k + 1 && \text{if } \dot{\chi}_2(t_i) > 0 \\ k &:= k + \text{sgn}(k_3 \sin x_3(t_i)) && \text{if } x_2(t_i) = 0. \end{aligned} \quad (4.24)$$

Under these conditions, x_1 and x_2 converge to the origin exponentially. Moreover, when the poles are such that $r_1 = r_2$, $r_3 = r_4$ and $r_1 > r_3$, x_3 converges to one of four values, namely $0 + 2k\pi$, $\pi + 2k\pi$, $\frac{\pi}{2} + 2k\pi$ and $-\frac{\pi}{2} + 2k\pi$. Therefore, the initial conditions become partitioned into four distinct sets depending on the asymptotic value of their corresponding trajectory, namely the sets of initial conditions \mathcal{X}_0 , \mathcal{X}_π , $\mathcal{X}_{\frac{\pi}{2}}$, and $\mathcal{X}_{-\frac{\pi}{2}}$.

The proof of this proposition is given in Section 4.5.5 and is based on lemmas given in Sections 4.5.1-4.5.3. A short guide through these results is provided in Section 4.5.

Remark 4.3 *The final heading angle is determined by the rate of convergence of the controlled linear equivalent system. The precise statement is given in Lemma 4.12 in Section 4.5.3 and can be summarized as follows. It follows from $x_3 = \arctan(\dot{x}_1, \dot{x}_2)$ that the heading angle x_3 is determined by appropriate trajectories for x_1 and x_2 . If a faster pole is chosen for x_2 than for x_1 , i.e. x_2 approaches zero faster than x_1 , the robot reaches the origin with a horizontal tangent ($x_3 = 0 + 2k\pi$ or $\pi + 2k\pi$).*

Remark 4.4 $\mathcal{X}_{\frac{\pi}{2}}$, and $\mathcal{X}_{-\frac{\pi}{2}}$ are unusual sets in the sense that they represent much smaller regions of initial conditions than \mathcal{X}_0 and \mathcal{X}_π . A characterization of these sets is given in Lemma 4.13. Moreover, it will be seen that it is always possible to set the initial conditions $\chi_1(0)$ and $\chi_2(0)$ so as to avoid these sets. Therefore, upon this modification, the proposed controller converges to either $0 + 2k\pi$ or $\pi + 2k\pi$.

Remark 4.5 *The controller is a smooth controller except when the exceptional set defined by the simultaneous vanishing of χ_1 and χ_2 is visited. Smoothness is ensured along the discontinuity of the arctan function (transition set) despite the hard discontinuity of the arctan function. The update involving the parameter k guarantees the required smoothness by ensuring continuity when crossing between the second and third quadrants.*

Remark 4.6 *The expression (4.21) for u_2 contains two terms. The second one, $-k_p(\bar{x}_3 - \arctan(\chi_1, \chi_2))$ is never singular (as long as $\arctan(0, 0)$ is defined to be 0), whereas the first one becomes singular when χ_1 and χ_2 go to zero simultaneously. The first term is not really necessary to achieve convergence as long as k_p is chosen sufficiently large (as will be seen in Section 4.6), but it is very useful for ensuring the asymptotic behavior of the controller, i.e. it guarantees asymptotic convergence for all values of k_p . Moreover, as shown in Section 4.5.2, the singularity does not cause any hindrance in achieving asymptotic convergence.*

Remark 4.7 *As with the controller proposed in [65], x_3 is determined modulo 2π . Therefore, depending on the initial value of k , the final angle is a multiple of 2π and the final value depends on the entire closed-loop dynamics. It is not possible to know a priori the number of times k will be updated. However, under the additional condition that k_p is chosen large enough, it is possible to both replace \bar{x}_3 by x_3 in equation (4.21) and suppress the k parameter.*

Unfortunately, Remark 4.7 cannot be turned into a proposition for all initial conditions. This is discussed in detail after the proof of Proposition 4.1, since it relies on the whole machinery presented in Section 4.5. Nevertheless, the modification proposed in Remark 4.7 is illustrated through simulation in Section 4.4.2.

Robustness property

The proportion term $-k_p(\bar{x}_3 - \arctan(\chi_1, \chi_2))$ appearing in (4.21) is an added benefit of the jet-scheduling methodology over classical dynamical feedback. For this nonholonomic robot, it helps in rejecting constant disturbances on the rotational axis that might appear in practice. This the case, when the robot is equipped with a radar consisting of a rotary turret that operates at constant angular velocity. Indeed, viscous friction inherent to this device induces a constant torque that must be rejected. To take into account this disturbance, the robot model is modified to

$$\begin{aligned} \dot{x}_1 &= u_1 \cos x_3 \\ \dot{x}_2 &= u_1 \sin x_3 \\ \dot{x}_3 &= u_2 + \delta, \end{aligned} \tag{4.25}$$

where δ is a constant but unknown disturbance term. We have the following proposition whose proof is positioned to Section 4.5

Proposition 4.2 *Let the hypothesis and the controller be as in Proposition 4.1. Moreover, consider the system (4.25) together with this controller. Under these conditions both x_1 and x_2 converge to 0 exponentially as long as k_p is chosen sufficiently large. Moreover, when the poles are such that $r_1 = r_2$, $r_3 = r_4$ and $r_1 > r_3$, x_3 converges to one of two values, namely $0 + 2k\pi + \frac{\delta}{k_p}$, $\pi + 2k\pi + \frac{\delta}{k_p}$.*

Proof: The proof is postponed to Section 4.5. ■

Discontinuous controller

By toggling between two symmetrical smooth controllers, according to a switching parameter β , it is possible to introduce a suitable discontinuity in the controller and address Zabczyk’s topological obstruction. Notice that this controller still does not achieve Lyapunov asymptotic stability, but it nevertheless guarantees that all states converge to zero except x_3 which converges to $0 + 2k\pi$.

Remark 4.8 *A discontinuous controller is obtained upon replacing Equations (4.20) and (4.21) by the following two equations*

$$u_1 = (-1)^\beta \sqrt{\chi_1^2 + \chi_2^2} \tag{4.26}$$

$$u_2 = \frac{\dot{\chi}_2 \chi_1 - \chi_2 \dot{\chi}_1}{\chi_1^2 + \chi_2^2} - k_p(\bar{x}_3 - \arctan(\chi_1, \chi_2) + \beta\pi), \tag{4.27}$$

and setting $\beta = 0$ (forward motion) or $\beta = 1$ (backward motion) according to a suitable switching rule.

When $(\bar{x}_3 - \arctan(\chi_1, \chi_2)) = 0$, Lemma 4.2 in Section 4.5.1 shows that the nonlinear dynamics are equivalent to linear ones. According to Lemma 4.11 in Section 4.5.3, the final converged values of these linear dynamics are fully characterized by the initial conditions. Now, using a singular perturbation argument (as $k_p \rightarrow \infty$), the states can be separated into a fast variable ξ and slow variables x_1 , χ_1 , x_2 , and χ_2 . It is reasonable to expect that, since the linear system is perfectly determined by its initial conditions, the topologies of \mathcal{X}_0 and \mathcal{X}_π should converge to those of the linear system as k_p tends to infinity. This would mean that it is possible to adapt the β parameter based on the partition of the initial conditions to which the state

belongs. This leads to the following conjecture concerning the switching rule.

Conjecture 2 *The discontinuous controller (4.26)-(4.27), together with the switching law*

$$\begin{aligned} \beta &= 1 && \text{for } -r_1x_1(t) + \chi_1(t) \geq 0 \\ \beta &= 0 && \text{for } -r_1x_1(t) + \chi_1(t) < 0, \end{aligned}$$

makes the states of both the system (4.2-4.4) and the dynamic state extension (4.22) and (4.23) converge to the origin.

Unfortunately, this conjecture is false, in general, even for large values of k_p . The reason is that a Zeno-like phenomenon (that is, a very large number of switchings during a small interval of time) cannot be excluded. This indeed occurs in practice and will be illustrated in the simulation section. Nevertheless, it is possible to modify the switching rule by simply waiting a prescribed amount of time before changing the mode according to the above rule. This results in the following transition policy:

Algorithm 1 *Let β be an integer variable equal to 0 or 1. Let $t_\beta \in \mathbb{R}$ denote the time at which the last transition (i.e. change in β) took place. Also, let $T_\beta \in \mathbb{R}$ denote a parameter.*

Initialization: *Set $t_\beta = 0$. If $-r_1x_1(t) + \chi_1(t) \geq 0$ then set $\beta := 0$ else $\beta := 1$.*

Induction: *while $t < \infty$ do*

- *if $-r_1x_1(t) + \chi_1(t) < 0$ and $(t - t_\beta) > T_\beta$ then set both $\beta := 1$ and $t_\beta := t$.*
- *if $-r_1x_1(t) + \chi_1(t) \geq 0$ and $(t - t_\beta) > T_\beta$ then set both $\beta := 0$ and $t_\beta := t$.*
- *Set the inputs u_1 and u_2 according to expressions (4.26) and (4.27).*

Conjecture 3 *If*

1. $\xi(t) = 0$ and $-r_1x_1(t) + \chi_1(t) = 0$ do not hold simultaneously, and
2. T_β is sufficiently large,

Algorithm 1 ensures asymptotic convergence of x_1 , x_2 , χ_1 , χ_2 and x_3 to $0 + 2k\pi$.

Conjecture 4 *Setting $\chi_1(0)$ so that $-r_1x_1(0) + \chi_1(0) \neq 0$ guarantees that Assumption 1 of Conjecture 3 is always satisfied.*

This update policy removes the Zeno phenomenon as will be illustrated in the simulation section. Contrary to Conjecture 2, the last two conjectures have not been invalidated.

4.3.2 Trajectory tracking

The result obtained in Section 4.3.1 can be extended to trajectory tracking. In this case, the controller in Proposition 4.1 becomes:

$$u_1 = \sqrt{\chi_1^2 + \chi_2^2} \tag{4.28}$$

$$u_2 = \frac{\dot{\chi}_2\chi_1 - \dot{\chi}_1\chi_2}{\chi_1^2 + \chi_2^2} - k_p(x_3 - \arctan(\chi_1, \chi_2)) \tag{4.29}$$

$$\dot{\chi}_1 = k_1e_{x1} + k_2e_{\chi1} + \ddot{x}_{1ref} \tag{4.30}$$

$$\dot{\chi}_2 = k_3e_{x2} + k_4e_{\chi2} + \ddot{x}_{2ref} \tag{4.31}$$

Here x_{1ref} and x_{2ref} are the reference trajectories for x_1 and x_2 , respectively. The error variables are defined as:

$$e_{x1} \triangleq x_1 - x_{1ref} \tag{4.32}$$

$$e_{x2} \triangleq x_2 - x_{2ref} \tag{4.33}$$

$$e_{\chi1} \triangleq \chi_1 - \dot{x}_{1ref} \tag{4.34}$$

$$e_{\chi2} \triangleq \chi_2 - \dot{x}_{2ref} \tag{4.35}$$

With (4.32)-(4.33), the system (4.2)-(4.4) can be rewritten as:

$$\dot{e}_{x1} = u_1 \cos(x_3) - \dot{x}_{1ref} \tag{4.36}$$

$$\dot{e}_{x2} = u_1 \sin(x_3) - \dot{x}_{2ref} \tag{4.37}$$

$$\dot{x}_3 = u_2 \tag{4.38}$$

Proposition 4.3 *Consider the system (4.36) - (4.38) with the controller (4.28) - (4.31). Assume that the controller is such that the poles $r_1 = \frac{k_2 + \sqrt{4k_1 + k_2^2}}{2}$, $r_2 = \frac{k_2 - \sqrt{4k_1 + k_2^2}}{2}$, $r_3 = \frac{k_4 + \sqrt{4k_3 + k_4^2}}{2}$, $r_4 = \frac{k_4 - \sqrt{4k_3 + k_4^2}}{2}$ are negative and real. Then, x_1 and x_2 converge to x_{1ref} and x_{2ref} exponentially.*

The proof can be found in Section 4.5.5.

Remark 4.9 *Trajectory tracking is easier than point stabilization since, locally around a trajectory, u_1 is almost always different from 0. This way, the singularity is avoided and local linear controllability along the trajectory can be guaranteed.*

4.4 Simulation Study

In this section, the behavior of the controllers for both stabilization at the origin and trajectory tracking is tested and compared to that of dynamic feedback linearization in order to illustrate the theoretical results of Section 4.3.

4.4.1 Stabilization at the origin

The stabilizing behavior of the velocity-scheduling controller is tested first. The goal is to bring the mobile robot to the origin, i.e. $x_1 = 0, x_2 = 0$. The initial conditions and controller parameters are given in Table 4.1.

The first simulation is carried out without disturbance on the rotational axis of the robot. The results are shown in Figure 4.4. The trajectories for dynamic feedback linearization and velocity-scheduling control are similar, and all states converge adequately to the origin.

For the second experiment, the robot model is modified to

$$\begin{aligned}\dot{x}_1 &= u_1 \cos x_3 \\ \dot{x}_2 &= u_1 \sin x_3 \\ \dot{x}_3 &= u_2 + \delta,\end{aligned}$$

where δ is a constant but unknown disturbance term. For the value $\delta = 1$, the stabilization results are given in Figure 4.5. Although all translational positions go to zero in both cases, dynamic feedback linearization cannot handle the disturbance along the rotational axis, and the robot keeps turning on itself. In contrast, the velocity-scheduling controller ensures robust convergence to the origin.

The improvement can be explained as follows: Setting u_2 according to Proposition 4.1 leads to $\dot{\xi} = -k_p \xi + \delta$. Therefore, ξ converges to a constant value, that can be made as small as desired by increasing the gain k_p , thereby leading to arbitrary boundedness in ξ . It follows from total

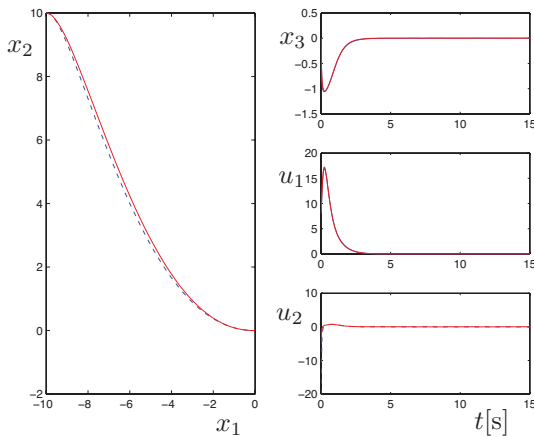


Figure 4.4: Stabilization at the origin for the case of no disturbance on the rotational axis: Dynamic feedback linearization (solid line) and velocity-scheduling control (dashed line) give nearly the same behavior.

Parameters	Values	Parameters	Values
$x_1(0)$	-10	$x_2(0)$	10
$\chi_1(0)$	0	$\chi_2(0)$	0
k_p	10	$x_3(0)$	0
k_1	-4	k_3	-16
k_2	-4	k_4	-8

Table 4.1: Initial conditions and controller parameters used in the simulation of stabilization at the origin

stability arguments [40] that the proof of Proposition 4.3 can be adapted to infer that x_1 , x_2 are also practically stable.

4.4.2 Modification of the smooth controller

It was proposed in Remark 4.7, to change \bar{x}_3 to x_3 in the controller of Proposition 4.1 and to suppress the updating mechanism involving k (see

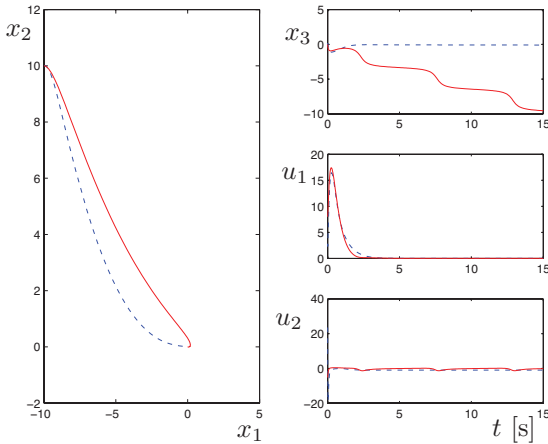


Figure 4.5: Stabilization at the origin for the case of a constant disturbance acting on the rotational axis of the robot: Dynamic feedback linearization cannot reject it (solid line), while velocity-scheduling control guarantees robust convergence.

also Conjecture 5 in Section 4.5.5). A simulation for the conditions given in Table 4.2 provides the results given in Figure 4.6. Thanks to the modification, the robot unwinds completely so that the angle x_3 goes to 0 exactly.

However, there are initial conditions for which the convergence of x_3 is prevented by a rapid succession of jumps along the discontinuity of the arctangent function. This is illustrated in Figure 4.7 and explained in detail after Conjecture 5. Nevertheless, increasing k_p allows induction of a satisfactory convergence of x_3 for the same initial conditions (Figure 4.8). Despite this improvement, it is still not possible to guarantee convergence for any initial condition since, for this new value of k_p , it is possible to find a particular set of initial conditions leading to a similar oscillatory behavior as the one illustrated in Figure 4.7.

It is important to underline that using the smooth controller, the initial conditions can be chosen so that the robot converges to $\pm\pi$. This is illustrated in Figure 4.9 and the initial conditions are given in Table 4.5.

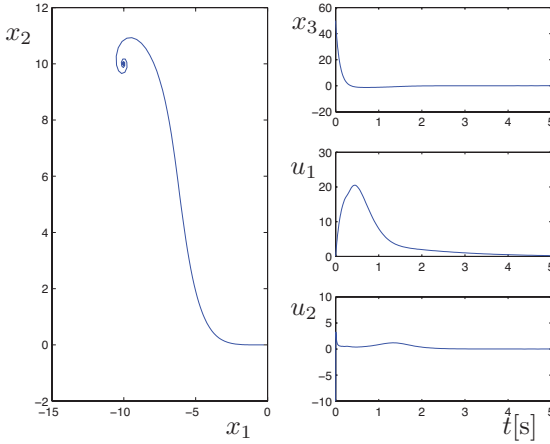


Figure 4.6: Stabilization at the origin with the modified smooth controller: Using the controller proposed in Remark 4.7, i.e. \bar{x}_3 is changed to x_3 , the robot unwinds completely from 50 rad to 0.

Parameters	Values	Parameters	Values
$x_1(0)$	-10	$x_2(0)$	10
$\chi_1(0)$	0	$\chi_2(0)$	0
k_p	10	$x_3(0)$	50
k_1	-4	k_3	-16
k_2	-4	k_4	-8

Table 4.2: Initial conditions and controller parameters used in the simulation illustrated in Figure 4.6

4.4.3 Discontinuous controller

The discontinuous controller presented in Section 4.3.1 brings the final heading angle correctly to the desired asymptotic value of 0 (Figure 4.10).

However, there exist initial conditions that can induce rapid switchings between the two modes $\beta = 0$ and $\beta = 1$. This phenomenon, called the Zeno effect, is illustrated in Figure 4.11. The corrective action, detailed

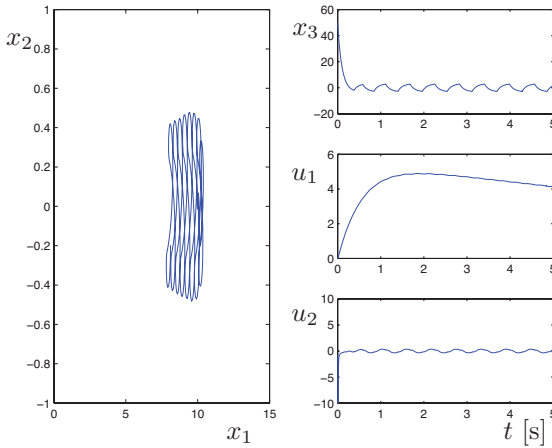


Figure 4.7: Stabilization at the origin with the modified smooth controller: Using the controller proposed in Remark 4.7, there are initial conditions for which the convergence of x_3 is prevented by a rapid succession of jumps along the discontinuity of the arctangent function.

Parameters	Values	Parameters	Values
$x_1(0)$	10	$x_2(0)$	0.1
$\chi_1(0)$	0	$\chi_2(0)$	0
k_p	10	$x_3(0)$	50
k_1	-4	k_3	-16
k_2	-4	k_4	-8

Table 4.3: Initial conditions and controller parameters used in the simulation illustrated in Figure 4.7

in Algorithm 1, consists in waiting for a sufficient amount of time before changing the mode. This is illustrated in Figure 4.12 for the same “poor” initial conditions as in Figure 4.11. Clearly, the Zeno effect has disappeared and the robot correctly converges to the origin.

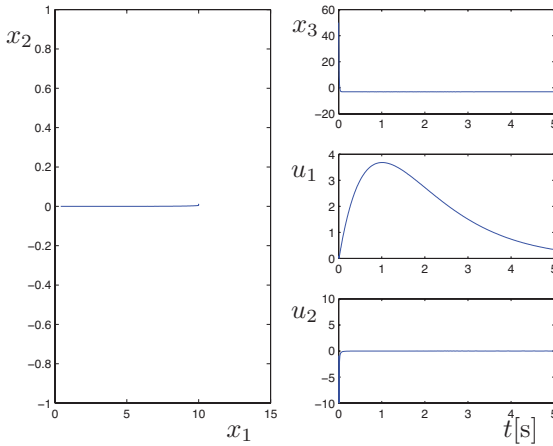


Figure 4.8: Stabilization at the origin with the modified smooth controller: Increasing k_p allows induction of satisfactory convergence of x_3 for the same initial conditions as those in Figure 4.7

Parameters	Values	Parameters	Values
$x_1(0)$	10	$x_2(0)$	0.1
$\chi_1(0)$	0	$\chi_2(0)$	0
k_p	100	$x_3(0)$	50
k_1	-4	k_3	-16
k_2	-4	k_4	-8

Table 4.4: Initial conditions and controller parameters used in the simulation illustrated in Figure 4.8

4.4.4 Circular trajectory tracking

Figure 4.13 illustrates the theoretical development of Section 4.3.2. Velocity-scheduling control is compared to dynamic feedback linearization. As in [45], the problem of tracking a circle with a constant velocity is considered. A circle centered at the origin with a unity radius is defined. As can be seen,

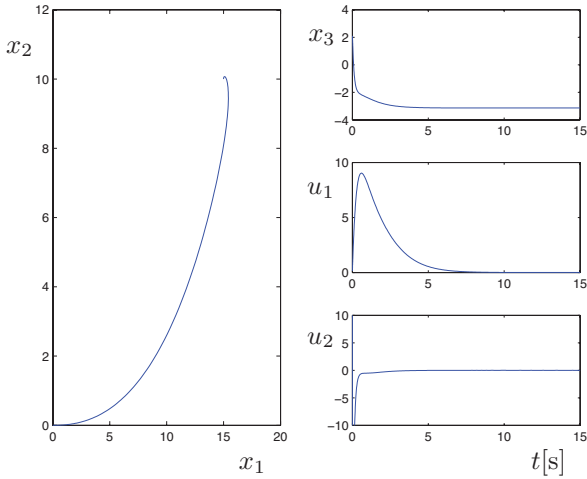


Figure 4.9: Stabilization at the origin with the modified smooth controller (i.e. without switching mode): The robot converges to $(x_1 = 0, x_2 = 0)$, with a heading angle x_3 equal to $-\pi$. In a “strict” stabilization problem, x_3 should converge to 0.

Parameters	Values	Parameters	Values
$x_1(0)$	15	$x_2(0)$	10
$\chi_1(0)$	0	$\chi_2(0)$	0
k_p	10	$x_3(0)$	2
k_1	-4	k_3	-16
k_2	-4	k_4	-8

Table 4.5: Initial conditions and controller parameters used in the simulation illustrated in Figure 4.9

the robot converges nicely to the trajectory with both dynamic feedback linearization and velocity-scheduling control. The input u_2 is smoother with velocity-scheduling control than with dynamic feedback linearization.

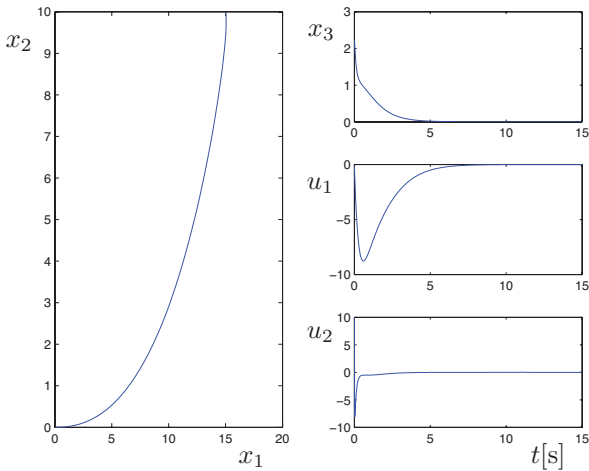


Figure 4.10: Stabilization at the origin using a discontinuous controller (i.e. switching controller): The robot can change its direction, i.e. either forward or backward motion. In this case, backward motion “ $u_1 < 0$ ” is selected by the controller since $t = 0$. The robot moves to the origin with a heading angle $x_3 = 0$.

The initial conditions and controller parameters used in the simulation are given in Table 4.6.

Parameters	Values	Parameters	Values
$x_1(0)$	1	$x_2(0)$	5
x_{1ref}	$\cos(0.2\pi t)$	x_{2ref}	$\sin(0.2\pi t)$
$x_3(0)$	0	k_p	10
k_1	-6	k_3	-6
k_2	-6	k_4	-6

Table 4.6: Initial conditions and controller parameters used in the simulation of circular trajectory tracking

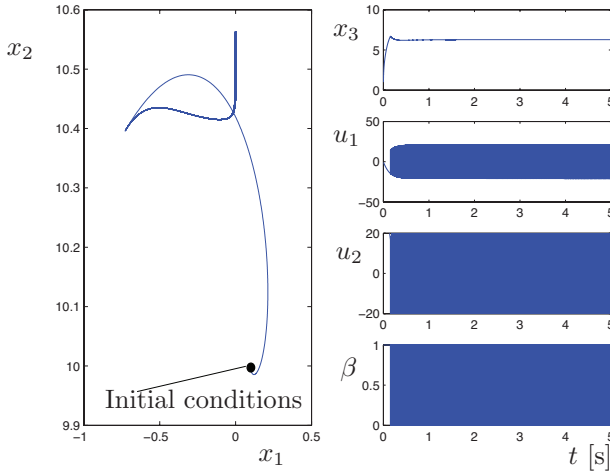


Figure 4.11: Stabilization at the origin using a discontinuous controller (i.e. switching controller): With the switching time T_β chosen equal to zero, the robot can get stuck in an infinite number of switchings, with no convergence to the origin.

Similarly to the stabilization problem, the robot model is modified to

$$\begin{aligned}\dot{x}_1 &= u_1 \cos x_3 \\ \dot{x}_2 &= u_1 \sin x_3 \\ \dot{x}_3 &= u_2 + \delta,\end{aligned}$$

where δ is a constant but unknown disturbance term. For the value $\delta = 3$, the tracking results are given in Figure 4.14. In both cases, the controllers ensure stability, but the performances are quite different. Velocity scheduling control performs significantly better than dynamic feedback linearization. This is due to the fact that there is the additional feedback gain k_p at the user's disposal. This extra degree of freedom achieves (as in the stabilization problem) practical asymptotic stability, meaning that it is possible to reduce the tracking error by increasing k_p .

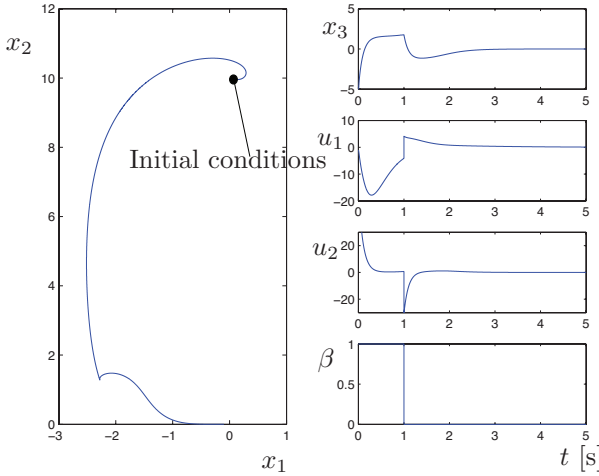


Figure 4.12: Thanks to the delay in the switching time introduced by T_β , the infinite number of switchings is avoided. This leads to convergence despite the same “poor” initial conditions as in Figure 4.11.

4.5 Stability Analysis

This section serves as a guide through the proofs of the lemmas and propositions given in this section.

Combining the robot equations (4.2)-(4.4) and the dynamic extension (4.22)-(4.23) with the inputs defined in (4.20)-(4.21) leads to the so-called S_{x_3} -system:

$$\dot{x}_1 = \sqrt{\chi_1^2 + \chi_2^2} \cos x_3 \quad (4.39)$$

$$\dot{\chi}_1 = k_1 x_1 + k_2 \chi_1 \quad (4.40)$$

$$\dot{x}_2 = \sqrt{\chi_1^2 + \chi_2^2} \sin x_3 \quad (4.41)$$

$$\dot{\chi}_2 = k_3 x_2 + k_4 \chi_2 \quad (4.42)$$

$$\dot{x}_3 = \frac{\dot{\chi}_2 \chi_1 - \chi_2 \dot{\chi}_1}{\chi_1^2 + \chi_2^2} - k_p (\bar{x}_3 - \arctan(\chi_1, \chi_2)). \quad (4.43)$$

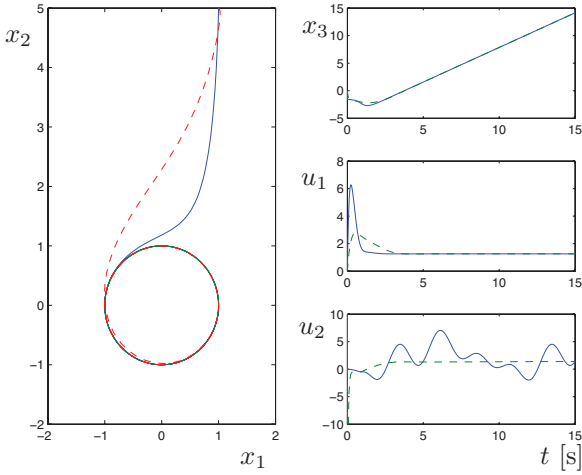


Figure 4.13: Circular trajectory tracking for the case of no disturbance on the rotational axis: Both dynamic feedback linearization (solid line) and velocity-scheduling control (dashed line) achieve good trajectory tracking.

Then, consider the change of coordinates:

$$\xi = \bar{x}_3 - \arctan(\chi_1, \chi_2) = x_3 + 2k\pi - \arctan(\chi_1, \chi_2), \quad (4.44)$$

where k is the discrete variable defined in (4.24). Defining the new state space $(x_1, \chi_1, x_2, \chi_2, \xi)^T$, the S_{x_3} -system (4.39)-(4.43) is equivalent to the S_ξ -system defined as:

$$\dot{x}_1 = \sqrt{\chi_1^2 + \chi_2^2} \cos(\xi + \arctan(\chi_1, \chi_2)) \quad (4.45)$$

$$\dot{\chi}_1 = k_1 x_1 + k_2 \chi_1$$

$$\dot{x}_2 = \sqrt{\chi_1^2 + \chi_2^2} \sin(\xi + \arctan(\chi_1, \chi_2)) \quad (4.46)$$

$$\dot{\chi}_2 = k_3 x_2 + k_4 \chi_2$$

$$\dot{\xi} = -k_p \xi. \quad (4.47)$$

Equivalence means here that the trajectories of the S_{x_3} -system are in diffeomorphic correspondence with those of the S_ξ -system. This can be verified by differentiating the change of coordinates (4.44).

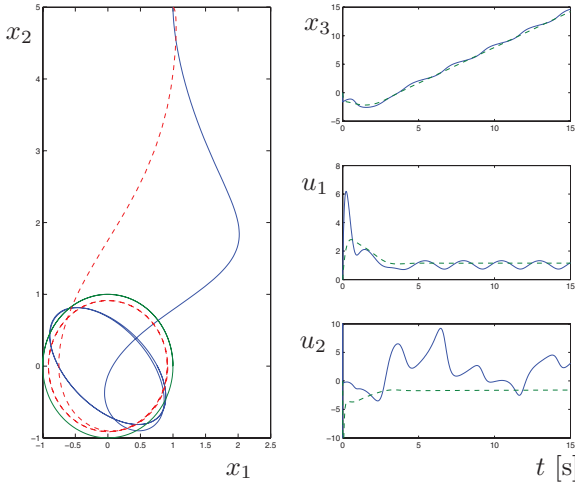


Figure 4.14: Circular trajectory tracking for the case of a constant disturbance acting on the rotational axis: Velocity-scheduling control (dashed line) exhibits better performance than dynamic feedback linearization (solid line).

Since χ_1 and χ_2 are functions of time, the time derivative of $\arctan(\chi_1(t), \chi_2(t))$ has been studied. Due to the discontinuity, the derivation is performed using distributions [82]. The time instant \bar{t}_i is defined such that $\chi_1(\bar{t}_i)$ and $\chi_2(\bar{t}_i)$ reach the set \mathcal{D} (i.e. $\{\chi_2(\bar{t}_i) = 0, \chi_1(\bar{t}_i) < 0\}$). Consequently, after introducing

$$\mathcal{B}(v, w) = \begin{cases} \operatorname{sgn}(v) & v \neq 0 \\ \operatorname{sgn}(k_3 \sin w) & v = 0, \end{cases}$$

the time derivative of the arctangent function along the closed-loop dynamics becomes

$$\begin{aligned} \frac{d}{dt} \arctan(\chi_1, \chi_2) &= \frac{\chi_1 \dot{\chi}_2 - \chi_2 \dot{\chi}_1}{\chi_1^2 + \chi_2^2} \\ &+ 2\pi \sum_{i=0}^N \mathcal{B}(\dot{\chi}_2(t), \chi_3(t)) \delta(t - \bar{t}_i), \end{aligned} \quad (4.48)$$

where N is the number of incursions in \mathcal{D} and δ is the Dirac delta distribution. $\mathcal{B}(\dot{\chi}_2(t), x_3(t))$ is used to determine if the “jump” is positive or negative.

For this purpose, let us recall that the discrete variable k is updated by the rules (4.24). Let \bar{t}_i denote the time instant at which k is updated. The differentiation of $k(t)$ can be considered using distributions:

$$\frac{dk(t)}{dt} = \sum_{i=0}^N \text{sgn}(\dot{\chi}_2(t))\delta(t - \bar{t}_i), \quad (4.49)$$

where N is the number of switches and δ the Dirac delta distribution. Differentiating (4.44) yields

$$\dot{\xi} = \dot{x}_3 + 2\frac{dk}{dt}\pi - \frac{d\arctan(\chi_1, \chi_2)}{dt} \quad (4.50)$$

Using (4.48) and (4.49), one notices that the contributions of the Dirac delta distribution terms are cancelled. It follows that

$$\dot{\xi} = \dot{x}_3 - \frac{\dot{\chi}_2\chi_1 - \chi_2\dot{\chi}_1}{\chi_1^2 + \chi_2^2} \quad (4.51)$$

Putting (4.44) and (4.51) into (4.43) leads to (4.47).

Section 4.5.1 shows the convergence of the S_ξ -system. The states of the S_ξ -system can be decomposed into two sets, $\Upsilon = (\chi_1 \ x_1 \ \chi_2 \ x_2)^T$ and ξ . This gives the structure $\dot{\Upsilon} = F(\Upsilon, \xi)$ and $\dot{\xi} = -k_p\xi$. It is clear that the state ξ decreases exponentially to zero. The difficulty lies in establishing the type of convergence for Υ . Lemma 4.2 gives explicitly the value of F and examines the type of transient behavior of Υ based on the general Lemma 2.1 that concludes that there is no finite escape time, owing to the way ξ interconnects inside $\dot{\Upsilon} = F(\Upsilon, \xi)$ (a growth rate condition is satisfied). The asymptotic system $F(\Upsilon, 0)$ is shown to be a linear stable system. The only difficulty is then to ensure that $F(\Upsilon, \xi)$ eventually converges to the origin. This is guaranteed by Lemma 4.3. The proof follows lines similar to those appearing in the vanishing perturbation case unfolded in [47]. However, the associated Lyapunov function, labelled V_0 , can be constructed explicitly. The lemma shows that, after a finite time T , all states of the S_ξ -system are bounded by an exponential, i.e. there exist $c_1, c_2 > 0$ such that $\|(\chi \ \xi)^T\| \leq c_1 e^{-c_2 t}$, $\forall t > T$.

Now the convergence properties have been established for the S_ξ -system, we can draw our attention to the S_{x_3} -system. Both systems have continuous-time solutions. This follows from the fact that solutions approaching the discontinuity from one side leave the discontinuity from the other side and, therefore, solutions can be defined as absolutely continuous functions satisfying the equations almost everywhere [28]. Next, as long as the partial states $\chi_1(t)$ and $\chi_2(t)$ (corresponding to these solutions) do not enter the transition set $\{\chi_1 < 0, \chi_2 = 0\}$ nor the exceptional set $\{\chi_1 = 0, \chi_2 = 0\}$, the asymptotic property of the S_{x_3} -system can be directly inferred from the S_ξ -system. This follows from the diffeomorphic correspondence between solutions for these two systems starting from compatible initial conditions.

Section 4.5.2 pays special attention to the transition and exceptional sets. Two difficulties occur. First, the correspondence between ξ and x_3 is discontinuous on the transition set. Second, and most important, x_3 is undefined on the exceptional set. The first problem is addressed using a phase unwrapping technique described in Lemma 4.4. Depending on the sign of the velocity $\dot{\chi}_2$ while crossing the discontinuity, it is possible to reset the value of x_3 to maintain continuity before and after the transition. The second difficulty needs careful attention. Lemma 4.5 shows that the robot turns quickly on itself and the rotational angle jumps by a finite amount. The argument is based again on examining the trajectories before and after the transition. Even though the rotational velocity \dot{x}_3 tends to infinity, only a finite jump of $\pm\pi$ is induced on x_3 . In practice, it is not possible to either achieve an infinite velocity or be precisely on the set $\{\chi_1 = 0, \chi_2 = 0\}$. Nevertheless, this lemma, together with a continuity argument, ensures that the behavior of the robot close to this singularity remains satisfactory. Away from the singularity, the diffeomorphic correspondence guarantees that x_3 is continuous and follows reasonable trajectories due to the one-to-one correspondence with ξ , and ξ converges to zero.

Another interesting question concerns the number of times the exceptional set is visited. Lemma 4.6 shows that the vector field is transversal to the exceptional set (except for the origin which is the final desired equilibrium point). Therefore, the system visits the exceptional set only at specific time instants. The number of occurrences can be shown to be finite. To see this, Lemma 4.8 constructs a lower bounding Lyapunov function of χ_1 and χ_2 that decreases exponentially. This guarantees that χ_1 and χ_2 cannot vanish after time T_δ , which, in turn, guarantees that the exceptional set can be visited only a finite number of times. Indeed, Lemma 4.10 gives the result based on the observation that the time taken for the robot to revisit

the exceptional set is bounded from below, before time T_δ , and thus leads to a contradiction argument.

The only remaining question is the convergence of x_3 to the desired value of 0, which is discussed in Section 4.5.3. Depending on the initial conditions, the robot converges to either 0 or π , the actual value depending on which side of the state space the robot starts. Lemma 4.12 shows that all final values of the nonlinear system are those of a linear system that is guaranteed to converge to either 0 or π , depending on the initial conditions of this linear system. This prediction is accurate after some time T . In practice, if the robot starts with appropriate initial conditions, then the final robot angle will be 0. In case x_3 is poorly initialized, a high gain k_p makes it converge rapidly so that the nonlinear system quickly follows the linear system (this makes T small). Lemma 4.11 provides information regarding the final angle value that is going to be reached. This can be used to define a switching law, and a suitable waiting time is introduced to avoid the Zeno phenomenon. The bottom line is that ξ will eventually be small enough so that no more switching occurs. Thus, the asymptotic behavior will correspond to that of the linear system described in Lemma 4.11, for which the asymptotic zero angle is guaranteed to be reached. This sketches the proofs of the main propositions that can be found in Section 4.5.5.

4.5.1 Convergence of the S_ξ -system

Lemma 4.2 *The system (4.2)-(4.4) and the controller (4.20)-(4.23) can be put in the form (2.36), with $\chi = (\chi_1, x_1, \chi_2, x_2)$, where χ_1 and χ_2 are given by (4.22), (4.23), and $\xi = x_3 - \arctan(\chi_1, \chi_2)$. Moreover, both subsystems $\dot{\chi} = F(\chi, 0)$ and $\dot{\xi} = G(\xi)$ are globally exponentially stable at the origin and (2.39) is fulfilled with $c_1 \neq 0$, $\psi(\|\xi\|) = 0$ and $c_2 = 0$.*

Proof: Using (4.2), (4.3), (4.22) and (4.23), $F(\chi, \xi)$ can be written:

$$F(\chi, \xi) = \begin{pmatrix} \frac{k_1 x_1 + k_2 \chi_1}{\sqrt{\chi_1^2 + \chi_2^2}} \cos(\xi + \arctan(\chi_1, \chi_2)) \\ \frac{k_3 x_2 + k_4 \chi_2}{\sqrt{\chi_1^2 + \chi_2^2}} \sin(\xi + \arctan(\chi_1, \chi_2)) \end{pmatrix}$$

Using (4.4) and (4.21), $G(\xi)$ is obtained as:

$$G(\xi) = -k_p \xi \tag{4.52}$$

from which it follows that $\dot{\xi} = G(\xi)$ is exponentially stable.

When $\xi = 0$, and after some algebraic manipulations, $F(\chi, 0)$ becomes:

$$F(\chi, 0) = \begin{pmatrix} k_1 x_1 + k_2 \chi_1 \\ \chi_1 \\ k_3 x_2 + k_4 \chi_2 \\ \chi_2 \end{pmatrix} \quad (4.53)$$

As can be directly checked, $\dot{\chi} = F(\chi, 0)$ is globally exponentially stable at the origin.

Let

$$\Delta F(\chi, \xi) \triangleq F(\chi, \xi) - F(\chi, 0). \quad (4.54)$$

Now, (2.39) can be expressed using the canonical basis e_i , $i = 1, \dots, 4$:

$$\begin{aligned} \|\Delta F(\chi, \xi)\| &= \left\| \sum_{i=1}^4 \Delta F(\chi, \xi)_i e_i \right\| \\ &\leq \sum_{i=1}^4 \|\Delta F(\chi, \xi)_i\| = \\ &\|0\| + \left\| \sqrt{\chi_1^2 + \chi_2^2} \cos(\xi + \arctan(\chi_1, \chi_2)) - \chi_1 \right\| + \\ &\|0\| + \left\| \sqrt{\chi_1^2 + \chi_2^2} \sin(\xi + \arctan(\chi_1, \chi_2)) - \chi_2 \right\| \leq \\ &\|\chi_1\| + \|\chi_2\| + \|\chi_1\| + \|\chi_1\| + \|\chi_2\| + \|\chi_2\| \\ &= 3\|\chi_1\| + 3\|\chi_2\| \\ &\leq c_1 \|\chi\| \end{aligned} \quad (4.55)$$

Here $c_2 = 0$ and $\psi(\|\xi\|) = 0$. ■

Lemma 4.3 *Let $P = P^T > 0$, $Q = Q^T > 0$ be such that,*

$$\begin{aligned} P \begin{pmatrix} 0 & 1 \\ k_1 & k_2 \end{pmatrix} + \begin{pmatrix} 0 & 1 \\ k_1 & k_2 \end{pmatrix}^T P &= -I_2, \\ Q \begin{pmatrix} 0 & 1 \\ k_3 & k_4 \end{pmatrix} + \begin{pmatrix} 0 & 1 \\ k_3 & k_4 \end{pmatrix}^T Q &= -I_2. \end{aligned}$$

Then, there exists a time instant T such that

$$V_0 = (x_1 \quad \chi_1) P \begin{pmatrix} x_1 \\ \chi_1 \end{pmatrix} + (x_2 \quad \chi_2) Q \begin{pmatrix} x_2 \\ \chi_2 \end{pmatrix} \quad (4.56)$$

converges to zero exponentially, i.e. there exists a constant $c > 0$ such that $V_0 \leq V_0(T)e^{-ct} \quad \forall t \geq T$.

Proof: Notice first that

$$\dot{x}_1 = \chi_1 + \sqrt{\chi_1^2 + \chi_2^2} (\cos(\hat{x}_3 + \xi) - \cos(\hat{x}_3)) \quad (4.57)$$

$$\dot{\chi}_1 = k_1 x_1 + k_2 \chi_1 \quad (4.58)$$

$$\dot{x}_2 = \chi_2 + \sqrt{\chi_1^2 + \chi_2^2} (\sin(\hat{x}_3 + \xi) - \sin(\hat{x}_3)) \quad (4.59)$$

$$\dot{\chi}_2 = k_3 x_2 + k_4 \chi_2 \quad (4.60)$$

with $\hat{x}_3 = \arctan(\chi_1, \chi_2)$.

Then,

$$\begin{aligned} \dot{V}_0 &= -(x_1^2 + \chi_1^2) - (x_2^2 + \chi_2^2) \\ &+ 2 \begin{pmatrix} x_1 & \chi_1 \end{pmatrix} P \begin{pmatrix} 1 \\ 0 \end{pmatrix} \sqrt{\chi_1^2 + \chi_2^2} \cdot (\cos(\hat{x}_3 + \xi) - \cos \hat{x}_3) \\ &+ 2 \begin{pmatrix} x_2 & \chi_2 \end{pmatrix} Q \begin{pmatrix} 1 \\ 0 \end{pmatrix} \sqrt{\chi_1^2 + \chi_2^2} \cdot (\sin(\hat{x}_3 + \xi) - \sin \hat{x}_3) \end{aligned}$$

Since

$$\cos(\hat{x}_3 + \xi) - \cos \hat{x}_3 = -\xi \int_0^1 \sin(\hat{x}_3 + \lambda \xi) d\lambda \quad (4.61)$$

$$\sin(\hat{x}_3 + \xi) - \sin \hat{x}_3 = \xi \int_0^1 \cos(\hat{x}_3 + \lambda \xi) d\lambda, \quad (4.62)$$

it follows that

$$\begin{aligned} \dot{V}_0 &= -(x_1^2 + \chi_1^2) - (x_2^2 + \chi_2^2) \\ &- 2 \begin{pmatrix} x_1 & \chi_1 \end{pmatrix} P \begin{pmatrix} 1 \\ 0 \end{pmatrix} \sqrt{\chi_1^2 + \chi_2^2} \cdot \xi \int_0^1 \sin(\hat{x}_3 + \lambda \xi) d\lambda \\ &+ 2 \begin{pmatrix} x_2 & \chi_2 \end{pmatrix} Q \begin{pmatrix} 1 \\ 0 \end{pmatrix} \sqrt{\chi_1^2 + \chi_2^2} \cdot \xi \int_0^1 \cos(\hat{x}_3 + \lambda \xi) d\lambda. \end{aligned} \quad (4.63)$$

Now, using first the Cauchy-Schwarz inequality and then the arithmetic and geometric means inequality $\sqrt{ab} \leq \frac{a+b}{2}$ with $a = x_1^2 + \chi_1^2$ and $b = \chi_1^2 + \chi_2^2$,

leads to

$$\begin{aligned}
& |2(x_1 \ \chi_1) P \begin{pmatrix} 1 \\ 0 \end{pmatrix} \sqrt{\chi_1^2 + \chi_2^2} \xi| \\
& \leq 2 \|(x_1 \ \chi_1)\| \|P \begin{pmatrix} 1 \\ 0 \end{pmatrix}\| \sqrt{\chi_1^2 + \chi_2^2} |\xi| \\
& = 2\sqrt{x_1^2 + \chi_1^2} \sqrt{\chi_1^2 + \chi_2^2} \|P \begin{pmatrix} 1 \\ 0 \end{pmatrix}\| |\xi| \\
& \leq (x_1^2 + 2\chi_1^2 + \chi_2^2) \left\| P \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| |\xi|. \tag{4.64}
\end{aligned}$$

Similary,

$$\begin{aligned}
& |2(x_2 \ \chi_2) Q \begin{pmatrix} 1 \\ 0 \end{pmatrix} \sqrt{\chi_1^2 + \chi_2^2} \xi| \\
& \leq (x_2^2 + \chi_1^2 + 2\chi_2^2) \left\| Q \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| |\xi|. \tag{4.65}
\end{aligned}$$

Then, using (4.64), (4.65) and the fact that

$$\left| \int_0^1 \sin(\hat{x}_3 + \lambda\xi) d\lambda \right| \leq 1 \quad \left| \int_0^1 \cos(\hat{x}_3 + \lambda\xi) d\lambda \right| \leq 1, \tag{4.66}$$

both $\frac{1}{2}x_1^2 + \frac{1}{2}\chi_2^2 + \chi_1^2 < x_1^2 + x_2^2 + \chi_1^2 + \chi_2^2$ and $\frac{1}{2}x_2^2 + \frac{1}{2}\chi_1^2 + \chi_2^2 < x_1^2 + x_2^2 + \chi_1^2 + \chi_2^2$. Therefore, (4.63) becomes

$$\begin{aligned}
\dot{V}_0 & \leq \left(-1 + 4|\xi(t)| \left(\left\| P \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| + \left\| Q \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| \right) \right) \\
& \quad (x_1^2 + x_2^2 + \chi_1^2 + \chi_2^2). \tag{4.67}
\end{aligned}$$

Thanks to Lemma 4.2, there is no finite escape time. Since ξ is exponentially stable, picking $\tilde{c} > 0$ sufficiently small so that

$$1 - 4\tilde{c} \left(\left\| P \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| + \left\| Q \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| \right) > 0, \tag{4.68}$$

there will always exist a finite time T for which

$$|\xi(t)| \leq \tilde{c} \quad \forall t \geq T \tag{4.69}$$

It follows that (4.67) becomes

$$\begin{aligned} \dot{V}_0 &\leq \left(-1 + 4\tilde{c} \left(\left\| P \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| + \left\| Q \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| \right) \right) \\ &\quad \cdot (x_1^2 + x_2^2 + \chi_1^2 + \chi_2^2) \\ &\leq -cV_0 \quad \forall t \geq T. \end{aligned} \tag{4.70}$$

Thus, one sees that $V_0 \rightarrow 0$ exponentially. \blacksquare

4.5.2 Transition and exceptional sets

Lemma 4.4 *Let t_i be any time instant for which the solution of the S_{x_3} -system reaches the set $\{\chi_2 = 0, \chi_1 < 0\}$, then depending on the value of x_2 , it is possible to reset the state $x_3(t_i)$ to a new value $x_3(t_i^+)$ so as to maintain diffeomorphic equivalence between the S_{x_3} -system and the S_ξ -system:*

$$\begin{aligned} x_3(t_i^+) &= x_3(t_i) - 2\pi && \text{if } \dot{\chi}_2(t_i) < 0 \\ x_3(t_i^+) &= x_3(t_i) + 2\pi && \text{if } \dot{\chi}_2(t_i) > 0 \\ x_3(t_i^+) &= x_3(t_i) + 2\pi \operatorname{sgn}(k_3 \sin x_3(t_i)) && \text{if } \dot{\chi}_2(t_i) = 0 \end{aligned}$$

Proof: The transition set $\{\chi_1 < 0, \chi_2 = 0\}$ is not invariant and does not contain any proper invariant subset. To find a proper invariant subset within the transition set, one should display a trajectory such that $\chi_2(t) = 0$ over a certain nonvanishing time interval. This would imply $\dot{\chi}_2 = 0$. Examining Equation (4.42) leads to the conclusion that $x_2(t)$ should also vanish over this time interval for such a trajectory to be possible. This would mean, after examining (4.41), that $\sin x_3$ should also vanish, since $\chi_1 < 0$ on the transition set (i.e. the square root appearing cannot vanish). However, on the transition set, $\arctan(\chi_1, 0) = \pi$ and therefore equation (4.21) cannot vanish, leading to a contradiction.

Hence, the system exits the transition set at an infinitesimal time instant after entering it, i.e. at time $t_1 + \epsilon$. These considerations then lead to the technique for updating the variable x_3 , once the transition set is encountered. For example, when $k_3 x_2(t_i) > 0$, Equation (4.42) shows that $\lim_{\epsilon \rightarrow 0^+} \dot{\chi}_2(t_i + \epsilon) = \lim_{\epsilon \rightarrow 0^-} \dot{\chi}_2(t_i - \epsilon) > 0$. This means that the arctan function jumps from $-\pi$ to π , as χ_2 changes from negative to positive while crossing the transition set. Therefore, in order to maintain ξ continuous, x_3 has to increase to $x_3 + 2\pi$. The other cases follow similarly. \blacksquare

Nevertheless, due to inappropriate initial conditions of the S_{x_3} -system, both χ_1 and χ_2 can vanish simultaneously. This means that the controller is ill-defined, since the first term in (4.43) becomes infinite whenever $\chi_1 = \chi_2 = 0$. This is the case when the system starts anywhere in the set $\mathcal{W} = \{x_1, x_2, x_3, \chi_1, \chi_2 \mid \chi_1 = 0, \chi_2 = 0, x_1 \in \mathbb{R}, x_2 \in \mathbb{R}, x_3 \in \mathbb{R}\}$, which is termed the exceptional set.

All that remains is to find out exactly what is happening to x_3 after crossing this exceptional set.

Lemma 4.5 *Let t_i denote the time instant at which $\chi_1(t_i) = 0$ and $\chi_2(t_i) = 0$ with $x_1(t_i)$ and $x_2(t_i)$ both not being zero. Then, the following discontinuity in x_3 occurs:*

$$\left| \lim_{\epsilon \rightarrow 0_-} x_3(t_i + \epsilon) - \lim_{\epsilon \rightarrow 0_+} x_3(t_i + \epsilon) \right| = \pi \quad (4.71)$$

Proof: Consider the S_ξ -system when such a crossing occurs. Suppose it happens at time t_i . This means $x_1(t_i) = \bar{x}_1$, $x_2(t_i) = \bar{x}_2$, $\chi_1(t_i) = 0$ and $\chi_2(t_i) = 0$, where at least \bar{x}_1 and \bar{x}_2 are nonzero. All states are well defined and continuous for the S_ξ -system except possibly for their time derivative at t_i where discontinuities can occur. Since arctan is ill-defined at that time instant, the trajectory is split into two parts; one part for the time interval $[t_i - \epsilon, t_i)$ and another for $(t_i, t_i + \epsilon]$. For small ϵ , only the first-order terms are considered, i.e. $\xi(t) = \xi(t_i) + k_p \xi(t - t_i)\epsilon + O(\epsilon)$, $x_1(t) = \bar{x}_1$, $x_2(t) = \bar{x}_2$. $\chi_1 = 0$, $\chi_2 = 0$. Consider a small $\delta > 0$, for which

$$\begin{aligned} \chi_1(t_i - \epsilon) &= \chi_1(t_i - \epsilon - \delta) + \dot{\chi}_1(t_i - \epsilon)\delta + O(\delta) \\ \chi_1(t_i + \epsilon + \delta) &= \chi_1(t_i + \epsilon) + \dot{\chi}_1(t_i + \epsilon)\delta + O(\delta). \end{aligned}$$

Then, taking the limit $\epsilon \rightarrow 0$ so as to join both trajectory segments gives:

$$\begin{aligned} \chi_1(t_1 - \delta) &= -\dot{\chi}_1(t_i)\delta + O(\delta) = -k_1 \bar{x}_1 \delta + O(\delta) \\ \chi_1(t_1 + \delta) &= \dot{\chi}_1(t_i)\delta + O(\delta) = k_1 \bar{x}_1 \delta + O(\delta). \end{aligned}$$

A similar development can be undertaken for χ_2 . Thus, when approaching the set (prior to t_i), $x_3(t_i - \delta) = \xi(t_i - \delta) + \arctan(-k_1 \bar{x}_1, -k_2 \bar{x}_2)$, and when quitting the set (after t_i) $x_3(t_i + \delta) = \xi(t_i + \delta) + \arctan(k_1 \bar{x}_1, k_2 \bar{x}_2)$. Therefore, by taking the limit $\delta \rightarrow 0$, the net difference is π and the result follows. ■

Therefore, even though \dot{x}_3 might become arbitrarily large (the robot spins quickly on itself), this does not have dramatic consequences on the

translational position of the robot (both translational velocities vanish). Moreover, the system exits the exceptional set right away and restarts on a “regular” trajectory as will be shown in the following Lemma.

Lemma 4.6 *If the robot enters the exceptional set, both velocities \dot{x}_1 and \dot{x}_2 vanish.*

Proof: Squaring equations (4.39) and (4.41) and adding the result gives $\chi_1^2 + \chi_2^2 = \dot{x}_1^2 + \dot{x}_2^2$. Therefore, both $\dot{x}_1 = 0$ and $\dot{x}_2 = 0$ when the robot enters the exceptional manifold. ■

Remark 4.10 *The result of Lemma 4.5 shows that close to the exceptional manifold, the \dot{x}_3 velocity can become arbitrarily large over a small interval of time. The overall effect is a limited rotation of roughly half a turn. An exact half turn comes when the robot is exactly on the exceptional manifold. Another insight is also given in the proof: In order to find appropriate initial conditions for displaying the behavior close to the exceptional manifold, the robot can be placed at any point different from the origin (i.e. at \bar{x}_1 , and \bar{x}_2); then set $\chi_1(0) = -\epsilon k_1 \bar{x}_1$ and $\chi_2(0) = -\epsilon k_3 \bar{x}_2$, where ϵ is a small number, simulate the (x_3) -system over a time interval $T = [0, 2\epsilon]$.*

We now draw our attention to the number of times the exceptional set is visited. The next lemma shows that these visits occur at specific time instants.

Lemma 4.7 *Let $\Phi(t)$ denote the solution to system (4.2-4.4) together with the controller (4.20-4.23), i.e. $\Phi(t) = (\chi_1(t) \ \chi_2(t) \ x_1(t) \ x_2(t) \ x_3(t))^T$. Let $\mathcal{V} = \{x \mid x_1 = x_2 = \chi_1 = \chi_2 = 0\}$. If there exists a time instant $t_1 > 0$ for which $\Phi(t_1) \in \mathcal{W} \setminus \mathcal{V}$, then there exists an $\epsilon > 0$ for which $\Phi(t) \notin \mathcal{W} \setminus \mathcal{V}$ for all $t_1 - \epsilon < t < t_1 + \epsilon$, $t \neq t_1$.*

Proof: The proof consists in showing that the solution manifold is transversal to the set $\mathcal{W} \setminus \mathcal{V}$. To see this, we will first show that $T_{\Phi(\cdot)}(x) \notin T_{\mathcal{W}}$ for each $x \in \mathcal{W} \setminus \mathcal{V}$. Then, since the vector field defining the dynamics is continuous, the above transversality is guaranteed, i.e. there exists a certain $\epsilon > 0$ for which $\Phi(t) \notin \mathcal{W}$ for all $t_1 - \epsilon < t < t_1 + \epsilon$, $t \neq t_1$.

The manifold \mathcal{W} is defined by $\chi_1 = 0$ and $\chi_2 = 0$. Then, $T_{\Phi(\cdot)}(x)$ for $x \in \mathcal{W}$ is given by

$$T_{\Phi(\cdot)}(x) = (k_1 x_1 \quad k_2 x_2 \quad 0 \quad 0 \quad \beta(x))^T \quad (4.72)$$

where $x = (\chi_1 \ \chi_2 \ x_1 \ x_2 \ x_3)$ and $\beta(x)$ is a corresponding scalar function of x . Now $x \in \mathcal{W}$ means that $\chi_1 = 0$ and $\chi_2 = 0$. Hence,

$$T_{\mathcal{W}} = \text{span} \left\{ \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \right\} \quad (4.73)$$

clearly exhibits the fact that, as long as both x_1 and x_2 do not cancel simultaneously, i.e. as long as $x \in \mathcal{W} \setminus \mathcal{V}$, then $T_{\Phi(\cdot)}(x) \notin T_{\mathcal{W}}(x)$ and the conclusion follows. ■

Thus, the robot exits instantaneously the exceptional set once it has entered it. This then raises the question of how many times the robot visits the exceptional set. It will be shown that the robot can only visit this set a finite number of times. However, it is first necessary to show that, after a given time, it is impossible to revisit the exceptional manifold. To prove this result, it is necessary to construct a lower bound for V_0 , much in the same way as Lemma 4.3 gave an upper bound for the Lyapunov function V_0 :

Lemma 4.8 *There exists a time instant T_δ sufficiently large and a constant $d > 0$ such that the Lyapunov function V_0 given in Lemma 4.3 is bounded by*

$$V_0 \geq V_0(T_\delta) \exp(-dt), \quad \forall t \geq T_\delta \quad (4.74)$$

Proof: The proof follows exactly the same lines as those of Lemma 4.3 except for the fact that a lower bound of V_0 is needed. Using the inequality detailed in Lemma 4.3, \dot{V}_0 becomes:

$$\begin{aligned} \dot{V}_0 \geq & \left(-1 - 2 |\xi(t)| \left(\left\| P \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| + \left\| Q \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| \right) \right) \\ & (x_1^2 + x_2^2 + \chi_1^2 + \chi_2^2). \end{aligned}$$

Since $\xi(t)$ is exponentially stable, there will always exist a finite time T_δ for which

$$|\xi(t)| \leq \tilde{d} \quad \forall t \geq T_\delta.$$

Therefore, picking $\tilde{d} > 0$ sufficiently small so that there exists a positive constant d such that

$$\begin{aligned} \dot{V}_0 &\geq \left(-1 - 2|\xi(t)| \left(\left\| P \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| + \left\| Q \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| \right) \right) \\ &\quad (x_1^2 + x_2^2 + \chi_1^2 + \chi_2^2) \\ &\geq -dV_0 \quad \forall t \geq T_\delta. \end{aligned} \quad (4.75)$$

It follows that $V_0 \geq V_0(T_\delta)e^{-dt}$, $\forall t > T_\delta$. This concludes the proof. \blacksquare

Lemma 4.9 *There exists a time instant T_δ sufficiently large such that the solution $\phi(t)$ to system (4.2-4.4) together with the controller (4.20-4.23) does not visit the set $\mathcal{W} \setminus \mathcal{V}$, $\forall t > T_\delta$. Moreover, $\chi_1 \neq 0, \forall t > T_\delta$ and $\chi_2 \neq 0, \forall t > T_\delta$.*

Proof: In order to avoid the set $\mathcal{W} \setminus \mathcal{V}$, $\forall t > T_\delta$, it is sufficient to guarantee that the solution $\phi(t)$ to system (4.2-4.4) together with the controller (4.20-4.23) never visits the set $\{x_1, \chi_1, x_2, \chi_2, x_3 | \chi_1^2 + \chi_2^2 = 0\}$, $\forall t > T_\delta$. Let

$$V_\chi = \chi_1^2 + \chi_2^2. \quad (4.76)$$

Using (4.20-4.23), it follows that

$$\dot{V}_\chi = k_1 x_1 \chi_1 + k_3 x_2 \chi_2 + k_2 \chi_1^2 + k_4 \chi_2^2 \quad (4.77)$$

Now, \dot{V}_χ can be bounded:

$$\dot{V}_\chi \geq -|k_1 x_1 \chi_1| - |k_3 x_2 \chi_2| - |k_2 \chi_1^2| - |k_4 \chi_2^2| \quad (4.78)$$

Using the fact that $\|x\|P\|x\| \geq \lambda_{\min}(P)\|x\|^2$ together with Lemma 4.3, it can be concluded that

$$\begin{aligned} |k_1 x_1 \chi_1| &\leq \gamma_1 V_0 \\ |k_2 x_2 \chi_2| &\leq \gamma_2 V_0 \\ |k_3 \chi_1^2| &\leq \gamma_3 V_0 \\ |k_4 \chi_2^2| &\leq \gamma_4 V_0 \end{aligned}$$

where V_0 is the Lyapunov function (4.56) and $\gamma_1 > 0$, $\gamma_2 > 0$, $\gamma_3 > 0$, $\gamma_4 > 0$. Thus, using (4.78), \dot{V}_χ is bounded by:

$$\dot{V}_\chi \geq -(\gamma_1 + \gamma_2 + \gamma_3 + \gamma_4)V_0 = -\gamma V_0. \quad (4.79)$$

Using Lemma 4.8, it follows that

$$\dot{V}_\chi \geq -\gamma V_0(T_\delta)e^{-dt}, \quad \forall t > T_\delta \quad (4.80)$$

The solution of (4.80) is given by

$$V_\chi \geq \frac{d\gamma V_0(T_\delta)e^{-dt}}{d} + C, \quad \forall t > T_\delta \quad (4.81)$$

Thanks to Lemma 4.3, we know that the Lyapunov function

$V_0(x_1, \chi_1, x_2, \chi_2)$ converges exponentially to 0. Therefore x_1, χ_1, x_2, χ_2 all converge to 0, leading to $V_\chi \rightarrow 0$ when $t \rightarrow \infty$. Thus, evaluating (4.81) at $t = \infty$, the constant C should satisfy $C \leq 0$. Now, picking $C = 0$, it follows that

$$V_\chi \geq \frac{d\gamma V_0(T_\delta)e^{(-dt)}}{d}, \quad \forall t > T_\delta. \quad (4.82)$$

This implies that $V_\chi \neq 0, \quad \forall t > T_\delta$.

Changing V_χ in (4.76) into $V_\chi = \chi_1^2$, and $V_\chi = \chi_2^2$, respectively leads to the last two assertions, using the exact same reasoning as above with appropriate adaptations. This concludes the proof. \blacksquare

Lemma 4.10 *The solution to the system (4.2-4.4) together with (4.20-4.23) visits a finite number of times the set $\mathcal{W} \setminus \mathcal{V}$.*

Proof: Let $V_\chi = \chi_1^2 + \chi_2^2$. The value of V_χ along the solution to the system (4.2-4.4) together with (4.20-4.23) is studied. A Taylor series expansion is performed around $t = T$. Using the Lie derivative formulation, this results in

$$V_\chi(t - T) = \sum_{k=0}^{\infty} \frac{1}{(k+1)!} (L_F^k V_\chi) \Big|_{t=T} (t - T)^k \quad (4.83)$$

where F is the vector field (4.52). Now, the solution of V_χ is developed around $V_\chi(t_i) = 0$ and described only with the first three terms of (4.83):

$$V_\chi(\delta) = \left(\frac{1}{2}k_1^2 x_1(t_i)^2 + k_3^2 x_2(t_i)^2 \right) \delta^2 + O(\delta^3)$$

where $\delta = t - t_i$. Choosing x_1, x_2 such that $|x_1(t_i)| > \epsilon$ and $|x_2(t_i)| > \epsilon$, where ϵ is an arbitrary small positive constant, it follows that:

$$V_\chi(\delta) \geq \left(\frac{1}{2}(k_1^2 + k_3^2)\epsilon^2 \right) \delta^2 + O(\delta^3) \quad (4.84)$$

(4.84) states that there exists a $\bar{\delta}$ sufficiently small for which

$$V_\chi(t) > 0 \quad \forall t \in]t_i; t_i + \bar{\delta}[. \quad (4.85)$$

Let us suppose that the solution $\phi(t)$ to the system (4.2-4.4) together with (4.20-4.23) visits an infinite number of times the set $\mathcal{W} \setminus \mathcal{V}$. Therefore, there exists an infinite sequence of $t_i, i = 1, \dots, \infty$ for which $\phi(t_i) \in \mathcal{W} \setminus \mathcal{V}$. Additionally, Lemma 4.9 shows that $\phi(t) \notin \mathcal{W} \setminus \mathcal{V} \quad \forall t > T_\delta$. It follows that there exists an infinite sequence of $t_i < T_\delta, i = 1, \dots, \infty$ for which $\phi(t_i) \in \mathcal{W} \setminus \mathcal{V}$. Moreover, (4.85) ensures that $\phi(t) \notin \mathcal{W} \setminus \mathcal{V}, \forall t \in]t_i; t_i + \bar{\delta}[$. This generates an infinite number of disjoint intervals, each one of which has a finite measure. But, the sum of these measures must be finite since the sequence $t_i, (i = 1, \dots, \infty)$ must belong to the interval of finite measure $[0, T_\delta]$, which leads to a contradiction since an infinite sum of finite lower bounded quantities cannot be finite. \blacksquare

4.5.3 Convergence of x_3

The following lemmas give the asymptotic value of x_3 . By choosing conveniently the parameters of a reduced system given in Lemma 4.11, the final value of x_3 converges for most initial conditions to either $0 + 2k\pi$ or to $\pi + 2k\pi$ without oscillating. Then, through setting k_p large enough, the asymptotic behavior of the S_ξ -system can be made to match the one of the reduced system. Then using the appropriate value of β , based on the initial conditions of the robot, it is possible to enforce convergence to $0 + 2k\pi$ instead of $\pi + 2k\pi$, in case the initial conditions are not suitable.

Lemma 4.11 *Given the system*

$$\frac{d}{dt} \begin{pmatrix} \check{x}_1 \\ \check{\chi}_1 \\ \check{x}_2 \\ \check{\chi}_2 \end{pmatrix} = \begin{pmatrix} \check{\chi}_1 \\ k_1\check{x}_1 + k_2\check{\chi}_1 \\ \check{\chi}_2 \\ k_3\check{x}_2 + k_4\check{\chi}_2 \end{pmatrix} \quad (4.86)$$

with $\check{x}_3 = \arctan(\check{\chi}_1, \check{\chi}_2)$ and under the conditions that the poles of system (4.86), $r_1 = \frac{k_2 + \sqrt{4k_1 + k_2^2}}{2}$, $r_2 = \frac{k_2 - \sqrt{4k_1 + k_2^2}}{2}$, $r_3 = \frac{k_4 + \sqrt{4k_3 + k_4^2}}{2}$, $r_4 = \frac{k_4 - \sqrt{4k_3 + k_4^2}}{2}$ are real negative and satisfy $r_1 = r_2$, $r_3 = r_4$ and $r_1 > r_3$ then:

(i) when $-r_1\check{x}_1(0) + \check{\chi}_1(0) \neq 0$

$$\lim_{t \rightarrow \infty} \check{x}_3(t) = \begin{cases} -\pi & \text{if } (-r_1\check{x}_1(0) + \check{\chi}_1(0)) > 0 \\ 0 & \text{if } (-r_1\check{x}_1(0) + \check{\chi}_1(0)) < 0 \end{cases}$$

(ii) when $-r_1\check{x}_1(0) + \check{\chi}_1(0) = 0$ and $\check{x}_1(0) \neq 0$

$$\lim_{t \rightarrow \infty} \check{x}_3(t) = \begin{cases} -\pi & \text{if } \check{\chi}_1(0) > 0 \\ 0 & \text{if } \check{\chi}_1(0) < 0 \end{cases}$$

(iii) when $-r_1\check{x}_1(0) + \check{\chi}_1(0) = 0$, $\check{x}_1(0) = \check{\chi}_1(0) = 0$, $-r_3\check{x}_2(0) + \check{\chi}_2(0) = 0$ and $\check{x}_2(0) \neq 0$

$$\lim_{t \rightarrow \infty} \check{x}_3(t) = \begin{cases} -\frac{\pi}{2} & \text{if } (-r_3\check{x}_2(0) + \check{\chi}_2(0)) > 0 \\ \frac{\pi}{2} & \text{if } (-r_3\check{x}_2(0) + \check{\chi}_2(0)) < 0 \end{cases}$$

(iii) when $-r_1\check{x}_1(0) + \check{\chi}_1(0) = 0$, $\check{x}_1(0) = \check{\chi}_1(0) = 0$, $-r_3\check{x}_2(0) + \check{\chi}_2(0) = 0$ and $\check{x}_2(0) \neq 0$

$$\lim_{t \rightarrow \infty} \check{x}_3(t) = \begin{cases} -\frac{\pi}{2} & \text{if } \check{\chi}_2(0) > 0 \\ \frac{\pi}{2} & \text{if } \check{\chi}_2(0) < 0 \end{cases}$$

Proof: Due to the linearity of System (4.86), when the poles are multiple, the solution $\check{\chi}(t)$ becomes:

$$\check{x}_1(t) = e^{r_1 t}(\check{x}_1(0) + t(-r_1\check{x}_1(0) + \check{\chi}_1(0))) \quad (4.87)$$

$$\check{\chi}_1(t) = e^{r_1 t}(\check{\chi}_1(0) + r_1 t(-r_1\check{x}_1(0) + \check{\chi}_1(0))) \quad (4.88)$$

$$\check{x}_2(t) = e^{r_3 t}(\check{x}_2(0) + t(-r_3\check{x}_2(0) + \check{\chi}_2(0))) \quad (4.89)$$

$$\check{\chi}_2(t) = e^{r_3 t}(\check{\chi}_2(0) + r_3 t(-r_3\check{x}_2(0) + \check{\chi}_2(0))) \quad (4.90)$$

Now, the limit of $\check{x}_3 = \arctan(\check{\chi}_1(t), \check{\chi}_2(t))$ for $t \rightarrow \infty$ is considered. Substituting (4.88) and (4.90) in $\arctan(\check{\chi}_1(t), \check{\chi}_2(t))$ and after a few algebraic manipulations together with the fact that $r_1 = r_2$, $r_3 = r_4$ and $r_1 > r_3$, the following result holds: when $-r_1\check{x}_1(0) + \check{\chi}_1(0) \neq 0$

$$\lim_{t \rightarrow \infty} \check{x}_3 = -\iota \ln \frac{-r_1\check{x}_1(0) + \check{\chi}_1(0)}{\sqrt{(-r_1\check{x}_1(0) + \check{\chi}_1(0))^2}}$$

$$\lim_{t \rightarrow \infty} \check{x}_3(t) = \begin{cases} 0 & \text{if } (-r_1\check{x}_1(0) + \check{\chi}_1(0)) > 0 \\ -\pi & \text{if } (-r_1\check{x}_1(0) + \check{\chi}_1(0)) < 0 \end{cases} \quad (4.91)$$

The other cases are studied along similar lines. ■

Remark 4.11 *The result of Lemma 4.11 means that the mobile robot reaches the origin with a horizontal tangent ($\check{x}_3 = 0$ or π), because \check{x}_2 approaches zero faster than \check{x}_1 thanks to its faster poles.*

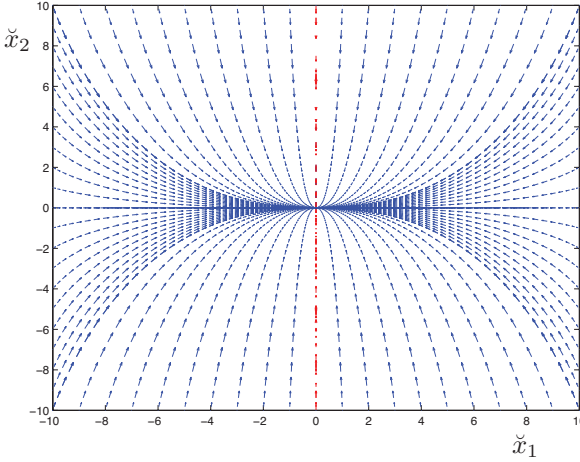


Figure 4.15: Solutions of system (4.86) when $\check{x}_1(0) = 0$ and $\check{x}_2(0) = 0$. In this case, one can see that when $\check{x}_1(0) < 0$, $\check{x}_3 = \arctan(\check{\chi}_1, \check{\chi}_2)$ converges to 0 and when $\check{x}_1(0) > 0$, \check{x}_3 converges to π . In the abnormal case where $\check{x}_1(0) = 0$, \check{x}_3 converges to either $\frac{\pi}{2}$ or to $-\frac{\pi}{2}$. This illustrates the results of Lemma 4.11.

Lemma 4.12 *There exists a certain $T_\delta = 0$ for which the system (4.2)-(4.4) with the controller of Proposition 4.1 ensures that*

$$\lim_{t \rightarrow \infty} x_3(t) = \begin{cases} \pi + 2k\pi & \text{if } (-r_1 x_1(T_\delta) + \chi_1(T_\delta)) > 0 \\ 0 + 2k\pi & \text{if } (-r_1 x_1(T_\delta) + \chi_1(T_\delta)) < 0 \end{cases}$$

Proof: Let $\chi(t)$ be a solution of System (4.39)-(4.42) and $\check{\chi}(t)$ be a solution to System (4.86). According to Lemma 4.9, there exists a T_δ such that $\forall t > T_\delta$, χ_1 and χ_2 do not visit the set $\mathcal{W} \setminus \mathcal{V}$. Additionally, this lemma also guarantees that χ_1 and χ_2 never become 0, $\forall t > T_\delta$. Consider the error between $\arctan(\check{\chi}_1, \check{\chi}_2)$ and $\arctan(\chi_1, \chi_2)$:

$$\| \arctan(\check{\chi}_1, \check{\chi}_2) - \arctan(\chi_1, \chi_2) \| \tag{4.92}$$

The error variables w_1 and w_2 are defined such that

$$\begin{aligned} \chi_1 &= \check{\chi}_1 + w_1 \\ \chi_2 &= \check{\chi}_2 + w_2 \end{aligned}$$

Therefore, (4.92) becomes

$$\begin{aligned} & \left\| \arctan(\check{\chi}_1, \check{\chi}_2) - \arctan(\check{\chi}_1 + w_1, \check{\chi}_2 + w_2) \right\| \leq \\ & \left\| \arctan(\check{\chi}_1, \check{\chi}_2) - \arctan(\check{\chi}_1 + w_1, \check{\chi}_2) \right\| + \\ & \left\| \arctan(\check{\chi}_1 + w_1, \check{\chi}_2) - \arctan(\check{\chi}_1 + w_1, \check{\chi}_2 + w_2) \right\| = \nu. \end{aligned}$$

In order to find an upper bound for ν , the mean value theorem is used. In this respect, the differentiability of the function $\arctan(a, b)$ needs to be discussed. Since $\frac{\partial \arctan(a, b)}{\partial a}$ and $\frac{\partial \arctan(a, b)}{\partial b}$ are well-defined on $\mathbb{R} \times \mathbb{R} \setminus \{0, 0\}$, the mean value theorem can be used only if the point $\{0, 0\}$ is avoided. Looking at (4.88) and (4.90), one can see that $\check{\chi}_1$ and $\check{\chi}_2$ never become 0. Since χ_1 and χ_2 also never become 0, $\forall t > T_\delta$, it follows that there exists a $\rho_1 \in [0, 1]$ and $\rho_2 \in [0, 1]$ for which

$$\begin{aligned} \nu & \leq \left\| \frac{\partial \arctan(\check{\chi}_1, \check{\chi}_2)}{\partial \check{\chi}_1} \right\|_{\check{\chi}_1 = \check{\chi}_1 + \rho_1 w_1} \left\| w_1 \right\| + \\ & \left\| \frac{\partial \arctan(\check{\chi}_1 + w_1, \check{\chi}_2)}{\partial \check{\chi}_2} \right\|_{\check{\chi}_2 = \check{\chi}_2 + \rho_2 w_2} \left\| w_2 \right\| \\ & \leq \|w_1\| + \|w_2\|, \quad \forall t > T_\delta \end{aligned} \tag{4.93}$$

Since χ and $\check{\chi}$ are exponentially decreasing $\forall t > T_\delta$ together with the fact that $\|\chi - \check{\chi}\| \leq \|\chi\| + \|\check{\chi}\|$, it follows that $\|w_1\| \leq \gamma_1 e^{-\gamma_2 t}$ and $\|w_2\| \leq \gamma_3 e^{-\gamma_4 t}$, with $\gamma_1, \gamma_3 \in \mathbb{R}$ and $\gamma_2, \gamma_4 \in \mathbb{R}_+$. This means that ν converges to zero owing to (4.93). Since $\xi = -k_p \xi$, together with the fact that $\xi = \bar{x}_3 - \arctan(\chi_1, \chi_2) = x_3 + 2k\pi - \arctan(\chi_1, \chi_2)$, \bar{x}_3 converges exponentially to $\arctan(\chi_1, \chi_2)$. Finally, since $\|\arctan(\check{\chi}_1, \check{\chi}_2) - \arctan(\chi_1, \chi_2)\|$ converges to zero, \bar{x}_3 converges to $\arctan(\check{\chi}_1, \check{\chi}_2)$. ■

4.5.4 Partition of the initial conditions

Lemma 4.13 $\lim_{t \rightarrow \infty} x_3$ converges to either $\frac{\pi}{2} + 2k\pi$ or $-\frac{\pi}{2} + 2k\pi$ only if $x_0 \in \mathcal{X}_p$, where

$$\begin{aligned} \mathcal{X}_p & = \{x_1 = 0, \chi_1 = 0, x_2 \in \mathbb{R}, \chi_2 \in \mathbb{R}, x_3 = \frac{\pi}{2} + 2k\pi\} \\ & \cup \{x_1 = 0, \chi_1 = 0, x_2 \in \mathbb{R}, \chi_2 \in \mathbb{R}, x_3 = -\frac{\pi}{2} + 2k\pi\} \end{aligned}$$

Proof:

Recall that ξ decreases quickly ($\dot{\xi} = -k_p \xi$) so that, after a given time T_δ , the system is guaranteed to converge to either $0 + 2k\pi$ or to $\pi + 2k\pi$, depending on the sign of $-r_1 x_1(T_\delta) + \dot{x}_1(T_\delta)$. Therefore, according to Lemma 4.12, the only way for x_3 to converge to a value different from $0 + 2k\pi$ or $\pi + 2k\pi$ is that the condition $-r_1 x_1(t) + \dot{x}_1(t) = 0$ holds for all $t \geq T_\delta$. This means studying the condition under which $\mathfrak{X} = \{x_1, \chi_1, x_2, \chi_2, x_3 \mid -r_1 x_1(t) + \dot{x}_1(t) = 0\}$ becomes an invariant set. Notice that (4.45) can be put in the form $\dot{x}_1 = \chi_1 \cos \xi - \chi_2 \sin \xi$. Also, without loss of generality consider (4.46) with the particular choice $k_1 = -k^2$ and $k_2 = -2k$. Moreover, set $r_1 = -k$, again without loss of generality. Now, constitute the two-dimensional vector field

$$f = \begin{pmatrix} \dot{x}_1 \\ \dot{\chi}_1 \end{pmatrix} = \begin{pmatrix} \chi_1 \cos \xi - \chi_2 \sin \xi \\ -k^2 x_1 - 2k \chi_1 \end{pmatrix} \quad (4.94)$$

If the system is contrived to remain in the set \mathfrak{X} , then the vector field describing the full five state system (i.e. with state space $x_3, x_1, x_2, \chi_1, \chi_2$) should belong to the set $\{x_1 = r_1 \alpha, \chi_1 = \alpha, x_2 = \alpha_1, \chi_2 = \alpha_2, x_3 = \alpha_3 \mid \alpha \in \mathbb{R}, \alpha_1 \in \mathbb{R}, \alpha_2 \in \mathbb{R}, \alpha_3 \in \mathbb{R}\}$. A necessary condition for this to occur is that the following determinant vanishes:

$$\begin{vmatrix} 1 & \chi_1 \cos \xi - \chi_2 \sin \xi \\ k & -k^2 x_1 - 2k \chi_1 \end{vmatrix} = -k \chi_1 + k \chi_1 \cos \xi - k \chi_2 \sin \xi = 0,$$

($x_1 = -\frac{\chi_1}{k}$). Now, there are three cases that satisfy this necessary condition:

1. $\xi = 0 \Rightarrow -k \chi_1 + k \chi_1 - 0 = 0$.
2. $\xi \neq 0, \chi_1 = 0 \Rightarrow$ either $\xi = \pm \frac{\pi}{2}$ or $\chi_2 = 0$.
3. $\xi \neq 0, \chi_2 = 0 \Rightarrow +k(\cos \xi - 1) = 0 \Rightarrow$ either $\xi = 0$ or $\xi = 2\pi$.
4. $k \chi_1(\cos \xi - 1) - k \chi_2 \sin \xi = 0$.

Case 4) cannot happen since this would contradict the dynamics. The conclusion follows after observing that case 2) with $\xi \pm \frac{\pi}{2}$ also cannot occur since $\xi = -k_p \xi$, meaning that ξ cannot stay at a constant value. Moreover, case 2) with $\chi_2 = 0$ corresponds to crossing the exceptional set. Then, due to transversality exposed in Lemma 4.7, the system will instantaneously leave this case and the asymptotic convergence properties are then resolved using the other cases. Now, only the case 1) and 3) holds which means that

$\xi = 0, \forall t$. Therefore, the S_ξ -system (4.45)-(4.46) becomes

$$\begin{aligned}\dot{x}_1 &= \chi_1 \\ \dot{\chi}_1 &= k_1 x_1 + k_2 \chi_1 \\ \dot{x}_2 &= \chi_2 \\ \dot{\chi}_2 &= k_3 x_2 + k_4 \chi_2\end{aligned}$$

which is nothing but (4.86) for which is known (Lemma 4.11) that whenever $-r_1 x_1 + \chi_1 = 0$, then

$$\lim_{t \rightarrow \infty} x_3(t) = \pm \frac{\pi}{2} \quad (4.95)$$

■

4.5.5 Proof of the main propositions

Proof of Proposition 4.1

Proposition 4.1 *Choose the controller gains $k_i, 1 \leq i \leq 4$, such that the poles $r_1 = \frac{k_2 + \sqrt{4k_1 + k_2^2}}{2}$, $r_2 = \frac{k_2 - \sqrt{4k_1 + k_2^2}}{2}$, $r_3 = \frac{k_4 + \sqrt{4k_3 + k_4^2}}{2}$, $r_4 = \frac{k_4 - \sqrt{4k_3 + k_4^2}}{2}$ are negative real, and consider the system (4.2)-(4.4) together with the following controller:*

$$u_1 = \sqrt{\chi_1^2 + \chi_2^2} \quad (4.96)$$

$$u_2 = \frac{\dot{\chi}_2 \chi_1 - \chi_2 \dot{\chi}_1}{\chi_1^2 + \chi_2^2} - k_p (\bar{x}_3 - \arctan(\chi_1, \chi_2)), \quad (4.97)$$

$$\dot{\chi}_1 = k_1 x_1 + k_2 \chi_1 \quad (4.98)$$

$$\dot{\chi}_2 = k_3 x_2 + k_4 \chi_2, \quad (4.99)$$

where \bar{x}_3 is given as $\bar{x}_3 = x_3 + 2k\pi$ with k a discrete state determined as follows: k is initialized at zero and updated at specific time instants. Let t_i denote any time instant for which the system states are such that $\chi_1(t_i) < 0$ and $\chi_2(t_i) = 0$. Then, update k according to the following rule:

$$\begin{aligned}k &:= k - 1 & \text{if } \dot{\chi}_2(t_i) < 0 \\ k &:= k + 1 & \text{if } \dot{\chi}_2(t_i) > 0 \\ k &:= k + \text{sgn}(k_3 \sin x_3(t_i)) & \text{if } x_2(t_i) = 0.\end{aligned} \quad (4.100)$$

Under these conditions, x_1 and x_2 converge to the origin exponentially. Moreover, when the poles are such that $r_1 = r_2$, $r_3 = r_4$ and $r_1 > r_3$, x_3 converges to one of four values, namely $0 + 2k\pi$, $\pi + 2k\pi$, $\frac{\pi}{2} + 2k\pi$ and $-\frac{\pi}{2} + 2k\pi$. Therefore, the initial conditions become partitioned into four distinct sets depending on the asymptotic value of their corresponding trajectory, namely the sets of initial conditions \mathcal{X}_0 , \mathcal{X}_π , $\mathcal{X}_{\frac{\pi}{2}}$, and $\mathcal{X}_{-\frac{\pi}{2}}$.

Proof: The proposition follows by the various lemmas and propositions. Lemma 2.1 guarantees that there is no finite escape time in the solutions of the S_ξ -system. This is also the case for the S_{x_3} -system, as long as the solutions to each of these systems are in diffeomorphic correspondence. This is true outside the transition and exceptional sets. Under such circumstances, Lemmas 2.1, 4.2 and 4.3, together with the diffeomorphic correspondence, guarantee that all states x_1 , x_2 , ξ_1 , ξ_2 and x_3 decrease. However, when crossing the transition set, the implicit resetting mechanism (see Lemma 4.4) shows that the solutions can be defined so as to maintain the diffeomorphic correspondence.

Nevertheless, the problem of visiting the exceptional set must still be clarified. Lemma 4.6 shows that, when crossing this set, only x_3 is affected and at most with a finite jump of $\pm\pi$, and therefore this does not affect the convergence process. Moreover, Lemma 4.10 shows that the number of visits of the exceptional set is only finite after considering the amount of time needed for the robot to be in a position to cross this set again. Finally, Lemma 4.12 guarantees that the angle x_3 of the nonlinear system converges to either $0 + 2k\pi$ or to $\pi + 2k\pi$, since the system trajectory converges to the one of the linear system which was, in turn, shown to converge to either one of these values. Therefore, the convergence depends on which side of the state space the system is initially in, i.e. in either \mathcal{X}_0 or \mathcal{X}_π . Nevertheless the topology of the sets \mathcal{X}_0 and \mathcal{X}_π is hard to characterize analytically. Moreover Lemma 4.13 states the conditions of the abnormal set of initial conditions for which x_3 converges to either $\frac{\pi}{2} + 2k\pi$ or to $-\frac{\pi}{2} + 2k\pi$. ■

Proof of Proposition 4.2

Proposition 4.2 *Let the hypothesis and the controller be as in Proposition 4.1. Moreover, consider the system (4.25) together with this controller. Under these conditions both x_1 and x_2 converge to 0 exponentially as long as k_p is chosen sufficiently large. Moreover, when the poles are such that*

$r_1 = r_2$, $r_3 = r_4$ and $r_1 > r_3$, x_3 converges to one of two values, namely $0 + 2k\pi + \frac{\delta}{k_p}$, $\pi + 2k\pi + \frac{\delta}{k_p}$.

Proof: The proof follows roughly the same lines as those above (Proposition 4.1). However, the S_ξ -system defined by (4.101)-(4.103) is slightly different considering the perturbed mobile robot (4.25): the S_ξ -system becomes

$$\dot{x}_1 = \sqrt{\chi_1^2 + \chi_2^2} \cos(\xi + \arctan(\chi_1, \chi_2)) \quad (4.101)$$

$$\dot{\chi}_1 = k_1 x_1 + k_2 \chi_1$$

$$\dot{x}_2 = \sqrt{\chi_1^2 + \chi_2^2} \sin(\xi + \arctan(\chi_1, \chi_2)) \quad (4.102)$$

$$\dot{\chi}_2 = k_3 x_2 + k_4 \chi_2$$

$$\dot{\xi} = -k_p \xi + \delta. \quad (4.103)$$

since δ is just added to \dot{x}_3 in (4.43). Equation (4.103), shows that the variable ξ converges exponentially to $\frac{\delta}{k_p}$. In particular, considering a given positive constant \tilde{c} arbitrary small, there will always exist a finite time T and a sufficiently large k_p for which

$$|\xi(t)| \leq \tilde{c} \quad \forall t \geq T \quad (4.104)$$

Therefore, Lemma 4.3 can be adapted by just adding the condition that k_p must be chosen sufficiently large. Indeed, in Lemma 4.3, the condition (4.104) needs to be satisfied (see (4.69)) to conclude on the exponential convergence of x_1 and x_2 to 0. Now, the asymptotic value of x_3 needs to be discussed. This can be done by modifying slightly Lemma (4.12), which leads to the following lemma

Lemma 4.14 *There exists a certain $T_\delta = 0$ for which the perturbed mobile robot (4.25) with the controller of Proposition 4.1 ensures that*

$$\lim_{t \rightarrow \infty} x_3(t) = \begin{cases} \pi + 2k\pi + \frac{\delta}{k_p} & \text{if } (-r_1 x_1(T_\delta) + \chi_1(T_\delta)) > 0 \\ 0 + 2k\pi + \frac{\delta}{k_p} & \text{if } (-r_1 x_1(T_\delta) + \chi_1(T_\delta)) < 0 \end{cases}$$

Proof: The proof is the same as in Lemma 4.12 except for the last arguments. Indeed, in this case, the ξ dynamics is $\dot{\xi} = -k_p \xi + \delta$, meaning that ξ converges to $\frac{\delta}{k_p}$. Since ξ is defined as $\xi = \bar{x}_3 - \arctan(\chi_1, \chi_2) = x_3 + 2k\pi - \arctan(\chi_1, \chi_2)$, \bar{x}_3 converges exponentially to $\arctan(\chi_1, \chi_2) +$

$\frac{\delta}{k_p}$. Finally, since $\|\arctan(\check{\chi}_1, \check{\chi}_2) - \arctan(\chi_1, \chi_2)\|$ converges to zero, \bar{x}_3 converges to $\arctan(\check{\chi}_1, \check{\chi}_2) + \frac{\delta}{k_p}$. ■

Lemma 4.14 does not consider the case $(-r_1 x_1(T_\delta) + \chi_1(T_\delta) = 0$. However, this is never fulfilled contrary to Lemma 4.13 (e.g. $\delta = 0$). Indeed, following the same reasoning as in the proof of Lemma 4.13, we get that ξ should be equal to 0 for all $t > T_\delta$ to match this exceptional conditions. However this is impossible since $\xi = -k_p \xi + \delta$. Using Lemma 4.14 and the fact that x_1 and x_2 have been shown to converge exponentially to 0 concludes the proof. ■

4.5.6 A modification of Proposition 4.1

Replacing \bar{x}_3 by x_3 in the controller of Proposition 4.1 leads unfortunately to a false conjecture concerning its convergence. However, the controller was satisfactory for most initial conditions, as it has been shown in the simulation section.

Conjecture 5 *Let all Hypotheses of Proposition 4.1 hold except that \bar{x}_3 is replaced by x_3 in equation (4.21), k_p is chosen sufficiently large, and the updating of k is suppressed. Under these new assumptions, x_1 and x_2 converge to the origin for most initial conditions. Moreover, when the poles are such that $r_1 = r_2$, $r_3 = r_4$ and $r_1 > r_3$, x_3 then x_3 converges to 0, π , $\frac{\pi}{2}$, and $-\frac{\pi}{2}$ depending on the initial conditions being respectively in the four sets \mathcal{X}_0 , \mathcal{X}_π , $\mathcal{X}_{\frac{\pi}{2}}$, and $\mathcal{X}_{-\frac{\pi}{2}}$.*

A tentative proof of this conjecture could run along the following lines. By defining

$$\xi = x_3 - \arctan(\chi_1, \chi_2),$$

instead of (4.44), ξ and x_3 are both in diffeomorphic correspondence as long as the transition set is not reached. However, when the system reaches the transition set $\chi_2 = 0$, $\chi_1 < 0$, then the discontinuity in the arctangent function prevents both ξ and x_3 from being continuous. The controller in Proposition 4.1 handled the problem by admitting a discontinuity in \bar{x}_3 and defined ξ to be equal to (4.44), maintaining both x_3 and ξ continuous. This had the disadvantage of only guaranteeing convergence of the true angle x_3 to the desired one modulo 2π .

This time, the discontinuity is forced on ξ , and therefore all previous arguments concerning the convergence of the states should be adapted to

take into consideration the fact that ξ , while obeying $\dot{\xi} = k_p \xi$, is reinitialized each time the transition set is encountered.

This means that the convergence of x_3 could be guaranteed if V_0 would decrease between two visits to the transition set. Although the increase in V_0 due to the transient (caused by the discontinuity in ξ) can be ensured by simply increasing the gain k_p , it is not easy to guarantee that V_0 decreases sufficiently before another discontinuity in ξ occurs. Indeed, consider the condition of visiting the transition set $\chi_2 = 0$, $\chi_1 < 0$, which means studying the dynamics of χ_1 and χ_2 . Therefore, the time instants t_k for which the system visits the transition set are defined by $\chi_2(t_k) = 0$, $\chi_1(t_k) < 0$. Now, the dynamics of χ_2 around $\chi_2(t_k)$ is obtained by differentiating (4.42) and using (4.41), so that

$$\ddot{\chi}_2 = k_1 \sqrt{\chi_1^2 + \chi_2^2} \sin(x_3) + k_2 \dot{\chi}_2 \quad (4.105)$$

Using (4.67), it can be concluded that there exist two positive constants a and b such that $\sqrt{\chi_1^2 + \chi_2^2} \leq ae^{bt}$. It follows that (4.105) becomes

$$k_1 ae^{bt} + k_2 \dot{\chi}_2 \leq \ddot{\chi}_2 \leq -k_1 ae^{bt} + k_2 \dot{\chi}_2 \quad (4.106)$$

If it were possible, using the two bounding solutions in (4.106), to find a lower bound on the time for which $\chi_2(t)$ revisits the transition set (i.e. $\exists \delta > 0$ such that whenever $\chi_2(t_{k+1}) = 0$ implies that $t_{k+1} > t_k + \delta$) then it would be possible to find a gain k_p that is sufficiently large to guarantee convergence irrespective of the initial conditions. However, this is not possible since when $\chi_2 = 0$ then $\dot{\chi}_2 = -k_1 x_2$ meaning that by choosing x_2 to be small, the solutions of (4.105) cannot be limited by a solution guaranteeing the existence of the bound δ . It is this argument that helped in finding the initial conditions that led to the behavior illustrated in Figure 4.7.

4.5.7 Proof of Proposition 4.3

Proposition 4.3 *Consider the system (4.36)-(4.38) with the controller (4.28)-(4.31). Assume that the controller is such that the poles $r_1 = \frac{k_2 + \sqrt{4k_1 + k_2^2}}{2}$, $r_2 = \frac{k_2 - \sqrt{4k_1 + k_2^2}}{2}$, $r_3 = \frac{k_4 + \sqrt{4k_3 + k_4^2}}{2}$, $r_4 = \frac{k_4 - \sqrt{4k_3 + k_4^2}}{2}$ are negative and real. Then, x_1 and x_2 converge to x_{1ref} and x_{2ref} exponentially.*

Proof: The proof follows roughly the same lines as those above (Proposition 4.1). However, Lemma 4.3 is replaced by Lemma 4.15. ■

Lemma 4.15 *Let P and Q be chosen to be identical to those for Lemma 4.3 Then,*

$$V_0 = \begin{pmatrix} e_{x_1} & e_{\chi_1} \end{pmatrix} P \begin{pmatrix} e_{x_1} \\ e_{\chi_1} \end{pmatrix} + \begin{pmatrix} e_{x_2} & e_{\chi_2} \end{pmatrix} Q \begin{pmatrix} e_{x_2} \\ e_{\chi_2} \end{pmatrix}$$

converges to zero exponentially.

Proof: This proof follows similar lines to those for Lemma 4.3. For this reason, only the main differences are briefly set out below. The first difference stems from the fact that (4.57)-(4.60) changes into

$$\begin{aligned} \dot{e}_{x_1} &= e_{\chi_1} + \sqrt{(\dot{x}_{1ref} + e_{\chi_1})^2 + (\dot{x}_{2ref} + e_{\chi_2})^2} \\ &\quad (\cos(\hat{x}_3 + \xi) - \cos(\hat{x}_3)) \\ \dot{e}_{\chi_1} &= k_1 e_{x_1} + k_2 e_{\chi_1} \\ \dot{e}_{x_2} &= e_{\chi_2} + \sqrt{(\dot{x}_{1ref} + e_{\chi_1})^2 + (\dot{x}_{2ref} + e_{\chi_2})^2} \\ &\quad (\sin(\hat{x}_3 + \xi) - \sin(\hat{x}_3)) \\ \dot{e}_{\chi_2} &= k_3 e_{x_2} + k_4 e_{\chi_2}. \end{aligned}$$

Then, the reference terms that appear under the square root lead to a modification of (4.64) to

$$\begin{aligned} &\|2 \begin{pmatrix} e_{x_1} & e_{\chi_1} \end{pmatrix} P \begin{pmatrix} 1 \\ 0 \end{pmatrix} \sqrt{\chi_1^2 + \chi_2^2} \xi\| \\ &\leq | \alpha_1 e_{x_1} + \alpha_2 e_{\chi_1} | (| \chi_1 | + | \chi_2 |) | \xi | \\ &= | \alpha_1 e_{x_1} + \alpha_2 e_{\chi_1} | \\ &\quad (| e_{x_1} + \dot{x}_{1ref} | + | e_{\chi_2} + \dot{x}_{2ref} |) | \xi | \\ &\leq 2\tilde{c}_1 \left(V_0 + \sqrt{V_0} \right) | \xi | . \end{aligned}$$

By analogy, a similar result can be found for the term involving Q . Finally, after a few algebraic manipulations, there exist positive constants c_1, c_2 such that

$$\dot{V}_0 \leq -2c_1 V_0 + 2c_2 (V_0 + \sqrt{V_0}) | \xi | \quad (4.107)$$

Here one can see that (4.107) has the same structure as (46), page 750, in [43]. Now using the result in [43], it can be concluded that (4.107) exhibits no finite escape time and converges exponentially to 0. \blacksquare

4.6 Experimental Verification

In this section, the control methodology is validated using real-time experiments performed with the autonomous mobile robot *Fouzy III* depicted in Figure 4.16. The robot weighs about 15 kg for dimensions of 0.3 m by 0.3 m and a height of 0.4 m. Its mass and inertia are not negligible. Due to space limitations, two independent motors, each with a power consumption of 23 W, drive the wheels through belt transmissions. The center of mass lies in front of the wheel axle.

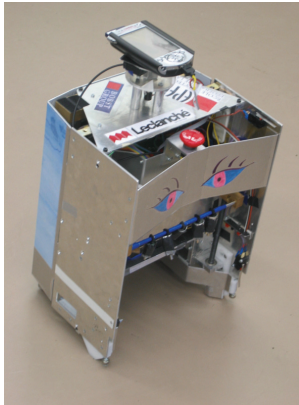


Figure 4.16: The robot *Fouzy III*.

The electronic components include an ARM processor, a low-level microcontroller and two pulse-width modulation devices (PWM), each one supplying adequate power to its corresponding motor. The processors are programmed in C language. The control loop runs at 100 Hz. The controller implements the algorithm described in Section 4.3, with some important differences detailed next.

4.6.1 Model mismatch and hardware limitations

The high-level controller given in Section 4.3 is based on a kinematic model of the robot. Hence, it assumes that the velocities u_1 and u_2 can be imposed directly on the robot. However, this is experimentally impossible due to inertia and the motor characteristics. Hence, it is necessary to have a

low-level, high-gain controller that speeds up the low-level activities and makes the higher level look like the kinematic model that has been assumed. A linear quadratic regulator (LQR) with integral action is used to this effect. In addition, this low-level controller decreases the effect of nonlinearity stemming from motor dead-zone and belt transmission. Nevertheless, the velocities are inherently limited by the motor characteristics, which is another difference with the theoretical part of the thesis. These velocity bounds can be enforced in an open-loop manner by the controller if the robot slips on the surface on which it moves.

Note that the robot knows its position through the use of encoders mounted on the motors. It computes its odometry with a discrete model. This way of estimating the position is not ideal since the robot can slip. Although the position measurement may be crude, it is sufficient for our purpose.

4.6.2 Control implementation

The implementation of the high-level controller has been done on the ARM processor. However, since the architecture lacks a floating point unit, it is very slow in processing numbers that are not integers. Some adaptations have been necessary. For example, mathematical functions have been re-programmed (i.e. sin and cos) and set in a table in RAM memory and optimized to work with integer arithmetic.

Apart from the intrinsic model mismatch and hardware limitations described in the previous section, the equations given in the theoretical part need some further processing. Indeed, they are given in continuous time, but the robot interface is discrete. The following adaptations have been made:

- The time variable t becomes kh , where h is the sampling period (10 ms).
- Using Euler's first-order approximation, Equations (4.30) and (4.31) become (after setting $e_{\chi_j}(k) = \chi_j(k) - \dot{\chi}_{jref}(k)$):

$$\chi_1(k+1) = \chi_1(k) + (k_1 e_{\chi_1}(k) + k_2 e_{\chi_1}(k) + \ddot{\chi}_{1ref}(k)) h \quad (4.108)$$

$$\chi_2(k+1) = \chi_2(k) + (k_3 e_{\chi_2}(k) + k_4 e_{\chi_2}(k) + \ddot{\chi}_{2ref}(k)) h, \quad (4.109)$$

where $x_{jref}(k)$, $j = 1, 2$, and their derivatives are sampled versions of their continuous-time analog.

- Two modifications are introduced in (4.29): (i) The term $(x_3 - \arctan(\chi_1, \chi_2))$ is bounded between $[-\pi; \pi[$ to avoid trigonometric circle discontinuity; (ii) The continuous part

$$\psi(t) = \frac{\dot{\chi}_2\chi_1 - \dot{\chi}_1\chi_2}{\chi_1^2 + \chi_2^2}, \quad (4.110)$$

is approximated by the discretized version

$$\psi(k) = \frac{\chi_1(k-1)\chi_2(k) - \chi_2(k-1)\chi_1(k)}{h(\chi_1^2(k) + \chi_2^2(k))}. \quad (4.111)$$

4.6.3 Experimental results

Two experiments have been carried out, one in stabilization at the origin and one in circular trajectory tracking.

Stabilization at the origin with smooth controller

The parameters used in this experiment are given in Table 4.7. The controller parameters k_i , $i = 1, \dots, 4$ are different from those used in simulation. Here, a sufficiently high ratio must exist between the dominant poles (the slow ones) r_1 and r_3 . This ratio has been set to 4, making the robot go faster along x_2 than along x_1 . The fast poles r_2 and r_4 are set equal.

Figure 4.17 shows the way the controller brings the system to $x_1 = 0$ and $x_2 = 0$. No difficulty is encountered, and x_3 converges to 0 according to Lemma 4.12. However, Part (iv) of Figure 4.17 shows that u_2 can get quite noisy.

This noise amplification stems from the computation in (4.111) and has two causes. On the one hand, (4.111) is a numerical differentiation and is inherently sensitive to noise. On the other hand, when $\chi_1^2(k) + \chi_2^2(k)$ becomes small, (4.111) approaches singularity. This last point is responsible for the strong noise amplification visible in Part (iv) around $t = 10$ s.

An interesting observation, already mentioned in the theoretical part of the paper, is that singularity crossing has little effect on the robot performance for two reasons:

- Firstly, Lemma 4.10 ensures that this can happen only a finite number of times. The consequence of getting close to the singular manifold, or

even momentarily entering it, is an abrupt increase in angular velocity (i.e. high \dot{x}_3). However, as mentioned previously, such a high velocity cannot be imposed instantaneously by the low-level controller despite its high gain.

- Secondly, two important aspects limit the effect of high angular velocity on x_1 and x_2 . This can be seen through examination of (4.2) and (4.3). On the one hand, $\sin(x_3)$ and $\cos(x_3)$ remain bounded. On the other hand, u_1 stays small in the vicinity of the exceptional set. Indeed, singularity crossing happens at zero u_1 velocity since, when $\chi_1 = \chi_2 = 0$, $u_1 = \sqrt{\chi_1^2 + \chi_2^2} = 0$.

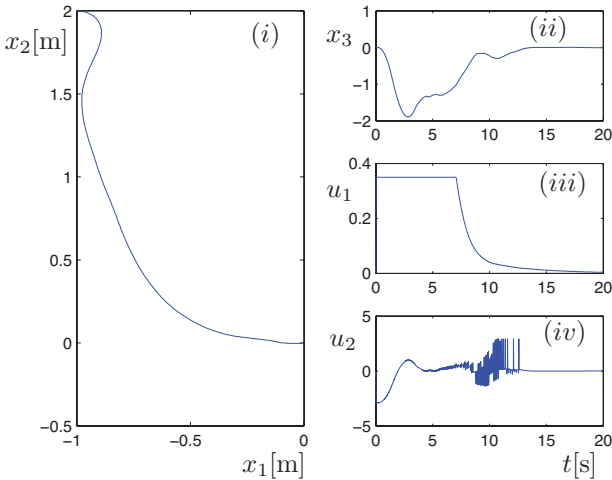


Figure 4.17: Experimental stabilization at the origin using a smooth controller with $\psi(k) = 0$ given by (4.111).

Noise reduction for u_2 : The results in Figure 4.17 show a large sensitivity of u_2 to noise. The control performance is not degraded itself, but a noisy input can rapidly wear down the actuator. For this reason, Equation (4.111) is modified to

$$\psi(k) = \frac{\ddot{x}_{2ref}(k)\dot{x}_{1ref}(k) - \ddot{x}_{1ref}(k)\dot{x}_{2ref}(k)}{\dot{x}_{1ref}^2(k) + \dot{x}_{2ref}^2(k)}, \quad (4.112)$$

where the noisy terms χ_1 and χ_2 have been replaced by the reference values \dot{x}_{1ref} and \dot{x}_{2ref} .

Remark 4.12 For stabilization, $\psi(k) = 0$ since $x_{1ref} = 0$ and $x_{2ref} = 0$.

Since Equation (4.112) is a feedforward term that does not depend on measurements, it is not sensitive to noise. Equation (4.112) is an approximation of (4.111) with the assumption that $\chi_1 \cong \dot{x}_{1ref}$ and $\chi_2 \cong \dot{x}_{2ref}$. This is a strong assumption, which is validated by very convincing experimental results. A comparison of the trajectories in Figure 4.17 using (4.111) with those in Figure 4.18 using (4.112) shows that the robot behaves nearly identically. Moreover, the noise on the input signal u_2 is strongly reduced.

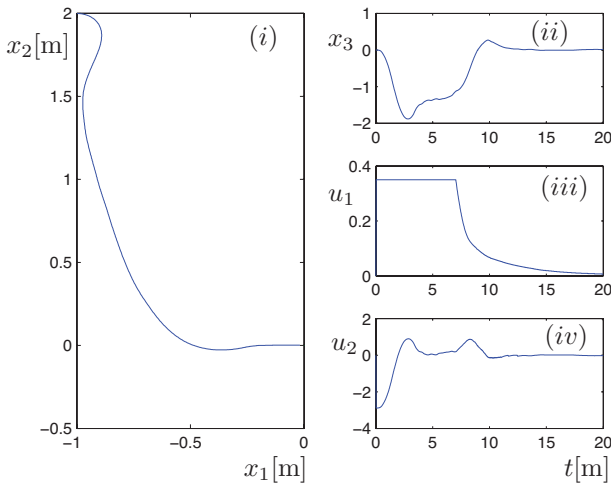


Figure 4.18: Experimental stabilization at the origin using a smooth controller with $\psi(k) = 0$ given by (4.112).

Stabilization at the origin with discontinuous controller

As explained in Section 4.3, when the initial conditions are not suitable, it is possible through appropriate switchings to ensure that the robot's heading angle converges to the desired value of 0. A typical trajectory is portrayed

Parameters	Values	Parameters	Values
$x_1(0)$	-1 [m]	$x_2(0)$	2 [m]
$\chi_1(0)$	0 [m/s]	$\chi_2(0)$	0 [m/s]
$x_3(0)$	$-\pi$ [rad]	k_p	2
k_1	-10	k_3	-39
k_2	-40	k_4	-40

Table 4.7: Initial conditions and controller parameters used in the stabilization experiments

in Figure 4.19. The initial conditions and controller parameters are given in Table 4.8.

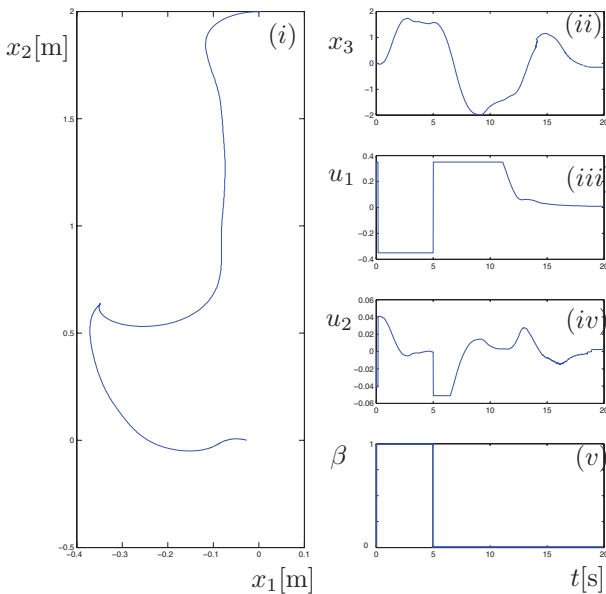


Figure 4.19: Experimental stabilization using switching provided by the discontinuous controller.

Parameters	Values	Parameters	Values
$x_1(0)$	0.01 [m]	$x_2(0)$	2 [m]
$\chi_1(0)$	0 [m/s]	$\chi_2(0)$	0 [m/s]
$x_3(0)$	$-\pi$ [rad]	k_p	2
k_1	-10	k_3	-39
k_2	-40	k_4	-40

Table 4.8: Initial conditions and controller parameters used in the stabilization experiments with the discontinuous controller

Circular trajectory tracking

For this experiment, the feedforward term is computed according to (4.112). The robot is able to reach and then follow the circular trajectory as illustrated in Figure 4.20. The parameters used for this experiment are shown in Table 4.9. The distance

$$\epsilon(t) = \sqrt{(x_1(t) - x_{1ref}(t))^2 + (x_2(t) - x_{2ref}(t))^2} \quad (4.113)$$

between the robot position and the reference trajectory is shown as a function of time in Figure 4.21. Fast convergence to zero with a very low residual error of about 3 mm is observed.

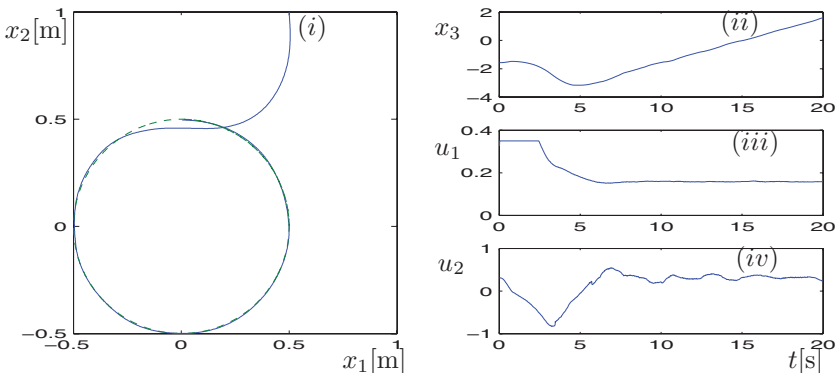


Figure 4.20: Experimental results for circular trajectory tracking.

Parameters	Values	Parameters	Values
$x_1(0)$	0.5 [m]	$x_2(0)$	1 [m]
$\chi_1(0)$	0 [m]	$\chi_2(0)$	0 [m]
x_{1ref}	$\cos(0.1\pi t)$ [m]	x_{2ref}	$\sin(0.1\pi t)$ [m]
$x_3(0)$	$-\frac{\pi}{2}$ [rad]	k_p	2
k_1	-39	k_3	-39
k_2	-40	k_4	-40

Table 4.9: Initial conditions and controller parameters used in the circular trajectory tracking experiment

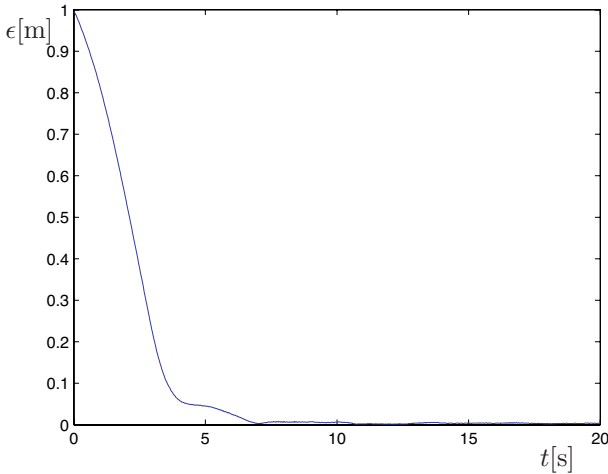


Figure 4.21: Error in measured position results for circular trajectory tracking.

4.6.4 Discussion

The following general observations are valid for all experiments that were performed:

- The robot converges even when both $\chi_1(0)$ and $\chi_2(0)$ vanish simultaneously.

- The time to converge is longer than in the simulation case. This is due to the presence of bounds on the velocities u_1 and u_2 that are clearly visible in Figures 4.19 (iii) and 4.19 (iv).
- The coefficient k_p is difficult to tune, and the results are strongly dependent on its value.

Nevertheless, the results agree well with the theory, and the control performance is excellent. The differences between simulation and experiments are mainly due to the robot imperfections and unmodeled dynamics.

4.7 Conclusions

Jet-scheduling control is applied successfully in simulation to a nonholonomic mobile robot. For this system, the jet-scheduling controller can be interpreted as a two-dimensional state extension, which provides a reference heading angle that is tracked using a proportional controller. Once tracking convergence has been achieved, the third-order mobile robot, together with the second-order dynamic extension (i.e. an extended 5th-order system) becomes equivalent to a 4th-order linear system that can be easily stabilized. During the transient period, i.e. when the heading angle has not yet reached its reference value, the robot position may very well drift away due to poor initial conditions. Nevertheless, it was shown that this excursion remains finite, before the robot eventually goes back to its reference position.

Additionally, the control method proposed in this thesis has been shown to be more robust in rejecting a constant perturbation appearing on the rotational axis of the robot.

The two open issues mentioned in the conclusion of Chapter 3 (choice of p and the design of the low-order controller) have been addressed in the present context almost trivially. Indeed, p is chosen as small as possible yet retaining the advantage of the jet-scheduling process, and the low-order controller involves a single differential equation that is directly controllable through the second input u_2 . The computations are direct and the pseudo-inverse trivial. The low-order controller leads to a straightforward physical interpretation since it steers the robot angle directly to the reference value provided by the scheduler.

The low-order controller also explains why it can be more robust. Indeed, putting a high gain on the steering angle of the robot allows variety of perturbations to be handled. The final error is inversely proportional to the

gain, which can be increased independently from the manner in which the robot converges to the origin, a benefit not easily achievable using classical feedback linearization using dynamical extension.

Chapter 5

Application to Cranes

5.1 Introduction

The previous chapter showed how jet-scheduling control could be applied to an essentially kinematic system. Indeed, only the velocities were controlled to steer the mobile robot to the origin. In this chapter, a dynamical mechanical system is considered where the inputs impose accelerations rather than velocities. A crane is a typical example of such systems with the added difficulty that the number of independent inputs are fewer than the number of degrees of freedom.

From a purely practical viewpoint, the stabilization of loads that are carried by cranes is tedious, and the lack of truly efficient strategies implies a large economic loss due to the additional time required for transferring loads. In various industries, such as construction and naval transport, the crane drivers move the load in a quasi-static way, i.e. by keeping the cable vertical in order not to induce oscillations. To improve the work rate, it is necessary to abandon the quasi-static approach and introduce a control law that can cope with dynamic couplings.

For these reasons, the dynamic behavior and the control of cranes have been abundantly studied. In particular, a fundamental paper [49] defines a class of systems called “crane”. This definition gives a general framework that includes all the classical cranes like the overhead crane, the cantilever and the US-Navy crane. The dynamical equations and the flatness property of “crane” are given as well.

The problem of crane stabilization has often been addressed and can be split in three main categories: i) controllers based on the linearized

model [38, 16, 80, 70, 9], ii) Lyapunov-based controllers [25, 41, 53], and iii) controllers using differential flatness and dynamical feedback linearization.

We will spend some time on the third category since the jet-scheduling controller belongs to it. The control of cranes with dynamic inversion techniques tends to generate non-asymptotically stable internal dynamics, when the outputs are not appropriately chosen. However, using flat outputs, the inversion technique has very good performance. In particular, the flatness formalism is ideally suited to handle the motion planning problem: Trajectory generation and input computation are performed without integrating differential equations. Interestingly, part of the flat output corresponds to the position of the load. This simplifies the motion planning problem, since the load has to be positioned accurately.

The jet scheduling for cranes takes advantage of the general definition of cranes as proposed in [49]. However, before both proceeding along the general definition and presenting the essential features of the jet scheduler when applied to cranes, an introductory example is given so as to set the concepts and notation used in the chapter. While presenting the main issues in a straightforward and simple way, it gives the necessary insight for generalizing jet scheduling control to the whole crane class throughout the chapter. The ideas will be presented using the simplified example first, before extending them to the general class using the universal definition.

This chapter is organized as follows. Section 5.2 presents respectively the crane and its model which is rigorously the same as that appearing in [49]. It is included in this thesis for the sake of completeness. The flatness property of cranes are then presented with a geometrical interpretation. The jet-scheduling controller for cranes is developed in Section 5.3. Section 5.4 illustrates the effectiveness of the proposed control scheme. In Section 5.5, the proofs of stability are detailed. An experimental verification on a laboratory-scale setup is provided in Section 5.6. Finally, Section 5.7 concludes the chapter.

5.2 Preliminaries

As mentioned in the introduction, we will first describe the overhead crane with its model before giving the general crane definition and its model. The overhead crane is a very popular 2-dimensional crane. The flatness property is also recalled for this particular example and for the general class as well. Another important crane will then be presented to which the methodology of the thesis is particularly efficient, namely the SpiderCrane. This crane

operates in the 3 dimensional space. The particular structure of this crane lies in the absence of a heavy mobile component, which allows the crane to work at a high pace, taking full advantage of the dynamics.

5.2.1 Crane description

The overhead crane

The popular overhead crane is chosen as the first example. This crane is particularly simple and allows an easy illustration of the general crane definition that will be given next (Definition 5.1).

The overhead crane (Fig. 5.1) comprises:

- A working load with mass m whose position is denoted by (x_1, x_2) .
- A main pulley mounted on a trolley that moves on a rail for which m_0 denotes the mass of the trolley and (x_{01}, x_{02}) its position.

The input T_1 is the force on the cable L_1 connected to the trolley. The input T_2 is the force that pulls the working load through the main cable L_2 . The forces T_i , $i = 1, 2$, are delivered by two motors that are located at the origin $(0,0)$ and at $(1,0)$, respectively. The total inertia with respect to the variables L_1 and L_2 , are denoted m_1 and m_2 , respectively. The following properties describe on the variables appearing in Definition 5.1:

- The overhead crane works in a plane and the dimension of the working space is $q = 2$.
- The rigid structure has no degree of freedom, consequently $d = 0$.
- There are two motors, so that $h = 1$.
- The main pulley moves in a manifold of dimension $n = 1$

General crane description

In [49], the following crane definition is given. The dimension q of the working space is 2 when the crane operates in a plane and 3 otherwise ($q \in \{2, 3\}$).

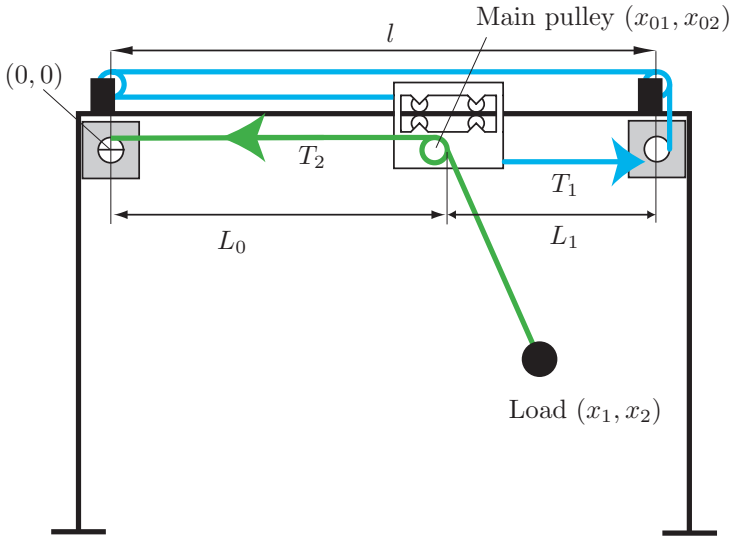


Figure 5.1: Overhead crane

Definition 5.1 (Crane) A crane is constituted by the following elements: *i*) a rigid articulated actuated mechanical system with $d \in \{0, 1\}$ degrees of freedom, *ii*) motors, *iii*) cables, *iv*) pulleys, *v*) a load, and it enjoys the following topographic properties:

1. There are $h + 1$ motors fixed on the articulated structure.
2. There are as many cables as motors.
3. A motor is linked to a pulley or to the load with a cable.
4. h cables end on a unique pulley, called the main pulley. If $h = 0$ there is no main pulley. Every other pulley is fixed to the structure.
5. There is a unique cable going through the main pulley and ending on the load.
6. Between the load and the main pulley there is no other pulley.

Moreover, the following physical property is assumed. The main pulley moves in a manifold of dimension $n \in (q-1, q)$. This manifold is determined

through the constraints imposed by the cables and by the restriction of the main pulley to move along a rail. If $n=q-1$, the manifold is transversal to the gravitational field.

SpiderCrane

As a second example, a new crane design labeled SpiderCrane is proposed. Its main particularity is that it is devoid of any heavy mobile components. Since the problem of classical cranes is the large inertia of the boom, which limits the crane dynamics, SpiderCrane can work at a considerably higher speed.

SpiderCrane is made of three fixed pylons and a fixed gibbet. A pulley is mounted at the top of each pylon, allowing a cable to slide. These three cables are attached to a ring, and by varying their length, the ring can be moved in the surrounding space. The end of the gibbet is above the plane formed by the three pulleys and at the centre of the triangle formed by the pylons. At the end of the gibbet, another pulley is mounted, guiding the main cable. This cable goes through the centre of the ring and is attached to the load. The positioning of the load in space is done by adjusting both the positioning of the ring and the length of the main cable.

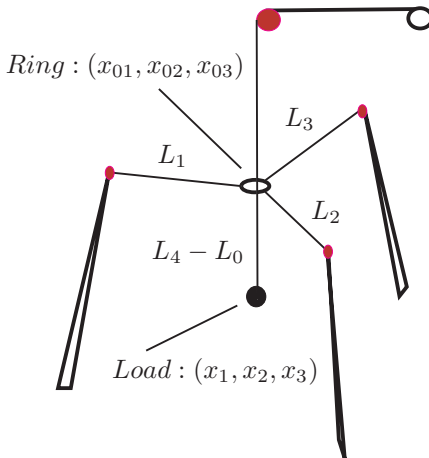


Figure 5.2: SpiderCrane

The position of the load of mass m is given by (x_1, x_2, x_3) , that of the ring (main pulley) of mass m_0 by (x_{01}, x_{02}, x_{03}) . The position of the main motor is (x_{41}, x_{42}, x_{43}) and its equivalent inertia m_4 . The positions of the three secondary motors are (x_{11}, x_{12}, x_{13}) , (x_{21}, x_{22}, x_{23}) , and (x_{31}, x_{32}, x_{33}) , respectively, and their equivalent inertias with respect to the cables m_1 , m_2 and m_3 . The length of the cable connecting the main motor to the load is L_4 , with the portion going from the motor to the ring being L_0 . The lengths of the cables connecting the secondary motors to the ring are given by L_1 , L_2 and L_3 , respectively. The four motors are torque controlled and provide the forces T_1 , T_2 , T_3 and T_4 .

- The dimension of the working space is $q = 3$.
- There is no rigid articulated actuated system $d = 0$.
- The number of motors is $h + 1 = 4$.
- The main pulley (i.e the ring) moves in a manifold of dimension $d = 3$

5.2.2 Crane modeling

In Section 5.2.1, a general crane description is given. However, this description does not explain its dynamical behavior and in particular the complex dynamic couplings between the different crane elements. To quantify this behavior, a general dynamic model is developed in this section. Then, in order to illustrate this general model, the model of the overhead crane is derived.

Model of the overhead crane

In Sections 5.2.1 and 5.2.2, a general framework has been introduced. Meanwhile, the concepts are quite abstract and in order to exemplify and make both Definition 5.1 and Theorem 5.1 more intuitive, the overhead crane is considered here.

Based on the description of Section 5.2.1, the constraint on the cable of length L_1 terminating at the main pulley reads

$$C_1 = l - L_1 - x_{01} = 0 \tag{5.1}$$

The cable of length L_2 , which passes through the main pulley and terminates at the work load, generates the following constraints

$$C_2 = L_0 - x_{01} = 0 \quad (5.2)$$

$$C_3 = \frac{(x_1 - L_0)^2 + x_2^2 - (L_2 - L_0)^2}{2} = 0 \quad (5.3)$$

Finally, since the main pulley moves along a rail, the last constraint is

$$C_4 = x_{02} = 0 \quad (5.4)$$

which is established using Theorem 5.1 together with the constraints (5.1)-(5.4). The result is

$$m\ddot{x}_1 = \lambda_3(x_1 - L_0) \quad (5.5)$$

$$m\ddot{x}_2 = \lambda_3 x_2 - mg \quad (5.6)$$

$$m_0\ddot{x}_{01} = -\lambda_1 - \lambda_2 \quad (5.7)$$

$$m_0\ddot{x}_{02} = \lambda_4 - m_0 g \quad (5.8)$$

$$0 = \lambda_2 + \lambda_3(x_1 - L_2) \quad (5.9)$$

$$m_1\ddot{L}_1 = -\lambda_1 + T_1 \quad (5.10)$$

$$m_2\ddot{L}_2 = \lambda_3(L_2 - L_0) + T_2. \quad (5.11)$$

The expressions (5.5)-(5.11) are constrained by (5.1)-(5.4), which are associated with the Lagrange multiplier λ_1 , λ_2 , λ_3 and λ_4 , respectively. However, in order to have a more compact model, the constraints (5.1), (5.2) and (5.4) can be injected into (5.5)-(5.11). The resulting dynamics are given by

$$m\ddot{x}_1 = \lambda_3(x_1 - l + L_1) \quad (5.12)$$

$$m\ddot{x}_2 = \lambda_3 x_2 - mg \quad (5.13)$$

$$(m_1 + m_0)\ddot{L}_1 = \lambda_3(x_1 - L_2) + T_1 \quad (5.14)$$

$$m_2\ddot{L}_2 = \lambda_3(l - L_1 - L_2) + T_2 \quad (5.15)$$

under the constraint

$$\frac{(x_1 - l + L_1)^2 + x_2^2 - (L_1 + L_2 - l)^2}{2} = 0 \quad (5.16)$$

associated with the Lagrange multiplier λ_3 .

Notice that λ_3 can be calculated explicitly by differentiating the constraint (5.16) twice, introducing the dynamics (5.12)-(5.15) and solving for λ_3 . The result is

$$\lambda_3 = \frac{A}{B}, \quad (5.17)$$

where

$$\begin{aligned} A &= m(-l(m_0 + m_1)T_1 + (m_0 + m_1)L_1T_1 + (m_0 + m_1 + m_2)L_2T_1 + \\ &\quad m_2((m_0 + m_1)(\dot{L}_2 - \dot{x}_1)(2\dot{L}_1 + \dot{L}_2 + \dot{x}_1) - \dot{x}_2^2) - T_1x_1) \\ B &= (m_0 + m_1)(m + m_2)(l^2 + L_1^2) + m(m_0 + m_1 + m_2)L_2^2 + \\ &\quad 2(m_0 + m_1)L_1(-l(m + m_2) + mL_2 + m_2x_1) - \\ &\quad 2L_2(lm(m_0 + m_1) + mm_2x_1) + m_2((m + m_0 + m_1)x_1^2 - \\ &\quad 2l(m_0 + m_1)x_1 + (m_0 + m_1)x_2(x_2 - gm)) \end{aligned}$$

General crane modeling

In [49], the dynamic equations of the crane are established using the Lagrangian formalism. The model is given in Theorem 5.1 below. An inertial base frame is considered such that its q th axis points in the direction opposite to g , the gravity acceleration. The following coordinates are introduced:

1. The position of the work load: (x_1, \dots, x_q) .
2. The position of the main pulley (if it exists): (x_{01}, \dots, x_{0q}) .
3. The positions of the motors: (x_{i1}, \dots, x_{iq}) for $i = 1 \dots h + 1$.
4. The positions of the fixed pulleys: $(w_{ij1}, \dots, w_{ijq})$ for $i = 1 \dots h + 1$ and $j = 1 \dots r_i$.
5. The cable lengths: L_i for $i = 1 \dots h + 1$.
6. The cable length L_0 between the main pulley (if it exists) and the motor winching the work load.

The load mass is m and the main pulley mass is m_0 . To each motor fixed on the structure, there is a corresponding equivalent mass m_i , $i = 1 \dots m + 1$. The coordinate L_0 is not associated to any mass. It is assumed that the rigid body with at most one degree of freedom has an equivalent mass M and its coordinates coincide with those of the motor winching the load, namely $(x_{(h+1)1}, \dots, x_{(h+1)q})$. The following assumptions are made.

Assumptions

- (A1) *The main pulley is present. Consequently, $h \geq 1$.*
- (A2) *The angular velocities of the fixed pulleys are small enough to neglect their quadratic effects w.r.t. the structure. All the motors are located on the structure along a line determined by the origin of the base frame and the position of the motor winching the load: $x_{ji} = \alpha_j x_{(h+1)i}$ for $j = 1 \dots h$ and $i = 1 \dots q$.*
- (A3) *If the main pulley moves along a rail, the rail coincides with the above line. Let us introduce a parameter c such that $c = 1$ if the rail is present and $c = 0$ otherwise.*
- (A4) *The crane has no redundant actuator or motor: $h = q - d - c$.*
- (A5) *If $d = 1$, the origin of the base frame is on the joint axis of the articulated mechanical structure. The articulated mechanical structure consists of either a rotational joint, in which case the joint axis is colinear with g , or a prismatic joint, in which case the joint axis is orthogonal to g . This assumption eliminates the variable $x_{(h+1)q}$. The vertical position of the motor winching the load remains constant.*

Constraints on the cable lengths are due to either cables terminating at the main pulley:

$$C_j(x_{01}, \dots, x_{0q}, x_{(h+1)1}, \dots, x_{(h+1)q-1}, L_j) = 0 \quad j = 1 \dots h, \quad (5.18)$$

or the cable terminating at the working load, one for the total length between the main pulley and the corresponding motor, and one for the length between the load and the main pulley:

$$C_{h+1}(x_{01}, \dots, x_{0q}, x_{(h+1)1}, \dots, x_{(h+1)q-1}, L_0) = 0 \quad (5.19)$$

$$C_{h+2}(x_{01}, \dots, x_{0q}, x_1, \dots, x_q, L_0, L_{h+1}) = 0. \quad (5.20)$$

An additional constraint is imposed by the motion compatible with the degree of freedom of the structure. In view of the above assumptions, the following constraint exists only if $q = 3$:

$$C_{h+3}(x_{(h+1)1}, \dots, x_{(h+1)q-1}) = 0. \quad (5.21)$$

The motion of the main pulley along the rail (if it is present) is of the form:

$$C_{h+k}(x_{0k}, x_{0q}, x_{(h+1)k}) = 0 \quad k = 1 \dots q - 1. \quad (5.22)$$

Let denote l the total number of constraints. If (5.22) is present, $l = h + 2q - 1$, otherwise $l = h + q$.

Instead of obtaining an explicit differential model, an implicit formulation with additional variables, known as *Lagrange multipliers*, is preferred.

Theorem 5.1 *Assume that the constraints are independent in an open subset of the generalized coordinate space. The dynamic model associated with the crane given by Definition 5.1 reads:*

$$m\ddot{x}_i = \lambda_{h+2} \frac{\partial C_{h+2}}{\partial x_i} - \delta_{iq} mg \quad i = 1 \dots q \quad (5.23)$$

$$m_0\ddot{x}_{0i} = \sum_{j=1}^l \lambda_j \frac{\partial C_j}{\partial x_{0i}} - \delta_{iq} m_0 g \quad i = 1 \dots q \quad (5.24)$$

$$0 = \sum_{j=1}^l \lambda_j \frac{\partial C_j}{\partial L_0} \quad (5.25)$$

$$m_i \ddot{L}_i = \sum_{j=1}^l \lambda_j \frac{\partial C_j}{\partial L_i} + T_i \quad i = 1 \dots h + 1 \quad (5.26)$$

$$M\ddot{x}_{(h+1)i} = \sum_{j=1}^l \lambda_j \frac{\partial C_j}{\partial x_{(h+1)i}} + F_i(T_{h+2}) \quad i = 1 \dots q - 1 \quad (5.27)$$

subject to Constraints (5.18)–(5.22), where $\delta_{iq} = 1$ if $i = q$ and $\delta_{iq} = 0$ otherwise. T_1, \dots, T_{h+1} are the forces produced by the motors on the structure and T_{h+2} that produced by the structure actuator.

Proof: See [49]. ■

Essential characteristics of the crane model

The overhead crane model obtained in the previous section clearly underlines the difficulties that might appear in the control-design stage. Indeed, equations (5.12)–(5.15) explicit the nonlinear nature of the model, because the Lagrange multipliers (5.17) stem from quadratic constraints. Once they are computed explicitly, they introduce gyroscopical and centrifugal couplings that will be difficult to handle using a local linearization based design.

Moreover, the variables L_1 and L_2 vary when trying to follow complex trajectories. This also induces a fundamental shift in the frequency characteristic of the linearized model. Namely, the fundamental frequency linked to the natural pendulum oscillation changes significantly according to the length of the main cable.

These two considerations stress the need to take advantage of the nonlinear model presented and to compensate these effects adequately. Notice that both characteristics apply to all the models that can be obtained through the application of Theorem 5.1.

In the forthcoming section we will see how nonlinear terms can be compensated so as to obtain a feedforward open-loop control law that makes the payload efficiently and elegantly follow smoothly planned motions.

5.2.3 Flatness

The models presented in the previous sections underline the presence of complex-dynamical-coupling terms in their respective dynamics. Fortunately, these couplings can be taken care of, once a useful essential structural property is pinpointed. This helps in finding out the right way of cancelling and compensating these nonlinear couplings. This property is the differential flatness property. Indeed, the flatness property, as it was the case with the nonholonomic robot, helps in finding the right input to be applied so as to steer the load adequately. As it has been done so far, checking flatness will first be carried out on the overhead crane before recalling it for the general class of cranes.

The overhead crane

In order to explain intuitively the concept of flatness for the cranes, Theorem 5.2 will be interpreted from a geometrical point of view (Figure 5.3).

For this purpose, a trajectory $y(t) \in \mathcal{C}^4$ for the flat output is chosen, as illustrated in Figure 5.3(a).

For a given time instant $t = t_i$, the flat output determines the load position $y = (y_1, y_2) = (x_1, x_2)$ (Figure 5.3(b)).

Then, considering the acceleration of the flat output $\ddot{y} = (\ddot{y}_1, \ddot{y}_2) = (\ddot{x}_1, \ddot{x}_2)$ and knowing the direction and intensity of gravity, it is possible to compute the resulting force acting on the load (Figure 5.3(c)):

$$\begin{aligned} F_{x_1} &= m\ddot{y}_1 \\ F_{x_2} &= m\ddot{y}_2 + mg \end{aligned}$$

Using (5.13), the Lagrange multiplier λ_3 can be determined

$$\lambda_3 = \frac{m(\ddot{y}_2 + g)}{y_2} \tag{5.28}$$

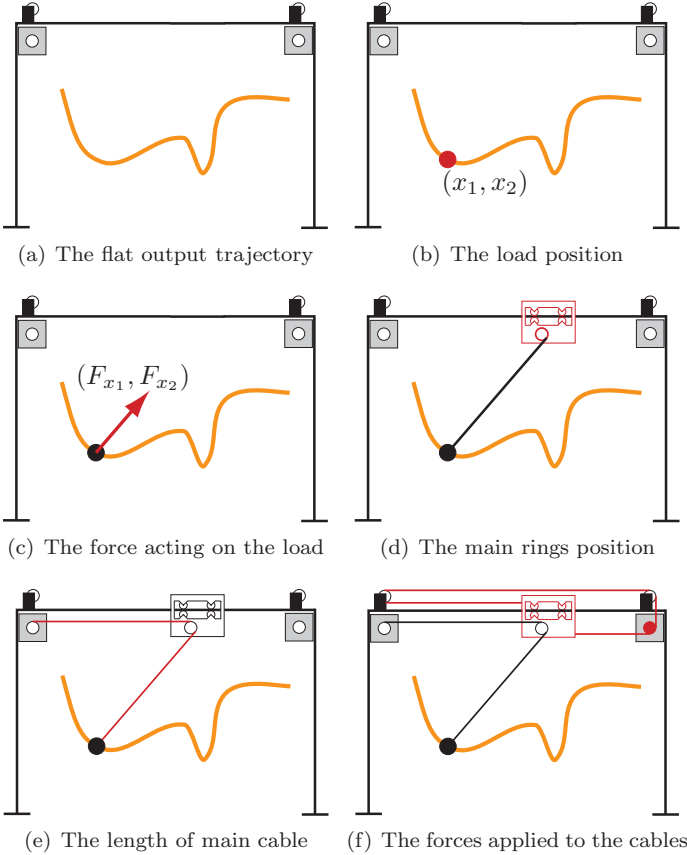


Figure 5.3: The flatness of the overhead crane from a geometrical point of view

As shown in Figure 5.3(d), the force on the load lies necessarily along the direction of the cable and hence after intersection with the rail of the

trolley, the main-ring position is inferred. This is expressed algebraically using (5.12),

$$L_1 = \frac{m\ddot{y}_1}{\lambda_3} + l - y_1. \quad (5.29)$$

Notice that L_1 is only expressed with $(y_1, \dot{y}_1, y_2, \dot{y}_2)$ since λ_3 is given by (5.28).

Taking advantage of the constraint (5.16), the length of the main cable L_2 is computed (Figure 5.3(e))

$$L_2 = l - L_1 + (y_1^2 + y_2^2 + L_1^2 + l^2 + 2y_1L_1 - 2ly_1 - 2lL_1)^{\frac{1}{2}} \quad (5.30)$$

It is then possible to determine the forces to be applied to the cables, by first differentiating (5.29) and (5.30) and then using (5.14) and (5.15), as illustrated in Figure 5.3(f)

$$\begin{aligned} \ddot{L}_1 &= \varphi_{\ddot{L}_1}(y_1, \dot{y}_1, \ddot{y}_1, y_1^{(3)}, y_1^{(4)}, y_2, \dot{y}_2, \ddot{y}_2, y_2^{(3)}) \\ \ddot{L}_2 &= \varphi_{\ddot{L}_2}(y_1, \dot{y}_1, \ddot{y}_1, y_1^{(3)}, y_1^{(4)}, y_2, \dot{y}_2, \ddot{y}_2, y_2^{(3)}) \\ T_1 &= (m_1 + m_0)\ddot{L}_1 - \lambda_3(y_1 - L_2) \\ &= \varphi_{T_1}(y_1, y_2, \dot{y}_1, \dot{y}_2, \ddot{y}_1, \ddot{y}_2, y_1^{(3)}, y_2^{(3)}, y_1^{(4)}, y_2^{(4)}) \\ T_2 &= m_2\ddot{L}_2 + \lambda_3(L_2 + L_1 - l) \\ &= \varphi_{T_2}(y_1, y_2, \dot{y}_1, \dot{y}_2, \ddot{y}_1, \ddot{y}_2, y_1^{(3)}, y_2^{(3)}, y_1^{(4)}, y_2^{(4)}) \end{aligned}$$

In a nutshell, the states $x_1, \dot{x}_1, x_2, \dot{x}_2, L_1, \dot{L}_1, L_2, \dot{L}_2$ and the inputs T_1 and T_2 of the overhead crane are constructed as functions of the flat output y and its derivatives $\dot{y}, \ddot{y}, y^{(3)}, y^{(4)}$;

$$\begin{aligned}
x_1 &= y_1 \\
\dot{x}_1 &= \dot{y}_1 \\
x_2 &= y_2 \\
\dot{x}_2 &= \dot{y}_2 \\
\lambda &= \varphi_\lambda(x_1, x_2, \ddot{x}_1, \ddot{x}_2) \\
L_1 &= \varphi_{L_1}(y_1, y_2, \ddot{y}_1, \ddot{y}_2) \\
\dot{L}_1 &= \varphi_{\dot{L}_1}(y_1, y_2, \dot{y}_1, \dot{y}_2, \ddot{y}_1, \ddot{y}_2, y_1^{(3)}, y_2^{(3)}) \\
L_2 &= \varphi_{L_2}(y_1, y_2, \ddot{y}_1, \ddot{y}_2) \\
\dot{L}_2 &= \varphi_{\dot{L}_2}(y_1, y_2, \dot{y}_1, \dot{y}_2, \ddot{y}_1, \ddot{y}_2, y_1^{(3)}, y_2^{(3)}) \\
T_1 &= \varphi_{T_1}(y_1, y_2, \dot{y}_1, \dot{y}_2, \ddot{y}_1, \ddot{y}_2, y_1^{(3)}, y_2^{(3)}, y_1^{(4)}, y_2^{(4)}) \\
T_2 &= \varphi_{T_2}(y_1, y_2, \dot{y}_1, \dot{y}_2, \ddot{y}_1, \ddot{y}_2, y_1^{(3)}, y_2^{(3)}, y_1^{(4)}, y_2^{(4)})
\end{aligned} \tag{5.31}$$

The general class of cranes

In [49], it is stated that the cranes are flat in the sense of Chapter 2. This result is established in the following theorem, assuming that trajectories in free fall are excluded, namely those for which $\ddot{x}_p = -g$.

Theorem 5.2 *Cranes obeying Definition 5.1 and satisfying (A1)–(A5) are differentially flat. The flat output can be chosen as (x_1, \dots, x_q) , the coordinates of the load, and $h+d+1-q$ coordinates of the main pulley.*

The proof is given in [49] and will not be repeated here. However, an intuitive sketch of the proof is done using a geometrical interpretation for the overhead crane.

5.2.4 Dynamical feedback linearization for the overhead crane

In order to construct a dynamical feedback controller for the overhead crane (see Section 2.1.5), the first step is to find an endogenous feedback

$$\dot{\gamma} = \beta(x, \gamma, w) \tag{5.32}$$

$$u = \alpha(x, \gamma, w) \tag{5.33}$$

such that the extended states (x, γ) are in one-to-one correspondence with the states $(y_1, y_2, \dot{y}_1, \dot{y}_2, \ddot{y}_1, \ddot{y}_2, y_1^{(3)}, y_2^{(3)})$ of the trivial system

$$\begin{aligned} y_1^{(4)} &= v_1 \\ y_2^{(4)} &= v_2 \end{aligned} \quad (5.34)$$

Moreover, the the new input w needs to be expressed as $(y, \dot{y}, \dots, y^{(4)} = v)$ (cf. (2.22) and (2.23)).

In the case of the overhead crane, the original states are $(x_1, \dot{x}_1, x_2, \dot{x}_2, L_1, \dot{L}_1)$ and the original inputs T_1, T_2 . Notice that L_2 and \dot{L}_2 are not taken into account since the constraint (5.16) and its derivatives allows reducing the degree of freedom. The dynamical extension γ is chosen as

$$\gamma = (\lambda_3, \dot{\lambda}_3),$$

meaning that there should be a one-to-one correspondence between

$$(x_1, \dot{x}_1, x_2, \dot{x}_2, L_1, \dot{L}_1, \lambda_3, \dot{\lambda}_3)$$

and

$$(y_1, y_2, \dot{y}_1, \dot{y}_2, \ddot{y}_1, \ddot{y}_2, y_1^{(3)}, y_2^{(3)}).$$

We will show that this is indeed the case. Moreover, whenever this correspondence exists then the new input can chosen as

$$w = (T_1, \ddot{\lambda}_3).$$

Since the flat output is $y = (y_1, y_2) = (x_1, x_2)$ by definition and considering (5.16), it follows that

$$\begin{aligned} y_1 &= x_1 \\ \dot{y}_1 &= \dot{x}_1 \\ y_2 &= x_2 \\ \dot{y}_2 &= \dot{x}_2 \end{aligned}$$

Then, using (5.28) and (5.29)

$$\ddot{y}_1 = \frac{-\lambda_3 L_1 - l + x_1}{m} \quad (5.35)$$

$$\ddot{y}_2 = \frac{-\lambda_3 x_2 - mg}{m} \quad (5.36)$$

By differentiating (5.35) and (5.36),

$$y_1^{(3)} = \frac{-\dot{\lambda}_3 L_1 - \lambda_3 \dot{L}_1 + \dot{x}_1}{m} \quad (5.37)$$

$$y_2^{(3)} = \frac{-\dot{\lambda}_3 x_2 - \lambda_3 \dot{x}_2}{m}. \quad (5.38)$$

At this point, one can notice that the extended states $(x_1, \dot{x}_1, L_1, \dot{L}_1, L_2, \dot{L}_2, \lambda_3, \dot{\lambda}_3)$ are in one-to-one correspondence with $(y_1, y_2, \dot{y}_1, \dot{y}_2, \ddot{y}_1, \ddot{y}_2, y_1^{(3)}, y_2^{(3)})$.

Now, differentiating again (5.37) and (5.38) and introducing the dynamics (5.12) and (5.14)

$$y_1^{(4)} = \frac{-\ddot{\lambda}_3 L_1 - \dot{\lambda}_3 \dot{L}_1 - \dot{\lambda}_3 \dot{L}_1 - \lambda_3 \frac{\lambda_3(x_1 - L_2 + T_1)}{(m_1 + m_0)} + \dot{x}_1}{m} \quad (5.39)$$

$$y_2^{(4)} = \frac{-\ddot{\lambda}_3 x_2 - \dot{\lambda}_3 \dot{x}_2 - \dot{\lambda}_3 \dot{x}_2 - \lambda_3 \frac{\lambda_3 x_2 - mg}{m}}{m}, \quad (5.40)$$

one can see that the new input $w = (w_1, w_2) = (T_1, \ddot{\lambda}_3)$ appears. Notice that L_2 can be expressed using the constraint (5.16),

$$L_2 = l - L_1 + \sqrt{(-l + L_1 + x_1)^2 + x_2^2}.$$

Therefore, the dynamical extension for the overhead crane reads

$$\begin{aligned} \ddot{\lambda}_3 &= w_2 \\ T_1 &= w_1 \\ T_2 &= m_2 \ddot{L}_2 + \lambda_3 \sqrt{(-l + L_1 + x_1)^2 + x_2^2} \end{aligned} \quad (5.41)$$

where \ddot{L}_2 can be expressed with $(x_1, \dot{x}_1, L_1, \dot{L}_1, L_2, \dot{L}_2, \lambda_3, T_1)$ after differentiating the constraint (5.16) twice, introducing the dynamics (5.12)-(5.15), and solving for \ddot{L}_2 .

In summary, the overhead crane (5.12)-(5.15) together with the dynamical extension (5.41) is equivalent to the trivial system (5.34).

Then, stable dynamics are imposed to (5.34) thanks to the following control law

$$\begin{aligned} v_1 &= k_{11} y_1 + k_{12} \dot{y}_1 + k_{13} \ddot{y}_1 + k_{14} y_1^{(3)} \\ v_2 &= k_{12} y_2 + k_{22} \dot{y}_2 + k_{23} \ddot{y}_2 + k_{24} y_2^{(3)} \end{aligned} \quad (5.42)$$

with k_{ij} , $i = 1, 2, j = 1, \dots, 4$ suitable gains to achieve stable dynamics..

Finally, inverting (5.39) and (5.40) so as to express the new input $w = (w_1, w_2) = (T_1, \ddot{\lambda}_3)$ a function of $v_1 = y_1^{(4)}$ and $v_2 = y_2^{(4)}$, and introducing (5.42), we get the dynamic feedback linearizing controller

$$\begin{aligned} w_1 &= \frac{1}{\lambda_2} \left((mv_1 - \frac{mv_2 + \dot{\lambda}_3 \dot{x}_2 + \dot{\lambda}_3 \dot{x}_2 + \lambda_3 \frac{\lambda_3 x_2 - mg}{m}}{x_2} L_1 \right. \\ &\quad \left. + \dot{\lambda}_3 \dot{L}_1 + \dot{\lambda}_3 \dot{L}_1 - \dot{x}_1) \frac{(m_1 + m_0)}{-\lambda_3} - \lambda_3 (x_1 - L_2) \right) \\ w_2 &= - \frac{mv_2 + \dot{\lambda}_3 \dot{x}_2 + \dot{\lambda}_3 \dot{x}_2 + \lambda_3 \frac{\lambda_3 x_2 - mg}{m}}{x_2}. \end{aligned} \quad (5.43)$$

5.3 Jet-Scheduling Control for Cranes

In Section 5.2.3, the flat property of cranes has been established. Thanks to this property, it is possible to compensate the dynamical coupling using a feedforward input. Clearly, in practice, this approach is not satisfactory because an open-loop solution is incapable of rejecting perturbations, for instance a gust of wind. For this reason, a closed-loop controller dealing with dynamic couplings needs to be designed. This section proposes the jet-scheduling controller developed in Chapter 3 as a possible solution.

Notice that the construction of the jet scheduler and especially the low-level controller takes full advantage of the general definition of cranes as it will be derived at the end of this section. However, establishing this relation directly is quite demanding. Therefore, we will first present the jet-scheduler for the overhead crane.

5.3.1 Jet-scheduling control for the overhead crane

In this example, a dynamical feedback linearization controller will first be constructed for the overhead crane. Then, a jet scheduling controller will be synthesized following the procedure given in Section 3.3.4. In order to construct a jet scheduling controller for the overhead crane, the principal steps developed in Chapter 3 will be addressed, namely:

- The construction of the jet scheduler.
- The definition of the coordinates Ξ , which represent the sub-manifold \mathcal{S} .

- The synthesis of a control law enforcing the convergence of Ξ to 0.

The first step so as to construct the jet scheduler is to select the flat outputs of the overhead crane (which are $y_1 = x_1$ and $y_2 = x_2$ as stated in Section 5.2.3). In this case, the order of derivation for the flat outputs is $r = 3$. Therefore, the jet scheduler is built exploiting (3.13) and reads

$$\begin{aligned}\ddot{\chi}_1 &= k_{11}y_1 + k_{12}\dot{y}_1 + k_{13}\chi_1 + k_{14}\dot{\chi}_1 \\ \ddot{\chi}_2 &= k_{12}y_2 + k_{22}\dot{y}_2 + k_{23}\chi_2 + k_{24}\dot{\chi}_2,\end{aligned}\quad (5.44)$$

with k_{ij} , $i = 1, 2$ $j = 1, \dots, 4$ suitable gains to achieve stable dynamics. Notice that, in this case, the parameter p is chosen equal to 1, meaning that the scheduled jets are based on the position and the velocity of the payload.

As discussed in the previous paragraph, the states of the overhead crane are $(x_1, \dot{x}_1, L_1, \dot{L}_1, L_2, \dot{L}_2)$ and the extended states are $(\lambda, \dot{\lambda})$, which guarantees the following one-to-one correspondence with $(y_1, y_2, \dot{y}_1, \dot{y}_2, \ddot{y}_1, \ddot{y}_2, y_1^{(3)}, y_2^{(3)})$.

$$\begin{aligned}x_1 &= y_1 \\ \dot{x}_1 &= \dot{y}_1 \\ x_2 &= y_2 \\ \dot{x}_2 &= \dot{y}_2 \\ L_1 &= \varphi_{L_1}(y_1, y_2, \ddot{y}_1, \ddot{y}_2) \\ \dot{L}_1 &= \varphi_{\dot{L}_1}(y_1, y_2, \dot{y}_1, \dot{y}_2, \ddot{y}_1, \ddot{y}_2, y_1^{(3)}, y_2^{(3)}) \\ \lambda &= \varphi_{\lambda}(x_1, x_2, \ddot{x}_1, \ddot{x}_2) \\ \dot{\lambda} &= \varphi_{\dot{\lambda}}(y_1, y_2, \dot{y}_1, \dot{y}_2, \ddot{y}_1, \ddot{y}_2, y_1^{(3)}, y_2^{(3)})\end{aligned}$$

Now, setting

$$\begin{aligned}\bar{\varphi}_{x_1} &= y_1 \\ \bar{\varphi}_{\dot{x}_1} &= \dot{y}_1 \\ \bar{\varphi}_{x_2} &= y_2 \\ \bar{\varphi}_{\dot{x}_2} &= \dot{y}_2 \\ \bar{\varphi}_{L_1} &= \varphi_{L_1}(y_1, y_2, \chi_1 + \alpha_{0,1}, \chi_2 + \alpha_{0,2}) \\ \bar{\varphi}_{\dot{L}_1} &= \varphi_{\dot{L}_1}(y_1, y_2, \dot{y}_1, \dot{y}_2, \chi_1 + \alpha_{0,1}, \chi_2 + \alpha_{0,2}, \dot{\chi}_1 + \alpha_{1,1}, \dot{\chi}_2 + \alpha_{1,2}) \\ \bar{\varphi}_{\lambda} &= \varphi_{\lambda}(y_1, y_2, \chi_1 + \alpha_{0,1}, \chi_2 + \alpha_{0,2}) \\ \bar{\varphi}_{\dot{\lambda}} &= \varphi_{\dot{\lambda}}(y_1, y_2, \dot{y}_1, \dot{y}_2, \chi_1 + \alpha_{0,1}, \chi_2 + \alpha_{0,2}, \dot{\chi}_1 + \alpha_{1,1}, \dot{\chi}_2 + \alpha_{1,2}),\end{aligned}$$

it follows that the submanifold \mathcal{S} defined by (3.24) becomes

$$\mathcal{S} = \{ \bar{\varphi}(\alpha_{0,1}, \alpha_{0,2}, \alpha_{1,1}, \alpha_{1,2}) \mid \alpha_{i,j} \in \mathbb{R}, i = 0, 1, j = 1, 2 \}. \quad (5.45)$$

Choosing the map $\sigma : \mathbb{R}^4 \rightarrow \mathbb{R}^8$ given by the following six components

$$\begin{aligned} \sigma_{x_1} &= y_1 \\ \sigma_{\dot{x}_1} &= \dot{y}_1 \\ \sigma_{x_2} &= y_2 \\ \sigma_{\dot{x}_2} &= \dot{y}_2 \\ \sigma_{L_1} &= \varphi_{L_1}(y_1, y_2, \chi_1, \chi_2) + \xi_{x,1} \\ \sigma_{\dot{L}_1} &= \varphi_{\dot{L}_1}(y_1, y_2, \dot{y}_1, \dot{y}_2, \chi_1, \chi_2, \dot{\chi}_1, \dot{\chi}_2) + \xi_{x,2} \\ \sigma_\lambda &= \varphi_\lambda(y_1, y_2, \chi_1, \chi_2) + \xi_{\gamma,1} \\ \sigma_{\dot{\lambda}} &= \varphi_{\dot{\lambda}}(y_1, y_2, \dot{y}_1, \dot{y}_2, \chi_1, \chi_2, \dot{\chi}_1, \dot{\chi}_2) + \xi_{\gamma,2} \end{aligned}$$

implies a split among a newly defined set of coordinates $\xi_{x,1}, \xi_{x,2}, \xi_{\gamma,1}, \xi_{\gamma,2}$ such that $\xi_{\gamma,1}, \xi_{\gamma,2}$ only influence the extended states λ and $\dot{\lambda}$ without affecting the original state $x_1, \dot{x}_1, L_1, \dot{L}_1, L_2, \dot{L}_2$.

As stated in Section 3.3.4, the objective is now to find a controller that stabilizes exponentially the origin of \mathcal{S} . The first step is to calculate the reduced dynamics of the ξ 's associated with the original states (i.e $\xi_{x,1}$ and $\xi_{x,2}$). These dynamics are given in (3.41) and read

$$\begin{aligned} \frac{d}{dt} \begin{pmatrix} \xi_{x,1} \\ \xi_{x,2} \end{pmatrix} &= \left[\begin{pmatrix} \frac{\partial \sigma_x}{\partial(\xi_{x,1}, \xi_{x,2})} \end{pmatrix}^T \begin{pmatrix} \frac{\partial \sigma_x}{\partial(\xi_{x,1}, \xi_{x,2})} \end{pmatrix} \right]^{-1} \\ &\quad \begin{pmatrix} \frac{\partial \sigma_x}{\partial(\xi_{x,1}, \xi_{x,2})} \end{pmatrix}^T \left(f(x, u) - \frac{\partial \sigma_x}{\partial \Upsilon} \frac{d\Upsilon}{dt} \right) \end{aligned} \quad (5.46)$$

where $\Upsilon = (y_1, y_2, \dot{y}_1, \dot{y}_2, \chi_1, \chi_2, \dot{\chi}_1, \dot{\chi}_2)$. Developing (5.46) gives

$$\frac{d}{dt} \begin{pmatrix} \xi_{x,1} \\ \xi_{x,2} \end{pmatrix} = \begin{pmatrix} \dot{L}_1 - \hat{\dot{L}}_1 \\ \ddot{L}_1 - \hat{\ddot{L}}_1 \end{pmatrix} = \begin{pmatrix} \dot{L}_1 - \hat{\dot{L}}_1 \\ \frac{\lambda_3(x_1 - L_2) + T_1}{(m_1 + m_0)} - \hat{\dot{L}}_1 \end{pmatrix} \quad (5.47)$$

Notice that $\dot{\xi}_{x,1} = \xi_{x,2}$.

Then, following the methodology proposed in Section 3.3.4, the dynamics (5.47) need to be stabilized using the input T_1 . The following structure for T_1 is chosen

$$T_1 = \hat{T}_1 + K_{T,1} = (m_1 + m_0)\hat{L}_1 - \hat{\lambda}_3(x_1 - \hat{L}_2) + K_{T,1} \quad (5.48)$$

Therefore, (5.47) reads

$$\ddot{\xi}_{x,1} = \frac{\lambda_3(x_1 - L_2) - \hat{\lambda}_3(x_1 - \hat{L}_2) + K_{T,1}}{(m + m_0)}. \quad (5.49)$$

Moreover, the control law is now synthesized considering the new inputs $K_{T,1}$. Setting

$$K_{T,i} = k_p \xi_{x,1} k_d \dot{\xi}_{x,1} - \lambda_3(x_1 - L_2) + \hat{\lambda}_3(x_1 - \hat{L}_2), \quad (5.50)$$

transforms the dynamics (5.47) into

$$\ddot{\xi}_{x,1} = \frac{-k_p \xi_{x,1} - k_d \dot{\xi}_{x,1}}{(m + m_0)}, \quad (5.51)$$

which are stable for suitable values of k_p and k_d .

Therefore, the input T_1 reads

$$T_1 = \hat{T}_1 + k_p \xi_{x,1} + k_d \dot{\xi}_{x,1} - \lambda_3(x_1 - L_2) + \hat{\lambda}_3(x_1 - \hat{L}_2). \quad (5.52)$$

Remark 5.1 Notice that T_2 is not used to stabilize (5.47), since there are many different ways of choosing it. Two direct choices are

$$T_2 = \hat{T}_2 \quad (5.53)$$

$$T_2 = \hat{T}_2 + k_p(L_2 - \hat{L}_2) - k_d(\dot{L}_2 - \hat{\dot{L}}_2). \quad (5.54)$$

As it will be seen in the stability-analysis section (Section 5.4), the input (5.53) can lead to unsatisfactory behavior for some initial conditions. However, the input (5.54) leads to excellent results and it is used for both the simulation and the experimental parts. Nevertheless, it still provides difficulties in establishing a stability proof as it will be discussed in Section 5.4

Summary of the jet scheduling controller for the overhead crane

Based on the jet scheduler

$$\begin{aligned}\ddot{\chi}_1 &= k_{11}y_1 + k_{12}\dot{y}_1 + k_{13}\chi_1 + k_{14}\dot{\chi}_1 \\ \ddot{\chi}_2 &= k_{12}y_2 + k_{22}\dot{y}_2 + k_{23}\chi_2 + k_{24}\dot{\chi}_2,\end{aligned}\quad (5.55)$$

which provides the desired accelerations in order to reach the origin, the reduced dynamics (5.47) are calculated. Notice that the values \hat{L}_1 , $\dot{\hat{L}}_1$ and $\ddot{\hat{L}}_1$ are compatible with the jet scheduler. These quantities are obtained through the φ -map given by the flatness property and read explicitly as

$$\begin{aligned}\hat{L}_1 &= \frac{\chi_1 y_2}{\chi_2 + g} + l - y_1 \\ \dot{\hat{L}}_1 &= -\frac{y_2((g + \chi_2)\dot{\chi}_1 - \chi_1\dot{\chi}_2) + \chi_1(g + \chi_2)\dot{y}_2}{(g + \chi_2)^2} - \dot{y} \\ \ddot{\hat{L}}_1 &= \varphi_{\hat{L}_1}(y_1, y_2, \dot{y}_1, \dot{y}_2, \chi_1, \chi_2, \dot{\chi}_1, \dot{\chi}_2, \ddot{\chi}_1, \ddot{\chi}_2).\end{aligned}\quad (5.56)$$

Then, in order to stabilize exponentially the reduced dynamics (5.47), the following input is chosen

$$T_1 = -k_p \xi_{x,1} - k_d \dot{\xi}_{x,1} + \hat{T}_1 - \lambda_3(x_1 - L_2) + \hat{\lambda}_3(x_1 - \hat{L}_2), \quad (5.57)$$

where the reference input \hat{T}_1 is given explicitly as

$$\hat{T}_1 = (m_1 + m_0)\hat{L}_1 + \frac{m(\chi_2 + g)}{y_2}(y_1 - l + L_1 - \sqrt{(-l + L_1 + x_1)^2 + x_2^2}).$$

Since, T_2 is not used to stabilize (5.47), it is simply set to

$$T_2 = \hat{T}_2 + k_p(L_2 - \hat{L}_2) + k_d(\dot{L}_2 - \dot{\hat{L}}_2), \quad (5.58)$$

where the input compatible with the jet scheduler \hat{T}_2 is given by

$$\hat{T}_2 = \varphi_{T_2}(y_1, y_2, \dot{y}_1, \dot{y}_2, \chi_1, \chi_2, \dot{\chi}_1, \dot{\chi}_2, \ddot{\chi}_1, \ddot{\chi}_2)$$

Finally, the jet scheduling controller for the overhead crane is composed of (5.55), (5.57) and (5.58).

Remark 5.2 Notice that, in (5.57) the term $-\lambda_3(x_1 - L_2) + \hat{\lambda}_3(x_1 - \hat{L}_2)$ equals to 0 when the origin of \mathcal{S} is reached. For this reason, in practice, this term is often small and can be neglected leading to the controller

$$\begin{aligned} T_1 &= \hat{T}_1 + k_p \xi_{x,1} + k_d \dot{\xi}_{x,1} \\ T_2 &= \hat{T}_2 + k_p(L_2 - \hat{L}_2) + k_d(\dot{L}_2 - \dot{\hat{L}}_2). \end{aligned} \quad (5.59)$$

This simple controller is used both for simulation and on the physical application (see Sections 5.4 and 5.6).

A physical insight can be given with respect to formulas (5.46) and (5.59). In (5.46), $\xi_{x,1}$ represents the error between the true trolley position and an ideal one given by the jet scheduler. This is illustrated in Figure 5.4. Now (5.59) is responsible for inducing the true trolley position to converge to its ideal value. Indeed, as described in Section 5.3, the scheduled jets (χ_1, χ_2) fix the desired position of the trolley \hat{L}_1 and the desired length of the main cable \hat{L}_2 . Consequently, the problem is reduced to impose the position L_1 of the trolley to \hat{L}_1 , which is ensured with (5.59).

5.3.2 Jet-scheduling control for cranes

In the previous section, a jet scheduling controller for the overhead crane has been developed. In this section a generalization of this controller is described for the overall class of crane given in Definition 5.1. Basically, In order to construct a jet scheduling controller for cranes, the principal steps developed in Chapter 5.3.1 will be addressed, namely:

- The construction of the jet scheduler.
- The definition of the coordinates Ξ that represent the sub-manifold \mathcal{S} .
- Design of a control law so as to reach the origin of \mathcal{S} .

Before starting the synthesis the jet-scheduling control, a careful counting of the independent states which is needed. Without loss of generality, we will treat the case where:

- The dimension of the work space is $q = 3$ (i.e. a 3D crane).
- The rigid structure of the crane has $d = 1$ degree of freedom is treated here.

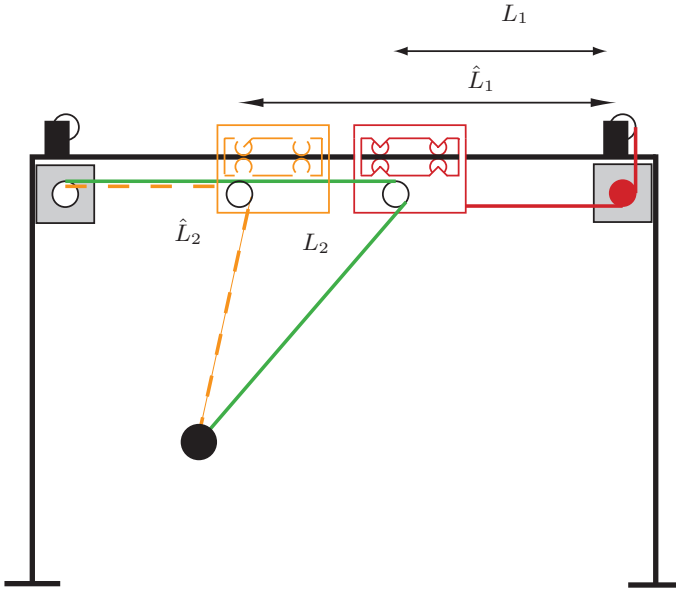


Figure 5.4: Jet-scheduling control for the overhead crane, physical interpretation: The jet-scheduler provides an ideal load acceleration χ . An ideal trolley position \hat{L}_1 compatible with χ is then computed using the flat correspondence. Then, using (5.59), the position L_1 of the trolley converges to \hat{L}_1 .

- The main pulley does not move along a rail $c = 0$.

The counting is done considering the dynamics of cranes that are given by (5.23)-(5.27) under the constraints (5.18)-(5.22). Then, it is important to notice that:

- The position of the main pulley (x_{01}, \dots, x_{0q}) is completely determined by the length of the cables L_1, \dots, L_h (i.e those which are linked to the main pulley) and the position of the rigid body $(x_{(h+1)1}, \dots, x_{(h+1)1q})$ (cf. (5.18)).
- The length of the main cable L_{h+1} is determined by the position of the load together with the position of the main pulley (x_{01}, \dots, x_{0q}) .

Therefore, it is also determined by the cable lengths L_1, \dots, L_h (cf. (5.19)-(5.20)):

- The constraint (5.21) (which is related to the degree of freedom of the structure) impose that $x_{(h+1)q-1}$ can be expressed as a function of $x_{(h+1)1}, \dots, x_{(h+1)q-2}$
- The constraints (5.22) are optional constraints and will not be taken into account in the following development. However, all the results can be adapted without loss of generality for the other cases .

Therefore, the set of independent states of the crane are $x_1, \dot{x}_1, \dots, x_q, \dot{x}_q, L_1, \dot{L}_1, \dots, L_h, \dot{L}_h, x_{(h+1)1}, \dot{x}_{(h+1)1}, \dots, x_{(h+1)q-2}, \dot{x}_{(h+1)q-2}$. This means that there are $2(q + h + q - 2)$ independent states.

As stated in (Theorem 5.2), a crane is a flat system and their flat outputs are given by

$$y = (y_1, \dots, y_q, y_{q+1}, \dots, y_{h+d+1}) = (x_1, \dots, x_q, x_{0a_1}, \dots, x_{0a_{h+d+1-q}}),$$

where $a = 1, \dots, h + d + 1 - q$ are distinct element in the set $\{1, \dots, 3\}$. Moreover, since a crane is a flat system and according to [49], the flat map φ becomes a multi-valued function of the variable Y defined as

$$Y = (y_1, \dots, y_q, \dot{y}_1, \dots, \dot{y}_q, \dots, y_1^{(3)}, \dots, y_q^{(3)}, y_{q+1}, \dots, y_{h+d+1}, \dot{y}_{q+1}, \dots, \dot{y}_{h+d+1}).$$

This means that φ is summarized as

$$\begin{pmatrix} x_1 \\ \dot{x}_1 \\ \vdots \\ x_q \\ \dot{x}_q \\ L_1 \\ \dot{L}_1 \\ \vdots \\ L_h \\ \dot{L}_h \\ x_{(h+1)1} \\ \dot{x}_{(h+1)1} \\ \vdots \\ x_{(h+1)q-2} \\ \dot{x}_{(h+1)q-2} \end{pmatrix} = \begin{pmatrix} \varphi_{x_1}(Y) \\ \varphi_{\dot{x}_1}(Y) \\ \vdots \\ \varphi_{x_q}(Y) \\ \varphi_{\dot{x}_q}(Y) \\ \varphi_{L_1}(Y) \\ \varphi_{\dot{L}_1}(Y) \\ \vdots \\ \varphi_{L_h}(Y) \\ \varphi_{\dot{L}_h}(Y) \\ \varphi_{x_{(h+1)1}}(Y) \\ \varphi_{\dot{x}_{(h+1)1}}(Y) \\ \vdots \\ \varphi_{x_{(h+1)q-2}}(Y) \\ \varphi_{\dot{x}_{(h+1)q-2}}(Y) \end{pmatrix} \quad (5.60)$$

Remark 5.3 *The order of differentiation needed for the position of the load y_1, \dots, y_q is 3 while for the coordinate of the main pulley $y_{q+1}, \dots, y_{h+d+1}$ the differentiation order is 1*

Notice that the size of Y , is $4q + 2(h + 2 - q)$ (since d is considered equal to 1), while the size of the original states is $2(q + h + q - 2)$. Therefore, the size difference is equal to 2, meaning that in order to have a one-to-one correspondence between the original states and Y , it is necessary to add a state extension

$$\begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix} = \begin{pmatrix} \varphi_{\gamma_1}(Y) \\ \varphi_{\gamma_2}(Y) \end{pmatrix} \quad (5.61)$$

of size 2. The existence of this extension is guaranteed by the flatness property.

Now, the jet scheduler follows the general procedure developed in Chapter 3. Indeed, using (3.13) and assuming that y and \dot{y} are measured (i.e. $p = 1$), the jet scheduler becomes

$$\begin{aligned} \ddot{\chi}_i &= k_{i1}y_i + k_{i2}\dot{y}_i + k_{i3}\chi_i + k_{i4}\dot{\chi}_i \quad i = 1, \dots, q \\ \chi_i &= k_{i1}y_i + k_{i2}\dot{y}_i \quad i = q + 1, \dots, h + d + 1 \end{aligned} \quad (5.62)$$

with $k_{ij}, j = 1, \dots, 4$ suitable gains to achieve stable dynamics.

The next step consist in defining the manifold \mathcal{S} . As stated in Chapter 3, this manifold is defined through a map $\bar{\varphi} : \bar{Y} \rightarrow (\hat{x}, \hat{\gamma})$ which is based on the map $\varphi : Y \rightarrow (x, \gamma)$ but where some components of Y are fixed. These fixed components are the measured flat outputs and their measured derivatives:

$$(y_1, \dots, y_q, y_{q+1}, \dots, y_{h+d+1}, \dot{y}_1, \dots, \dot{y}_q, \dot{y}_{q+1}, \dots, \dot{y}_{h+d+1}).$$

Hence, the other $2q$ components (i.e. $\ddot{y}_1, \dots, \ddot{y}_q, y_1^{(3)}, \dots, y_q^{(3)}$) are free parameters. As explain in Chapter 3, the manifold \mathcal{S} is described through the map $\bar{\varphi}$

$$\begin{aligned} \bar{\varphi}(\alpha_{0,1}, \dots, \alpha_{0,q}, \alpha_{1,1}, \dots, \alpha_{1,q}) = \\ \varphi(y_1, \dots, y_q, \dot{y}_1, \dots, \dot{y}_q, \chi_1 + \alpha_{0,1}, \dots, \chi_q + \alpha_{0,q}, \dot{\chi}_1 + \alpha_{1,1}, \dots, \dot{\chi}_q + \alpha_{1,q}, \\ y_{q+1}, \dots, y_{h+d+1}, \dot{y}_{q+1}, \dots, \dot{y}_{h+d+1}) \end{aligned}$$

The $2q$ coordinates α define \mathcal{S}

$$\mathcal{S} = \left\{ \bar{\varphi}(\alpha_{0,1}, \dots, \alpha_{0,q}, \alpha_{1,1}, \dots, \alpha_{1,q}) \mid \alpha_{i,j} \in \mathbb{R}, i = 0, 1, j = 1, \dots, q \right\} \quad (5.63)$$

Since φ is a diffeomorphic map between Y and (x, γ) , \mathcal{S} is a $2q$ -dimension sub-manifold within the manifold described by (x, γ) . The manifold \mathcal{S} is

expressed using the coordinates (x, γ) as

$$\begin{pmatrix} x_1 \\ \dot{x}_1 \\ \vdots \\ x_q \\ \dot{x}_q \\ \bar{\varphi}_{L_1}(\alpha) \\ \bar{\varphi}_{\dot{L}_1}(\alpha) \\ \vdots \\ \bar{\varphi}_{L_h}(\alpha) \\ \bar{\varphi}_{\dot{L}_h}(\alpha) \\ \varphi_{x_{(h+1)1}}(\alpha) \\ \varphi_{\dot{x}_{(h+1)1}}(\alpha) \\ \vdots \\ \varphi_{x_{(h+1)q-2}}(\alpha) \\ \varphi_{\dot{x}_{(h+1)q-2}}(\alpha) \\ \varphi_{\gamma_1}(\alpha) \\ \varphi_{\gamma_2}(\alpha) \end{pmatrix} \tag{5.64}$$

Notice that using both Assumptions (A4) and the fact that $q = 3, d = 1, c = 0$, the α affect only the last $2h + 2(q - 2) + 2 = 6$ coordinates of (5.64). It is important to underline that the size of α is also $2q = 6$. This means that the sub-manifold \mathcal{S} can be expressed using only the last $2q$ components of $(\hat{x}, \hat{\gamma})$. This consideration leads to the construction of the coordinate set Ξ

as proposed in Section 3.3.4. This yields the map

$$\begin{pmatrix} \sigma_{x_1} \\ \sigma_{\dot{x}_1} \\ \vdots \\ \sigma_{x_q} \\ \sigma_{\dot{x}_q} \\ \sigma_{L_1} \\ \sigma_{\dot{L}_1} \\ \vdots \\ \sigma_{L_{h+1}} \\ \sigma_{\dot{L}_{h+1}} \\ \sigma_{x_{(h+1)1}} \\ \sigma_{\dot{x}_{(h+1)1}} \\ \vdots \\ \sigma_{x_{(h+1)q-1}} \\ \sigma_{\dot{x}_{(h+1)q-1}} \\ \sigma_{\gamma_1} \\ \sigma_{\gamma_2} \end{pmatrix} = \begin{pmatrix} y_1 \\ \dot{y}_1 \\ \vdots \\ y_q \\ \dot{y}_q \\ \hat{L}_1 + \xi_{L_1,1} \\ \hat{L}_1 + \xi_{L_1,2} \\ \vdots \\ \hat{L}_h + \xi_{L_h,1} \\ \hat{L}_h + \xi_{L_h,2} \\ \hat{x}_{(h+1)1} + \xi_{x_{(h+1)1},1} \\ \hat{x}_{(h+1)1} + \xi_{x_{(h+1)1},2} \\ \vdots \\ \hat{x}_{(h+1)q-1} + \xi_{x_{(h+1)q-1},1} \\ \hat{x}_{(h+1)q-1} + \xi_{x_{(h+1)q-1},2} \\ \hat{\gamma}_1 + \xi_{\gamma,1} \\ \hat{\gamma}_2 + \xi_{\gamma,2} \end{pmatrix} \quad (5.65)$$

Recall that the point $(\hat{x}, \hat{\gamma}) = \varphi(y, \dot{y}, \chi, \dot{\chi})$ corresponds to the origin of \mathcal{S} (e.g. (3.23)).

The condition (3.26) is clearly fulfilled using (5.65). The next step consists in enforcing the convergence of the ξ_x 's to 0. For this purpose, the dynamical behavior of ξ_x 's is studied. Similar expression as those of the

overhead-crane example are obtained, namely

$$\frac{d}{dt} \begin{pmatrix} \xi_{L,1} \\ \xi_{\dot{L},1} \\ \vdots \\ \xi_{L,h} \\ \xi_{\dot{L},h} \\ \xi_{x,(h+1)1} \\ \xi_{\dot{x},(h+1)1} \\ \vdots \\ \xi_{x,(h+1)q-2} \\ \xi_{\dot{x},(h+1)q-2} \end{pmatrix} = \begin{pmatrix} \dot{L}_1 - \hat{\dot{L}}_1 \\ \ddot{L}_1 - \hat{\ddot{L}}_1 \\ \vdots \\ \dot{L}_h - \hat{\dot{L}}_h \\ \ddot{L}_h - \hat{\ddot{L}}_h \\ \dot{x}_{(h+1)1} - \hat{\dot{x}}_{(h+1)1} \\ \ddot{x}_{(h+1)1} - \hat{\ddot{x}}_{(h+1)1} \\ \vdots \\ \dot{x}_{(h+1)q-2} - \hat{\dot{x}}_{(h+1)q-2} \\ \ddot{x}_{(h+1)q-2} - \hat{\ddot{x}}_{(h+1)q-2} \end{pmatrix} \quad (5.66)$$

and need to be stabilized according to the general jet scheduling methodology.

Remark 5.4 *As for the overhead crane case, the reduced dynamics describe basically the behavior of the error variable $\xi_{L_i,1} = L_i - \hat{L}_i$ between the secondary cables and their scheduled values and respectively the error variable $\xi_{x_{(h+1)i},1} = x_{(h+1)i} - \hat{x}_{(h+1)i}$ between the position of the structure and their scheduled values.*

Now, following the same steps as those (5.47) to (5.52), the controllers

$$\begin{aligned} T_i &= \hat{T}_i - k_p \xi_{L_i} - k_d \dot{\xi}_{L_i} - \sum_{j=1}^l \lambda_j \frac{\partial C_j}{\partial L_i} + \sum_{j=1}^l \hat{\lambda}_j \frac{\partial \hat{C}_j}{\partial \hat{L}_i} \quad i = 1 \dots h \\ F_i &= \hat{F}_i - k_p \xi_{x_{(h+1)i}} - k_d \dot{\xi}_{x_{(h+1)i}} \\ &\quad - \sum_{j=1}^l \lambda_j \frac{\partial C_j}{\partial x_{(h+1)i}} + \sum_{j=1}^l \hat{\lambda}_j \frac{\partial \hat{C}_j}{\partial \hat{x}_{(h+1)i}} \quad i = 1 \dots q-1. \end{aligned} \quad (5.67)$$

are obtained. Notice that:

- The terms \hat{T}_i and \hat{F}_i are chosen in order to respect Assumption 3.1.
- The PD controller enforces the stability of $\xi_{L_i,1} = L_i - \hat{L}_i$ and $\xi_{x_{(h+1)i},1} = x_{(h+1)i} - \hat{x}_{(h+1)i}$

- The other terms are present so as to compensate for the dynamical couplings generated by the constraints.

As for the overhead case, the input on the main cable T_{h+1} is not used to stabilize the reduced dynamics. Therefore, it is simply set to

$$T_{h+1} = \hat{T}_{h+1}.$$

Remark 5.5 Notice that, in (5.67) the term $\sum_{j=1}^l \hat{\lambda}_j \frac{\partial \hat{C}_j}{\partial L_i} - \sum_{j=1}^l \lambda_j \frac{\partial C_j}{\partial L_i}$ and $\sum_{j=1}^l \hat{\lambda}_j \frac{\partial \hat{C}_j}{\partial \hat{x}_{(h+1)i}} - \sum_{j=1}^l \lambda_j \frac{\partial C_j}{\partial x_{(h+1)i}}$ are equal to 0 when the origin of \mathcal{S} is reached. For this reason, in practice, this term can be neglected leading to the controller

$$\begin{aligned} T_i &= \hat{T}_i - k_p \xi_{L_i} - k_d \dot{\xi}_{L_i} & i = 1 \dots h \\ F_i &= \hat{F}_i - k_p \xi_{x_{(h+1)i}} - k_d \dot{\xi}_{x_{(h+1)i}} & i = 1 \dots q - 1, \end{aligned} \quad (5.68)$$

Remark 5.6 Recall that in Chapter 3 the construction of the ξ coordinates was not explicitly given. For cranes however, the explicit expressions (5.65) are obtained. Moreover, each ξ gives rise to an independent second order differential equation with a separate input. This stems from the definition given in (5.66). Indeed each cable (of length of L_i) is independently actuated using a separate motor (Axioms 3 and 4 of the crane definition (Definition 5.1)). In the same way, the mobile structure (coordinate $x_{(h+1)i}$) is also actuated by definition.

5.3.3 Jet-scheduling control for cranes: extension to trajectory tracking

The jet-scheduling controller for cranes described in Section 5.3.2 can be easily extended to the case of trajectory tracking. The unique modification appears in the jet scheduler. Indeed, (5.62) becomes

$$\begin{aligned} \ddot{\chi}_i &= k_{i1}(y_i - y_{i,ref}) + k_{i2}(\dot{y}_i - \dot{y}_{i,ref}) + \\ & k_{i3}(\chi_i - \ddot{y}_{i,ref}) + k_{i4}(\dot{\chi}_i - y_{i,ref}^{(3)}) + y_{i,ref}^{(4)} & i = 1, \dots, q \\ \chi_i &= k_{i1}(y_i - y_{i,ref}) + k_{i2}(\dot{y}_i - \dot{y}_{i,ref}) + \ddot{y}_{i,ref} & i = q + 1, \dots, h + d + 1 \end{aligned} \quad (5.69)$$

where $y_{i,ref}$ is the reference trajectory for the flat output.

5.4 Simulation Study

In this section, the behavior of the Jet Scheduling Controller for the overhead crane (5.59) is tested and compared with dynamic feedback linearization (5.40).

5.4.1 Stabilization

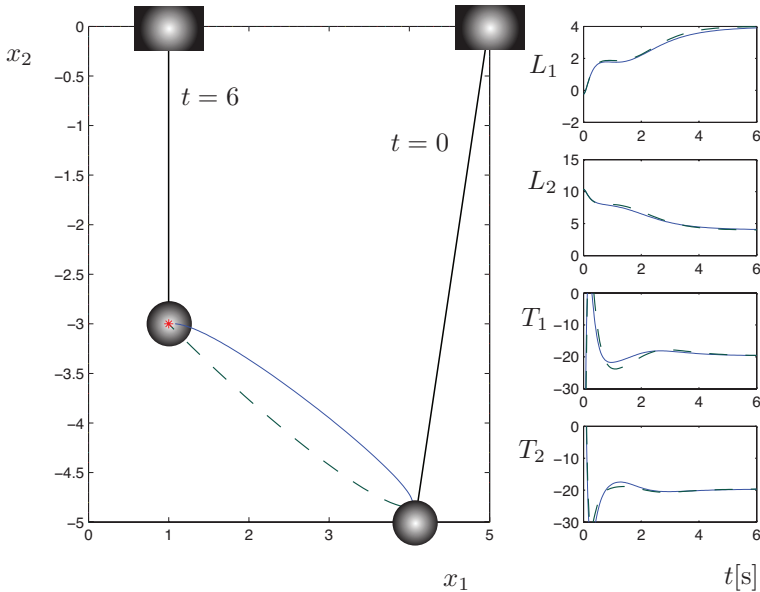


Figure 5.5: Stabilization at the reference position $x_1 = 1, x_2 = -3$ for the case of no Coulomb friction on the trolley: Dynamic feedback linearization (dashed line) and jet-scheduling control (plain line) give both a good behavior.

The stabilizing behavior of the overhead crane together with the jet scheduling controller is tested first. The goal is to bring the load to the reference position $x_1 = 1, x_2 = -3$. The initial conditions and controller parameters are given in Table 5.1.

The first simulation is carried out without disturbance. The results are shown in Figure 5.5. The trajectories for dynamic feedback linearization

and jet-scheduling controller are similar, and all states converge adequately.

For the second experiment, the overhead crane model (5.12)-(5.15) is modified to

$$m\ddot{x}_1 = \lambda_3(x_1 - l + L_1) \quad (5.70)$$

$$m\ddot{x}_2 = \lambda_3x_2 - mg \quad (5.71)$$

$$(m_1 + m_0)\ddot{L}_1 = \lambda_3(x_1 - L_2) + T_1 - \delta \operatorname{sgn}(\dot{L}_1) \quad (5.72)$$

$$m_2\ddot{L}_2 = \lambda_3(l - L_1 - L_2) + T_2 \quad (5.73)$$

where δ is a constant but unknown disturbance term.

Remark 5.7 *The added term $-\delta \operatorname{sgn}(\dot{L}_1)$ models a Coulomb friction on the trolley. As it will be shown in Section 5.6, the Coulomb friction may be important and difficult to identify on a real system.*

For the Coulomb friction value chosen as $\delta = 1, 5[N]$, the stabilization results are given in Figure 5.6. Both controllers stabilize the overhead crane with a residual static error. However, in the case dynamic feedback, this error is three times larger than with the jet-scheduler controller.

Parameters	Values	Parameters	Values	Parameters	Values
$x_1(0)$	4	$x_2(0)$	-5	$L_1(0)$	-0.24
$\dot{x}_1(0)$	0	$\dot{x}_2(0)$	1.25	$\dot{L}_1(0)$	0.06
$\chi_1(0)$	0	$\chi_2(0)$	0	$L_2(0)$	10.39
$\dot{\chi}_1(0)$	0	$\dot{\chi}_2(0)$	0	$\dot{L}_2(0)$	-1.29
m	2	$m_0 + m_1$	1	m_2	1
k_{i1}	-16	k_{i2}	-32	k_{i3}	-24
k_{i4}	-8	k_p	100	k_d	20

Table 5.1: Initial conditions and controller parameters used in the simulation of stabilization at the reference position $x_1 = 1, x_2 = -3$ (Figures 5.5 and 5.6). $i = 1, 2$.

5.4.2 Circular trajectory tracking

Figure 5.7 illustrates the behavior of the jet scheduling controller in trajectory tracking. This controller is compared to dynamic feedback linearization. The problem of tracking a circle with a constant velocity is considered.

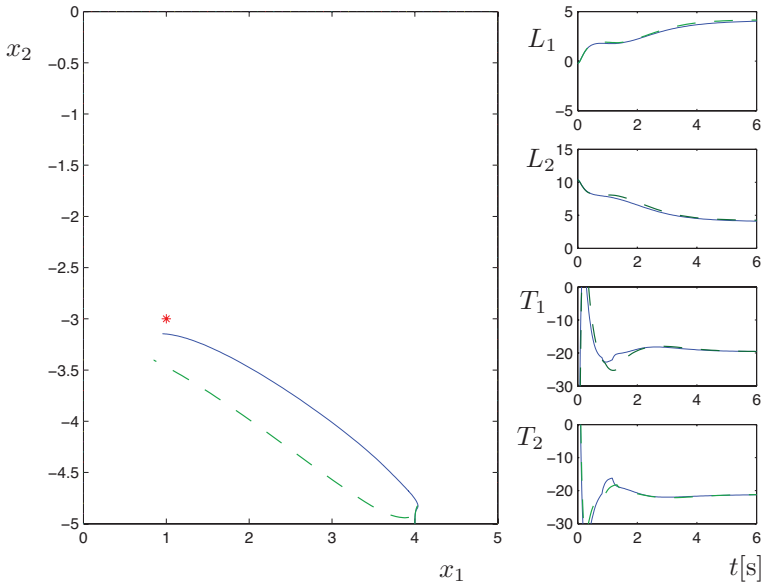


Figure 5.6: Stabilization at the reference position $x_1 = 1, x_2 = -3$ for the case of a Coulomb friction acting on the trolley: Dynamic feedback linearization (dashed line) and jet-scheduling (plain line) control stabilize the overhead crane with a residual static error. However, in the case dynamic feedback, this error is three times larger than with the jet-scheduling controller.

A circle centered at $(x_1 = 2, x_2 = -3)$ with a unity radius is defined. As can be seen, the load converges nicely to the trajectory with both dynamic feedback linearization and jets-scheduling control. The initial conditions and controller parameters used in the simulation are given in Table 5.2.

Similarly to the stabilization problem, the overhead crane model is modified to (5.70)-(5.73) where δ is a constant but unknown disturbance term. For the value $\delta = 1.5$, the tracking results are given in Figure 5.8. In both cases, the controllers ensure stability, but the performances are quite different as shown in Figure 5.9 where the distance,

$$\epsilon(t) = \sqrt{(x_1(t) - x_{1ref}(t))^2 + (x_2(t) - x_{2ref}(t))^2} \quad (5.74)$$

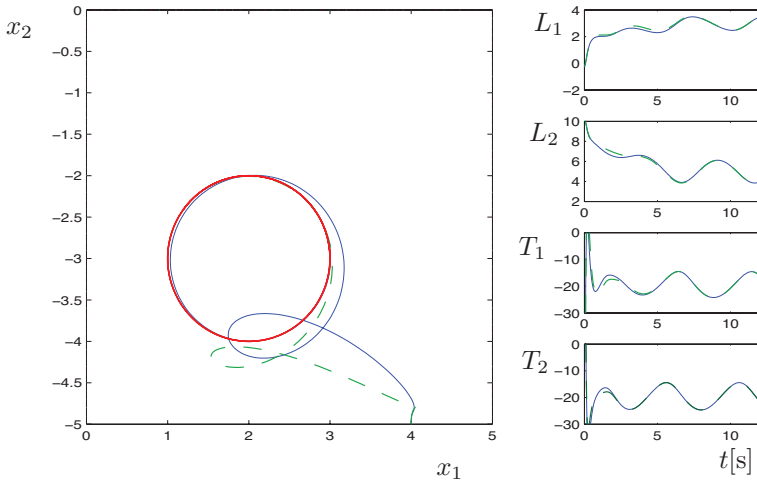


Figure 5.7: Circular trajectory tracking for the case of no Coulomb friction on the trolley: Dynamic feedback linearization (dashed line) and jet-scheduling (plain line) control give both a good result.

Parameters	Values	Parameters	Values	Parameters	Values
$x_1(0)$	4	$x_2(0)$	-5	$L_1(0)$	-0.24
$\dot{x}_1(0)$	0	$\dot{x}_2(0)$	1.25	$\dot{L}_1(0)$	0.06
$\chi_1(0)$	0	$\chi_2(0)$	0	$L_2(0)$	10.39
$\dot{\chi}_1(0)$	0	$\dot{\chi}_2(0)$	0	$\dot{L}_2(0)$	-1.29
x_{1ref}	$\cos(0.2\pi t)$	x_{2ref}	$\sin(0.2\pi t)$	l	5
m	2	$m_0 + m_1$	1	m_2	1
k_{i1}	-16	k_{i2}	-32	k_{i3}	-24
k_{i4}	-8	k_p	100	k_d	20

Table 5.2: Initial conditions and controller parameters used in the simulation of circular trajectory tracking (Figures 5.7 and 5.8). $i = 1, 2$.

between the load position and the reference trajectory is plotted. The jet-scheduling controller performs significantly better than dynamic feedback linearization. In particular, the residual error is smaller for the jet-

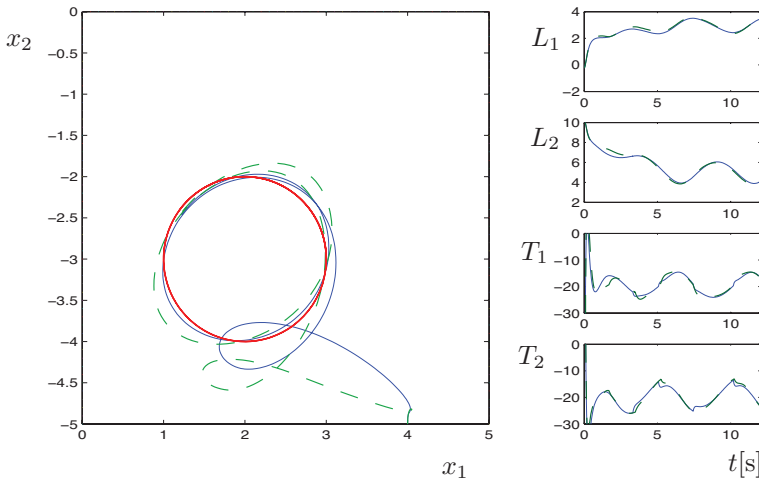


Figure 5.8: Circular trajectory tracking for the case of a Coulomb friction acting on the trolley: The jet-scheduling controller (plain line) performs significantly better than dynamic feedback linearization (dashed line). In particular, the residual error is smaller for the jet-scheduling controller.

scheduling controller. This is due to the fact that there are additional feedback gains (k_p , k_d) at the user's disposal. These additional degrees of freedom achieve better performance (as in the stabilization problem), meaning that it is possible to reduce the tracking error by increasing k_p , k_d .

5.5 Stability Analysis

As shown in Chapter 3, the stability analysis for the jet-scheduling controller is usually complex and need to be addressed from case to case. In this section, only the stability of the overhead crane controlled by the jet scheduling controller is studied. The first step is to discuss about the singularities appearing in the control law and its consequent limitations for the stability. Then some modifications on the input T_2 based on the stability analysis will be proposed. Indeed, as discuss in Remark 5.1, the jet scheduling methodology allows some degree of freedom in the synthesis of T_2 . Finally, a stability analysis following basically the same steps that are

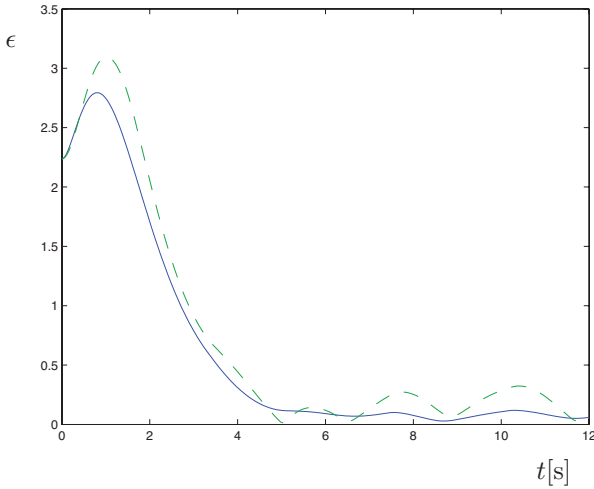


Figure 5.9: Circular trajectory tracking for the case of a Coulomb friction acting on the trolley. The distance, $\epsilon(t) = \sqrt{(x_1(t) - x_{1ref}(t))^2 + (x_2(t) - x_{2ref}(t))^2}$ between the load position and the reference trajectory is illustrated: The jet-scheduling controller (plain line) performs significantly better than dynamic feedback linearization (dashed line). In particular, the residual error is smaller for the jet-scheduling controller.

described in Section 3.4 is done.

5.5.1 Discussion about singularities

Before entering the discussion on the stability analysis of the overhead crane controlled by the jet scheduling controller, a study about the singularities appearing in the control law needs to be undertaken. The jet scheduling controller is given by (5.57)-(5.58). These expressions contain respectively

the terms \hat{T}_1 and \hat{T}_2 , which are given explicitly by

$$\begin{aligned}
\hat{T}_1 &= (m_1 + m_0)\hat{L}_1 - \hat{\lambda}_3(x_1 - \hat{L}_2) \\
&= \frac{\chi_2\chi_1}{\chi_2 + g} - \chi_1 + \frac{2\left((\chi_2 + g)\dot{\chi}_1 - \chi_1\dot{\chi}_2\right)\dot{x}_2}{(\chi_2 + g)^2} + \\
&\quad \frac{x_2\left(\dot{\chi}_1(\chi_2 + g)^2 - 2\dot{\chi}_1\dot{\chi}_2(\chi_2 + g) + \chi_1\left(2\dot{\chi}_2^2 - (\chi_2 + g)\ddot{\chi}_2\right)\right)}{(\chi_2 + g)^3} + \\
&\quad \frac{m(\chi_2 + g)}{x_2}\left(-\frac{\chi_1x_2}{\chi_2 + g} + (x_1^2 + x_2^2 + \left(\frac{\chi_1x_2}{\chi_2 + g} + l - x_1\right)^2\right. \\
&\quad \left.+ l^2 + 2x_1\left(\frac{\chi_1x_2}{\chi_2 + g} + l - x_1\right) - 2lx_1 - 2l\left(\frac{\chi_1x_2}{\chi_2 + g} + l - x_1\right)\right)^{\frac{1}{2}} \\
\hat{T}_2 &= m_2(\hat{L}_2 + \frac{m\chi_1}{\lambda_3} + l - x_1 - l)\hat{\lambda}_3 \\
&= \left(-\left(\frac{\chi_1x_2}{\chi_2 + g} + l - x_1\right) + (x_1^2 + x_2^2 + \left(\frac{\chi_1x_2}{\chi_2 + g} + l - x_1\right)^2\right. \right. \\
&\quad \left. \left.+ l^2 + 2x_1\left(\frac{\chi_1x_2}{\chi_2 + g} + l - x_1\right) - 2lx_1 - 2l\left(\frac{\chi_1x_2}{\chi_2 + g} + l - x_1\right)\right)^{\frac{1}{2}} + \right. \\
&\quad \left.\frac{\chi_1x_2}{\chi_2 + g} + l - x_1\right)\frac{m(\chi_2 + g)}{x_2}
\end{aligned}$$

These expressions become singular when $x_2 = 0$ or $\chi_2 + g = 0$. In order to understand the physical meaning of these singularities, it is important to recall that \hat{T}_1 and \hat{T}_2 are based on the flatness equivalence, and the singularities are a consequence of this equivalence. As seen in Chapter 2, the flatness property allows to reconstruct the input u through the flat output y and its derivatives (i.e. $u = \varphi_u(y, \dot{y}, \dots, y^{(r+1)})$). The singularities appearing in this equivalence means that there exist some points where it is locally indeterminate.

The first singularity $x_2 = 0$ corresponds to the physical situation where the load is aligned with the rail of the trolley (see figure 5.10). In this case, the inputs T_1 and T_2 can act on the load only along the horizontal direction. Therefore, the inputs T_1 and T_2 can not provide an instantaneous effect on the vertical acceleration of the load. This loss of direct force along the vertical axis induces indetermination in the flat equivalence.

The second singularity $\chi_2 + g = 0$ appears when the jet scheduler plans a desired vertical acceleration equal to the gravity. This case corresponds to

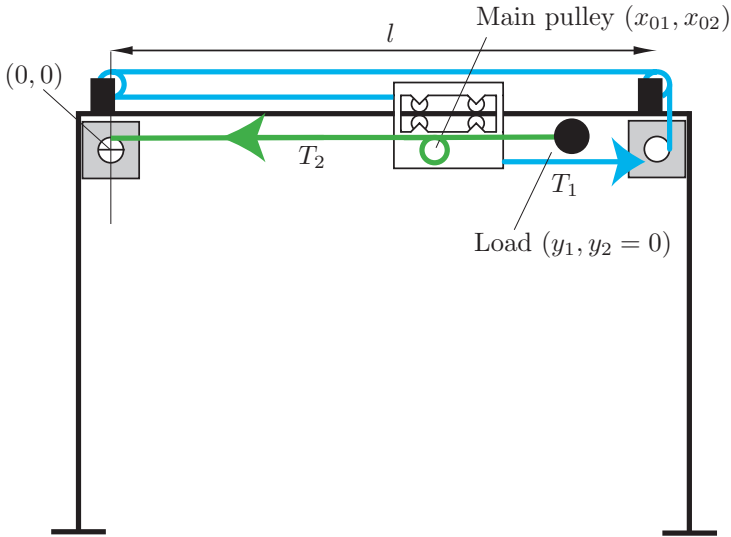


Figure 5.10: Overhead crane

a situation of weightlessness. In this case, the load floats and is not linked to the rest of the crane. Therefore, this local loss of connection with the crane structure induces an indetermination on the inputs T_1 and T_2 .

Naturally, both of these singularities could ruin the stability of the closed-loop system. However, the position $x_2 = 0$ is a pathological case which can be reached only with a very strong gust of wind. In such a case, the working of the overhead crane must be stopped and the control law shut down. Naturally, the reference position y_{2ref} must be chosen in the working space (i.e. $y_{2ref} < 0$).

The second singularity can easily be avoided through saturating $\chi_2 < -g$ using a sigmoid function as proposed in [63]. However in practice, choosing the gains of the jet scheduler satisfactorily, this saturation is not reached for the conditions within the working space.

Therefore, we will consider that these singularities are not reached. This is summarized in the following assumption:

Assumption 5.1 $\exists c_1, c_2 \in \mathbb{R}^+$ such that

$$x_2(t) < -c_1, \quad \forall t \geq 0 \quad (5.75)$$

$$|\chi_2(t) + g| < c_2, \quad \forall t \geq 0 \quad (5.76)$$

5.5.2 Stability analysis

The aim of this section, is to discuss the stability of the overhead crane (5.12)-(5.15) controlled by the jet-scheduling controller (5.52) and (5.69). T_2 is first not specified. The set of reference trajectory (y_{1ref}, y_{2ref}) have been introduced in the jet scheduler so as to choose the position of the load. Therefore, the closed-loop system reads

$$m\ddot{x}_1 = \lambda_3(x_1 - l + L_1) \quad (5.77)$$

$$m\ddot{x}_2 = \lambda_3x_2 - mg \quad (5.78)$$

$$(m_1 + m_0)\ddot{L}_1 = \lambda_3(x_1 - L_2) + T_1 \quad (5.79)$$

$$m_2\ddot{L}_2 = \lambda_3(l - L_1 - L_2) + T_2 \quad (5.80)$$

$$\begin{aligned} \ddot{\chi}_1 = & k_{11}(x_1 - y_{1ref}) + k_{12}(x_{1d} - \dot{y}_{1ref}) + \\ & k_{13}(\chi_1 - \ddot{y}_{1ref}) + k_{14}(\chi_{1d} - y_{1ref}^{(3)}) + y_{1ref}^{(4)} \end{aligned} \quad (5.81)$$

$$\begin{aligned} \ddot{\chi}_2 = & k_{21}(x_2 - y_{2ref}) + k_{22}(x_{2d} - \dot{y}_{2ref}) + \\ & k_{23}(\chi_2 - \ddot{y}_{2ref}) + k_{24}(\chi_{2d} - y_{2ref}^{(3)}) + y_{2ref}^{(4)} \end{aligned} \quad (5.82)$$

$$T_1 = -k_p\xi_{x,1} - k_d\dot{\xi}_{x,1} + \hat{T}_1 - \lambda_3(x_1 - L_2) + \hat{\lambda}_3(x_1 - \hat{L}_2). \quad (5.83)$$

Now, introducing the input T_1 (5.83) into (5.79), the closed-loop system becomes

$$m\ddot{x}_1 = \lambda_3(x_1 - l + L_1) \quad (5.84)$$

$$m\ddot{x}_2 = \lambda_3x_2 - mg \quad (5.85)$$

$$\ddot{\xi}_{x,1} = -k_p\xi_{x,1} - k_d\dot{\xi}_{x,1} \quad (5.86)$$

$$m_2\ddot{L}_2 = \lambda_3(l - L_1 - L_2) + T_2 \quad (5.87)$$

$$\begin{aligned} \ddot{\chi}_1 = & k_{11}(x_1 - y_{1ref}) + k_{12}(x_{1d} - \dot{y}_{1ref}) + \\ & k_{13}(\chi_1 - \ddot{y}_{1ref}) + k_{14}(\chi_{1d} - y_{1ref}^{(3)}) + y_{1ref}^{(4)} \end{aligned} \quad (5.88)$$

$$\begin{aligned} \ddot{\chi}_2 = & k_{21}(x_2 - y_{2ref}) + k_{22}(x_{2d} - \dot{y}_{2ref}) + \\ & k_{23}(\chi_2 - \ddot{y}_{2ref}) + k_{24}(\chi_{2d} - y_{2ref}^{(3)}) + y_{2ref}^{(4)} \end{aligned} \quad (5.89)$$

Let us define the error variables e as

$$e_{11} = x_1 - y_{1ref} \quad (5.90)$$

$$e_{12} = \dot{x}_1 - \dot{y}_{1ref} \quad (5.91)$$

$$e_{13} = \chi_1 - \ddot{y}_{1ref} \quad (5.92)$$

$$e_{14} = \dot{\chi}_1 - y_{1ref}^{(3)} \quad (5.93)$$

$$e_{21} = x_2 - y_{2ref} \quad (5.94)$$

$$e_{22} = \dot{x}_2 - \dot{y}_{2ref} \quad (5.95)$$

$$e_{23} = \chi_2 - \ddot{y}_{2ref} \quad (5.96)$$

$$e_{24} = \dot{\chi}_2 - y_{2ref}^{(3)}. \quad (5.97)$$

Therefore, the closed-loop system (5.84)-(5.89) reads

$$\dot{e}_{12} = \frac{\lambda_3}{m}(x_1 - l + L_1) - \ddot{y}_{1ref} \quad (5.98)$$

$$\dot{e}_{11} = e_{12} \quad (5.99)$$

$$\dot{e}_{22} = \frac{\lambda_3 x_2}{m} - g - \ddot{y}_{2ref} \quad (5.100)$$

$$\dot{e}_{21} = e_{22} \quad (5.101)$$

$$\ddot{\xi}_{x,1} = -k_p \xi_{x,1} - k_d \dot{\xi}_{x,1} \quad (5.102)$$

$$m_2 \ddot{L}_2 = \lambda_3(l - L_1 - L_2) + T_2 \quad (5.103)$$

$$\dot{e}_{14} = k_{11}e_{11} + k_{12}e_{12} + k_{13}e_{13} + k_{14}e_{14} \quad (5.104)$$

$$\dot{e}_{13} = e_{14} \quad (5.105)$$

$$\dot{e}_{24} = k_{21}e_{21} + k_{22}e_{22} + k_{23}e_{23} + k_{24}e_{24} \quad (5.106)$$

$$\dot{e}_{23} = e_{24} \quad (5.107)$$

A simplified case: First, stability analysis is carried on a simplified model of the overhead crane. The assumption made in order to simplify the model is to consider the inertia m_2 of the motor winching the main cable negligible with respect to the mass of the load m and the mass of the trolley $m_0 + m_1$. Consequently, let us consider

$$m_2 = 0.$$

Therefore, the expression (5.103) becomes

$$\lambda_3 = \frac{T_2}{(L_2 + L_1 - l)}.$$

This expression allows to control λ_3 directly through T_2 . Then, setting $T_2 = \hat{T}_2$ as proposed in Section 5.3.1, we get

$$\lambda_3 = -\frac{\hat{\lambda}_3(\hat{L}_2 + \hat{L}_1 - l)}{L_2 + L_1 - l}.$$

In this way, $\lambda_3 = \hat{\lambda}_3$ only asymptotically. However, it is possible to impose $\lambda_3 = \hat{\lambda}_3$ at every time instant. Indeed, setting T_2 as

$$T_2 = \hat{T}_2 \frac{L_2 + L_1 - l}{\hat{L}_2 + \hat{L}_1 - l},$$

we get that

$$\lambda_3 = \hat{\lambda}_3 \quad \forall t.$$

Remark 5.8 *The input T_2 defined in this way fulfills Assumption 3.2 because once $L_1 = \hat{L}_1$ and $L_2 = \hat{L}_2$ (that is, the origin of \mathcal{S} is reached) then $T_2 = \hat{T}_2$.*

Therefore, the closed-loop system (5.98)-(5.107) reads

$$\dot{e}_{12} = \frac{\hat{\lambda}_3}{m}(x_1 - l - \hat{L}_1 + \xi_{x,1}) - \ddot{y}_{1ref} = e_{13} + \xi_{x,1} \frac{\hat{\lambda}_3}{m} \quad (5.108)$$

$$\dot{e}_{11} = e_{12} \quad (5.109)$$

$$\dot{e}_{22} = \frac{\hat{\lambda}_3 x_2}{m} - g - \ddot{y}_{2ref} = e_{23} \quad (5.110)$$

$$\dot{e}_{21} = e_{22} \quad (5.111)$$

$$\ddot{\xi}_{x,1} = k_p \xi_{x,1} + k_d \dot{\xi}_{x,1} \quad (5.112)$$

$$\dot{e}_{14} = k_{11} e_{11} + k_{12} e_{12} + k_{13} e_{13} + k_{14} e_{14} \quad (5.113)$$

$$\dot{e}_{13} = e_{14} \quad (5.114)$$

$$\dot{e}_{24} = k_{21} e_{21} + k_{22} e_{22} + k_{23} e_{23} + k_{24} e_{24} \quad (5.115)$$

$$\dot{e}_{23} = e_{24} \quad (5.116)$$

Notice that the simplifications appearing in (5.108) and (5.110) are possible since the explicit definition of $\hat{\lambda}_3$ and \hat{L}_1 given by the jet-scheduling controller are $\hat{\lambda}_3 = \frac{m(\chi_2 + g)}{x_2}$ and $\hat{L}_1 = \frac{m\chi_1}{\hat{\lambda}_3} + l - x_1$. The equations (5.118)-(5.116)

can be reordered as

$$\dot{e}_{11} = e_{12} \quad (5.117)$$

$$\dot{e}_{12} = e_{13} + \xi_{x,1} \frac{e_{23} + \ddot{y}_{2ref}}{e_{21} + y_{2ref}} \quad (5.118)$$

$$\dot{e}_{13} = e_{14} \quad (5.119)$$

$$\dot{e}_{14} = k_{11}e_{11} + k_{12}e_{12} + k_{13}e_{13} + k_{14}e_{14} \quad (5.120)$$

$$\dot{e}_{21} = e_{22} \quad (5.121)$$

$$\dot{e}_{22} = e_{23} \quad (5.122)$$

$$\dot{e}_{23} = e_{24} \quad (5.123)$$

$$\dot{e}_{24} = k_{21}e_{21} + k_{22}e_{22} + k_{23}e_{23} + k_{24}e_{24} \quad (5.124)$$

$$\ddot{\xi}_{x,1} = k_p \xi_{x,1} + k_d \dot{\xi}_{x,1} \quad (5.125)$$

so that

- (5.125) is an independent-linear equation.
- Equations (5.121)-(5.124) constitute an independent-linear system.
- Equations (5.117)-(5.120) constitute a perturbed-linear system.

Theorem 5.3 *In case the inertia of the winching motor m_2 is 0, the overhead crane with the jet-scheduler controller (5.117)-(5.125) guarantees that x_1 and x_2 converge respectively to y_{1ref} and y_{2ref} exponentially whenever the following four conditions holds:*

- *The controller parameters $k_{ij}, i = 1, 2, j = 1, \dots, 4$ are chosen such that the matrix*

$$A_i = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ k_{i1} & k_{i2} & k_{i3} & k_{i4} \end{pmatrix}, \quad i = 1, 2 \quad (5.126)$$

is Hurwitz (i.e. $\max(\text{eig}(A_i)) < 0$).

- *k_p and k_d are chosen such that the matrix*

$$\begin{pmatrix} 0 & 1 \\ k_p & k_d \end{pmatrix} \quad (5.127)$$

is Hurwitz.

- Assumption 5.1 holds.
- $y_{2ref} < 0$ and $|\ddot{y}_{2ref}| < c_3$, where $c_3 > 0$

Proof: Since (5.127) is Hurwitz, k_p and k_d guarantee that (5.125) is exponentially stable, meaning that there exists two strictly positive constants $C_\xi, \lambda_\xi > 0$ such that

$$|\xi_{x,1}| < C_\xi |\xi_{x,1}(0)| e^{-\lambda_\xi t}.$$

In the same way, since (5.126) is Hurwitz, the system (5.121)-(5.124) is exponentially stable. Therefore, there exists four strictly positive constant $C_{e_{21}}, \lambda_{e_{21}}, C_{e_{23}}, \lambda_{e_{23}} > 0$ such that

$$|e_{21}| < C_{e_{21}} |e_{21}(0)| e^{-\lambda_{e_{21}} t} \quad (5.128)$$

$$|e_{23}| < C_{e_{23}} |e_{23}(0)| e^{-\lambda_{e_{23}} t}. \quad (5.129)$$

Since the perturbing term $\xi_{x,1} \frac{e_{23} + \ddot{y}_{2ref}}{e_{21} + y_{2ref}}$ can be considered as an exosignal in Equation (5.118), we define

$$u(t) = \xi_{x,1} \frac{e_{23} + \ddot{y}_{2ref}}{e_{21} + y_{2ref}}.$$

Then, the system (5.117)-(5.120) can be rewritten in matrix form

$$\frac{d}{dt} \begin{pmatrix} e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ k_{11} & k_{12} & k_{13} & k_{14} \end{pmatrix} \begin{pmatrix} e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} u \quad (5.130)$$

or in a more compact form

$$\dot{E} = A_1 E + B u. \quad (5.131)$$

The explicit solution of (5.131) is given by

$$E(t) = e^{A_1 t} E(0) + \int_0^t e^{A_1(t-\tau)} B u(\tau) d\tau. \quad (5.132)$$

Since A_1 is Hurwitz and defining $\min(\text{eig}(A_1)) = -q$, the forced response $\int_0^t e^{A_1(t-\tau)} Bu(\tau) d\tau$ is bounded as follows

$$\begin{aligned} \left\| \int_0^t e^{A_1(t-\tau)} Bu(\tau) d\tau \right\| &\leq \int_0^t \|e^{A_1(t-\tau)}\| \|B\| |u(\tau)| d\tau \\ &\leq \int_0^t \|e^{A_1(t-\tau)}\| |u(\tau)| d\tau \\ &\leq \int_0^t |e^{-q(t-\tau)}| |u(\tau)| d\tau. \end{aligned} \quad (5.133)$$

Notice that:

- By definition $e_{21} + y_{2ref} = x_2$. Therefore, using Assumption 5.1 it follows that $e_{21} + y_{2ref} < -c_1$, where c_1 is a positive constant.
- Using (5.129), it follows $|e_{23}| < C_{e_{23}} |e_{23}(0)|$
- y_{2ref} is chosen such that $|\ddot{y}_{2ref}| < Y$, where $Y > 0$ is a positive constant.

Therefore, there exists a $D > 0$ such that the perturbing term u can be bounded by

$$|u(t)| = |\xi_{x,1}| \frac{e_{23} + \ddot{y}_{2ref}}{e_{21} + y_{2ref}} < |\xi_{x,1}| \frac{C_{e_{23}} |e_{23}(0)| Y}{c_1} = D e^{-\lambda_\xi t}. \quad (5.134)$$

Introducing (5.134) into (5.133), the forced response is bounded by

$$\begin{aligned} \left\| \int_0^t e^{A(t-\tau)} Bu(\tau) d\tau \right\| &\leq \int_0^t |e^{-q(t-\tau)}| D e^{-\lambda_\xi \tau} d\tau \\ &= D \frac{e^{-qt} - e^{-\lambda_\xi t}}{\lambda_\xi - q} \end{aligned} \quad (5.135)$$

Therefore, the forced response (5.133) of (5.132) vanishes to zero exponentially. Moreover, because A is Hurwitz, the free response $e^{At} E(0)$ also vanishes to zero exponentially, meaning that $E(t) = (e_{11}, e_{12}, e_{13}, e_{14})^T$ converges to zero exponentially as well and consequently x_1 converges to y_{1ref} also. Finally, using (5.128), it follows that $e_{21} = x_2 - y_{2ref}$ converges to zero exponentially. This concludes the proof. \blacksquare

General case The first variant has shown that (i) neglecting the inertia m_2 of the motor winching the main cable and (ii) choosing the input T_2 as $T_2 = \hat{T}_2 \frac{L_2 + L_1 - l}{\hat{L}_2 + \hat{L}_1 - l}$, guarantees that $\lambda_3 = \hat{\lambda}_3 \quad \forall t$. This property has led to many simplifications, so as to find a proof of convergence for the closed-loop system (see Theorem 5.5.2).

However, it would be very interesting to have a similar result for the general case $m_2 \neq 0$. Therefore, considering (5.103),

$$m_2 \ddot{L}_2 = \lambda_3(l - L_1 - L_2) + T_2,$$

and recalling that

$$m_2 \hat{\dot{L}}_2 = \hat{\lambda}_3(l - \hat{L}_1 - \hat{L}_2) + \hat{T}_2,$$

it is possible using T_2 to guarantee $\lambda_3 = \hat{\lambda}_3 \quad \forall t$ by combining these two equations, the result is:

$$T_2 = (\hat{T}_2 - m_2 \hat{\dot{L}}_2) \frac{L_2 + L_1 - l}{\hat{L}_2 + \hat{L}_1 - l} + m_2 \ddot{L}_2. \quad (5.136)$$

In this case, the same result as that of Theorem 5.5.2 holds. This control law needs measuring \dot{L}_2 , which is possible thanks to an accelerometer. However, from a practical point of view, this solution is not really efficient due to the measurement noise. Therefore, so as to design an implementable control law, a first guess would be to replace (5.136) by

$$T_2 = (\hat{T}_2 - m_2 \hat{\dot{L}}_2) \frac{L_2 + L_1 - l}{\hat{L}_2 + \hat{L}_1 - l} + m_2 \hat{\dot{L}}_2. \quad (5.137)$$

Although satisfactory for all practical initial conditions whenever the mass parameter m_2 is equal to its nominal value, it is nevertheless possible to induce unboundedness of all signals using a sufficiently large m_2 and bad initial conditions. This is illustrated in Figure 5.11 (the initial conditions and controller parameters used in the simulation are given in Table 5.11). This is also the case with the feedback laws $T_2 = \hat{T}_2$ and $T_2 = \hat{T}_2 \frac{L_2 + L_1 - l}{\hat{L}_2 + \hat{L}_1 - l}$. However, even for these extreme conditions, the classical control law

$$T_2 = \hat{T}_2 + k_p(L_2 - \hat{L}_2) + k_d(\dot{L}_2 - \hat{\dot{L}}_2) \quad (5.138)$$

works properly in the simulation and experimentally (see Sections 5.4 and 5.6). Indeed, all signals converge asymptotically to their reference values in Figure 5.12.

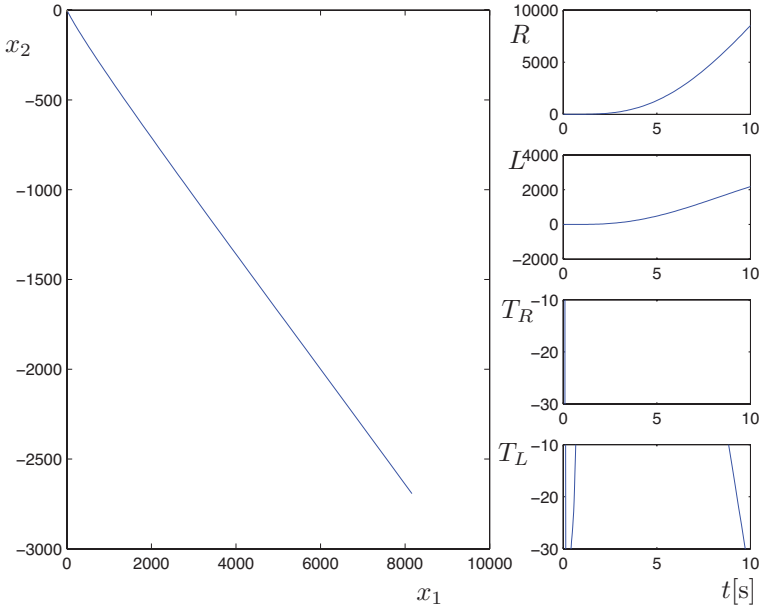


Figure 5.11: Instability occurs with the feedback law (5.137) whenever m_2 is sufficiently large

Parameters	Values	Parameters	Values	Parameters	Values
$x_1(0)$	4	$x_2(0)$	-5	$L_1(0)$	-0.53
$\dot{x}_1(0)$	0	$\dot{x}_2(0)$	1.26	$\dot{L}_1(0)$	0.07
$\chi_1(0)$	0	$\chi_2(0)$	0	$L_2(0)$	10.39
$\dot{\chi}_1(0)$	0	$\dot{\chi}_2(0)$	-6	$\dot{L}_2(0)$	-1.30
x_{1ref}	1	x_{2ref}	-3	l	5
m	2	$m_0 + m_1$	1	m_2	10
k_{i1}	-16	k_{i2}	-32	k_{i3}	-24
k_{i4}	-8	k_p	100	k_d	20

Table 5.3: Initial conditions and controller parameters used in the simulation illustrated in Figures 5.11 and 5.12. $i = 1, 2$.

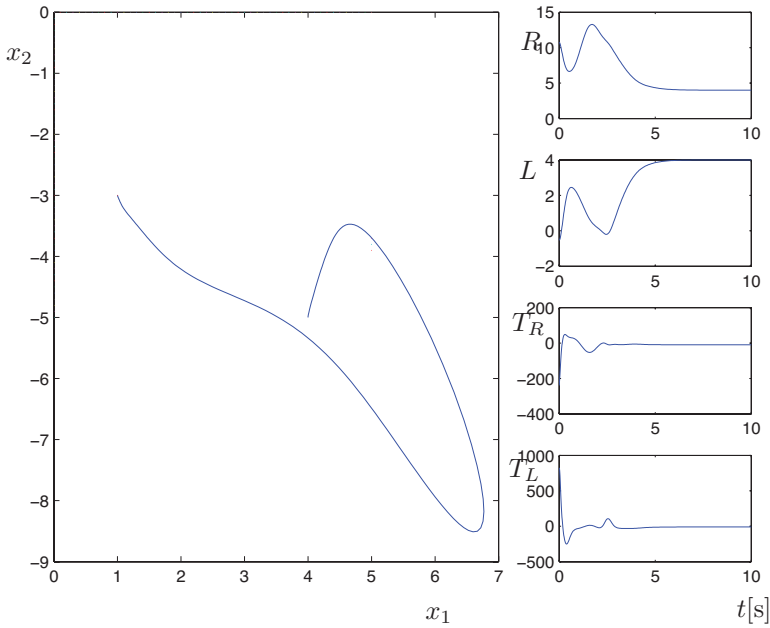


Figure 5.12: Even when m_2 is large, the control law (5.138) guarantees asymptotic convergence. The initial conditions are the same as those in Figure 5.11 (see Table 5.3).

5.6 Experimental Verification

A slight modification of the SpiderCrane setup described in Section 5.2.1 has recently been built in the Automatic Control Laboratory of EPFL (see Figure 5.13). The main difference between the two designs lies in the absence of the fourth pylon (the one guiding the hosting cable). Instead, the three secondary cables are directly attached to the ring so that the load can be hoisted and lowered through a combination of the three cable lengths that can be adjusted through the motor positions. The length of the main cable between the ring and the load is fixed. A short description of the setup is given next.

5.6.1 Setup description

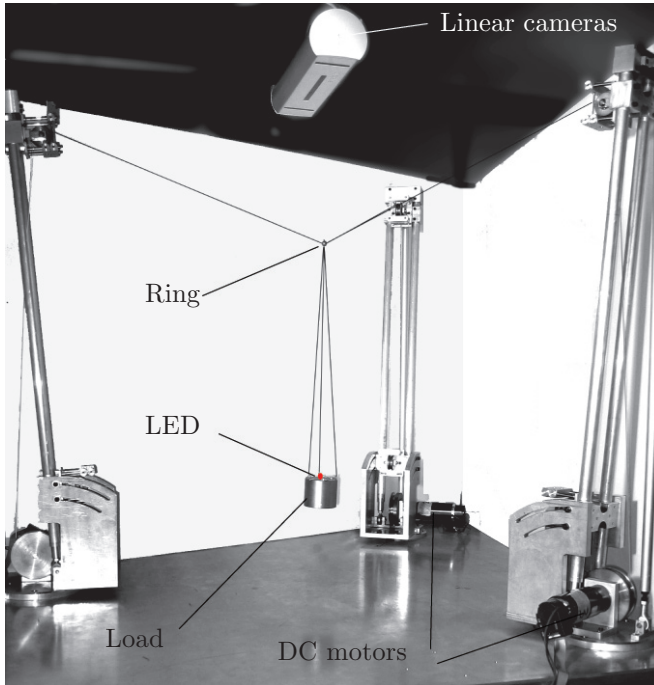


Figure 5.13: Experimental setup with the three winching motors, the load with the LED, the ring, and the linear cameras.

Contrary to the normal setup, the three cables are attached to a ring, and by varying their length, the ring can be moved in the surrounding space. A main cable goes through the centre of the ring and is attached to the load. The positioning of the load in space is done by adjusting the position of the ring. The position of the load of mass m is given by (x_1, x_2, x_3) , that of the ring of mass m_0 by (x_{01}, x_{02}, x_{03}) . The positions of the three motors are (x_{11}, x_{12}, x_{13}) , (x_{21}, x_{22}, x_{23}) and (x_{31}, x_{32}, x_{33}) , respectively. Furthermore, the motor inertias are considered to be equivalent to the masses m_1 , m_2 and m_3 , respectively, suspended to the cables. The length of the cable connecting the ring to the load is L_0 . The geometrical and inertial values

of SpiderCrane are given in Table 5.4.

Param.	Values	Param.	Values	Param.	Values
x_{11}	0[m]	x_{12}	0[m]	x_{13}	0[m]
x_{21}	-0.36[m]	x_{22}	0.64[m]	x_{23}	0[m]
x_{31}	0.36[m]	x_{32}	0.64[m]	x_{33}	0[m]
L_0	0.34[m]	m	0.49[kg]	m_0	0.02[kg]
m_1	0.54[kg]	m_2	0.54[kg]	m_3	0.54[kg]

Table 5.4: Geometrical and inertial values of SpiderCrane

The cables to the ring of length L_1 , L_2 and L_3 are controlled by means of DC motors equipped with encoders, making it possible to measure the length as well as the speed of the cables. The load position (x_1, x_2, x_3) is measured through a sensor consisting of three linear cameras. The position of an infrared LED positioned on the load can be reconstructed with a precision smaller than 1 [mm].

The measurement readings, the control law, and the voltages applied to the motors are handled by a real-time kernel implemented in C. The control loop runs at 100 Hz. The user interface that exchanges information between the user and the real-time kernel is implemented in LabVIEW. For the interested readers, all the implementation details regarding the real-time kernel can be found in [71].

5.6.2 Dynamic model

The mathematical model of this version of SpiderCrane along the same lines as those in Section 5.2.2. A set q of coordinates are defined, the cardinality of which exceeds the minimal number of required generalized coordinates:

$$q = (x_1, x_2, x_3, x_{01}, x_{02}, x_{03}, L_1, L_2, L_3).$$

This set of coordinates is constrained by a set of holonomic constraints:

$$C_1 = \sum_{i=1}^3 (x_i - x_{0i})^2 - (L_0)^2 = 0 \quad (5.139)$$

$$C_{j+1} = \sum_{i=1}^3 (x_{0i} - x_{ji})^2 - L_j^2 = 0 \quad j = 1, \dots, 3 \quad (5.140)$$

describing the geometrical relationship between the position of the crane components and the length of the cables. The external forces acting in the directions associated with the variables q are given by the three motors:

$$F_{ext} = (0, 0, 0, 0, 0, 0, T_1, T_2, T_3)$$

The Lagrange method of analytical mechanics is applied, and suitable Lagrange multipliers are introduced to handle the constraints ([37]). For SpiderCrane, this yields:

$$m\ddot{x}_1 = (x_1 - x_{01})\lambda_1, \quad (5.141)$$

$$m\ddot{x}_2 = (x_2 - x_{02})\lambda_1, \quad (5.142)$$

$$m\ddot{x}_3 = (x_3 - x_{03})\lambda_1 - gm, \quad (5.143)$$

$$m_0\ddot{x}_{01} = (x_{01} - x_1)\lambda_1 + (x_{01} - x_{11})\lambda_2 + (x_{01} - x_{21})\lambda_3 + (x_{01} - x_{31})\lambda_4 + \quad (5.144)$$

$$m_0\ddot{x}_{02} = (x_{02} - x_2)\lambda_1 + (x_{02} - x_{12})\lambda_2 + (x_{02} - x_{22})\lambda_3 + (x_{02} - x_{32})\lambda_4 \quad (5.145)$$

$$m_0\ddot{x}_{03} = (x_{03} - x_3)\lambda_1 + (x_{03} - x_{13})\lambda_2 + (x_{03} - x_{23})\lambda_3 + (x_{03} - x_{33})\lambda_4 - gm_0, \quad (5.146)$$

$$m_1\ddot{L}_1 = T_1 - L_1\lambda_2 - L_0 \quad (5.147)$$

$$m_2\ddot{L}_2 = T_2 - L_2\lambda_3 - L_0 \quad (5.148)$$

$$m_3\ddot{L}_3 = T_3 - L_3\lambda_4 - L_0 \quad (5.149)$$

where λ_j with $j = 1, \dots, 4$ are the Lagrange multipliers.

5.6.3 Model discrepancies

The two main model discrepancies are:

- The cables are considered as being perfectly rigid (meaning that they can pull and push) in the mathematical model while in practice the cables can only pull.
- There is a large amount of Coulomb friction on the DC motors.

The DC motor used for the experimental setup is controlled in voltage, meaning that a tension u is imposed to the motor in order to obtain an electromechanical torque to pull the cable. However, in presence of Coulomb friction, this electromechanical torque is not completely transmitted to the

cable, leading to a loss of sensitivity of the input. Indeed, a variation of the input may induce no variation on the torque applied to the cable. In order to illustrate this phenomenon the following experiment is done: A load of mass $m = 0.9$ [Kg] suspended by a cable is winched through a DC motor. The available voltage range is comprised between -5 [V] and 5 [V]. The experiment begins by finding the minimal voltage needed to raise the load without inducing a displacement h , meaning that the force applied by the motor compensates perfectly the weight of the load; this voltage is equal to 0.42 [V] as illustrated in Figure 5.14. Then, the voltage is increased progressively. However, the load still does not move until the voltage reaches 0.85 [V]. The difference of voltage between 0.42 [V] and 0.85 [V] corresponds to the Coulomb friction presents in the motor. As it will be shown in Section 5.6.7, the standard amplitude of the inputs of SpiderCrane during a working phase is smaller than 2 [V]. This means that the Coulomb friction is around 20 per cent of the standard amplitude of the inputs. This loss of transmitted torque sensitivity due to the Coulomb friction makes the control of the crane difficult because a control law may impose an input that will not have any influence on the system.

5.6.4 Flatness of SpiderCrane

The choice of the flat output y and the explicit calculation of the function φ_x and φ_u are usually not trivial. In the case of SpiderCrane, one has:

$$\begin{aligned} x &= (x_1, x_2, x_3, x_{01}, x_{02}, x_{03}, L_1, L_2, L_3 \\ &\quad \dot{x}_1, \dot{x}_2, \dot{x}_3, \dot{x}_{01}, \dot{x}_{02}, \dot{x}_{03}, \dot{L}_1, \dot{L}_2, \dot{L}_3) \\ y &= (x_1, x_2, x_3) \\ u &= (T_1, T_2, T_3) \end{aligned}$$

Using (5.141), (5.142) and (5.143), x_{01} , x_{02} and λ_1 can be expressed as:

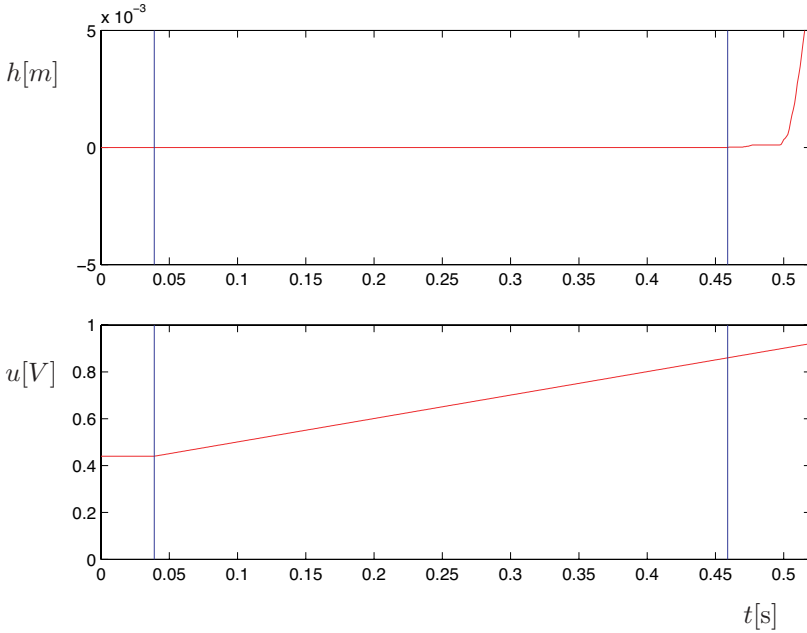


Figure 5.14: Illustration of the Coulomb friction presents in the DC motor: The experiment begins by finding the minimal voltage needed to raise the load without inducing any displacement h . This voltage is equal to $0.42[V]$. Then, the voltage is increased progressively until the load starts moving.

$$\begin{aligned}
 x_{01} &= x_1 - \frac{m\ddot{x}_1}{\lambda_1} \\
 &= \varphi_{x_1}(x_1, \ddot{x}_1)
 \end{aligned} \tag{5.150}$$

$$\begin{aligned}
 x_{02} &= x_2 - \frac{m\ddot{x}_2}{\lambda_2} \\
 &= \varphi_{x_2}(x_2, \ddot{x}_2)
 \end{aligned} \tag{5.151}$$

$$\begin{aligned}
 \lambda_1 &= \frac{m\ddot{x}_3 + gm}{x_3 - x_{03}} \\
 &= \varphi_{\lambda_1}(x_3, x_{03}, \ddot{x}_3)
 \end{aligned} \tag{5.152}$$

Differentiating (5.150) and (5.151) gives:

$$\dot{x}_{01} = \varphi_{\dot{x}_1}(x_1, \dot{x}_1, \dots, x_1^{(3)}) \quad (5.153)$$

$$\dot{x}_{02} = \varphi_{\dot{x}_2}(x_2, \dot{x}_2, \dots, x_2^{(3)}) \quad (5.154)$$

Solving the constraint equations (5.139)-(5.140) for L_j with $j = 1, \dots, 3$, and using (5.150) and (5.151), leads to:

$$L_j = \varphi_{L_j}(x_1, \ddot{x}_1, x_2, \ddot{x}_2, x_3, \ddot{x}_3) \quad j = 1, \dots, 3 \quad (5.155)$$

Time differentiation of (5.155) gives:

$$\dot{L}_j = \varphi_{\dot{L}_j}(x_1, \dots, x_1^{(3)}, x_2, \dots, x_2^{(3)}, x_3, \dots, x_3^{(3)}) \quad j = 1, \dots, 3. \quad (5.156)$$

Equations (5.150)-(5.156) establish that the states can be expressed as functions of the flat outputs and their derivatives.

Now, it remains to express the inputs as functions of the outputs and their derivatives and, for this purpose, (5.153), (5.154) and (5.156) need to be differentiated with respect to time:

$$\ddot{x}_{01} = \varphi_{\ddot{x}_{01}}(x_1, \dot{x}_1, \dots, x_1^{(4)}) \quad (5.157)$$

$$\ddot{x}_{02} = \varphi_{\ddot{x}_{02}}(x_2, \dot{x}_2, \dots, x_2^{(4)}) \quad (5.158)$$

$$\ddot{L}_j = \varphi_{\ddot{L}_j}(x_1, \dots, x_1^{(4)}, x_2, \dots, x_2^{(4)}, x_3, \dots, x_3^{(4)}) \quad j = 1, \dots, 3 \quad (5.159)$$

Solving (5.144)-(5.146) for $\lambda_2, \lambda_3, \lambda_4$ and λ_5 , and using (5.150), (5.151), (5.152), (5.155), (5.157) and (5.158), gives:

$$\lambda_i = \varphi_{\lambda_i}(x_1, \dots, x_1^{(4)}, x_2, \dots, x_2^{(4)}, x_3, \dots, x_3^{(4)}) \quad i = 2, \dots, 4 \quad (5.160)$$

Finally, solving (5.147)-(5.149) for T_1, T_2, T_3 and T_4 , and using (5.152), (5.155), (5.156), (5.159) and (5.160), results in:

$$T_j = \varphi_{T_j}(x_1, \dots, x_1^{(4)}, x_2, \dots, x_2^{(4)}, x_3, \dots, x_3^{(4)}) \quad j = 1, \dots, 3 \quad (5.161)$$

Formally, the following expressions hold:

$$\begin{aligned}
 L_j &= \varphi_{L_j}(x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3, \ddot{x}_1, \ddot{x}_2, \ddot{x}_3) & j = 1, \dots, 3 \\
 \dot{L}_j &= \varphi_{\dot{L}_j}(x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3, \ddot{x}_1, \ddot{x}_2, \ddot{x}_3, x_1^{(3)}, x_2^{(3)}, x_3^{(3)}) \\
 & & j = 1, \dots, 3 \\
 T_j &= \varphi_{T_j}(x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3, \ddot{x}_1, \ddot{x}_2, \ddot{x}_3, \dots, x_1^{(4)}, x_2^{(4)}, x_3^{(4)}) \\
 & & j = 1, \dots, 3
 \end{aligned}$$

These relationships show that there exists a correspondence between the load position (and their time derivatives) and the original inputs and states of SpiderCrane, which means that the system is indeed flat.

5.6.5 Jet-scheduling control

Successful implementation of feedforward control needs to consider the discrepancies between the mathematical model and the experimental setup. For SpiderCrane, the main discrepancy relates to the characteristics of the winching mechanism. Indeed, the motors are mounted on gears that introduce a large amount of dry friction that cannot easily be compensated for through feedforward control. Furthermore, flatness-based control is inappropriate to reject disturbances, e.g. sudden unpredictable forces acting either on the load or on the motors. Hence, some feedback is necessary. One is naturally led to consider dynamic feedback linearization, i.e. using endogeneous dynamic feedback ([35]). However, this technique has a few drawbacks. The first one is the need to find the dynamic extension, which complicates the controller and especially its implementation. The second, and most important one, lies in the difficulty of separating the closed-loop dynamics in two parts, one governing the motors and the other responsible for the sway and load positioning. Such a separation would allow increasing the gains for the motors without necessarily imposing a violent load reaction.

Jet-scheduling control can answer the aforementioned drawbacks. The basic idea is to measure the load position and its derivatives and generate appropriate references for the three cable lengths. Jet-scheduling control has three parts:

1. The first part calculates appropriate load accelerations (the jets χ_1 , χ_2 and χ_3) as stated in Section 5.3 to reach the load reference (x_{1ref} ,

x_{2ref}, x_{3ref}). These jets are updated regularly based on the measurements of the load position (x_1, x_2, x_3) and its derivatives $(\dot{x}_1, \dot{x}_2, \dot{x}_3)$. The regeneration of the scheduled jets upon measurements introduces the element of feedback that is needed to reject disturbances. The jets are computed using the following dynamic filter:

$$\begin{aligned}\ddot{\chi}_1 &= -k_1^4(x_1 - x_{1ref}) - 4k_1^3(\dot{x}_1 - \dot{x}_{1ref}) \\ &\quad - 6k_1^2(\chi_1 - \ddot{x}_{1ref}) - 4k_1(\dot{\chi}_1 - \dot{x}_{1ref}^{(3)}) + x_{1ref}^{(4)} \\ \ddot{\chi}_2 &= -k_2^4(x_2 - x_{2ref}) - 4k_2^3(\dot{x}_2 - \dot{x}_{2ref}) \\ &\quad - 6k_2^2(\chi_2 - \ddot{x}_{2ref}) - 4k_2(\dot{\chi}_2 - \dot{x}_{2ref}^{(3)}) + x_{2ref}^{(4)} \\ \ddot{\chi}_3 &= -k_3^4(x_3 - x_{3ref}) - 4k_3^3(\dot{x}_3 - \dot{x}_{3ref}) \\ &\quad - 6k_3^2(\chi_3 - \ddot{x}_{3ref}) - 4k_3(\dot{\chi}_3 - \dot{x}_{3ref}^{(3)}) + x_{3ref}^{(4)}\end{aligned}$$

These expressions are independent of SpiderCrane dynamics. They are stabilized chain of integrators the inputs of which are the load positions and velocities. The coefficients of the characteristic polynomial are chosen so that the corresponding eigenvalues are the same and equal to $\lambda = -k_i$, so as to have few design parameters.

The above expressions should not be mistaken with linearizing dynamic extensions.

2. The second part uses the flatness property to compute references for the cable lengths. The acceleration and the higher derivatives in the flatness correspondences are replaced by the ideally scheduled variables χ_1, χ_2 and χ_3 and their time derivatives:

$$\begin{aligned}\hat{L}_j &= \varphi_{L_j}(x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3, \chi_1, \chi_2, \chi_3) \quad j = 1, \dots, 3 \\ \hat{\dot{L}}_j &= \varphi_{\dot{L}_j}(x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3, \chi_1, \chi_2, \chi_3, \dot{\chi}_1, \dot{\chi}_2, \dot{\chi}_3) \\ &\quad j = 1, \dots, 3\end{aligned}$$

Also, direct feedforward control on the inputs can be computed in a similar manner:

$$\begin{aligned}\hat{T}_j &= \varphi_{T_j}(x_1, x_2, x_3, \dot{x}_1, \dot{x}_2, \dot{x}_3, \chi_1, \chi_2, \chi_3, \dots, \ddot{\chi}_1, \ddot{\chi}_2, \ddot{\chi}_3) \\ &\quad j = 1, \dots, 3\end{aligned}$$

3. The third part consists of feedback controllers that track the computed cable lengths. High-gains PD controllers can be used to compensate the effect of dry friction and achieve a desired convergence:

$$\begin{aligned} T_1 &= -k_{p1}(L_1 - \hat{L}_1) - k_{d1}(\dot{L}_1 - \dot{\hat{L}}_1) + \hat{T}_1 \\ T_2 &= -k_{p2}(L_2 - \hat{L}_2) - k_{d2}(\dot{L}_2 - \dot{\hat{L}}_2) + \hat{T}_2 \\ T_3 &= -k_{p3}(L_3 - \hat{L}_3) - k_{d3}(\dot{L}_3 - \dot{\hat{L}}_3) + \hat{T}_3. \end{aligned}$$

5.6.6 SpiderCrane implementation

Force-controlled setup The jet-scheduling control law uses as inputs the forces T_1 , T_2 and T_3 that are applied to the three cables. However the physical inputs of the SpiderCrane setup are the voltages u_1 , u_2 and u_3 to the three DC motors. For this reason, a low-level control is designed to impose the desired forces.

The torque c_i provided by each DC motor is given by

$$c_i = K_{mi} \frac{u_i - K_{ni}\omega_i}{R_i}, \quad i = 1, \dots, 3 \quad (5.162)$$

where u_i is voltage input in [V], R_i is the coil resistance in [Ω], K_{mi} is the torque constant, K_{ni} is the velocity constant and ω_i is the motor velocity. The motor characteristics are given in Table 5.5.

Parameters	Values	Parameters	Values
Power	90[W]	Time const.	$6 \cdot 10^{-3}$ [s]
Torque cst. K_m	$19.4 \cdot 10^{-3}$ [Nm/A]	Vel. cst. K_n	29460[deg/Vs]

Table 5.5: Motor characteristics

The velocity of the cable \dot{L}_i is directly proportional to the motor velocity ω trough the pulley radius r_i ,

$$\dot{L}_i = r_i \omega_i. \quad (5.163)$$

In the same way, the force T_i is directly proportional to the torque c_i trough the pulley radius r_i ,

$$T_i = c_i r_i. \quad (5.164)$$

Now, inverting (5.162) and using (5.163) and (5.164) leads to the control law:

$$u_i = \frac{T_i c_i R_i}{r_i K_{mi}} + K_{ni} \frac{\dot{L}_i}{r_i} \quad (5.165)$$

The voltage u_i allows pulling on the cable L_i with the force T_i . In the sequel, we will consider the forces T_i , $i = 1, \dots, 3$, as the inputs to SpiderCrane.

5.6.7 Experimental results

In this section, experimental results for both load stabilization and trajectory tracking are presented. The numerical values of the controller parameters used for these experiments are given in Table 5.6.

Param.	Values	Param.	Values	Param.	Values
k_{pi} [V/m]	80	k_{di} [Vs/m]	15	k_i	8[1/s]

Table 5.6: Controller parameters ($i = 1, \dots, 3$).

The results described next are also available in movie form¹.

Stabilization.

Figure 5.15 illustrates the way in which the jet-scheduling controller stabilizes the load at the reference point $(x_{1ref}, x_{2ref}, x_{3ref})$. The experiment has two phases: (i) without control, the load oscillates strongly, and (ii) at time 3.5[s], the controller is switched on. The controller stabilizes nicely the load at its reference point. Moreover, the performance is excellent since the time needed for stabilization is of the order of 1.5 [s].

Figure 5.16 illustrates the controller behavior following a disturbance that imposes a load position different from the reference value. This corresponds to the situation where the load is being blocked by some obstacle, or a human operator pulls and holds the load away from the reference value. As can be seen by the small values of the inputs u_1 , u_2 , and u_3 , the controller does not over-react. The controller knows that, under normal conditions, small forces are sufficient to go back to the reference position. The fact that small forces are not able to move the load indicates the presence of

¹<http://lawwww.epfl.ch/page4506.html>

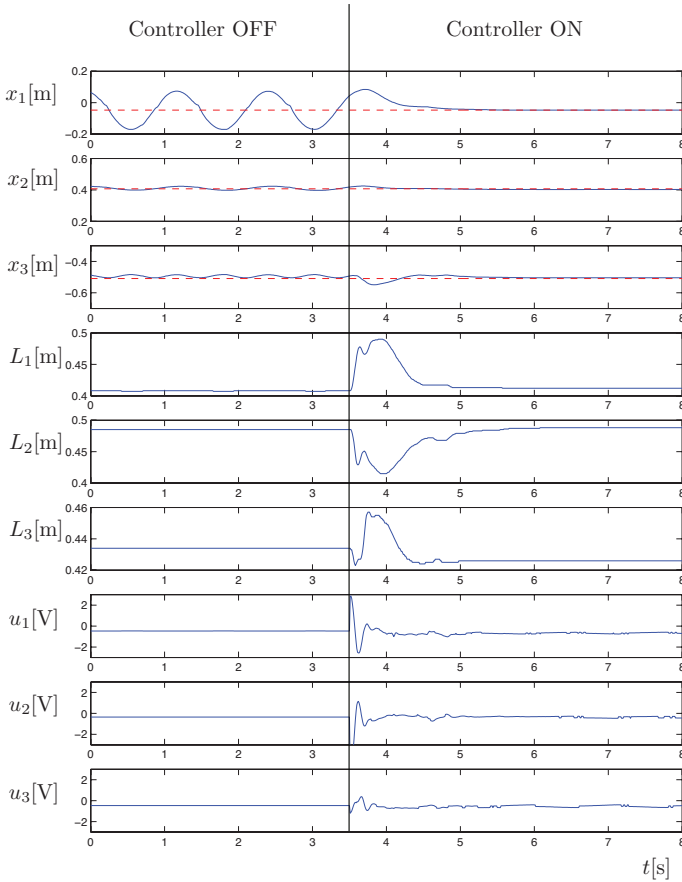


Figure 5.15: Load oscillation without control and load stabilization with jet-scheduling control. Reference values are represented by dashed lines.

an "unusual" situation. The controller, which works with higher derivatives of the position error, does not compute the large control effort that a proportional-like controller would. The figure also shows that, once the load is released, it goes back swiftly to its equilibrium position without any oscillation.

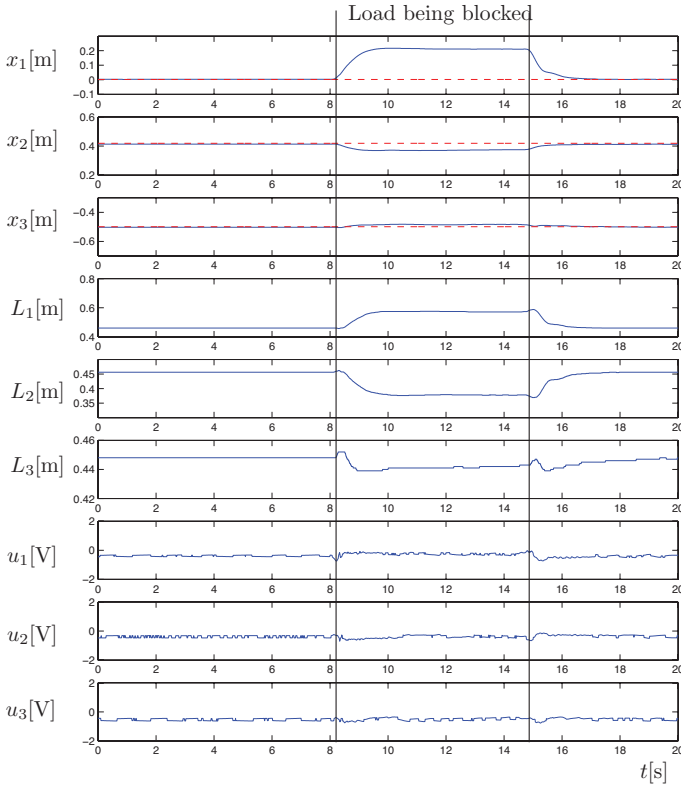


Figure 5.16: Performance when the load is being blocked, and subsequent load stabilization. The dashed-lines represent the reference values.

Trajectory tracking.

A circular reference trajectory is provided. Figures 5.17 and 5.18 show that the load position tracks the reference even after a sudden disturbance takes place at time $t = 2.7$ s. Again, the load rapidly catches up with the reference in a highly dynamic fashion. This can also be seen in the 3D Figure 5.18 where, once the disturbance takes place, the load rapidly cuts across the circle, along the diameter, to catch back with the reference.

Careful examination shows that there remains a slight tracking error along the x_3 -axis, which is not the same for each rotation. However, the

x_1 and x_2 -axes are in perfect agreement with the references. This can be explained by the following geometrical consideration. Table 5.4 shows that the chosen ring position $x_{3ref} = -0.49$ [m] is close to that of the fixed pylons. Hence, this requires a large force along the horizontal cables, and leads to a loss of sensitivity.

5.7 Conclusion

Crane control using the jet-scheduling mechanism improves over the traditional feedback linearization technique, especially regarding robustness when there is heavy dry friction on the trolleys and winching mechanism. These unmodelled dynamics can be handled by the high-gain low-order controller that forces the true system to match the jets provided by the scheduler. In particular, the Coulomb friction on the motors and on the trolley is important and difficult to identify. Due to these unmodeled (or poorly modeled) friction effects, jet-scheduling control for cranes can improve significantly the performance of cranes compared to exact dynamical feedback linearization.

The low-order controller is quite easy to design, mainly because it has a simple physical interpretation. The ξ variables associated with the part of \mathcal{S} involving the original states are directly the trolley position x_{0j} , ($j=1, 2$ for planar 2D systems, and $j=1,2,3$ for the 3D case) and the cable lengths L_i , $i=1, \dots, m+1$. These quantities can then be trivially controlled using the original inputs. The underactuated characteristics of the crane disappear at this stage since there are enough inputs to specify these quantities, a benefit of the jet-scheduling control methodology.

The complex nonlinear couplings, which are well modeled (there are mainly of geometric nature for which accurate measurements are the only prerequisite) are handled through the scheduling mechanism. The trivial, yet strongly uncertain dynamics are compensated through the low-order high-gain controller. The balance between the jet-scheduling part and the low-level controller is clearly linked to the choice of p .

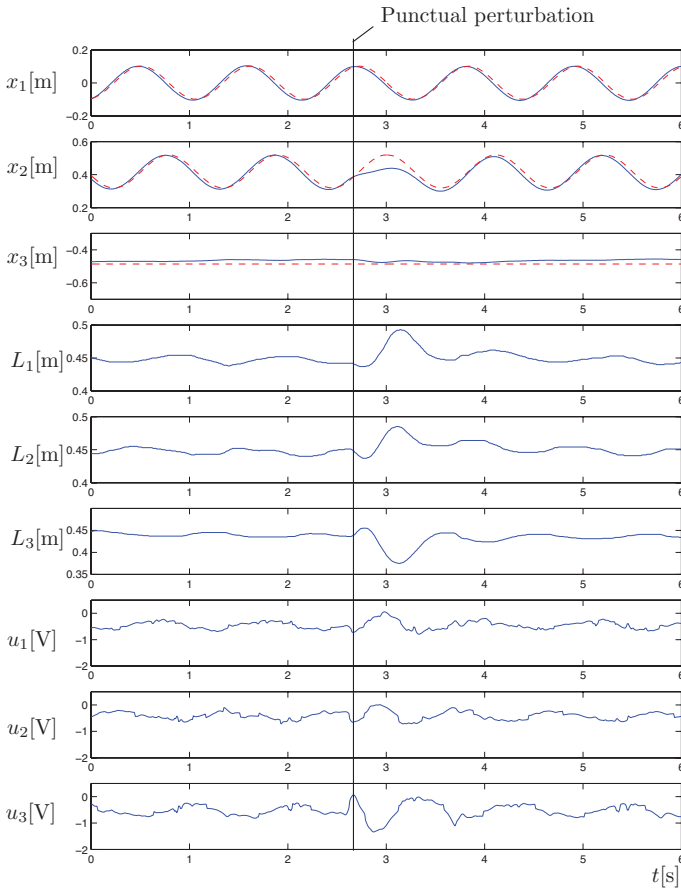


Figure 5.17: Tracking of a circular reference (height $x_{3ref} = -0.49$ [m], center at $x_{1ref} = 0$ [m], $x_{2ref} = 0.41$ [m], radius 0.1 [m], frequency 0.9 [Hz]). A sudden and short perturbation is applied at time $t = 2.7$ [s]. The dashed-lines represent the reference values.

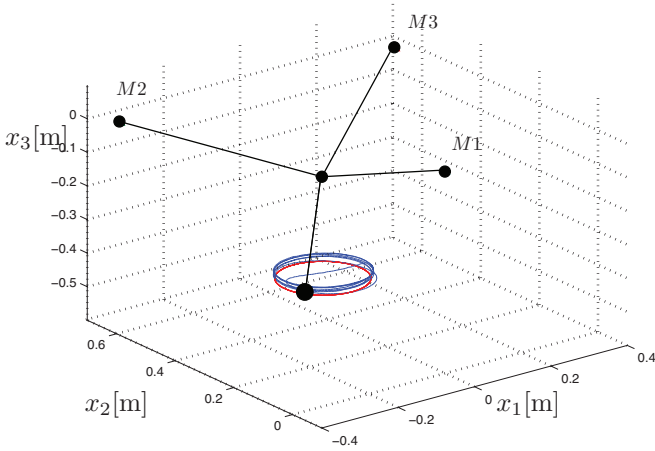


Figure 5.18: Three-dimensional view of the tracking of a circular reference. The reference is in solid red. The three motors are labeled M1, M2 and M3.

Chapter 6

Conclusions

The point of view adopted in this thesis is that, rather than tracking a full trajectory explicitly, a dynamical system coined “the jet scheduler” is set up to provide the derivatives (jets) of an ideal stabilizing trajectory. The jets are updated continuously according to the measured position of the system. The jets provide the ideal position of all the subparts of the system through the flat correspondence. They specify the ideal states that the system should have to converge smoothly towards the origin or along the reference trajectory to be tracked (depending upon whether point regulation or trajectory tracking is sought). Then, simple high-gain controllers are designed to enforce the real states to match the ones provided by the jets of the jet scheduler.

The interplay between the jet scheduler (which provides feedforward information, although this information is corrected through output measurements) and the high-gain controllers gives a control mechanism that is more robust than classical feedback linearization based on flatness. The only and main drawback of the proposed method is that the correspondence between the controlled system and a linear stable one is generally lost, contrary to dynamic linearization using flatness. This leads to a more subtle stability analysis than for classical feedback linearization schemes.

The two classes of systems addressed in this thesis illustrate how the general methodology can be applied in practice. Of course, the chosen examples are not exhaustive, and the idea proposed in Chapter 3 can be applied to many different flat systems.

From a formal standpoint, the thesis leaves some open issues, the main one being an abstract treatment of the low-order controller together with

associated criteria for choosing the value of p .

Both examples show improved robustness. The improvement can be explained largely by the intrinsic split that exists in the jet scheduler with one part imposing the jets (handling the nonlinear couplings) and the low-order controller ensuring that the system does match the jets. This low-level controller can be designed with high gain without disturbing the dominant time constants imposed by the scheduler. Such a splitting is difficult to achieve with flatness-based feedback linearization.

Another benefit of the jet-scheduling methodology over classical feedback linearization is that it gives simple and physically well-grounded controllers, which considerably simplifies the implementation step. Apart from the jet scheduler itself, the low-level controllers are P or PD controllers that can be designed easily for a particular purpose. This considerably simplifies the debugging stage.

The methodological Chapter 3 underlines however the challenges involved in defining such a procedure. Indeed, the flatness property together with the prolonged structure of the classical dynamical extension must be used to define the submanifold \mathcal{S} , from which the key variables ξ can be defined. They contrast with the natural α variables that result from flatness. These variables, in turn, give the structure of the low-order controller. This structure with the separation existing naturally in the prolonged dynamic extensions can be used to reduce the order of the controller.

However, the precise way and reason for the design of the controller is still obscured by insufficient formal treatment of the resulting dynamical structure. This direction could be investigated. This could also lead to possible extension of the method beyond the class of flat systems to some that are “almost” flat at least to a definable extent.

From a practical point of view, the jet-scheduling method for the class of cranes has shown how important the scheduling mechanism is in achieving fast positioning without induced oscillation. This is the case mainly because the load can be considered as a single point-mass payload. In the case of more complicated payloads such as vertical heavy beams, additional modeling is needed, mainly by considering a cascade of two flat systems, one for the crane and the other for the payload. The method should adapt to such a scenario with no major difficulty.

Another application directly linked to those presented is the coordinated steering of a platoon of vehicles (flocking for instance). Indeed, the nested structure of flat systems results in an overall flat parameterization well suited to the jet-scheduling approach.

Bibliography

- [1] M. Reyhanoglu A. M. Bloch and N. H. McClamroch. Control and stabilization of nonholonomic dynamic systems. *IEEE Transactions on Automatic Control*, 37:1746–1757, 1992.
- [2] D. Angelis, A. Casavola, and E. Mosca. A predictive command governor for nonlinear systems under constraints. In F. Allgöwer and A. Zheng, editors, *Nonlinear Model Predictive Control*, pages 115–126, Basel, Switzerland, December 2000.
- [3] A. Astolfi. Discontinuous control of nonholonomic systems. *Systems and Control Letters*, 27:37–45, 1997.
- [4] M. Bak, N. K. Poulsen, and O. Ravn. Path following mobile robot in the presence of velocity constraints. In *Technical Report, Informatics and Mathematical Modelling, Technical University of Denmark*, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, 2001.
- [5] L. T. Biegler. Efficient solution of dynamic optimization and nmpc problems. In F. Allgöwer and A. Zheng, editors, *Nonlinear Model Predictive Control*, pages 219–243, Basel, Switzerland, December 2000.
- [6] A. M. Bloch. *Nonholonomic Mechanics and Control*. Springer, 175, Fifth Avenue, New York 10010,USA, 2003.
- [7] T. J. J Boom, J. B. Klaassens, and R. Meiland. *Informatics in Control, Automation and Robotics II*, chapter Real-Time Time-Optimal Control for a Nonlinear Container Crane Using a Neural Network, pages 79–84. Springer Netherlands, 2007.

- [8] R. W. Brockett. Asymptotic stability and feedback stabilization. In *Differential Geometric Control Theory*, Progress in Mathematics, pages 181–191. Birkhauser, Boston, Brockett, Millman, Sussman edition, 1983.
- [9] H. Butler, G. Honderd, and J. Van Amerongen. Model reference adaptive control of a gantry crane scale model. *IEEE Control System Magazine*, page 5762, January 1991.
- [10] C. Canudas de Wit and G Bastin. *Theory of Robot Control*. Springer, New York, USA, 1996.
- [11] C. Canudas de Wit and O. J. Sordalen. Exponential stabilization of mobile robots with nonholonomic constraints. *IEEE Transactions on Automatic Control*, 37:1791–1797, 1992.
- [12] B. Charlet, J. Lévine, and R. Marino. Sufficient conditions for dynamic state feedback linearization. *SIAM J. Control Optimization*, 29(1):38–57, 1991.
- [13] C.-T. Chen. *Linear System Theory and Design*. Oxford University Press, New York, 3rd edition, 1999.
- [14] H. Chen and F. Allgöwer. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34:1205–1218, 1998.
- [15] W.-H. Chen, D. J. Balance, and J. O’Reilly. Model predictive control of nonlinear systems: Computational burden and stability. *IEE Proceedings in Control Theory and Applications*, 147:387–394, 2000.
- [16] G. Corrigan, A. Giua, and G Usai. An implicit gain-scheduling controller for cranes. *IEEE Transactions on Control Systems Technology*, 6(1):15–10, January 1998.
- [17] E. Delaleau and V. Hagenmeyer. Commande prédictive non linéaire fondée sur la platitude du moteur à induction. application au positionnement de précision. *Journal Européen des Systèmes Automatisés*, 36(5):737–748, 2002.
- [18] T. Devos and J. Lévine. A flatness-based nonlinear predictive approach for crane control. In *Proceedings of IFAC Workshop on Nonlinear Model Predictive Control for Fast Systems 2006*, Grenoble, France, october 2006.

- [19] W.E. Dixon, D.M. Dawson, E. Zergeroglu, and A. Behal. *Nonlinear Control of Wheeled Mobile Robots*. Springer, New York, USA, 2001.
- [20] W.E. Dixon, D.M. Dawson, F. Zhang, and E. Zergeroglu. Global exponential tracking control of a mobile robot system via a pe condition. In *In Proceedings of the 38th IEEE Conference on Decision and Control*, pages 4822–4827, Phoenix, Arizona, 1999.
- [21] K. D. Do, Z. P. Jiang, and J. Pan. A global output-feedback controller for simultaneous tracking and stabilization of unicycle-type mobile robots. *IEEE Transactions on Robotics and Automation*, 20:589–594, 2004.
- [22] K. D. Do, Z. P. Jiang, and J. Pan. Simultaneous tracking and stabilization of mobile robots: An adaptive approach. *IEEE Transactions on Automatic Control*, 49:1147–1152, 2004.
- [23] G. Dongbing and H. Huosheng. A stabilizing receding horizon regulator for nonholonomic mobile robots. *IEEE Transactions on Robotics*, 21(5):1022 – 1028, October 2005.
- [24] R. C. Dorf and R. H. Bishop. *Modern Control Systems*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2000.
- [25] Y. Fang, W. E. Dixon, D. M. Dawson, and E. Zergeroglu. Nonlinear coupling control laws for an underactuated overhead crane system. *IEEE/ASME Transactions on Mechatronics*, 8(3):418–423, 2003.
- [26] J. Ferruz and A. Ollero. Visual generalized predictive path tracking. In *5th International Workshop on Advanced Motion Control, AMC*, Coimbra, Portugal, 1998.
- [27] R. Fierro and F. Lewis. Control of a nonholonomic mobile robot: Backstepping kinematics into dynamics. In *In Proceedings of the 34th IEEE Conference on Decision and Control*, pages 3805–3810, New Orleans, LA, 1995.
- [28] A. F. Filippov. *Differential Equations with Discontinuous Righthand Sides*. Kluwer Academic Publishers, Dordecht, The Netherlands, 1988.
- [29] R. Findeisen and F. Allgöwer. A nonlinear model predictive control scheme for the stabilization of set point families. *Quarterly Journal on Automatic Control*, 41(1):37–45, 2000.

- [30] R. Findeisen, M. Diehl, Z. Nagy, F. Allgöwer, H. G. Bock, and J. P. Schlöder. Computational feasibility and performance of nonlinear model predictive control schemes. In *In Proceedings of the European Control Conference, ECC*, Porto, Portugal, 2001.
- [31] M. Fliess, J. Lévine, Ph. Martin, and P. Rouchon. Linéarisation par bouclage dynamique et transformations de Lie-Bäcklund. *C.R. Acad. Sci. Paris*, I-317:981–986, 1993.
- [32] M. Fliess, J. Lévine, Ph. Martin, and P. Rouchon. Nonlinear control and Lie-Bäcklund transformation: Toward a new differential standpoint. In *In Proceedings of the 33rd IEEE Conference on Decision and Control*, pages 339–344, Lake Buena Vista, FL, 1994.
- [33] M. Fliess, J. Levine, Ph. Martin, and P. Rouchon. Design of trajectory stabilizing feedback for driftless flat systems. In *In Proceedings of the 3rd European Control Conference*, page 1882 1887, Rome, Italy, 1995.
- [34] M. Fliess, J. Lévine, Ph. Martin, and P. Rouchon. Flatness and defect of nonlinear systems: Introductory theory and examples. *International Journal of Control*, 61(6):1327–1361, 1995.
- [35] M. Fliess, J. Lévine, Ph. Martin, and P. Rouchon. A Lie-Bäcklund approach to equivalence and flatness of nonlinear systems. *IEEE Transactions on Automatic Control*, 44:922–937, 1999.
- [36] M. Fliess and R. Marquez. Continuous-time linear predictive control and flatness: a module-theoretic setting with examples. *International Journal of Control*, 73(7):606–623, May 2000.
- [37] D. T. Greenwood. *Classical Dynamics*. Prentice-Hall, Englewood Cliffs, N.J., 1977.
- [38] T. Gustafsson. On the design and implementation of a rotary crane controller. *European J. Control*, 2(3):166–175, March 1996.
- [39] V. Hagenmeyer. *Robust Nonlinear Tracking Control Based on Differential Flatness*. Ph. D. thesis, Université Paris XI Orsay, Paris, France, 2002.
- [40] W. Hahn. *Stability of Motion*. Springer-Verlag, 1967.

- [41] K. S Hong, J. H. Kim, and K. I Lee. Control of a container crane: Fast traversing, and residual sway control from the perspective of controlling an underactuated system. In *In Proceedings of the American Control Conference*, pages 1294–1298, Philadelphia, PA, June 1998.
- [42] A. Isidori. *Nonlinear Control Systems*. Springer-Verlag, Berlin, Heidelberg, New York, 2nd edition, 1989.
- [43] Z-P. Jiang. Lyapunov design of global state and output feedback trackers for non-holonomic control systems. *International Journal of Control*, 73:744–761, 2000.
- [44] Z. P. Jiang. Robust exponential regulation of nonholonomic systems with uncertainties. *Automatica*, 36:189–209, 2000.
- [45] Z. P. Jiang and H. Nijmeijer. Tracking control of mobile robots: A case study in backstepping. *Automatica*, 33:1393–1399, July 1997.
- [46] Y. Kanayama, Y. Kimura, F. Miyazaki, and Noguchi T. A stable tracking control scheme for an autonomous mobile robot. In *In Proceedings of the IEEE International Conference on Robotics and Automation*, pages 384–389, Cincinnati, Ohio, 1990.
- [47] H. K. Khalil. *Nonlinear Systems*. Prentice-Hall, Englewood Cliffs, N.J., 3rd edition, 2002.
- [48] B Kimiaghalam and A. Homaifar. An application of model predictive control for a shipboard crane. In *Proceedings of the American Control Conference*, pages 929–934, Arlington, VA, June 2001.
- [49] B. Kiss, J. Lévine, and Ph. Mullhaupt. Modelling, flatness and simulation of a class of cranes. *Periodica Polytechnica Ser. El. Eng.*, 43(3):215–225, 1999.
- [50] B. Kiss, J. Lévine, and Ph. Mullhaupt. Global stability without motion planing may be worse than local tracking. In *the Proceedings of the European Control Conference*, pages 1106–1110, Porto, Portugal, June 2001.
- [51] I. Kolmanovsky and N. H. McClamroch. Developments in nonholonomic control problems. *IEEE Control Systems Magazine*, 15:20–36, December 1995.

- [52] F. Kuhne, W. Fetter Lages, and F. M. Gomes da Silva Jr. Point stabilization of mobile robots with nonlinear model predictive control. In *Proceedings of the IEEE International Conference on Mechatronics and Automation*, pages 1163–1168, Niagara Falls, Canada, July 2005.
- [53] H.-H. Lee. A new approach for the anti-swing control of overhead cranes with high-speed load hoisting. *International Journal of Control*, 76,(15):1493 – 1499, October 2003.
- [54] J. Lévine. *Analyse et commande des systèmes non linéaires*. Notes de cours de l'École Nationale des Ponts et Chaussées, Paris, 2004.
- [55] J. Lévine. On the equivalence between differential flatness and dynamic feedback linearization. In *3rd IFAC Symposium on System, Structure and Control*, Iguassu Falls, Brazil, October 2007.
- [56] R. Longchamp. *Commande numérique de systèmes dynamique*. Presses polytechniques et universitaires romandes, Lausanne, Suisse, 2ème édition, 2006.
- [57] R. Mahadevan, S.K. Agrawal, and F.J Doyle III. Differential flatness based nonlinear predictive control of fed-batch bioreactor. *Control Engineering Practice*, 9(8):889–899, 2001.
- [58] Ph. Martin, R.M. Murray, and P. Rouchon. A simple output feedback controller for nonlinear cranes. In *Mini-Course ECC'97*, pages 211–258, Brussels, Belgium, July 1997.
- [59] L. F. Mendonca, J. M. Sousa, and J. M. G. Sa da Costa. Optimization problems in multivariable fuzzy predictive control. *International Journal of Approximate Reasoning*, 36(3):199 – 221, July 2004.
- [60] H. Nijmeijer and A.J. van der Schaft. *Nonlinear Dynamical Control Systems*. Springer-Verlag, New York, Berlin, Heidelberg, 1990.
- [61] J. Normey-Rico, J. Gomez-Ortega, and E. Camacho. Smith-predictor-based generalized predictive controller for mobile robot path-tracking. *Control Engineering Practice*, 7(6):729–740, 1999.
- [62] W. Oelen and J. van Amerongen. Robust tracking control of two-degrees-of-freedom mobile robots. *Contr. Eng. Prac.*, 2:333–340, 1994.

- [63] R. Olfati-Saber. Global configuration stabilization for the vtol aircraft with strong input coupling. *IEEE Transactions on Automatic Control*, 47(11):1949–1952, November 2002.
- [64] A. Ollero and O. Amidi. Predictive path tracking of mobile robots. In *Proceedings of 5th International Conference on Advanced Robotics, Robots in Unstructured Environments (ICAR '91)*, June 1991.
- [65] G. Oriolo, A. De Luca, and M. Vendittelli. Wheeled mobile robot control via dynamic feedback linearization: Design, implementation, and experimental validation. *IEEE Transactions on Control Systems Technology*, 10(6):835–852, November 2002.
- [66] J. B. Pomet. Explicit design of time-varying stabilizing control laws for a class of controllable systems without drift. *Systems and Control Letters*, 18:147–158, 1992.
- [67] J. Richalet. *Pratique de la commande prédictive*. Hermès, Paris, 1993.
- [68] P. Rouchon, M. Fliess, J. Lévine, and Martin. Flatness, motion planning and trailer systems. In *In Proceedings of the 32nd IEEE Conference on Decision and Control*, pages 2700–2705, San Antonio, TX, 1993.
- [69] E. P. Ryan. On Brockett’s condition for smooth stabilizability and its necessity in a context of nonsmooth feedback. *SIAM J. Control Optimization*, 32(6):1597–1604, 1994.
- [70] Y. Sakawa and H. Sano. Nonlinear model and linear robust control of overhead traveling cranes. *Nonlinear Analysis*, 30(11):2197–2207, 1997.
- [71] Ch. Salzmann, D. Gillet, and P. Huguenin. Introduction to real-time control using LabVIEW with an application to distance learning. *International Journal of Engineering Education*, 16(3):255–272, 2000.
- [72] C. Samson. Control of chained systems application to path following and time-varying point-stabilization of mobile robots. *IEEE Transactions on Automatic Control*, 40:64–77, Jan 1995.
- [73] C. Samson and K. Ait-Abderrahim. Feedback control of a nonholonomic wheeled cart in cartesian space. In *Proc. 1991 IEEE Conf. Robotics and Automation*, pages 1136–1141, Sacramento, 1991.

- [74] R Soeterboek. *Predictive Control: A Unified Approach*. Prentice-Hall, New York, 1992.
- [75] E. D. Sontag. *Mathematical Control Theory*. Springer Verlag, 2nd edition, 1998.
- [76] L. Ti-Chung, S. Kai-Tai, L. Ching-Hung, and T. Ching-Cheng. Tracking control of unicycle-modeled mobile robots using a saturation feedback controller. *IEEE Transactions on Control Systems Technology*, 9:305–318, 2001.
- [77] M. Vidyasagar. *Nonlinear Systems Analysis*. Prentice-Hall, Englewood Cliffs, N.J., second edition, 1993.
- [78] G. Walsh, D. Tilbury, S. Sastry, R. Murray, and J. Laumond. Stabilization of trajectories for systems with nonholonomic constraints. *IEEE Transactions on Automatic Control*, 39:216–222, 1994.
- [79] J. Wan, C.G. Quintero, N. Luo, and J. Vehi. Predictive motion control of a mirosot mobile robot. In *Proceedings in World Automation Congress*, volume 21, pages 325–330, June 2004.
- [80] K. Yoshida and H. Kawabe. A design of saturating control with a guaranteed cost and its application to the crane control system. *IEEE Transactions on Automatic Control*, 37(1):121–127, January 1992.
- [81] J. Zabczyk. Some comments on stabilizability. *Applied Mathematics and Optimization*, 19(1):1–9, 1989.
- [82] A. H. Zemanian. *Distribution Theory and Transform Analysis: An Introduction to Generalized Functions, with Applications (Paperback)*. Dover Publications, Inc, New York, 2nd edition, 1987.

Davide Buccieri

4 Bd de la Liberté
2300 La Chaux-de-fonds (Suisse)
+ 41 926 89 23

Born in La Chaux-de-fonds the 28th November 1979
Nationality: Italian

Education

- 2003-2007: École Polytechnique Fédérale de Lausanne, **Doctoral studies and research** in Automatic Control.
- 1998-2003: École Polytechnique Fédérale de Lausanne, **Master of Science** in Mechanical Engineering.

Publications

- D. Buccieri, Ph. Mullhaupt, and D. Bonvin. SpiderCrane: Model and Properties of a Fast Weight-Handling Equipment. In *16th IFAC World Congress*, Prague, 2005.
- D. Buccieri, Ph. Mullhaupt, Z.P. Jiang, and D. Bonvin. Velocity Scheduling Controller for a Nonholonomic Mobile Robot. In *Chinese Control Conference*, Harbin, 2006.
- D. Buccieri, D. Perritaz, Ph. Mullhaupt, Z. P. Jiang, and D. Bonvin. Velocity-Scheduling Control for a Nonholonomic Mobile Robot: Theoretical and Experimental Results. *IEEE Transactions on Robotics and Automation*.(In revision).
- D. Buccieri, C. Salzmann, Ph. Mullhaupt, and D. Bonvin. Jet-Scheduling Control for SpiderCrane: Experimental Results. In *IFAC World Congress*, Seoul, 2008. (Submitted).
- D. Buccieri, J. Sanchez, S. Dormido, Ph. Mullhaupt, and D. Bonvin. Interactive 3D Simulation of Flat Systems: The SpiderCrane as a Case Study. In *CDC/ECC*, Seville, 2005.

- D. Bucciari, D. Perritaz, Ph. Mullhaupt, Z. P. Jiang, and D. Bonvin. Experimental Results for a Nonholonomic Mobile Robot Controller Enforcing Linear Equivalence Asymptotically. In *IEEE Conference on Industrial Electronics and Applications*, Singapore, 2006.
- S. Gros, D. Bucciari, Ph. Mullhaupt, and D. Bonvin. A Two-Time-Scale Control Scheme for Fast Systems. In *Int. Workshop on Non-linear Model Predictive Control*, Freudenstadt-Lauterbad, 2005.
- S. Gros, D. Bucciari, Ph. Mullhaupt, and D. Bonvin. A Two-Time-Scale Control Scheme for Fast Systems. *Springer Lecture Notes in Control and Information Sciences Series (LNCIS)*, 358:551–563, 2007.
- Ph. Mullhaupt, D. Bucciari, and D. Bonvin. A Numerical Sufficiency Test for the Asymptotic Stability of Linear Time-Varying Systems. *Automatica*, 43(4):631–638, 2007.