

**POLITECNICO DI TORINO**

**Master's Degree in Computer Engineering**



**Master's Degree Thesis**

# **Nonlinear and LPV Decoupling Control Strategies for Robotic Manipulators**

**Supervisor**

Prof. Marina Indri

**Candidate**

Davide Fricano

**Supervisors at Comau**

Alberto Gagliano

Giuseppe Avallone

**April 2025**



# Abstract

The thesis arises from the need of the robotics company Comau to address the issue of dynamic inertial coupling between the links of a robotic manipulator. The objective is to analyze the theory and experimentally validate the application of diagonalization techniques on the dynamic model of an industrial robotic manipulator, based on the design of a pre-compensator acting on the robotic plant. This decoupling compensator allows the implementation of independent joint control, treating each link as a SISO subsystem, thereby simplifying the controller synthesis and improving tracking accuracy.

In particular, the thesis explores different decoupling strategies in both the time domain and the frequency domain. The study begins with the modeling of robotic manipulators, considering various dynamic representations, including rigid-body models, elastic joint models, and damped joint models, leading to the formulation of the model used in Comau's industrial simulations. For time-domain decoupling, the nonlinear feedback linearization technique is applied to derive a nonlinear control law that effectively decouples the system for different manipulator models. On the other hand, the frequency-domain decoupling approach relies on the design of linear compensators, investigating both static and dynamic techniques, including static, SVD-based decoupling, ideal decoupling, simplified decoupling, and inverted decoupling, with the goal of minimizing interaction effects and enhancing closed-loop performance.

Extensive simulations and experimental validations are carried out on a Comau robotic manipulator model to evaluate the effectiveness of each method, highlighting the trade-offs between decoupling, computational complexity and robustness to modeling uncertainties.

# Acknowledgements

Dedico questa tesi a pochi ma essenziali pilastri.

Alla mia famiglia: a mamma che è sempre stata presente in qualsiasi momento; a papà che pur non approvando questo percorso lo ha finanziato; alla nonna e il nonno che contribuendo ci hanno creduto fino in fondo; alla zia Francesca che è stata di fondamentale supporto; allo Zio Giampiero e alla Zia Marinella per l'affetto.

Ai miei amici: Alessio, come caro amico e compagno di eventi; Giulia, come amica e coinquilina, per essere stata un punto fermo nel percorso universitario; Simone, come amico e collega, sempre brillante; Edoardo, come collega, capro espiatorio di tutti i mali del mondo; Miriana, come amica, con cui ho condiviso momenti difficili...

Una menzione speciale va alla mia cara Iris, che ha aggiunto un pizzico di colore ad una tela ingrigita, testimone del fatto che la razionale fortuna, intesa come *da etimo*, non è solo costruita sullo sforzo e sul sacrificio ma è soggetta a fenomeni non controllabili, deterministici e aleatori.

*Considerate la vostra semenza:  
fatti non foste a viver come bruti,  
ma per seguir virtute e canoscenza.*  
Dante Alighieri



# Table of Contents

<b>List of Tables</b>	VII
<b>List of Figures</b>	IX
<b>Acronyms</b>	XII
<b>1 Introduction</b>	1
1.1 Motivation and Objectives . . . . .	1
1.2 Literature Review . . . . .	3
1.3 Thesis Structure . . . . .	5
<b>2 Robot Modelling</b>	7
2.1 Robot Manipulator Dynamics . . . . .	7
2.1.1 Rigid Robot Dynamics . . . . .	7
2.1.2 Elastic Joint Robot Dynamics . . . . .	13
2.1.3 Relative Elastic-Dumped Joint Robot Dynamics . . . . .	15
2.1.4 Direct Elastic-Dumped Joint Robot Dynamics . . . . .	17
2.1.5 Complete Elastic-Dumped Joint Robot Dynamics . . . . .	18
<b>3 Time-Domain Decoupling Control</b>	23
3.1 Elastic-Joint Robots - Input-State Feedback Linearization . . . . .	24
3.2 Direct Elastic-Dumped Joint Robots - Input-State Feedback Linearization . . . . .	27
3.3 Relative Elastic-Damped Joint Robots - Input-Output Feedback Linearization . . . . .	29
<b>4 Frequency-Domain Decoupling Control</b>	31
4.1 Linear Decoupling Overview . . . . .	31
4.2 Static Decoupling . . . . .	34
4.2.1 Static Decoupling . . . . .	35
4.2.2 SVD Decoupling . . . . .	37

4.3	Dynamic Decoupling . . . . .	40
4.3.1	Ideal Decoupling . . . . .	40
4.3.2	Simplified Decoupling . . . . .	42
4.3.3	Inverted Decoupling . . . . .	45
<b>5</b>	<b>Simulations and Results</b>	<b>50</b>
5.1	Robot Dynamics Simulation . . . . .	51
5.1.1	Comau Robot Manipulator Model . . . . .	51
5.1.2	Comau Robot Linearized Model . . . . .	57
5.2	Decoupling Metrics . . . . .	61
5.2.1	Transfer Function Matrix . . . . .	61
5.2.2	Relative Gain Array . . . . .	62
5.2.3	Gershgorin Radius . . . . .	63
5.2.4	Control Response . . . . .	64
5.3	Decentralized and Decoupled LTI Dynamics . . . . .	65
5.3.1	Motor Torque $\tau_m$ to joint position $\mathbf{q}$ dynamics . . . . .	66
5.3.2	Motor Torque $\tau_m$ to Motor Angles $\theta$ dynamics . . . . .	75
5.4	Control response of Decentralized and Decoupled LTI Dynamics . .	76
5.4.1	Joint Step Response of $q/\tau_m$ Dynamics . . . . .	77
5.4.2	Motor Step Response of $\theta/\tau_m$ Dynamics . . . . .	80
5.5	Decentralized and Decoupled LPV Dynamics . . . . .	83
5.6	Control response of Decentralized and Decoupled LPV Dynamics .	87
5.7	Control response of Decentralized and LPV Decoupled Nonlinear Dynamics . . . . .	92
<b>6</b>	<b>Conclusions and Future Works</b>	<b>97</b>
6.1	Conclusions . . . . .	97
6.2	Future Works . . . . .	98
<b>A</b>	<b>Supplementary Tables</b>	<b>99</b>
A.1	Step Response of Joint velocity $\dot{q}/\tau_m$ and Motor velocity $\dot{\theta}/\tau_m$ Dy- namics . . . . .	99
	<b>Bibliography</b>	<b>106</b>

# List of Tables

5.1	joint position $\mathbf{q}$ response comparison: RMSE [rad] with different reference steps (target [rad]) for each output (n) . . . . .	77
5.2	joint position $\mathbf{q}$ response comparison: overshoot [rad] with different reference steps (target [rad]) for each output (n) . . . . .	78
5.3	joint position $\mathbf{q}$ response comparison: settling time with different reference steps (target [rad]) for each output (n) . . . . .	79
5.4	motor shaft position $\theta$ response comparison: RMSE [rad] with different reference steps (target [rad]) for each output (n) . . . . .	83
5.5	motor shaft position $\theta$ response comparison: overshoot [rad] with different reference steps (target [rad]) for each output (n) . . . . .	84
5.6	motor shaft position $\theta$ response comparison: settling time with different reference steps (target [rad]) for each output (n) . . . . .	85
5.7	LPV robot model $\mathbf{q}$ response comparison: RMSE, overshoot and settling time with reference steps (t.) of 1 rad for axis 1 and 2 rad for axis 2 . . . . .	92
5.8	joint position $\mathbf{q}$ nonlinear response comparison: RMSE [rad] with different reference steps (target [rad]) for each output (n) . . . . .	94
5.9	joint position $\mathbf{q}$ nonlinear response comparison: absolute overshoot [rad] with different reference steps (target [rad]) for each output (n)	94
5.10	joint position $\mathbf{q}$ nonlinear response comparison: settling time [s] with different reference steps (target [rad]) for each output (n) . . .	95
A.1	link velocity $\dot{\mathbf{q}}$ response comparison: RMSE [rad] with different reference steps (target [rad]) for each output (n) . . . . .	100
A.2	link velocity $\dot{\mathbf{q}}$ response comparison: overshoot [rad] with different reference steps (target [rad]) for each output (n) . . . . .	101
A.3	link velocity $\dot{\mathbf{q}}$ response comparison: settling time [s] with different reference steps (target [rad/s]) for each output (n) . . . . .	102
A.4	motor velocity $\dot{\theta}$ response comparison: RMSE [rad] with different reference steps (target [rad/s]) for each output (n) . . . . .	103



A.5	motor velocity $\dot{\theta}$ response comparison: overshoot [rad] with different reference steps (target [rad]) for each output (n) . . . . .	104
A.6	motor velocity $\dot{\theta}$ response comparison: settling time [s] with different reference steps (target [rad/s]) for each output (n) . . . . .	105

# List of Figures

4.1	General Control structure of MIMO system . . . . .	32
4.2	Decoupled Control structure of MIMO system . . . . .	32
4.3	Decentralized Control structure of TITO system . . . . .	33
4.4	Static Decoupling Control structure of MIMO system . . . . .	36
4.5	SVD Decoupling Control structure of MIMO system . . . . .	38
4.6	Ideal Decoupling Control structure of TITO system . . . . .	42
4.7	Simplified Decoupling Control structure of TITO system . . . . .	44
4.8	Inverted Decoupling Control structure of TITO system . . . . .	47
5.1	Elasticity behaviour in terms of deflection angle with respect to torque	53
5.2	Non-linear friction torque with respect to motor speed for a RR robot	54
5.3	Illustration of a 2-DOF RR robot manipulator with flexible joints [33]	54
5.4	Flexible-Joint Robot Model (EJM) block diagram . . . . .	55
5.5	Rigid-robot model block diagram . . . . .	56
5.6	Elastic transmission block diagram . . . . .	56
5.7	Motors' block diagram . . . . .	57
5.8	Simulink block diagram of the 2-axes LPV robot model . . . . .	61
5.9	Bode plot of the transfer function matrix $\mathbf{q}/\boldsymbol{\tau}_m$ with Static decoupling	67
5.10	RGA of the transfer function matrix $\mathbf{q}/\boldsymbol{\tau}_m$ with Static decoupling .	67
5.11	Gershgorin Radius of the transfer function matrix $\mathbf{q}/\boldsymbol{\tau}_m$ with Static decoupling . . . . .	68
5.12	Bode plot of the transfer function matrix $\mathbf{q}/\boldsymbol{\tau}_m$ with SVD static decoupling . . . . .	69
5.13	RGA of the transfer function matrix $\mathbf{q}/\boldsymbol{\tau}_m$ with SVD static decoupling	69
5.14	Gershgorin Radius of the transfer function matrix $\mathbf{q}/\boldsymbol{\tau}_m$ with SVD static decoupling . . . . .	70
5.15	Bode plot of the transfer function matrix $\mathbf{q}/\boldsymbol{\tau}_m$ with Ideal decoupling	70
5.16	RGA of the transfer function matrix $\mathbf{q}/\boldsymbol{\tau}_m$ with Ideal decoupling . .	71
5.17	Gershgorin Radius of the transfer function matrix $\mathbf{q}/\boldsymbol{\tau}_m$ with Ideal decoupling . . . . .	71

5.18	Bode plot of the transfer function matrix $\mathbf{q}/\tau_m$ with Simplified decoupling . . . . .	72
5.19	RGA of the transfer function matrix $\mathbf{q}/\tau_m$ with Simplified decoupling	72
5.20	Gershgorin Radius of the transfer function matrix $\mathbf{q}/\tau_m$ with Simplified decoupling . . . . .	73
5.21	Bode plot of the transfer function matrix $\mathbf{q}/\tau_m$ with Inverted decoupling . . . . .	73
5.22	RGA of the transfer function matrix $\mathbf{q}/\tau_m$ with Inverted decoupling	74
5.23	Gershgorin Radius of the transfer function matrix $\mathbf{q}/\tau_m$ with Inverted decoupling . . . . .	74
5.24	Response plots of the transfer function matrix $\mathbf{q}/\tau_m$ providing reference step values 1 rad for axis 1 and 0 rad for axis 2 . . . . .	80
5.25	Response plots of the transfer function matrix $\boldsymbol{\theta}/\tau_m$ providing reference step values 0 rad for motor 1 and 1 rad for motor 2 . . . . .	81
5.26	Response plots of the transfer function matrix $\mathbf{q}/\tau_m$ providing reference step values 0 rad for axis 1 and 1 rad for axis 2 . . . . .	82
5.27	Response plots of the transfer function matrix $\boldsymbol{\theta}/\tau_m$ providing reference step values 1 rad for motor 1 and 0 rad for motor 2 . . . . .	86
5.28	Bode plots of the LPV Static decoupled transfer function matrix $\mathbf{q}/\tau_m$ for a specified trajectory . . . . .	86
5.29	Bode plots of the LPV SVD decoupled transfer function matrix $\mathbf{q}/\tau_m$ for a specified trajectory . . . . .	87
5.30	Bode plots of the LPV Ideal decoupled transfer function matrix $\mathbf{q}/\tau_m$ for a specified trajectory . . . . .	87
5.31	Bode plots of the LPV Simplified decoupled transfer function matrix $\mathbf{q}/\tau_m$ for a specified trajectory . . . . .	88
5.32	Bode plots of the Inverted decoupled LPV transfer function matrix $\mathbf{q}/\tau_m$ for a specified trajectory . . . . .	88
5.33	Step response of the LPV system vs the LTI system, around $\mathbf{q}_0$ . .	89
5.34	LPV Simplified decoupler . . . . .	90
5.35	LPV Inverted decoupler . . . . .	90
5.36	Step response of the LPV, Simplified and Inverted decoupled systems	91
5.37	Step response of the Decentralized, Simplified decoupled and Inverted decoupled control systems on the nonlinear robot plant . . . . .	93



# Acronyms

**MIMO**

Multi-Input Multi-Output

**TITO**

Two-Input Two-Output

**SISO**

Single-Input Single-Output

**LTI**

Linear Time-Invariant

**LPV**

Linear Parameter-Varying

**PID**

Proportional-Integrative-Derivative

**SVD**

Singular Value Decomposition

**RMSE**

Root Mean Square Error

**RGA**

Relative Gain Array

**SCARA**

Selective Compliance Assembly Robot Arm

# Chapter 1

## Introduction

The dynamics of multivariable systems, such as robotic manipulators, are complex due to nonlinearities, varying parameters, and input-output couplings, for which a change in one input affects several outputs. To simplify control, decoupled transfer behaviors can be designed, where each output variable is influenced by a single reference signal, allowing control through SISO controllers. Two possible strategies to address dynamic couplings are proposed: extending LTI decoupling techniques to the LPV robot model or using a nonlinear approach to obtain independent and decoupled linear systems.

### 1.1 Motivation and Objectives

The thesis focuses on the problem of dynamic inertial coupling in industrial manipulators, encountered in many Comau robotic systems. In fact, the interaction between different links of the kinematic chain of a robotic manipulator leads to coupled dynamics, which complicates control design and affects precision in trajectory tracking for all axes, including not only the controlled one but also the response of the other links.

Comau's control architecture for a robot with  $n$  joints, employs a decentralized approach, namely an independent-joint control based on a gain scheduling robust technique, whose synthesis involves the design of  $n$  different linear controllers which are tuned on the dynamics of the relative link that must be actuated, meaning that no behaviour is imposed explicitly on the cross channel dynamics, for computational efficiency, differently from the centralized strategies. The issue of coupling inertial effects is an important topic which has not been explicitly faced at Comau yet. Until now, the independent-joint control decentralized approach provided sufficiently good performance for most of user cases, mainly in a low-frequency bandwidth, however in fast movement applications the linear control could lack

of accuracy, due to the coupling effects among different links, indeed. And this is the reason for which Comau needs to look for a system that would allow it to "diagonalize" the robotic system, ideally as an additive control block which inhibits the mutual couplings, to be placed inside the multi-loop cascade control system before the plant.

Therefore the objective of this work is to diagonalize the robotic system, enabling each joint to function as an isolated Single-Input Single-Output (SISO) system. By achieving decoupling, implemented as a pre-compensation control law for the plant, not only are performance and robustness improved, but the controller design process is also simplified due to the reduced influence of other joints.

A first theoretical idea to face the problem was the diagonalization of the inertia matrix, which is the main source of the couplings, apart from centrifugal and Coriolis effects, that being negligible are treated as disturbances. This technique, which can be seen as a Diagonalizing Modal Control, consists of a control space transformation where the inertia tensor becomes diagonal. However, a change of coordinates-based approach, although it simplifies the controller synthesis by obtaining independent SISO systems in the modal space and enhances the performance, does not achieve the physical decoupling, as the mutual couplings are introduced back due to the inverse mapping of control signals on the joint space.

Thus, the problem must be solved designing on purpose a specific pre-compensator which reduces, hopefully nullifies, the mutual torque of a link received from the other ones. A possible technique is a feedforward compensation on the motor, assuming that the torque measurements are provided. The only problem of this approach is the necessity of installing torque sensors or estimating indirectly the cross-axes mutual forces. In order to achieve dynamic decoupling, it is possible to arrange the two basic ideas: on one hand the diagonalization of the dynamics and on the other hand the design of a compensator. Anyway, the diagonalization is not referred to the inertia tensor, but it is rather extended to the whole system dynamics, and specifically, if the linear model is available, to the transfer function matrix. In this way, the aim is to devise a control algorithm which provides off-diagonal feedforward compensation to reduce the mutual dynamics, in order to eventually obtain a diagonalized transfer function matrix. Based on the latter intuition, a deep literature review has been conducted to explore the linear frequency-domain decoupling approaches, noticing that the theoretical and experimental applications of these strategies were not a common practice in robotics literature, which is mainly focused on nonlinear techniques. For this reason, the study elaborates and analyses the theoretical assumptions to guarantee the application of diagonalizing algorithms to the robotic case, and deepens decoupling linear strategies in practice experimenting them on a simulated Comau robot model. Furthermore, the idea of the diagonalization has been theoretically generalized to the nonlinear case,

aiming to obtain a decoupling nonlinear law which computes the needed torque to compensate the overall couplings, originated from the cross accelerations, mainly from the inertia tensor, but also from velocity effects as centrifugal-Coriolis forces.

## 1.2 Literature Review

The study of advanced control systems, particularly for robotic manipulators with rigid and elastic joints, has been widely explored in the literature of the last thirty years. Foundational books such as Slotine and Li (1991) [1] laid a solid groundwork for control of Nonlinear systems, while Skogestad and Postlethwaite (2006) [2] for advanced multivariable control methods applied to Linear models. On this general background, the modeling and control of rigid (Siciliano & Sciavicco, 2010 [3]), flexible-joint manipulators (Spong et al., 2005 [4]; De Luca, 2011 [5], 2012 [6]; Moberg, 2010 [7]), and flexible-link manipulators (De Luca, 2016 [8], 2023 [9]) have been studied to understand the specific architecture used by Comau in order to realize a simulative environment.

The robotics literature focuses on two families of control strategies in the joint space: the decentralized, like the independent-joint cascade control based on linear regulator synthesis, and the centralized control approaches, e.g. the Computed Torque Control, founded on more general and computationally heavy nonlinear laws. Nevertheless, at least as far as the literature on linear approaches is concerned, there is also a middle ground between these two extremes, which is the decoupled control, that is, the implementation in a decentralized context of a pre-compensator to be inserted before the plant that allows to benefit from the advantages of centralized control, that guarantees the control of mutual input-output channels, without however explicitly designing ad hoc regulators for each pair, saving computational resources, and thus allowing a low-cost implementation in industry.

The coupling problem, which is one of the most crucial problems in Multi-Input Multi-Output systems, was first mentioned by Boksenbom and Hood in 1950 [10], as reported by Liu et al. [11]. In fact, at that time it was just treated as a complicated design idea though and has not been widely explored, thus for this reason new strategies aiming to eliminate or decrease multi-loop interactions have been developed in the past decades. The basic intuition of the decoupling algorithm proposed by Boksenbom and Hood (1950) is making the overall closed-loop transfer function of the controlled MIMO system diagonal. Based on the knowledge of the LTI process to control in the frequency-domain, different types of decoupling have been proposed by Luyben (1970) [12], in the pillar work of chemical process decoupling, generalized by Liu [13] and further developed by Shinskey [14], Wade [15] and Gagnon [16]: Static Decouplers (Luyben, 1970 [12]; Lee, 2005 [17]), which provide only a steady-state compensation at a certain frequency but being efficient;



Ideal Decouplers (Luyben, 1970 [12]; Liu [13]), that diagonalizes dynamically the system but being complex to implement; Simplified Decouplers (Luyben, 1970 [12]; Waller, 2003 [18]), which offers a diagonalization at the cost of a change on the direct transfer functions; Inverted Decoupler (Shinskey, 1990 [14]; Wade, 1997 [15]; Gagnon, 1998 [16]) which implements a feedforward compensation by exploiting a simplified structure to ideally diagonalize the system, further extended to include a feedback also (Garrido, 2010 [19]); SVD Decouplers (Hung and MacFarlane, 1982 [20]; Skogestad et al., 2006 [2]; Dehaeze, 2021 [21]) which diagonalize statically or dynamically the transfer function matrix through the Singular Value Decomposition. These methods have been proposed for Linear Time-Invariant system (LTI), and with the development in the last few years on the field of Linear Parameter-Varying systems (LPV) the static decoupling approaches, like the inversion-based and SVD-based ones, have been extended to the LPV case, as proposed by Mohammadpour et al. (2009) [22] and refined by Baár et al. (2020) [23]. The linear decoupling can be achieved also in the time-domain, as explained by Wang (2002) [24] and Hippe (2006) [25], where a decoupling linear feedback law is realized in the state-space.

However, the decoupling control literature mainly addresses linear systems, so that the nonlinear decoupling is not explicitly treated due to the variety and complexity of a generic non-linear system, apart some works which apply the linear techniques like Lupu (2008) [26], indeed. In reality, for non-linear systems in an affine form with respect to the input, there is a fundamental control technique called input-output feedback linearization [1], which is often included in centralized approaches, that can be actually interpreted as a decoupling strategy.

In robotics literature there are two ways to dynamically diagonalize the robot dynamics: via control or mechanical design. The coupling issue is typically addressed with the design of a regulator by employing the latter nonlinear control algorithm, proposed in the state-space by Spong (1987) [27] and in mechanical form by De Luca (1996) [28]. Another possible control approach is the Modal Control which consists of a space transformation in order to diagonalize the inertia tensor, since couplings are mainly originated from the cross-acceleration proportional to the off-diagonal terms of this matrix, as done by Jain & Rodriguez (1995) [29] and more recently by Song et al. (2019) [30]. Despite the advantages of designing SISO controllers in a modal decoupled space, enhancing control performance, the Modal Diagonalizing Control does not realize a cross compensation to explicitly inhibit the mutual couplings in the physical space, so it is not considered in this thesis. Finally, in recent books as Arakelian (2018) [31] different mechatronic approaches to decouple the robot equations have been proposed, based on the physical diagonalization of the inertia tensor through mechanical design, via mass redistribution, via actuator relocation, and via addition of auxiliary links, however since they involve the modification of the robot structure are not considered.

## 1.3 Thesis Structure

The study explores two different families of approaches in order to decouple the robot dynamics, which the author named Time-Domain Decoupling, corresponding to the Nonlinear approach, and Frequency-Domain Decoupling, i.e. the Linear approach.

**Chapter 2** introduces a comprehensive analysis of robot dynamics, starting from rigid-body models to more complex representations incorporating elasticity and damping effects in the joints, in order to obtain the mathematical model used by Comau, with the purpose to implement the experimental simulator and to provide, moreover, the foundation for understanding and designing decoupling strategies, specifically to develop the nonlinear approaches which exploit the system nature. In particular, the following models are derived: rigid manipulator model, where links are assumed to be perfectly stiff; elastic-joint model, accounting for flexibility in gear transmissions; damped joint model, which includes both elastic and viscous effects.

**Chapter 3** studies decoupling in the nonlinear domain, where the idea of feedback linearization is used to achieve decoupling, which is a side effect of the exact linearization, being instead crucial to solve the coupling issue. This technique cancels out nonlinear dynamics by designing a control law that transforms the system into a linear and decoupled form. To achieve this purpose, a detailed overview of the application of the nonlinear decoupling is proposed on different models of a robotic manipulator, to show the specific properties of each decoupled plant by applying the feedback linearization. This leads to the theoretical adaptation of a partial feedback linearization on the Comau model, which decouples the most influent inertial links.

**Chapter 4** explores decoupling in the linear domain, where it is possible to exploit the mutual dynamics among links, expressed by the off-diagonal terms of the transfer function matrix, given by the linear MIMO representation of the robotic system. Assuming the transfer matrix is known for each joint configuration, it is possible designing linear compensators to minimize cross-coupling effects. However these techniques are proposed in literature as LTI decoupling methods, while the robot model is highly non-linear, representing the worst scenario. Actually, Comau still exploits the linear design to synthesize controllers, following an independent-joint decentralized architecture, based on a LPV model of the robot dynamics. Thus, the purpose of this thesis is to study the generalization of the MIMO LTI decoupling strategies to a LPV framework, and in detail on the robotic field. In particular, various decoupling control algorithms are analysed: Static Decoupling, which pre-compensates for steady-state interactions around a frequency; SVD-Based Decoupling, which uses Singular Value Decomposition to extract dominant modes

of the system, realizing a bilinear pre and post compensator; Ideal Decoupling, based on exact inversion of the system's transfer function matrix; Simplified Decoupling, which approximates ideal decoupling while reducing complexity; Inverted Decoupling, which introduces a feedforwarding structure to improve robustness. Each method is theoretically evaluated in terms of effectiveness, computational cost, and robustness to model uncertainties.

**Chapter 5** collects all the simulations conducted on a Comau robotic manipulator model to test the proposed decoupling techniques. The performance metrics used for the experiments include: tracking accuracy (comparison between reference and actual trajectories), decoupling effectiveness (measured through interaction terms in the system response) and computational complexity (how difficult it is to implement each strategy in real-time control). Results show that while ideal decoupling provides the best theoretical performance, it is highly sensitive to modelling errors. In contrast, SVD-based and inverted decoupling methods offer a good balance between performance and robustness. Lastly, this study proves that decoupling strategies significantly enhance the performance of robotic manipulator control, particularly in tracking accuracy and ease of controller design, taking into account trade-offs between accuracy, implementation complexity, and robustness of the proposed strategies.

**Chapter 6** ultimately presents the conclusions and outlines future work, proposing to improve the models, investigate the robustness of linear decoupling approaches, apply feedback linearization in simulations, and, finally, study decoupling techniques on a real robotic manipulator.

# Chapter 2

## Robot Modelling

A robotic manipulator system may be modeled by integrating mechanical dynamics, which describes the motion of the structure under joint torques, with the electrical dynamics of the actuators driving the links, aiming for a complete state-space representation.

### 2.1 Robot Manipulator Dynamics

The dynamics of a robot manipulator in joint space, describing the relationship among joint actuator torques and the robot motion, can be obtained from sundry dynamic models: **rigid dynamic models**, assuming perfect stiffness; **flexible-joint models**, approximating elasticity through gearbox flexibility or discretized link deformation with a subset of lumped parameters; **flexible-link models**, explicitly capturing link deformations using methods like Assumed Modes Models, derived from PDEs modal truncation for efficient simulations and control, or Finite Element Models for precise mechanical analysis.

#### 2.1.1 Rigid Robot Dynamics

Given an open chain manipulator with  $n$  joints driven by electric actuators and  $n + 1$  links, using the Lagrange formulation of classical mechanics, it is possible to systematically obtain the dynamic model of the robot, particularly when considering rigid links and no backlash in the gearboxes [3].

By assuming as coordinates the joint variables

$$q_i \quad \forall i = 1, 2, \dots, n \quad \implies \quad \mathbf{q} = [q_1 \quad q_2 \quad \dots \quad q_n]^\top \quad (2.1)$$

since they describe the positions of the links for an  $n$ -DOF serial open-chain manipulator, the Lagrangian of the mechanical system being defined as a function

of the generalized coordinates, is the difference of the total kinetic energy  $\mathcal{K}$  and potential energy  $\mathcal{U}$ .

$$\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) = \mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) - \mathcal{U}(\mathbf{q}) \quad (2.2)$$

According to the stationary-action principle which claims that an infinitesimal perturbation of the action  $\mathcal{S}$  is null

$$\mathcal{S}[\mathbf{q}(t)] = \int_{t_1}^{t_2} \mathcal{L}(\mathbf{q}(t), \dot{\mathbf{q}}(t), t) dt \quad \delta \mathcal{S} = 0 \quad (2.3)$$

the Euler-Lagrange equations are derived from the Lagrangian as

$$\frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{q}_i} \right) - \left( \frac{\partial \mathcal{L}}{\partial q_i} \right) = \tau_i \quad \forall i = 1, \dots, n \implies \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top = \boldsymbol{\tau} \quad (2.4)$$

Thus the derivation of the relations between the applied generalized forces  $\boldsymbol{\tau} \in \mathbb{R}^n$  and the joint variables  $\mathbf{q} \in \mathbb{R}^n$  for an open kinematic chain-based robot is possible computing the expressions of kinetic energy and potential energy.

The total kinetic energy is given by the sum of the kinetic energy of the body  $l$  and the motor  $m$  over all links  $i$ , where the roto-translation of each link is described by its translational and rotational energy, given the linear velocity  $\dot{x}_{l_i}$  and angular velocity  $\omega_{l_i}$ , with mass  $m_{l_i}$  and the inertia tensor of the link with respect to the base frame  $\mathcal{R}_0$ , denoted by  $\mathcal{I}_{l_i} = \mathbf{R}_i^0 \mathcal{I}_{l_i}^{\mathcal{R}_i} (\mathbf{R}_i^0)^\top$ , and the roto-translation of each motor, defined by the linear velocity  $\dot{x}_{m_i}$  and the angular velocity  $\omega_{m_i}$ , with mass  $m_{m_i} = m_{s_i} + m_{r_i}$  and rotor inertia tensor  $\mathcal{I}_{m_i} = \mathbf{R}_i^0 \mathcal{I}_{m_i}^{\mathcal{R}_i} (\mathbf{R}_i^0)^\top$ .

$$\begin{aligned} \mathcal{K} = \sum_{i=1}^n (\mathcal{K}_{l_i} + \mathcal{K}_{m_i}) &= \sum_{i=1}^n \left[ \left( \frac{1}{2} m_{l_i} \dot{\mathbf{x}}_{l_i}^\top \dot{\mathbf{x}}_{l_i} + \frac{1}{2} \boldsymbol{\omega}_{l_i}^\top \mathcal{I}_{l_i} \boldsymbol{\omega}_{l_i} \right) + \left( \frac{1}{2} m_{m_i} \dot{\mathbf{x}}_{m_i}^\top \dot{\mathbf{x}}_{m_i} \right. \right. \\ &+ \left. \left. \frac{1}{2} \boldsymbol{\omega}_{m_i}^\top \mathcal{I}_{m_i} \boldsymbol{\omega}_{m_i} \right) \right] = \sum_{i=1}^n \left[ \left( \frac{1}{2} m_{l_i} \dot{\mathbf{q}}^\top \mathbf{J}_L^{(l_i)\top} \mathbf{J}_L^{(l_i)} \dot{\mathbf{q}} + \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{J}_L^{(l_i)\top} \mathcal{I}_{l_i} \mathbf{J}_A^{(l_i)} \dot{\mathbf{q}} \right) + \right. \\ &+ \left. \left( \frac{1}{2} m_{m_i} \dot{\mathbf{q}}^\top \mathbf{J}_L^{(m_i)\top} \mathbf{J}_L^{(m_i)} \dot{\mathbf{q}} + \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{J}_A^{(m_i)\top} \mathcal{I}_{m_i} \mathbf{J}_A^{(m_i)} \dot{\mathbf{q}} \right) \right] \end{aligned} \quad (2.5)$$

where the geometric Jacobian matrix for the  $i$ -th link is given by

$$\mathbf{J}_L^{(l_i)} = [\mathbf{J}_{L_1}^{(l_i)} \quad \dots \quad \mathbf{J}_{L_i}^{(l_i)} \quad \mathbf{0} \quad \dots \quad \mathbf{0}] \quad \mathbf{J}_A^{(l_i)} = [\mathbf{J}_{A_1}^{(l_i)} \quad \dots \quad \mathbf{J}_{A_i}^{(l_i)} \quad \mathbf{0} \quad \dots \quad \mathbf{0}] \quad (2.6)$$

$$\mathbf{J}_{L_j}^{(l_i)} = \begin{cases} \mathbf{k}_{j-1} & j \text{ prismatic} \\ \mathbf{k}_{j-1} \times \mathbf{r}_{j-1, c_i} & j \text{ revolute} \end{cases} \quad \mathbf{J}_{A_j}^{(l_i)} = \begin{cases} \mathbf{0} & j \text{ prismatic} \\ \mathbf{k}_{j-1} & j \text{ revolute} \end{cases} \quad (2.7)$$

while the columns of Jacobian matrix for the  $i$ -th rotor are calculated as follows, denotating the reduction ratio of the  $i$ -th gearmotor as  $k_{r_i}$

$$\mathbf{J}_L^{(m_i)} = [\mathbf{J}_{L_1}^{(m_i)} \quad \dots \quad \mathbf{J}_{L_{i-1}}^{(m_i)} \quad \mathbf{0} \quad \dots \quad \mathbf{0}] \quad \mathbf{J}_A^{(m_i)} = [\mathbf{J}_{A_1}^{(m_i)} \quad \dots \quad \mathbf{J}_{A_i}^{(m_i)} \quad \mathbf{0} \quad \dots \quad \mathbf{0}] \quad (2.8)$$

$$\begin{aligned} \mathbf{J}_{L_j}^{(m_i)} &= \begin{cases} \mathbf{k}_{j-1} & j \text{ prismatic} \\ \mathbf{k}_{j-1} \times \mathbf{r}_{j-1, m_i} & j \text{ revolute} \end{cases} \\ \mathbf{J}_{A_j}^{(m_i)} &= \begin{cases} \mathbf{0} & j \text{ prismatic} \\ \mathbf{k}_{j-1} & j \text{ revolute} \end{cases} & j = 1, 2, \dots, i-1 \\ & k_{r_i} \mathbf{k}_{m_i} & j = i \end{aligned} \quad (2.9)$$

It is possible to define the generalized inertia matrix  $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$  such that the total kinetic energy can be expressed in a quadratic form with respect to the derivative of the joint variable vector  $\mathbf{q}$ .

$$\mathbf{M}(\mathbf{q}) \triangleq \sum_{i=1}^n \left( m_{l_i} \mathbf{J}_L^{(l_i)\top} \mathbf{J}_L^{(l_i)} + \mathbf{J}_L^{(l_i)\top} \mathcal{I}_{l_i} \mathbf{J}_A^{(l_i)} + m_{m_i} \mathbf{J}_L^{(m_i)\top} \mathbf{J}_L^{(m_i)} + \mathbf{J}_A^{(m_i)\top} \mathcal{I}_{m_i} \mathbf{J}_A^{(m_i)} \right) \quad (2.10)$$

The inertia matrix  $\mathbf{M}(\mathbf{q})$  has some interesting properties since it is

- symmetric:  $m_{ij} = m_{ji} \quad \forall i, j = 1, \dots, n$
- positive definite:  $\mathbf{x}^\top \mathbf{M}(\mathbf{q}) \mathbf{x} > 0 \quad \forall \mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$
- configuration-dependent:  $\mathbf{M}(\mathbf{q}) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$

The total kinetic energy can be expressed as a function of both the robot pose through inertia matrix  $\mathbf{M}(\mathbf{q})$  and joint velocity  $\dot{\mathbf{q}}$ . In fact, after summing the linear and angular kinetic contributes for both links and rotors, it is possible to factor out joint velocities  $\dot{\mathbf{q}}$ , working out the following quadratic form on matrix  $\mathbf{M}(\mathbf{q})$ .

$$\begin{aligned} \mathcal{K} &= \frac{1}{2} \dot{\mathbf{q}}^\top \sum_{i=1}^n \left[ \left( m_{l_i} \mathbf{J}_L^{(l_i)\top} \mathbf{J}_L^{(l_i)} + \mathbf{J}_A^{(l_i)\top} \mathcal{I}_{l_i} \mathbf{J}_A^{(l_i)} \right) + \left( m_{m_i} \mathbf{J}_L^{(m_i)\top} \mathbf{J}_L^{(m_i)} + \right. \right. \\ &\quad \left. \left. + \mathbf{J}_A^{(m_i)\top} \mathcal{I}_{m_i} \mathbf{J}_A^{(m_i)} \right) \right] \dot{\mathbf{q}} = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n m_{ij}(\mathbf{q}) \dot{q}_i \dot{q}_j = \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} \end{aligned} \quad (2.11)$$

The total potential energy stored in the manipulator is calculated by summing the gravity force contributions of each rigid body and the relative motor, mounted on the preceding link, over all  $n$  links, provided the gravity acceleration vector in the base frame defined as  $\mathbf{g}_0 = \begin{bmatrix} 0 & 0 & -g \end{bmatrix}^\top$ , the position of the centroid vector for a link  $x_{l_i}$  with mass  $m_{l_i}$  and the center of mass for a motor  $x_{m_i}$  weighting  $m_{m_i}$ .

$$\begin{aligned} \mathcal{U} &= \sum_{i=1}^n (\mathcal{U}_{l_i} + \mathcal{U}_{m_i}) = \sum_{i=1}^n \left( \int_{V_{l_i}} \mathbf{g}_0^\top \mathbf{x}_{l_i} \rho_l dV + \int_{V_{m_i}} \mathbf{g}_0^\top \mathbf{x}_{m_i} \rho_m dV \right) = \\ &= \sum_{i=1}^n \left( -m_{l_i} \mathbf{g}_0^\top \mathbf{x}_{l_i} - m_{m_i} \mathbf{g}_0^\top \mathbf{x}_{m_i} \right) = - \sum_{i=1}^n \left( m_{l_i} \mathbf{g}_0^\top \mathbf{x}_{l_i} + m_{m_i} \mathbf{g}_0^\top \mathbf{x}_{m_i} \right) \end{aligned} \quad (2.12)$$

Therefore the potential energy is a function with respect to joint variable  $\mathbf{q}$  only, being independent from joint velocities  $\dot{\mathbf{q}}$ , due solely to the effect of gravitational

force since the assumption of rigidity implies that links cannot store elastic energy. Once found the analytical expression of total kinetic and potential energy, it is possible to compute the Lagrangian as the difference

$$\mathcal{L} = \mathcal{K} - \mathcal{U} = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n m_{ij}(\mathbf{q}) \dot{q}_i \dot{q}_j - \sum_{i=1}^n \left( -m_{l_i} \mathbf{g}_0^\top \mathbf{x}_{l_i} - m_{m_i} \mathbf{g}_0^\top \mathbf{x}_{m_i} \right) \quad (2.13)$$

and solving Lagrange equations, the relation between joint position, velocity, acceleration and torques is derived by developing the two differential contributes applied for each joint.

$$\begin{aligned} \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{q}_i} \right) &= \frac{d}{dt} \left( \frac{\partial \mathcal{K}}{\partial \dot{q}_i} \right) = \frac{d}{dt} \frac{\partial}{\partial \dot{q}_i} \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n m_{ij}(\mathbf{q}) \dot{q}_i \dot{q}_j = \frac{d}{dt} \sum_{j=1}^n m_{ij}(\mathbf{q}) \dot{q}_j \\ &= \sum_{j=1}^n \left( m_{ij}(\mathbf{q}) \ddot{q}_j + \frac{dm_{ij}(\mathbf{q})}{dt} \dot{q}_j \right) = \sum_{j=1}^n m_{ij}(\mathbf{q}) \ddot{q}_j + \sum_{j=1}^n \sum_{k=1}^n \frac{\partial m_{ij}(\mathbf{q})}{\partial q_k} \dot{q}_k \dot{q}_j \end{aligned} \quad (2.14)$$

$$\begin{aligned} \left( \frac{\partial \mathcal{L}}{\partial q_i} \right) &= \left( \frac{\partial \mathcal{K}}{\partial q_i} \right) - \left( \frac{\partial \mathcal{U}}{\partial q_i} \right) = \frac{1}{2} \sum_{j=1}^n \sum_{k=1}^n \frac{\partial m_{jk}(\mathbf{q})}{\partial q_i} \dot{q}_k \dot{q}_j - \sum_{i=1}^n \left( -m_{l_i} \mathbf{g}_0^\top \frac{\partial \mathbf{x}_{l_i}}{\partial q_i} \right. \\ &\quad \left. - m_{m_i} \mathbf{g}_0^\top \frac{\partial \mathbf{x}_{m_i}}{\partial q_i} \right) = \frac{1}{2} \sum_{j=1}^n \sum_{k=1}^n \frac{\partial m_{jk}(\mathbf{q})}{\partial q_i} \dot{q}_k \dot{q}_j - \sum_{i=1}^n \left( -m_{l_i} \mathbf{g}_0^\top \mathbf{J}_{L_i}^{(l_j)} - \right. \\ &\quad \left. - m_{m_i} \mathbf{g}_0^\top \mathbf{J}_{L_i}^{(m_j)} \right) = \frac{1}{2} \sum_{j=1}^n \sum_{k=1}^n \frac{\partial m_{jk}(\mathbf{q})}{\partial q_i} \dot{q}_k \dot{q}_j - \mathbf{g}_i(\mathbf{q}) \end{aligned} \quad (2.15)$$

$$\frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{q}_i} \right) - \left( \frac{\partial \mathcal{L}}{\partial q_i} \right) = \sum_{j=1}^n m_{ij}(\mathbf{q}) \ddot{q}_j + \sum_{j=1}^n \sum_{k=1}^n \left( \frac{\partial m_{ij}(\mathbf{q})}{\partial q_k} - \frac{1}{2} \frac{\partial m_{jk}(\mathbf{q})}{\partial q_i} \right) \dot{q}_k \dot{q}_j + \mathbf{g}_i(\mathbf{q}) \quad (2.16)$$

where  $\mathbf{g}_i(\mathbf{q})$  is the gradient of the gravitational energy  $g_i(\mathbf{q}) \triangleq \left( \frac{\partial \mathcal{U}}{\partial \mathbf{q}} \right)^\top$ .

Therefore, equating this expression with the generalized force on the joint, the Lagrangian equation for the  $i$ -th link may be expressed as:

$$\sum_{j=1}^n m_{ij}(\mathbf{q}) \ddot{q}_j + \sum_{j=1}^n \sum_{k=1}^n \left( \frac{\partial m_{ij}(\mathbf{q})}{\partial q_k} - \frac{1}{2} \frac{\partial m_{jk}(\mathbf{q})}{\partial q_i} \right) \dot{q}_k \dot{q}_j + g_i(\mathbf{q}) = \tau_i \quad (2.17)$$

Defining the coefficient  $h_{ijk} \triangleq \frac{\partial m_{ij}(q)}{\partial q_k} - \frac{1}{2} \frac{\partial m_{jk}(q)}{\partial q_i}$  equations become

$$\sum_{j=1}^n m_{ij}(\mathbf{q}) \ddot{q}_j + \sum_{j=1}^n \sum_{k=1}^n h_{ijk}(\mathbf{q}) \dot{q}_k \dot{q}_j + g_i(\mathbf{q}) = \tau_i \quad i = 1, \dots, n \quad (2.18)$$

This equation of motion has an explicit physical interpretation since each term has a specific meaning:

- Acceleration coefficients ( $\ddot{\mathbf{q}}_j$ ):
  - $m_{ii}$ : equivalent moment of inertia for a link  $i$  when other joints are blocked
  - $m_{ij}$ : acceleration effect of joint  $j \neq i$  on link  $i$
- Quadratic velocity coefficients ( $\dot{\mathbf{q}}_k \dot{\mathbf{q}}_j$ ):
  - $h_{ijj} \dot{q}_j^2$ : centrifugal effect on link  $i$  due to velocity of joint  $j$
  - $h_{ijk} \dot{q}_j \dot{q}_k$ : Coriolis effect on link  $i$  from velocity of joints  $j, k$
- Configuration-dependent coefficients ( $\mathbf{q}$ ):
  - $g_i$ : moment on joint  $i$  z-axis due to the gravity force effect

The system of differential equations may be written in a more compact matrix form which represents the **rigid robot dynamic model**

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} \quad (2.19)$$

Eventually, it is possible to express the linear dependence of the velocity term  $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}})$  explicitly with respect to the joint velocity vector  $\dot{\mathbf{q}}$  by defining an appropriate matrix  $\mathbf{C} \in \mathbb{R}^{n \times n}$ . A particular solution can be obtained from  $h_{ijk}$  exploiting the symmetry of  $\mathbf{M}(\mathbf{q}) = [m_{ij}]_{n \times n}$ .

$$\begin{aligned} \sum_{j=1}^n c_{ij} \dot{q}_j &\stackrel{!}{=} \sum_{j=1}^n \sum_{k=1}^n h_{ijk} \dot{q}_k \dot{q}_j = \sum_{j=1}^n \sum_{k=1}^n \left( \frac{\partial m_{ij}(\mathbf{q})}{\partial q_k} - \frac{1}{2} \frac{\partial m_{jk}(\mathbf{q})}{\partial q_i} \right) \dot{q}_k \dot{q}_j = \\ &= \sum_{j=1}^n \sum_{k=1}^n \left[ \frac{1}{2} \left( \frac{\partial m_{ij}(\mathbf{q})}{\partial q_k} + \frac{\partial m_{ik}(\mathbf{q})}{\partial q_j} \right) - \frac{1}{2} \frac{\partial m_{jk}(\mathbf{q})}{\partial q_i} \right] \dot{q}_k \dot{q}_j = \end{aligned} \quad (2.20)$$

$$= \sum_{j=1}^n \sum_{k=1}^n \left[ \frac{1}{2} \left( \frac{\partial m_{ij}(\mathbf{q})}{\partial q_k} + \frac{\partial m_{ik}(\mathbf{q})}{\partial q_j} - \frac{\partial m_{jk}(\mathbf{q})}{\partial q_i} \right) \right] \dot{q}_k \dot{q}_j = \sum_{j=1}^n \sum_{k=1}^n c_{ijk} \dot{q}_k \dot{q}_j \quad (2.21)$$

so each term of the centrifugal-Coriolis matrix  $c_{ij} = [\mathbf{C}]_{ij}$  is computed exploiting Christoffel symbols of the first kind  $c_{ijk}$ , defined as follows

$$c_{ijk} \triangleq \frac{1}{2} \left( \frac{\partial m_{ij}(\mathbf{q})}{\partial q_k} + \frac{\partial m_{ik}(\mathbf{q})}{\partial q_j} - \frac{\partial m_{jk}(\mathbf{q})}{\partial q_i} \right) \implies c_{ij} = \sum_{k=1}^n c_{ijk} \dot{q}_k \quad (2.22)$$

The **rigid dynamic** equations can be written in Cauchy matrix form

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} \quad (2.23)$$

Moreover the generalized force vector  $\boldsymbol{\tau}$  is the sum of all torques so it can be expanded to explicitly represent all force effects acting dynamically and in static equilibrium, by applying the principle of virtual work, for each generalized coordinate  $q_i$   $i = 1, \dots, n$ :



- active input torques  $\boldsymbol{\tau}_l = [\tau_{c_1} \quad \tau_{c_2} \quad \dots \quad \tau_{c_n}]^\top$
- friction torques (Coulomb)  $\boldsymbol{\tau}_f = [\tau_{f_1} \quad \dots \quad \tau_{f_n}]^\top = \mathbf{D}_v \dot{\mathbf{q}} + \mathbf{D}_s \operatorname{sgn}(\dot{\mathbf{q}})$
- contact torques  $\boldsymbol{\tau}_e = [\tau_{e_1} \quad \tau_{e_2} \quad \dots \quad \tau_{e_n}]^\top = \mathbf{J}_g^\top \mathbf{F}_e$

where  $\mathbf{D}_v \in \mathbb{R}^{n \times n}$  and  $\mathbf{D}_s \in \mathbb{R}^{n \times n}$  are the diagonal damping matrices for viscous and static friction, while the geometric Jacobian  $\mathbf{J}_g \in \mathbb{R}^{n \times 6}$  applied to generalized external force on the end-effector  $\mathbf{F}_e \in \mathbb{R}^6$  represents joint torques in static equilibrium. So, adding all these force contributes, the total generalized force vector is calculated as

$$\boldsymbol{\tau} = \boldsymbol{\tau}_c - \boldsymbol{\tau}_f - \boldsymbol{\tau}_e = \boldsymbol{\tau}_c - \mathbf{D}_v \dot{\mathbf{q}} - \mathbf{D}_s \operatorname{sgn}(\dot{\mathbf{q}}) - \mathbf{J}_g^\top \mathbf{F}_e \quad (2.24)$$

and the **complete joint space rigid dynamic model** is obtained.

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}_v \dot{\mathbf{q}} + \mathbf{D}_s \operatorname{sgn}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}_c - \mathbf{J}_g^\top(\mathbf{q})\mathbf{F}_e \quad (2.25)$$

## Flexible Joint Robot Dynamics

Traditional dynamic models of robotic manipulators assume rigid joints, which is often a good approximation. However, when joints have elastic gearboxes, this assumption becomes inadequate for precise control, especially in high-speed applications or when handling heavy loads, where neglecting softness may generate vibrations and trajectory oscillations. In such cases, the rigid model needs to be extended to account for elastic effects, where joint flexibility can arise in general from elasticity in materials of gear transmission not completely rigid, or effects such as shaft windup in cable-based transmission [8]. The robot manipulator can be modeled as an open kinematic chain of rigid bodies connected by torsional spring-damper pairs [9], assuming small deflection on axis and axis-balanced motors mounted on the preceding links, such that inertial coupling between motors and rigid links are neglected, i.e. the angular kinetic energy of each motor is due only to its spinning, which is a reasonable approximation if gear ratios are high [27]. The model considers two main sets of coordinates:

- the flexible joint variables of the manipulator  $\mathbf{q} \in \mathbb{R}^n$ , representing the real position of manipulator joints, actuated by the gearbox output torque  $\boldsymbol{\tau}_q \in \mathbb{R}^n$
- the motor variables  $\boldsymbol{\theta} \in \mathbb{R}^n$ , denoting the rotor angular position, downstream gearbox transmission as reflected through gear ratio  $k_{r_i}$  w.r.t. motor shaft position  $\theta_{m_i} = k_{r_i} \theta_i$ , which may differ from  $\mathbf{q}$  due to flexibility, and set in motion by the rotor torque  $\boldsymbol{\tau}_\theta \in \mathbb{R}^n$

### 2.1.2 Elastic Joint Robot Dynamics

Considering the robot as a chain of rigid bodies connected by linear or torsional springs depending if a joint is prismatic or revolute, it is possible to derive the elastic model [6]. Let us re-define the lumped inertia matrix summing the linear  $\mathbf{M}_{l,L}$  and angular inertia  $\mathbf{M}_{l,A}$  of links plus the motor inertia  $\mathbf{M}_{m,L}$  only related to the linear velocity of motors with mass given by stators and rotors  $m_{m_i} = m_{s_i} + m_{r_i}$ .

$$\mathbf{M}(\mathbf{q}) \triangleq \mathbf{M}_l(\mathbf{q}) + \mathbf{M}_{m,L}(\mathbf{q}) = \sum_{i=1}^n \left[ \left( m_{l_i} \mathbf{J}_L^{(l_i)\top} \mathbf{J}_L^{(l_i)} + \mathbf{J}_L^{(l_i)\top} \mathbf{Z}_{l_i} \mathbf{J}_A^{(l_i)} \right) + m_{m_i} \mathbf{J}_L^{(m_i)\top} \mathbf{J}_L^{(m_i)} \right] \quad (2.26)$$

The kinetic energy of links is given by linear and angular contributes

$$\begin{aligned} \mathcal{K}_l &= \sum_{i=1}^n \mathcal{K}_{l_i} = \sum_{i=1}^n (\mathcal{K}_{l_i,L} + \mathcal{K}_{l_i,A}) = \sum_{i=1}^n \left( \frac{1}{2} m_{l_i} \dot{\mathbf{q}}^\top \mathbf{J}_L^{(l_i)\top} \mathbf{J}_L^{(l_i)} \dot{\mathbf{q}} + \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{J}_A^{(l_i)\top} \mathbf{Z}_{l_i} \mathbf{J}_A^{(l_i)} \dot{\mathbf{q}} \right) = \\ &= \frac{1}{2} \dot{\mathbf{q}}^\top \left( \sum_{i=1}^n m_{l_i} \mathbf{J}_L^{(l_i)\top} \mathbf{J}_L^{(l_i)} + \sum_{i=1}^n \mathbf{J}_A^{(l_i)\top} \mathbf{Z}_{l_i} \mathbf{J}_A^{(l_i)} \right) \dot{\mathbf{q}} = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n m_{l_{ij}}(\mathbf{q}) \dot{q}_i \dot{q}_j = \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M}_l(\mathbf{q}) \dot{\mathbf{q}} \end{aligned} \quad (2.27)$$

For motors the linear kinetic energy is the same as in (2.1.1) but the angular kinetic one is supposed only due to the spinning of each rotor

$$\mathcal{K}_{m,L} = \sum_{i=1}^n \mathcal{K}_{m_i,L} = \frac{1}{2} \dot{\mathbf{q}}^\top \left( \sum_{i=1}^n m_{m_i} \mathbf{J}_L^{(m_i)\top} \mathbf{J}_L^{(m_i)} \right) \dot{\mathbf{q}} = \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M}_{m,L}(\mathbf{q}) \dot{\mathbf{q}} \quad (2.28)$$

$$\mathcal{K}_{m,A} = \sum_{i=1}^n \mathcal{K}_{m_i,A} = \sum_{i=1}^n \frac{1}{2} I_{m_{i,zz}} \dot{\theta}_i^2 = \sum_{i=1}^n \frac{1}{2} I_{m_i} (k_{r_i} \dot{\theta}_i)^2 = \frac{1}{2} \sum_{i=1}^n (I_{m_i} k_{r_i}^2) \dot{\theta}_i^2 = \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \mathbf{B} \dot{\boldsymbol{\theta}} \quad (2.29)$$

The potential energy of both links, including the mass contribute from the carried stators of actuators, due to the gravity, is given by

$$\mathcal{U}_g = \sum_{i=1}^n (\mathcal{U}_{l_i} + \mathcal{U}_{m_i}) = - \sum_{i=1}^n \left( m_{l_i} \mathbf{g}_0^\top \mathbf{x}_{l_i} + m_{m_i} \mathbf{g}_0^\top \mathbf{x}_{m_i} \right) \quad (2.30)$$

while the potential energy due to the joint elasticity is calculated as

$$\mathcal{U}_e = \sum_{i=1}^n \mathcal{U}_{e_i} = \sum_{i=1}^n \frac{1}{2} K_i (q_i - \theta_i)^2 = \frac{1}{2} (\mathbf{q} - \boldsymbol{\theta})^\top \mathbf{K} (\mathbf{q} - \boldsymbol{\theta}) \quad (2.31)$$

So it is possible to write the Lagrangian of the elastic joint robot as

$$\begin{aligned} \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) &= \mathcal{K} - \mathcal{U} = (\mathcal{K}_l + \mathcal{K}_{m,L} + \mathcal{K}_{m,A}) - (\mathcal{U}_g + \mathcal{U}_e) = \\ &= \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} + \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \mathbf{B} \dot{\boldsymbol{\theta}} - \int \mathbf{g}(\mathbf{q}) d\mathbf{q} - \frac{1}{2} (\mathbf{q} - \boldsymbol{\theta})^\top \mathbf{K} (\mathbf{q} - \boldsymbol{\theta}) \end{aligned} \quad (2.32)$$

Exploiting Euler-Lagrange equations, separately for joint variables and motor variables, where the generalized force vector acts only on motors,  $2n$  equations of motion in Cauchy form are obtained.

$$\begin{cases} \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top = \boldsymbol{\tau}_q & \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right)^\top = \mathbf{0} \\ \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right)^\top = \boldsymbol{\tau}_\theta & \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top = \boldsymbol{\tau}_c \end{cases} \quad (2.33)$$

where  $\boldsymbol{\tau}_q$  and  $\boldsymbol{\tau}_d$  are generalized forces, and in particular as the only active forces are motors torques  $\boldsymbol{\tau}_l$  follows  $\boldsymbol{\tau}_q = \mathbf{0}$  and  $\boldsymbol{\tau}_\theta = \boldsymbol{\tau}_l$ .

Thus by solving derivatives and applying algebraic manipulations

$$\begin{aligned} \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top &= \frac{d}{dt} (\mathbf{M}(\mathbf{q})\dot{\mathbf{q}}) = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{M}}(\mathbf{q})\dot{\mathbf{q}} & \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top &= \frac{d}{dt} (\mathbf{B}\dot{\boldsymbol{\theta}}) = \mathbf{B}\ddot{\boldsymbol{\theta}} \\ \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top &= \frac{1}{2}\dot{\mathbf{q}}^\top \frac{\partial \mathbf{M}(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} - \mathbf{g}(\mathbf{q}) - \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) & \left( \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right)^\top &= \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) \end{aligned} \quad (2.34)$$

and, after summing up all these contributes, as done in Section 2.1.1

$$\begin{aligned} \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top &= \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{M}}(\mathbf{q})\dot{\mathbf{q}} - \frac{1}{2}\dot{\mathbf{q}}^\top \frac{\partial \mathbf{M}(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) \\ \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right)^\top &= \mathbf{B}\ddot{\boldsymbol{\theta}} - \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) \end{aligned} \quad (2.35)$$

the dynamics of a **robot with elastic joints** is given by the rigid manipulator equation of motion and the joint flexibility dynamics as

$$\begin{bmatrix} \mathbf{M}(\mathbf{q}) & \mathbf{0} \\ \mathbf{0} & \mathbf{B} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \ddot{\boldsymbol{\theta}} \end{bmatrix} + \begin{bmatrix} \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{g}(\mathbf{q}) \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) \\ \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\tau}_c \end{bmatrix} \quad (2.36)$$

where, being  $\mathbf{p} = [\mathbf{q}^\top \ \boldsymbol{\theta}^\top]^\top$  generalized coordinates, the total inertia matrix  $\boldsymbol{\mathcal{M}}(\mathbf{p}) = \text{diag}([\mathbf{M}(\mathbf{q}) \ \mathbf{B}])$  is a block diagonal matrix in  $\mathbb{R}^{2n \times 2n}$ , composed of the lumped inertia matrix  $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$  which represents the inertia of the manipulator system as seen from the flexible joints and by  $\mathbf{B} = \text{diag}([\mathcal{I}_{m_1} k_{r_1}^2 \dots \mathcal{I}_{m_n} k_{r_n}^2])$  i.e. the motors' inertia matrix as reflected through gear ratios, for which the generalized centrifugal-Coriolis term  $\mathbf{C}(\mathbf{p}, \dot{\mathbf{p}})\dot{\mathbf{p}}$  coincides with the one related to  $\mathbf{M}(\mathbf{q})$  only i.e.  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}$ , while  $\mathbf{K} = \text{diag}([k_1 \dots k_n])$  is the stiffness matrix which represents joints' elastic constants, and  $\boldsymbol{\theta} - \mathbf{q}$  describes how the joint position differs from the actuator position due to elasticity. Additionally  $\mathbf{0}_n$  is a null vector and  $\mathbf{0}_{n \times n}$  is

a block null matrix, based on the assumption of no coupling between joints and motors. The first matrix equation describes the manipulator's dynamics as if the joints were rigid but subject to a resulting force determined by the joint flexibility. While the second matrix equation supposes a 1-DOF joint deformation in the linear domain, modeling the joint flexibility as a torsional spring with constant stiffness which generates a torque proportional to the deflection between actuator  $\boldsymbol{\theta}$  and joint  $\mathbf{q}$  positions.

### 2.1.3 Relative Elastic-Dumped Joint Robot Dynamics

The robot manipulator can be modeled further as an open chain of rigid bodies connected by lumped torsional spring-dampers pairs, where the visco-elastic coupling force is assumed proportional to the relative velocity between the joint and the motor [8].

To develop this model, the damping is included into the Lagrangian framework by using the Rayleigh dissipation function, which allows to account for viscous friction in a manner consistent with Lagrangian mechanics, so the derivation is similar to the elastic joint case but includes viscous friction linearly proportional to the difference between joint and actuator variables.

The total kinetic energy of links and motors is given by

$$\mathcal{K} = \mathcal{K}_l + \mathcal{K}_m = \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} + \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \mathbf{B} \dot{\boldsymbol{\theta}} \quad (2.37)$$

The total potential energy due to gravity and elasticity is

$$\mathcal{U} = \mathcal{U}_g + \mathcal{U}_e = \sum_{i=1}^n \left( -m_{l_i} \mathbf{g}_0^\top \mathbf{x}_{l_i} - m_{m_i} \mathbf{g}_0^\top \mathbf{x}_{m_i} \right) + \frac{1}{2} (\mathbf{q} - \boldsymbol{\theta})^\top \mathbf{K} (\mathbf{q} - \boldsymbol{\theta}) \quad (2.38)$$

To include viscous friction on the relative motion between joints and motors, it is possible to exploit the Rayleigh dissipation function defined as a quadratic form with respect to the relative velocities through a diagonal damping matrix  $\mathbf{D} = \text{diag}([d_1 \ d_2 \ \dots \ d_n])$ ,

$$\mathcal{D} = \frac{1}{2} (\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}})^\top \mathbf{D} (\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) \quad (2.39)$$

where  $d_i$  coefficients correspond to the generalized damping forces  $\boldsymbol{\tau}_d = \mathbf{D}(\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}})$ , obtained as the negative gradient of the total dissipative energy with respect to the generalized joint and rotor velocities.

$$\boldsymbol{\tau}_{d_l} = -\frac{\partial \mathcal{D}}{\partial \dot{\mathbf{q}}} = -\mathbf{D}(\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) \quad \boldsymbol{\tau}_{d_\theta} = -\frac{\partial \mathcal{D}}{\partial \dot{\boldsymbol{\theta}}} = \mathbf{D}(\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) \quad (2.40)$$

The Lagrangian  $\mathcal{L}$  is computed as usual and it works out the same as for the elastic joint model, where  $\mathbf{g}(\mathbf{q}) \triangleq \left(\frac{\partial \mathcal{U}_g}{\partial \mathbf{q}}\right)^\top$  is the gravity term.

$$\mathcal{L}(\mathbf{q}, \boldsymbol{\theta}, \dot{\mathbf{q}}, \dot{\boldsymbol{\theta}}) = \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} + \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \mathbf{B} \dot{\boldsymbol{\theta}} - \int \mathbf{g}(\mathbf{q}) d\mathbf{q} - \frac{1}{2} (\mathbf{q} - \boldsymbol{\theta})^\top \mathbf{K} (\mathbf{q} - \boldsymbol{\theta}) \quad (2.41)$$

As friction is not a conservative force, it cannot be derived from a potential energy function and cannot be included directly in the Lagrangian function, hence, it is incorporated in the right hand side of the Euler-Lagrange equations as an active contribute. Thus, in the presence of damping represented by  $\mathcal{D}$ , Euler-Lagrange equations are

$$\begin{cases} \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top = \boldsymbol{\tau}_q & \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top + \left( \frac{\partial \mathcal{D}}{\partial \dot{\mathbf{q}}} \right)^\top = \mathbf{0} \\ \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right)^\top = \boldsymbol{\tau}_\theta & \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right)^\top + \left( \frac{\partial \mathcal{D}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top = \boldsymbol{\tau}_c \end{cases} \quad (2.42)$$

in which  $\boldsymbol{\tau}_q$  and  $\boldsymbol{\tau}_d$  are generalized non-conservative forces. In our case motor torques act only on motor angular positions  $\boldsymbol{\theta}$  thus  $\boldsymbol{\tau}_\theta = \boldsymbol{\tau}_l + \boldsymbol{\tau}_{D_m}$ , and since there is no external force on  $\mathbf{q}$  apart from damping accounted by  $\mathcal{D}$  and elasticity already included in  $\mathcal{L}$  then  $\boldsymbol{\tau}_q = \mathbf{0} + \boldsymbol{\tau}_{D_l}$ .

Now, computing derivatives for each contribute as following

$$\begin{aligned} \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top &= \frac{d}{dt} (\mathbf{M}(\mathbf{q}) \dot{\mathbf{q}}) = \mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \dot{\mathbf{M}}(\mathbf{q}) \dot{\mathbf{q}} & \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top &= \frac{d}{dt} (\mathbf{B} \dot{\boldsymbol{\theta}}) = \mathbf{B} \ddot{\boldsymbol{\theta}} \\ \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top &= \frac{1}{2} \dot{\mathbf{q}}^\top \frac{\partial \mathbf{M}(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} - \mathbf{g}(\mathbf{q}) - \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) & \left( \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right)^\top &= \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) \\ \left( \frac{\partial \mathcal{D}}{\partial \dot{\mathbf{q}}} \right)^\top &= \mathbf{D}(\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) & \left( \frac{\partial \mathcal{D}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top &= -\mathbf{D}(\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) \end{aligned} \quad (2.43)$$

and substituting all the derivatives into the Euler-Lagrange equations, the **relative elastic-dumped joint robot model** is obtained.

$$\begin{bmatrix} \mathbf{M}(\mathbf{q}) & \mathbf{0} \\ \mathbf{0} & \mathbf{B} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \ddot{\boldsymbol{\theta}} \end{bmatrix} + \begin{bmatrix} \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{D}(\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) \\ \mathbf{D}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) \end{bmatrix} + \begin{bmatrix} \mathbf{g}(\mathbf{q}) \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) \\ \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\tau}_c \end{bmatrix} \quad (2.44)$$

The first matrix equation represents the dynamics of the links, including inertia, Coriolis and centrifugal effects, gravity, elastic forces from torsional springs, and damping forces due to viscous friction while the second equation describes the dynamics of the motors, accounting for their inertia, elastic forces, damping opposing the relative motion, and applied control torques. Notice that the initial assumptions imply there is no direct inertial coupling among motors and links, however rotor variables indirectly affect the joints through the transmission.

### 2.1.4 Direct Elastic-Dumped Joint Robot Dynamics

The manipulator can alternatively be represented as a kinematic chain of rigid bodies affected by viscous friction, connected by motors with torsional springs with their own dampers, so that viscous friction is applied independently to both the links and the motors, acting directly on the joint velocity and the motor angular velocity [8]. To achieve this model, the damping must be directly included into the Lagrangian framework by exploiting again Rayleigh dissipation functions as done in Section 2.1.3, but separately for each vector variable.

The Lagrangian  $\mathcal{L}$  coincides with that of the elastic joint model.

$$\mathcal{L} = \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} + \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \mathbf{B} \dot{\boldsymbol{\theta}} - \int \mathbf{g}(\mathbf{q}) d\mathbf{q} - \frac{1}{2} (\mathbf{q} - \boldsymbol{\theta})^\top \mathbf{K} (\mathbf{q} - \boldsymbol{\theta}) \quad (2.45)$$

To account for viscous friction on both the joint and the motor side is necessary summing the dissipative energy related to joints and motors. So, given the damping matrix acting on the joints  $\mathbf{D}_l = \text{diag}([d_{l_1} d_{l_2} \dots d_{l_n}])$ , associated with the dissipative energy  $\mathcal{D}_l$ , and the matrix  $\mathbf{D}_m = \text{diag}([d_{m_1} d_{m_2} \dots d_{m_n}])$ , corresponding to the Rayleigh dissipation function for the motors  $\mathcal{D}_m$ , the total dissipative energy is

$$\mathcal{D} = \mathcal{D}_l + \mathcal{D}_m = \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{D}_l \dot{\mathbf{q}} + \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \mathbf{D}_m \dot{\boldsymbol{\theta}} \quad (2.46)$$

And the generalized dumping forces are derived from the partial derivative of the total dissipative energy w.r.t. joint and rotor speeds.

$$\boldsymbol{\tau}_{d_l} = -\frac{\partial \mathcal{D}}{\partial \dot{\mathbf{q}}} = -\mathbf{D}_l \dot{\mathbf{q}} \quad \boldsymbol{\tau}_{d_m} = -\frac{\partial \mathcal{D}}{\partial \dot{\boldsymbol{\theta}}} = -\mathbf{D}_m \dot{\boldsymbol{\theta}} \quad (2.47)$$

Thus, considering independent dumpers acting on joint and motor sides, Euler-Lagrange equations in Section 2.1.3 are formally the same, with the only difference that the friction forces are changed.

$$\begin{cases} \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top = \boldsymbol{\tau}_q & \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top + \left( \frac{\partial \mathcal{D}}{\partial \dot{\mathbf{q}}} \right)^\top = \mathbf{0} \\ \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right)^\top = \boldsymbol{\tau}_\theta & \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right)^\top + \left( \frac{\partial \mathcal{D}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top = \boldsymbol{\tau}_c \end{cases} \quad (2.48)$$

In fact taking the derivatives of the Lagrangian as follows

$$\begin{aligned} \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top &= \frac{d}{dt} (\mathbf{M}(\mathbf{q}) \dot{\mathbf{q}}) = \mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \dot{\mathbf{M}}(\mathbf{q}) \dot{\mathbf{q}} & \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top &= \frac{d}{dt} (\mathbf{B} \dot{\boldsymbol{\theta}}) = \mathbf{B} \ddot{\boldsymbol{\theta}} \\ \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top &= \frac{1}{2} \dot{\mathbf{q}}^\top \frac{\partial \mathbf{M}(\mathbf{q})}{\partial \mathbf{q}} - \mathbf{g}(\mathbf{q}) - \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) & \left( \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right)^\top &= \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) \\ \left( \frac{\partial \mathcal{D}}{\partial \dot{\mathbf{q}}} \right)^\top &= \mathbf{D}_l \dot{\mathbf{q}} & \left( \frac{\partial \mathcal{D}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top &= \mathbf{D}_m \dot{\boldsymbol{\theta}} \end{aligned} \quad (2.49)$$

the **direct elastic-dumped robot dynamic model** is obtained as

$$\begin{bmatrix} \mathbf{M}(\mathbf{q}) & \mathbf{0} \\ \mathbf{0} & \mathbf{B} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \ddot{\boldsymbol{\theta}} \end{bmatrix} + \begin{bmatrix} \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{D}_l \dot{\mathbf{q}} \\ \mathbf{D}_m \dot{\boldsymbol{\theta}} \end{bmatrix} + \begin{bmatrix} \mathbf{g}(\mathbf{q}) \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) \\ \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\tau}_c \end{bmatrix} \quad (2.50)$$

### 2.1.5 Complete Elastic-Dumped Joint Robot Dynamics

The complete flexible joint model is the lumped parameter model in which the robot manipulator is thought as composed of  $n$  rigid bodies subject to viscous friction forces, each one driven by an electric actuator placed in the previous link with a rotor that has center of mass on the rotation axis, and joined by  $n$  spring-dampers pairs modeling small deflections at joints through gear transmissions so that flexibility is limited in the linear domain, taking also into account the inertial coupling effects among motors and links.

The kinetic energy is the sum of the link contributions which is the same as the rigid case, given the link inertia matrix (2.26) of the elastic model

$$\begin{aligned} \mathcal{K}_l &= \sum_{i=1}^n \mathcal{K}_{l_i} = \sum_{i=1}^n (\mathcal{K}_{l_i,L} + \mathcal{K}_{l_i,A}) = \sum_{i=1}^n \left( \frac{1}{2} m_{l_i} \dot{\mathbf{x}}_{l_i}^\top \dot{\mathbf{x}}_{l_i} + \frac{1}{2} \boldsymbol{\omega}_{l_i}^\top \mathcal{I}_{l_i} \boldsymbol{\omega}_{l_i} \right) = \\ &= \sum_{i=1}^n \left( \frac{1}{2} m_{l_i} \dot{\mathbf{q}}^\top \mathbf{J}_L^{(l_i)\top} \mathbf{J}_L^{(l_i)} \dot{\mathbf{q}} + \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{J}_A^{(l_i)\top} \mathcal{I}_{l_i} \mathbf{J}_A^{(l_i)} \dot{\mathbf{q}} \right) = \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M}_l(\mathbf{q}) \dot{\mathbf{q}} \end{aligned} \quad (2.51)$$

and the rotor contributions, composed of two main components: the linear component, which is related to the linear velocity of the rotor's center of mass (assumed to lie on the rotation axis, ensuring that the rotor inertia matrix in the local frame  $\mathcal{I}_{m_i}^{\mathcal{R}_i} = \text{diag}([I_{m_{i,xx}} \ I_{m_{i,yy}} \ I_{m_{i,zz}}])$  is diagonal), and the angular kinetic energy which arises not only from the rotor's own spinning velocity but also from the coupling effects induced by the velocities of other joints. Based on the assumption that the actuator for the  $i$ -th link is mounted on the  $(i-1)$ -th preceding body, the angular velocity in the local frame  $\mathcal{R}_i$  is expressed as the sum of the contributions from the previous joint angular velocities and the rotation around the z-axis as

$$\boldsymbol{\omega}_{m_i}^{\mathcal{R}_i} = \sum_{j=1}^{i-1} \mathbf{J}_{A_j}^{(m_i)} \dot{\mathbf{q}}_j + [\mathbf{0} \ \mathbf{0} \ \dot{\boldsymbol{\theta}}_{m_i}]^\top \quad (2.52)$$

Under these assumptions, the following demonstrates the derivation of the kinetic energy of the motors.

$$\begin{aligned} \mathcal{K}_m &= \sum_{i=1}^n \mathcal{K}_{m_i} = \sum_{i=1}^n (\mathcal{K}_{m_i,L} + \mathcal{K}_{m_i,A}) = \sum_{i=1}^n \left( \frac{1}{2} m_{m_i} \dot{\mathbf{x}}_{m_i}^\top \dot{\mathbf{x}}_{m_i} + \frac{1}{2} \boldsymbol{\omega}_{m_i}^\top \mathcal{I}_{m_i} \boldsymbol{\omega}_{m_i} \right) = \\ &= \sum_{i=1}^n \left( \frac{1}{2} m_{m_i} \dot{\mathbf{x}}_{m_i}^\top \dot{\mathbf{x}}_{m_i} + \frac{1}{2} (\mathbf{R}_i^0 \boldsymbol{\omega}_{m_i}^{\mathcal{R}_i})^\top \mathcal{I}_{m_i} \mathbf{R}_i^0 \boldsymbol{\omega}_{m_i}^{\mathcal{R}_i} \right) = \\ &= \sum_{i=1}^n \left( \frac{1}{2} m_{m_i} \dot{\mathbf{x}}_{m_i}^\top \dot{\mathbf{x}}_{m_i} + \frac{1}{2} \boldsymbol{\omega}_{m_i}^{\mathcal{R}_i\top} (\mathbf{R}_i^0 \mathcal{I}_{m_i} \mathbf{R}_i^{0\top}) \boldsymbol{\omega}_{m_i}^{\mathcal{R}_i} \right) = \end{aligned}$$

$$\begin{aligned}
 &= \sum_{i=1}^n \left( \frac{1}{2} m_{m_i} \dot{\mathbf{x}}_{m_i}^\top \dot{\mathbf{x}}_{m_i} + \frac{1}{2} \boldsymbol{\omega}_{m_i}^{\mathcal{R}_i \top} \boldsymbol{\mathcal{I}}_{m_i}^{\mathcal{R}_i} \boldsymbol{\omega}_{m_i}^{\mathcal{R}_i} \right) = \frac{1}{2} \sum_{i=1}^n \left[ m_{m_i} \dot{\mathbf{q}}^\top \mathbf{J}_L^{(m_i) \top} \mathbf{J}_L^{(m_i)} \dot{\mathbf{q}} + \right. \\
 &\quad \left. + \left( \sum_{j=1}^{i-1} J_{A_j}^{(m_i)} \dot{q}_j + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_{m_i} \end{bmatrix} \right)^\top \boldsymbol{\mathcal{I}}_{m_i}^{\mathcal{R}_i} \left( \sum_{j=1}^{i-1} J_{A_j}^{(m_i)} \dot{q}_j + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_{m_i} \end{bmatrix} \right) \right] = \\
 &= \frac{1}{2} \sum_{i=1}^n \left[ m_{m_i} \dot{\mathbf{q}}^\top \mathbf{J}_L^{(m_i) \top} \mathbf{J}_L^{(m_i)} \dot{\mathbf{q}} + \left( \mathbf{J}_A^{(m_i)} \begin{bmatrix} \dot{\mathbf{q}}_{1:i-1} \\ \dot{\theta}_{m_i} \end{bmatrix} \right)^\top \boldsymbol{\mathcal{I}}_{m_i}^{\mathcal{R}_i} \left( \mathbf{J}_A^{(m_i)} \begin{bmatrix} \dot{\mathbf{q}}_{1:i-1} \\ \dot{\theta}_{m_i} \end{bmatrix} \right) \right] = \\
 &= \frac{1}{2} \dot{\mathbf{q}}^\top \left( \sum_{i=1}^n m_{m_i} \mathbf{J}_L^{(m_i) \top} \mathbf{J}_L^{(m_i)} \right) \dot{\mathbf{q}} + \frac{1}{2} \sum_{i=1}^n \begin{bmatrix} \dot{\mathbf{q}}_{1:i-1} \\ \dot{\theta}_{m_i} \end{bmatrix}^\top \left( \mathbf{J}_A^{(m_i) \top} \boldsymbol{\mathcal{I}}_{m_i}^{\mathcal{R}_i} \mathbf{J}_A^{(m_i)} \right) \begin{bmatrix} \dot{\mathbf{q}}_{1:i-1} \\ \dot{\theta}_{m_i} \end{bmatrix} \\
 &= \frac{1}{2} \dot{\mathbf{q}}^\top \left( \sum_{i=1}^n m_{m_i} \mathbf{J}_L^{(m_i) \top} \mathbf{J}_L^{(m_i)} \right) \dot{\mathbf{q}} + \frac{1}{2} \dot{\mathbf{q}}^\top \left( \sum_{i=1}^n \mathbf{J}_{A,q}^{(m_i) \top} \boldsymbol{\mathcal{I}}_{m_i}^{\mathcal{R}_i} \mathbf{J}_{A,q}^{(m_i)} \right) \dot{\mathbf{q}} + \\
 &\quad + \frac{1}{2} \left( \dot{\mathbf{q}}^\top \sum_{i=1}^n \mathbf{J}_{A,q}^{(m_i) \top} \boldsymbol{\mathcal{I}}_{m_i}^{\mathcal{R}_i} \mathbf{J}_{A,\theta}^{(m_i)} \dot{\boldsymbol{\theta}} + \dot{\boldsymbol{\theta}}^\top \sum_{i=1}^n \mathbf{J}_{A,\theta}^{(m_i) \top} \boldsymbol{\mathcal{I}}_{m_i}^{\mathcal{R}_i} \mathbf{J}_{A,q}^{(m_i)} \dot{\mathbf{q}} \right) + \frac{1}{2} \sum_{i=1}^n \dot{\boldsymbol{\theta}}^\top \mathbf{J}_{A,\theta}^{\mathcal{R}_i \top} \boldsymbol{\mathcal{I}}_{m_i}^{\mathcal{R}_i} \mathbf{J}_{A,\theta}^{(m_i)} \dot{\boldsymbol{\theta}} \\
 &= \frac{1}{2} \dot{\mathbf{q}}^\top \left( \sum_{i=1}^n m_{m_i} \mathbf{J}_L^{(m_i) \top} \mathbf{J}_L^{(m_i)} \right) \dot{\mathbf{q}} + \sum_{i=1}^n \mathbf{J}_{A,q}^{(m_i) \top} \boldsymbol{\mathcal{I}}_{m_i}^{\mathcal{R}_i} \mathbf{J}_{A,q}^{(m_i)} \dot{\mathbf{q}} + \\
 &\quad + \frac{1}{2} 2 \dot{\mathbf{q}}^\top \left( \sum_{i=1}^n \mathbf{J}_{A,q}^{(m_i) \top} \boldsymbol{\mathcal{I}}_{m_i}^{\mathcal{R}_i} \mathbf{J}_{A,\theta}^{(m_i)} \right) \dot{\boldsymbol{\theta}} + \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \left( \sum_{i=1}^n \mathbf{J}_{A,\theta}^{\mathcal{R}_i \top} \boldsymbol{\mathcal{I}}_{m_i}^{\mathcal{R}_i} \mathbf{J}_{A,\theta}^{(m_i)} \right) \dot{\boldsymbol{\theta}} = \\
 &= \frac{1}{2} \dot{\mathbf{q}}^\top \left[ \mathbf{M}_{m,L} + \mathbf{S}(\mathbf{q}) \mathbf{B}^{-1} \mathbf{S}^\top(\mathbf{q}) \right] \dot{\mathbf{q}} + \dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}) \dot{\boldsymbol{\theta}} + \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \mathbf{B} \dot{\boldsymbol{\theta}} = \\
 &= \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M}_m(\mathbf{q}) \dot{\mathbf{q}} + \dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}) \dot{\boldsymbol{\theta}} + \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \mathbf{B} \dot{\boldsymbol{\theta}}
 \end{aligned} \tag{2.53}$$

where  $\mathbf{J}_{A,q_j}^{(m_i)}$  and  $\mathbf{J}_{A,\theta_j}^{(m_i)}$  are extracted from the angular Jacobian  $\mathbf{J}_{A_j}^{(m_i)}$ , in particular the joint angular Jacobian is obtained taking the first  $i-1$  columns while the rotor angular Jacobian is derived taking just the  $i$ -th column, and keeping all the remaining columns zero, i.e.

$$\mathbf{J}_{A,q_j}^{(m_i)} \triangleq \begin{cases} \mathbf{k}_{j-1} & 1 \leq j \leq i \text{ revolute} \\ \mathbf{0} & 1 \leq j \leq i \text{ prism. } \forall j > i \end{cases} \quad \mathbf{J}_{A,\theta_j}^{(m_i)} \triangleq \begin{cases} k_{r_i} \mathbf{k}_{m_i} & j = i \\ \mathbf{0} & j \neq i \end{cases} \tag{2.54}$$

and  $\mathbf{M}_m(\mathbf{q})$  is the configuration-dependent motor inertia matrix, since it contains rotor masses and inertial components along the other axes, while the motor spinning inertia matrix  $\mathbf{B}$  represents the constant diagonal matrix collecting rotor inertial components  $I_{m_{i,zz}}$  times the squared reduction ratio  $k_{r_i}^2$  and, finally,  $\mathbf{S}(\mathbf{q})$  gathers all the inertial couplings among rotors' and previous links' velocities in the robot, being in general strictly upper triangular with elements dependent in cascade with respect to the preceding joint positions in the chain.



$$\mathbf{M}_{m,L}(\mathbf{q}) \triangleq \sum_{i=1}^n \left( m_{m_i} \mathbf{J}_L^{(m_i)\top} \mathbf{J}_L^{(m_i)} \dot{\mathbf{q}} + \sum_{i=1}^n \mathbf{J}_{A,q}^{(m_i)\top} \mathcal{I}_{m_i}^{\mathcal{R}_i} \mathbf{J}_{A,q}^{(m_i)} \right) \quad (2.55)$$

$$\mathbf{B} \triangleq \sum_{i=1}^n \mathbf{J}_{A,\theta}^{(m_i)\top} \mathcal{I}_{m_i}^{\mathcal{R}_i} \mathbf{J}_{A,\theta}^{(m_i)} = \sum_{i,j=1}^n \mathbf{J}_{A,\theta_j}^{(m_i)\top} \mathcal{I}_{m_i}^{\mathcal{R}_i} \mathbf{J}_{A,\theta_j}^{(m_i)} = \sum_{i=1}^n k_{r_i}^2 \mathbf{k}_{m_i}^\top \mathcal{I}_{m_i}^{\mathcal{R}_i} \mathbf{k}_{m_i} = \sum_{i=1}^n k_{r_i}^2 I_{m_{i,zz}}^{\mathcal{R}_i} \quad (2.56)$$

$$\mathbf{S}(\mathbf{q}) \triangleq \sum_{i=1}^n \mathbf{J}_{A,q}^{(m_i)\top} \mathcal{I}_{m_i}^{\mathcal{R}_i} \mathbf{J}_{A,\theta}^{(m_i)} = \begin{bmatrix} 0 & S_{12} & S_{13}(q_2) & \dots & S_{1N}(q_2, \dots, q_{N-1}) \\ 0 & 0 & S_{23} & \dots & S_{2N}(q_3, \dots, q_{N-1}) \\ 0 & 0 & 0 & \dots & S_{3N}(q_4, \dots, q_{N-1}) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & S_{N-2,N}(q_{N-1}) \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (2.57)$$

The total kinetic energy is the sum of the link and motor components,

$$\begin{aligned} \mathcal{K} &= \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M}_l(\mathbf{q}) \dot{\mathbf{q}} + \left( \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M}_m(\mathbf{q}) \dot{\mathbf{q}} + \dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}) \dot{\boldsymbol{\theta}} + \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \mathbf{B} \dot{\boldsymbol{\theta}} \right) \\ &= \frac{1}{2} \dot{\mathbf{q}}^\top (\mathbf{M}_l(\mathbf{q}) + \mathbf{M}_m(\mathbf{q})) \dot{\mathbf{q}} + \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{S}(\mathbf{q}) \dot{\boldsymbol{\theta}} + \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \mathbf{S}^\top(\mathbf{q}) \dot{\mathbf{q}} + \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \mathbf{B} \dot{\boldsymbol{\theta}} \\ &= \frac{1}{2} \begin{bmatrix} \dot{\mathbf{q}}^\top & \dot{\boldsymbol{\theta}}^\top \end{bmatrix} \begin{bmatrix} \mathbf{M}(\mathbf{q}) & \mathbf{S}(\mathbf{q}) \\ \mathbf{S}^\top(\mathbf{q}) & \mathbf{B} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\boldsymbol{\theta}} \end{bmatrix} = \frac{1}{2} \dot{\mathbf{p}}^\top \mathcal{M}(\mathbf{p}) \dot{\mathbf{p}} \end{aligned} \quad (2.58)$$

where the total configuration-dependent inertia has been defined as

$$\mathbf{M}(\mathbf{q}) \triangleq \mathbf{M}_l(\mathbf{q}) + \mathbf{M}_m(\mathbf{q}) = \mathbf{M}_l(\mathbf{q}) + \mathbf{M}_{m,L}(\mathbf{q}) + \mathbf{S}(\mathbf{q}) \mathbf{B}^{-1} \mathbf{S}^\top(\mathbf{q}) \quad (2.59)$$

The total potential energy, due to gravity force contributes of both links and motors and elasticity force between joints and rotors, is

$$\mathcal{U} = \mathcal{U}_g + \mathcal{U}_e = - \sum_{i=1}^n (m_{l_i} \mathbf{g}_0^\top \mathbf{x}_{l_i} + m_{m_i} \mathbf{g}_0^\top \mathbf{x}_{m_i}) + \frac{1}{2} (\mathbf{q} - \boldsymbol{\theta})^\top \mathbf{K} (\mathbf{q} - \boldsymbol{\theta}) \quad (2.60)$$

The Lagrangian  $\mathcal{L}$  is computed as the difference of  $\mathcal{K}$  and  $\mathcal{U}$ , resulting in a vector function of the generalized state  $\mathbf{p} \triangleq \begin{bmatrix} \mathbf{q}^\top & \boldsymbol{\theta}^\top \end{bmatrix}^\top$  and velocity  $\dot{\mathbf{p}} = \begin{bmatrix} \dot{\mathbf{q}}^\top & \dot{\boldsymbol{\theta}}^\top \end{bmatrix}^\top$

$$\mathcal{L} = \frac{1}{2} \begin{bmatrix} \dot{\mathbf{q}}^\top & \dot{\boldsymbol{\theta}}^\top \end{bmatrix} \begin{bmatrix} \mathbf{M}(\mathbf{q}) & \mathbf{S}(\mathbf{q}) \\ \mathbf{S}^\top(\mathbf{q}) & \mathbf{B} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\boldsymbol{\theta}} \end{bmatrix} - \int \mathbf{g}(\mathbf{q}) d\mathbf{q} - \frac{1}{2} (\mathbf{q} - \boldsymbol{\theta})^\top \mathbf{K} (\mathbf{q} - \boldsymbol{\theta}) \quad (2.61)$$

where as usual  $\mathbf{g}(\mathbf{q}) = \left( \frac{\partial \mathcal{U}_g}{\partial \mathbf{q}} \right)^\top$ .

To include damping effects, we define the Rayleigh dissipation function given by the sum of a relative linear viscous friction between motor output angular velocity and joint variable though the relative damping matrix  $\mathbf{D}_{lm}$ , with direct friction linear

contributes acting separately on joint side through viscous coefficients  $\mathbf{D}_l$  and on motor side by matrix of dumping  $\mathbf{D}_m$ , which are all diagonal and positive-definite.

$$\mathcal{D} = \mathcal{D}_{lm} + \mathcal{D}_l + \mathcal{D}_m = \frac{1}{2} (\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}})^\top \mathbf{D}_{lm} (\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) + \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{D}_l \dot{\mathbf{q}} + \frac{1}{2} \dot{\boldsymbol{\theta}}^\top \mathbf{D}_m \dot{\boldsymbol{\theta}} \quad (2.62)$$

The Euler-Lagrange equations are given by the following system:

$$\begin{cases} \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top = \boldsymbol{\tau}_q & \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right)^\top + \left( \frac{\partial \mathcal{D}}{\partial \dot{\mathbf{q}}} \right)^\top = \mathbf{0} \\ \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right)^\top = \boldsymbol{\tau}_\theta & \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top - \left( \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} \right)^\top + \left( \frac{\partial \mathcal{D}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top = \boldsymbol{\tau}_c \end{cases} \quad (2.63)$$

The time derivatives of the partial derivatives of the Lagrangian, i.e. only kinetic energy, with respect to the generalized velocities  $\dot{\mathbf{p}}$  are

$$\begin{aligned} \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right)^\top &= \frac{d}{dt} (\mathbf{M}(\mathbf{q}) \dot{\mathbf{q}} + \mathbf{S}(\mathbf{q}) \dot{\boldsymbol{\theta}}) = \dot{\mathbf{M}}(\mathbf{q}) \dot{\mathbf{q}} + \mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \dot{\mathbf{S}}(\mathbf{q}) \dot{\boldsymbol{\theta}} + \mathbf{S}(\mathbf{q}) \ddot{\boldsymbol{\theta}} \\ \frac{d}{dt} \left( \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{\theta}}} \right)^\top &= \frac{d}{dt} (\mathbf{S}^\top(\mathbf{q}) \dot{\mathbf{q}} + \mathbf{B} \dot{\boldsymbol{\theta}}) = \dot{\mathbf{S}}^\top(\mathbf{q}) \dot{\mathbf{q}} + \mathbf{S}^\top(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{B} \ddot{\boldsymbol{\theta}} \end{aligned} \quad (2.64)$$

The partial derivatives of the Lagrangian, being coincident with the derivation of potential energy with respect to  $\mathbf{p}$ , are

$$\frac{\partial \mathcal{L}}{\partial \mathbf{q}} = -\mathbf{g}(\mathbf{q}) - \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) + \frac{1}{2} \dot{\mathbf{q}}^\top \frac{\partial \mathbf{M}(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} + \dot{\boldsymbol{\theta}}^\top \frac{\partial \mathbf{S}^\top(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} \quad \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = -\mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) \quad (2.65)$$

The damping forces are calculated as minus the gradient of the dissipation function

$$\frac{\partial \mathcal{D}}{\partial \dot{\mathbf{q}}} = \mathbf{D}_{lm}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) + \mathbf{D}_l \dot{\mathbf{q}} \quad \frac{\partial \mathcal{D}}{\partial \dot{\boldsymbol{\theta}}} = -\mathbf{D}_{lm}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) + \mathbf{D}_m \dot{\boldsymbol{\theta}} \quad (2.66)$$

Substituting these expressions into the Euler-Lagrange equations  $2n$  differential equations arise, where the actuated torque  $\boldsymbol{\tau}_l$  is applied only to the motor variables. Eventually, this system may be expressed in block matrix form as done previously, allowing to obtain the **complete flexible-joint robot dynamic model**.

$$\begin{bmatrix} \mathbf{M}(\mathbf{q}) & \mathbf{S}(\mathbf{q}) \\ \mathbf{S}^\top(\mathbf{q}) & \mathbf{B} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \ddot{\boldsymbol{\theta}} \end{bmatrix} + \begin{bmatrix} \mathbf{c}_1(\mathbf{q}, \dot{\mathbf{q}}, \dot{\boldsymbol{\theta}}) \\ \mathbf{c}_2(\mathbf{q}, \dot{\mathbf{q}}) \end{bmatrix} + \begin{bmatrix} \mathbf{D}_{lm}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) + \mathbf{D}_l \dot{\mathbf{q}} \\ \mathbf{D}_{lm}(\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) + \mathbf{D}_m \dot{\boldsymbol{\theta}} \end{bmatrix} + \begin{bmatrix} \mathbf{g}(\mathbf{q}) + \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) \\ \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\tau}_c \end{bmatrix} \quad (2.67)$$

In this expression, given the generalized coordinates  $\mathbf{p} = [\mathbf{q}^\top \ \boldsymbol{\theta}^\top]^\top$ , the total inertia matrix  $\boldsymbol{\mathcal{M}}(\mathbf{p})$  is a block matrix composed by the inertia matrix of the links

$\mathbf{M}(\mathbf{q})$ , and by  $\mathbf{S}(\mathbf{q})$  that describes the coupling between links and motors, both dependent only on the configuration  $\mathbf{q}$  of the robot structure, while  $\mathbf{B}$  represents the constant inertia matrix of the motors. Notice that  $\mathcal{M}(\mathbf{p}) \equiv \mathcal{M}(\mathbf{q})$  since, due to the initial assumptions, the total inertia matrix depends solely on the robot pose  $\mathbf{q}$  and not on the rotor variable  $\boldsymbol{\theta}$ , because the inertial components of the rotors are confined to the z-axis of the revolute joints on which the rotor has the center of mass and around which it perfectly spins, allowing a simplified rotor shaft dynamics which neglect any possible precession effect on the overall dynamics.

The terms  $\mathbf{c}_1(\mathbf{q}, \dot{\mathbf{q}}, \dot{\boldsymbol{\theta}})$  and  $\mathbf{c}_2(\mathbf{q}, \dot{\mathbf{q}})$  collect the Coriolis and centrifugal terms and can be proved to be

$$\mathbf{c}_1(\mathbf{q}, \dot{\mathbf{q}}, \dot{\boldsymbol{\theta}}) = \dot{\mathbf{M}}(\mathbf{q}) \dot{\mathbf{q}} + \dot{\mathbf{S}}(\mathbf{q}) \dot{\boldsymbol{\theta}} - \frac{1}{2} \dot{\mathbf{q}}^\top \frac{\partial \mathbf{M}(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}^\top \frac{\partial \mathbf{S}^\top(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} \quad (2.68)$$

$$\mathbf{c}_2(\mathbf{q}, \dot{\mathbf{q}}) = \dot{\mathbf{S}}^\top(\mathbf{q}) \dot{\mathbf{q}} - \frac{1}{2} \dot{\mathbf{q}}^\top \frac{\partial \mathbf{S}^\top(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} \quad (2.69)$$

Where the first contribution (2.68) can be decomposed into the rigid centrifugal and Coriolis term  $\mathbf{c}$ , depending only on  $\mathbf{q}$  and  $\dot{\mathbf{q}}$ , and a motor-joint velocity contribution  $\hat{\mathbf{c}}_1$ , which is also a function of  $\dot{\boldsymbol{\theta}}$

$$\mathbf{c}_1(\mathbf{q}, \dot{\mathbf{q}}, \dot{\boldsymbol{\theta}}) = \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) + \hat{\mathbf{c}}_1(\mathbf{q}, \dot{\mathbf{q}}, \dot{\boldsymbol{\theta}}) \quad (2.70)$$

The overall Centrifugal-Coriolis vector term can be further expressed in a compact bilinear form with respect to the velocity vector  $\dot{\mathbf{p}} = [\dot{\mathbf{q}}^\top \ \dot{\boldsymbol{\theta}}^\top]^\top$  by exploiting the Christoffel symbols [8] as follows, such that each element of  $\mathbf{c}_{tot} \in \mathbb{R}^{2n}$  is

$$c_{tot_i}(\mathbf{p}, \dot{\mathbf{p}}) = \frac{1}{2} \dot{\mathbf{p}}^\top \left[ \frac{\partial \mathcal{M}_i}{\partial \mathbf{p}} + \left( \frac{\partial \mathcal{M}_i}{\partial \mathbf{p}} \right)^\top - \frac{\partial \mathcal{M}_i}{\partial p_i} \right] \dot{\mathbf{p}} \quad i = 1, \dots, 2n \quad (2.71)$$

where  $\mathcal{M}_i$  represents the i-th column of the total inertia matrix.

Let us notice that the velocity-dependent terms in  $\mathbf{c}_1$  and  $\mathbf{c}_2$  are independent of the motor positions  $\boldsymbol{\theta}$  while the specific dependence of  $\hat{\mathbf{c}}_1$  and  $\mathbf{c}_2$  arises only when  $\mathbf{S}(\mathbf{q})$  varies with the configuration  $\mathbf{q}$ , in fact if it is kept constant, both  $\hat{\mathbf{c}}_1$  and  $\mathbf{c}_2$  vanish, leaving only the rigid cross-velocities apparent forces.

Finally, the Coriolis and centrifugal terms may be factored linearly with respect to the velocity vector  $\dot{\mathbf{p}}$ , as done for the previous models

$$\mathbf{c}_{tot}(\mathbf{p}, \dot{\mathbf{p}}) = \mathbf{C}(\mathbf{p}, \dot{\mathbf{p}}) \dot{\mathbf{p}} \quad (2.72)$$

In conclusion, with this treatment we demonstrate the fundamental properties of a robot manipulator dynamics which is highly nonlinear and coupled. In particular, the coupling between input-output pairs is given by two sources of torque contributions: the linear dynamics influence due to the mutual acceleration among different joints originated from the inertia tensor, and the nonlinear cross-velocity effects produced by the centrifugal and Coriolis apparent forces.

## Chapter 3

# Time-Domain Decoupling Control

The dynamic decoupling of a flexible-joint robot can be realized on the overall nonlinear model, thus in the time domain, either in the state-space or equivalently in the joint space. The objective is designing a non-linear controller in order to compensate both the linear couplings, originated from the inertial, elastic and relative damping interactions, that the nonlinear dynamics, related to the centrifugal-Coriolis and gravitational forces plus nonlinear friction.

In detail, the decoupling of the flexible-joint dynamics in the time domain is achievable on the joint space through the nonlinear feedback linearization control approach, i.e. a non-linear state feedback law leading to a closed-loop system with  $n$  linear and decoupled systems. This controller functions as a nonlinear decoupler, able to transform the MIMO robot model into independent SISO systems, one for each link. In this way, given a reference trajectory, it is possible to devise  $n$  separated feedback controllers such that the tracking errors get forced to be globally exponentially stable, where the decaying rate can be specified by tuning the scalar feedback gains [9].

Notice that this approach can be seen as an extension of the well-known Computed Torque Control method [3] for rigid robots, to more sophisticated models including elastic and damping effects.

### 3.1 Elastic-Joint Robots - Input-State Feedback Linearization

Taking into consideration the Elastic Joint Robot Model (2.36), it is possible to generalize the idea of designing a control which provides a torque that makes the system decupled, even if it is not straightforward.

Let the desired trajectory be a smooth vector signal  $\mathbf{q}_d(t)$  for the robot links. The decoupling control design is based on the system inversion, i.e. the inverse dynamics computation, but rather using the current measures of the state variables  $[\mathbf{q}^\top \quad \boldsymbol{\theta}^\top \quad \dot{\mathbf{q}}^\top \quad \dot{\boldsymbol{\theta}}^\top]$  instead of the reference state evolution  $[\mathbf{q}_d^\top \quad \boldsymbol{\theta}_d^\top \quad \dot{\mathbf{q}}_d^\top \quad \dot{\boldsymbol{\theta}}_d^\top]$ .

Notice that there is no need to transform the robot equations into their state-space representation, that is the standard form exploited in control design for general nonlinear system, in fact it is possible to work on the robot model directly in its second-order differential form, namely the Cauchy's problem, widely used for mechanical systems.

Let us express the link equation in a compact form where we define the non-linear term  $\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) \triangleq \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q})$

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) = \mathbf{0} \quad (3.1)$$

Since none of the above quantities depends explicitly on the input torque, based on the feedback linearization theory, we have to differentiate (3.1) once with respect to time, obtaining

$$\mathbf{M}(\mathbf{q})\mathbf{q}^{[3]} + \dot{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{K}(\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) = \mathbf{0} \quad (3.2)$$

Applying again this principle until the output comes out, leads to

$$\mathbf{M}(\mathbf{q})\mathbf{q}^{[4]} + 2\dot{\mathbf{M}}(\mathbf{q})\mathbf{q}^{[3]} + \ddot{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} + \ddot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{K}(\ddot{\mathbf{q}} - \ddot{\boldsymbol{\theta}}) = \mathbf{0} \quad (3.3)$$

And finally  $\ddot{\boldsymbol{\theta}}$  shows explicitly. While in the motor equation notice that the motor acceleration is at the same differential level of  $\boldsymbol{\tau}$

$$\mathbf{B}\ddot{\boldsymbol{\theta}} + \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) = \boldsymbol{\tau} \quad (3.4)$$

from which it is possible to find the expression of  $\ddot{\boldsymbol{\theta}}$

$$\ddot{\boldsymbol{\theta}} = \mathbf{B}^{-1}(\boldsymbol{\tau} - \mathbf{K}(\boldsymbol{\theta} - \mathbf{q})) \quad (3.5)$$

to be replaced in the differentiated link equation (3.3), getting

$$\mathbf{M}(\mathbf{q})\mathbf{q}^{[4]} + 2\dot{\mathbf{M}}(\mathbf{q})\mathbf{q}^{[3]} + \ddot{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} + \ddot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{K}\ddot{\mathbf{q}} = \mathbf{K}\mathbf{B}^{-1}[\boldsymbol{\tau} - \mathbf{K}(\boldsymbol{\theta} - \mathbf{q})] \quad (3.6)$$

It can be expressed in function of  $\mathbf{q}$  only, replacing the last term  $\mathbf{K}(\boldsymbol{\theta} - \mathbf{q})$  with  $\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})$  obtained from the link equation (3.1).

$$\mathbf{M}(\mathbf{q})\mathbf{q}^{[4]} + 2\dot{\mathbf{M}}(\mathbf{q})\mathbf{q}^{[3]} + \ddot{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} + \ddot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{K}\ddot{\mathbf{q}} = \mathbf{K}\mathbf{B}^{-1}[\boldsymbol{\tau} - \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} - \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})] \quad (3.7)$$

It is important noting that each joint variable  $q_i$  needs to be differentiated to the fourth order, i.e.,  $\gamma_i = 4$ , thus, choosing as output the joint variables  $\mathbf{q}$ , the total relative degree of the system is  $\gamma = \sum_{i=1}^n \gamma_i = 4n$ . Recalling that the state of a robot with elastic joint is defined like  $[\mathbf{q}^\top \quad \dot{\mathbf{q}}^\top \quad \boldsymbol{\theta}^\top \quad \dot{\boldsymbol{\theta}}^\top] \in \mathbb{R}^{4n}$  which has the same dimension of the relative degree  $\gamma \equiv n_x = 4n$ , follows that the input-output linearization corresponds to a complete input-state linearization.

Given that the matrix  $\mathcal{D}(\mathbf{q}) \triangleq \mathbf{M}^{-1}(\mathbf{q})\mathbf{K}\mathbf{B}^{-1}$  is always invertible, it becomes possible to assign any desired value  $\mathbf{v}$  to the fourth derivative of  $\mathbf{q}$  by appropriately selecting the input torque  $\boldsymbol{\tau}$ . This matrix  $\mathcal{D}$ , referred to as the decoupling matrix of the system, must remain nonsingular, which is a necessary and sufficient condition to achieve decoupled input-output behavior through the nonlinear state feedback.

Based on these considerations, the input torque can be selected as

$$\boldsymbol{\tau} = \mathbf{B}\mathbf{K}^{-1}[\mathbf{M}(\mathbf{q})\mathbf{v} + \ddot{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} + 2\dot{\mathbf{M}}(\mathbf{q})\mathbf{q}^{[3]} + \ddot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}})] + [\mathbf{M}(\mathbf{q}) + \mathbf{B}]\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) \quad (3.8)$$

By feedbacking this non-linear torque signal it is possible to prove that the control law leads to a closed-loop robot system, which is linear and decoupled, namely corresponding to a quadruple integrator system, i.e. chains of 4 input-output integrators from each new input  $v_i$  to the relative link position  $q_i$  where  $i = 1, \dots, n$ .

$$\mathbf{q}^{[4]} = \mathbf{v} \quad (3.9)$$

The latter feedback law is a function of the so-called linearizing coordinates  $[\mathbf{q}^\top \quad \dot{\mathbf{q}}^\top \quad \ddot{\mathbf{q}}^\top \quad \mathbf{q}^{[3]\top}]$  only, which results practically quite hard to implement with just position encoders, since it is necessary to derive numerically velocity, acceleration and jerk which would be too noisy. As the state-representation is not unique, it is possible to express the linearizing coordinates in a equivalent way to the conventional state variables for a elastic-joint robot through invertible transformations.

$$[\mathbf{q}^\top \quad \dot{\mathbf{q}}^\top \quad \ddot{\mathbf{q}}^\top \quad \mathbf{q}^{[3]\top}] \iff [\mathbf{q}^\top \quad \dot{\mathbf{q}}^\top \quad \boldsymbol{\theta}^\top \quad \dot{\boldsymbol{\theta}}^\top] \quad (3.10)$$

In detail we can exploit the link equation (3.1) to obtain the acceleration

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1}(\mathbf{q})[\mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) - \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})] \quad (3.11)$$

and its first derivative (3.3) to compute the jerk, where  $\ddot{\mathbf{q}}$  has already been calculated as function of  $\mathbf{q}$ ,  $\dot{\mathbf{q}}$  and  $\boldsymbol{\theta}$  above

$$\mathbf{q}^{[3]} = \mathbf{M}^{-1}(\mathbf{q})[\mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) - \dot{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} - \dot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}})] \quad (3.12)$$

Hence, the exact linearizing and decoupling control law could be rewritten as a static feedback law in terms of the original state-space variables, by substituting the previous expressions.

$$\boldsymbol{\tau} = \boldsymbol{\tau}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \mathbf{q}^{[3]}, \mathbf{v}) = \boldsymbol{\tau}(\mathbf{q}, \boldsymbol{\theta}, \dot{\mathbf{q}}, \dot{\boldsymbol{\theta}}, \mathbf{v}) \quad (3.13)$$

For completeness, the analytic feedback control law is here reported.

$$\begin{aligned} \boldsymbol{\tau} = & \mathbf{B}\mathbf{K}^{-1} \left\{ \mathbf{M}(\mathbf{q})\mathbf{v} + \ddot{\mathbf{M}}(\mathbf{q}) \left[ \mathbf{M}^{-1}(\mathbf{q}) \left[ \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) - \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) \right] \right] + 2\dot{\mathbf{M}}(\mathbf{q}) \cdot \right. \\ & \cdot \left[ \mathbf{M}^{-1}(\mathbf{q}) \left[ \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) - \dot{\mathbf{M}}(\mathbf{q}) \left[ \mathbf{M}^{-1}(\mathbf{q}) \left[ \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) - \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) \right] \right] - \dot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) \right] \right] \\ & \left. + \ddot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) \right\} + \left[ \mathbf{M}(\mathbf{q}) + \mathbf{B} \right] \left[ \mathbf{M}^{-1}(\mathbf{q}) \left[ \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) - \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) \right] \right] + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) \end{aligned} \quad (3.14)$$

By the way, the implementation of this system can be enhanced in order to optimize computations by properly organizing the evaluation of derivatives, exploiting eventual already calculated quantities.

Thus, the inversion process results in a control torque  $\boldsymbol{\tau}$  that is expressed as a state feedback control law, which compensates for the elastic-joint robot dynamics and supplant it with a linear and decoupled system of an appropriate differential order [8]. From a physical standpoint, this control makes the dynamics of an elastic-joint robot rigid, irrespective of the values of the elastic constants.

A fundamental aspect about the feedback linearization is the feasibility of inverting the system from the output  $\mathbf{q}$  without causing instability problems, due to the presence of unobservable internal dynamics in the closed-loop system. In the elastic case, when inverting to determine the input  $\boldsymbol{\tau}$  that imposes  $\mathbf{q}^{[4]} = \mathbf{v}$ , it has been proved there is no dynamics left other than the one appearing in the closed-loop input-output mapping, guaranteeing a full input-state feedback linearization.

In the end, from a practical point of view, if compared to the computed torque method compatible with rigid robots, the feedback linearization control requires, beyond the inversion of the inertia tensor, also the additional evaluation of higher derivatives of the state and dynamic components, so it is computationally more expensive [9].

## 3.2 Direct Elastic-Dumped Joint Robots - Input-State Feedback Linearization

The feedback linearization approach may even also be applied in the presence of linear viscous friction, or in general to smooth friction terms, acting both on the link side and on the motor side. Let us consider again the Direct Elastic-Dumped Joint Robot Model (2.50) introduced in Section 2.1.4, whose link equation can be written, collecting the non-linear terms in  $\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})$ , as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{D}_l\dot{\mathbf{q}} + \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) = \mathbf{0} \quad (3.15)$$

Analogously to the purely elastic-joint scenario, we aim at designing a control law  $\boldsymbol{\tau}$  such that the resulting closed-loop system exhibits a linear input-output mapping of fourth order in the chosen output  $\mathbf{q}$ .

Firstly, let us define the extended non-linear term to include friction

$$\tilde{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) \triangleq \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{D}_l\dot{\mathbf{q}} \quad (3.16)$$

so that the link equation (3.15) can be rewritten more compactly like

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \tilde{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) = \mathbf{0} \quad (3.17)$$

Since none of the above quantities depends explicitly on the control input  $\tau$ , we must differentiate the link equation (3.17) until the torque appears. Following the same strategy described for the frictionless elastic-joint case, we differentiate once with respect to time, yielding

$$\mathbf{M}(\mathbf{q})\mathbf{q}^{[3]} + \dot{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\tilde{\mathbf{n}}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{K}(\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) = \mathbf{0} \quad (3.18)$$

and applying derivative to (3.18) a second time, leads to

$$\mathbf{M}(\mathbf{q})\mathbf{q}^{[4]} + 2\dot{\mathbf{M}}(\mathbf{q})\mathbf{q}^{[3]} + \ddot{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} + \ddot{\tilde{\mathbf{n}}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{K}(\ddot{\mathbf{q}} - \ddot{\boldsymbol{\theta}}) = \mathbf{0} \quad (3.19)$$

We now notice in equation (3.19) that the input variable of robot dynamics  $\ddot{\boldsymbol{\theta}}$  appears explicitly on the link side.

While reminding the motor equation including the viscous damping on the motor side, which is the following

$$\mathbf{B}\ddot{\boldsymbol{\theta}} + \mathbf{D}_m\dot{\boldsymbol{\theta}} + \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) = \boldsymbol{\tau} \quad (3.20)$$

it is possible to notice the output  $\boldsymbol{\theta}$  is at the same differential level as the input  $\boldsymbol{\tau}$  in motor equation (3.20), so we can solve for  $\ddot{\boldsymbol{\theta}}$  directly

$$\ddot{\boldsymbol{\theta}} = \mathbf{B}^{-1} \left[ \boldsymbol{\tau} - \mathbf{D}_m\dot{\boldsymbol{\theta}} - \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) \right] \quad (3.21)$$



Substituting the analytical expression of  $\ddot{\boldsymbol{\theta}}$  given by (3.21) back into (3.19) gives an equation where the output  $\boldsymbol{\tau}$  finally appears in the link equation.

$$\mathbf{M}(\mathbf{q})\mathbf{q}^{[4]} + 2\dot{\mathbf{M}}(\mathbf{q})\mathbf{q}^{[3]} + \ddot{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} + \ddot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{K}\ddot{\mathbf{q}} = \mathbf{K}\mathbf{B}^{-1}\left[\boldsymbol{\tau} - \mathbf{D}_m\dot{\boldsymbol{\theta}} - \mathbf{K}(\boldsymbol{\theta} - \mathbf{q})\right] \quad (3.22)$$

Then, for what regards the left-hand side, employing once again the elastic-dumped link equation from which isolate the term  $\mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \ddot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}})$ , and obtaining  $\boldsymbol{\theta}$  deriving once the same equation, the derived link equation can be expressed solely in terms of  $\mathbf{q}$ ,  $\dot{\mathbf{q}}$ ,  $\boldsymbol{\tau}$ .

As a result, each joint variable  $q_i$  must be differentiated four times ( $\gamma_i = 4$ ) in order for the input torque  $\tau_i$  to show up. Thus, choosing as output the joint variables  $q_i$ , the total relative degree of the system is  $\gamma = \sum_{i=1}^n \gamma_i = 4n$ . Since the state for the direct elastic-dumped model is  $[\mathbf{q}^\top \ \dot{\mathbf{q}}^\top \ \boldsymbol{\theta}^\top \ \dot{\boldsymbol{\theta}}^\top] \in \mathbb{R}^{4n}$ , which matches the dimension of the relative degree  $\gamma$ , the input-output linearization amounts to a complete input-state feedback linearization, as the simplified elastic model. Similarly, the decoupling matrix must be invertible to achieve the desired decoupling, in fact under assumptions of non-singularity of  $\mathbf{M}(\mathbf{q})$ ,  $\mathbf{B}$ , and  $\mathbf{K}$ , it is possible to invert  $\mathbf{D}(\mathbf{q}) = \mathbf{M}^{-1}(\mathbf{q})\mathbf{K}\mathbf{B}^{-1}$ , which is the same as the elastic case, and thus isolate  $\mathbf{q}^{[4]}$ . Thus, we can assign any desired input  $\mathbf{v}$  by an appropriate choice of the input torque  $\boldsymbol{\tau}$ , imposing that the snap must be equal to this new input

$$\mathbf{q}^{[4]} \stackrel{!}{=} \mathbf{v} \quad (3.23)$$

One can solve algebraically for  $\boldsymbol{\tau}$  to obtain the following type of feedback law

$$\boldsymbol{\tau} = \mathbf{B}\mathbf{K}^{-1}\left[\mathbf{M}(\mathbf{q})\mathbf{v} + \ddot{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} + 2\dot{\mathbf{M}}(\mathbf{q})\mathbf{q}^{[3]} + \ddot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}})\right] + [\mathbf{M}(\mathbf{q}) + \mathbf{B}]\ddot{\mathbf{q}} + \ddot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{D}_m\dot{\boldsymbol{\theta}} \quad (3.24)$$

that, after expanding the extended non-linear term, becomes

$$\begin{aligned} \boldsymbol{\tau} = \mathbf{B}\mathbf{K}^{-1}\left[\mathbf{M}(\mathbf{q})\mathbf{v} + 2\dot{\mathbf{M}}(\mathbf{q})\mathbf{q}^{[3]} + \left(\ddot{\mathbf{M}}(\mathbf{q}) + \mathbf{D}_l + \mathbf{K}\right)\ddot{\mathbf{q}} + \right. \\ \left. + \ddot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}})\right] + \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{D}_l\dot{\mathbf{q}} + \mathbf{D}_m\dot{\boldsymbol{\theta}} \end{aligned} \quad (3.25)$$

The torque expression can be equivalently recast in terms of the original state variables  $[\mathbf{q}^\top \ \dot{\mathbf{q}}^\top \ \boldsymbol{\theta}^\top \ \dot{\boldsymbol{\theta}}^\top]$  also, similarly as done in Section 3.1.

In the end, even when linear viscous friction appears both on the link and motor sides, the feedback linearization technique follows the same conceptual steps as in the purely elastic scenario. Friction does not affect the invertibility conditions, but simply enters as additional terms inside the non-linear functions and their time derivatives. Therefore, the final closed-loop system is still described by  $n$  decoupled fourth-order integrators from  $\mathbf{v}$  to  $\mathbf{q}$ , and leaving no internal dynamics unobserved or potentially unstable.

### 3.3 Relative Elastic-Damped Joint Robots - Input-Output Feedback Linearization

The feedback linearization decoupling technique can also be applied to an elastic-joint robot featuring mutual viscous damping on the elastic transmission, assumed as a damping proportional to the deflection between links and motors, whose model (2.44) is obtained in Section 2.1.3. On the link side the rigid robot equations are

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{D}(\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) + \mathbf{K}(\mathbf{q} - \boldsymbol{\theta}) = \mathbf{0} \quad (3.26)$$

As before, we firstly define the output of interest as the joint position  $\mathbf{y} \triangleq \mathbf{q}$ , in order to determine how many times  $\mathbf{q}$  must be differentiated for the control variable  $\mathbf{u} = \boldsymbol{\tau}$  to appear, thereby obtaining the feedback law that makes the input-output map linear and decoupled.

By differentiating the elastic-damped link equation (3.26) with respect to time, we derive the following equation

$$\mathbf{M}(\mathbf{q}) \mathbf{q}^{[3]} + \dot{\mathbf{M}}(\mathbf{q}) \ddot{\mathbf{q}} + \dot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{D}(\ddot{\mathbf{q}} - \ddot{\boldsymbol{\theta}}) + \mathbf{K}(\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) = \mathbf{0} \quad (3.27)$$

At this stage,  $\ddot{\boldsymbol{\theta}}$  appears, but  $\boldsymbol{\tau}$  is still absent so it is necessary to work algebraically on the motor dynamics, which is

$$\mathbf{B} \ddot{\boldsymbol{\theta}} + \mathbf{D}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) + \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) = \boldsymbol{\tau} \quad (3.28)$$

So, isolating  $\ddot{\boldsymbol{\theta}}$  from the motor equation (3.28) we obtain the relation

$$\ddot{\boldsymbol{\theta}} = \mathbf{B}^{-1} [\boldsymbol{\tau} - \mathbf{D}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) - \mathbf{K}(\boldsymbol{\theta} - \mathbf{q})] \quad (3.29)$$

where we observe that  $\ddot{\boldsymbol{\theta}}$  is already expressed at the same differential order as the input  $\boldsymbol{\tau}$ , differently from the previous treatment, and substituting (3.29) in the link equation (3.27), the relationship between  $\ddot{\boldsymbol{\theta}}$  and  $\boldsymbol{\tau}$  emerges.

$$\boldsymbol{\tau} = \mathbf{B} \mathbf{D}^{-1} \left[ \mathbf{M}(\mathbf{q}) \mathbf{q}^{[3]} + \dot{\mathbf{M}}(\mathbf{q}) \ddot{\mathbf{q}} + \dot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{D} \ddot{\mathbf{q}} + \mathbf{K}(\dot{\mathbf{q}} - \dot{\boldsymbol{\theta}}) \right] + \mathbf{D}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) + \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) \quad (3.30)$$

In such way, the variable  $\boldsymbol{\tau}$  becomes explicit already at  $\mathbf{q}^{[3]}$ , due to the effect of the mutual damping among each link-motor pair.

Since the relative degree of each link  $\mathbf{q}$  is limited to 3, rather than 4, for  $n$  joints we have an overall relative degree  $\gamma = 3n < 4n$ , so we cannot achieve full-state linearization but only input-output linearization. In other words, as the state dimension is  $4n$ , there remains a space of dimension  $4n - 3n = n$  which corresponds to the number of unobservable variables in the internal dynamics.

That internal dynamics can be derived by working on equation (3.27), firstly re-arranging the terms

$$\mathbf{D}\ddot{\boldsymbol{\theta}} + \mathbf{K}\dot{\boldsymbol{\theta}} = \mathbf{M}(\mathbf{q})\mathbf{q}^{[3]} + \dot{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{D}\ddot{\mathbf{q}} + \mathbf{K}\dot{\mathbf{q}} \quad (3.31)$$

so, assuming  $\boldsymbol{\theta}$  as state, we obtain a first-order linear system

$$\mathbf{D}\ddot{\boldsymbol{\theta}} + \mathbf{K}\dot{\boldsymbol{\theta}} = \boldsymbol{\omega} \quad (3.32)$$

which results asymptotically stable, since  $\mathbf{D}$  and  $\mathbf{K}$  are diagonal matrices with positive values, ensuring that the state matrix has all non-positive eigenvalues, where we defined the forcing input  $\boldsymbol{\omega}$  as

$$\boldsymbol{\omega} \triangleq \mathbf{M}(\mathbf{q})\mathbf{q}^{[3]} + [\dot{\mathbf{M}}(\mathbf{q}) + \mathbf{D}]\ddot{\mathbf{q}} + \dot{\mathbf{n}}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{K}\dot{\mathbf{q}} \quad (3.33)$$

Eventually, if we introduce a new input  $\mathbf{v}(t) \in \mathbb{R}^n$  as desired, we can impose the behaviour to the third derivative of the joint vector, similarly as done in previous sections but with a lower degree of derivation

$$\mathbf{q}^{[3]} \stackrel{!}{=} \mathbf{v} \quad (3.34)$$

thanks to a feedback law that renders the output  $\mathbf{q}$  equivalent to  $n$  chains of triple integrators, while leaving an internal dynamic of dimension  $n$ , which is unobservable from the output but asymptotically stable [8].

In conclusion, while the presence of dissipative terms separately for both link and motor sides allows them to be inserted in the computation without particular modifications in the structure of the feedback law, the inclusion of spring frictional forces on the transmission requires a whole different compensation, consisting only of a partial input-output feedback.

## Chapter 4

# Frequency-Domain Decoupling Control

In a general Multi-Input Multi-Output (MIMO) system, interactions between input and output variables impact the stability and overall control performance of the system. If the model is linear, the transfer function matrix can be derived and used to design various types of decoupling compensators in the frequency domain, aimed at reducing mutual coupling effects among different channels.

This chapter analyzes the theoretical foundation and practical implementations of various linear decoupling strategies, originally created for LTI system and extended to LPV system, which may be classified, based on time-variant characteristics, in static and dynamic approaches. The static decoupling strategies taken into account are the so-called static decoupling and the static singular value decomposition decoupling, offering an efficient but not precise compensation. About the dynamic decoupling techniques, so the ideal, simplified, and inverted decoupling, they provide a more effective approach, at a higher computational cost.

The robot plant can be modeled as a MIMO LPV system, used for controller synthesis, so a linear decoupler can be employed to compensate the linear dynamics, being computationally cheaper than nonlinear approaches. For theoretical simplicity and direct implementation, we focus on TITO systems, since the most interacting robot axes are represented by the first two, mainly due to inertial couplings.

### 4.1 Linear Decoupling Overview

Given a Multi-Input Multi-Output Linear Time-Invariant (MIMO LTI) system, with input  $\mathbf{u}(t) \in \mathbb{R}^n$  and output  $\mathbf{y}(t) \in \mathbb{R}^n$ , it can be described by the following input-output frequency relation

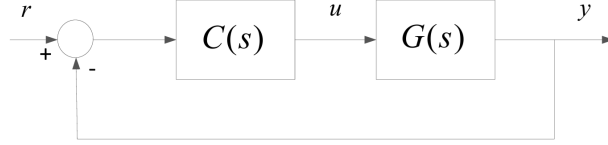
$$\mathbf{Y}(s) = \mathbf{G}(s)\mathbf{U}(s) \quad (4.1)$$

determined by the transfer function matrix  $\mathbf{G}(s) \in \mathbb{R}^{n \times n}$  as

$$\mathbf{G}(s) = \left\{ G_{ij}(s), i = 1, \dots, n, j = 1, \dots, n \right\}_{n \times n} \quad (4.2)$$

where the cross transfer functions of the system matrix  $G_{ij}(s)$   $i \neq j$  are the source of input-output couplings.

A controller, as in Figure 4.1, can be designed in order to impose a certain dynamics, based on three possible strategies: decentralized, decoupled or centralized control.



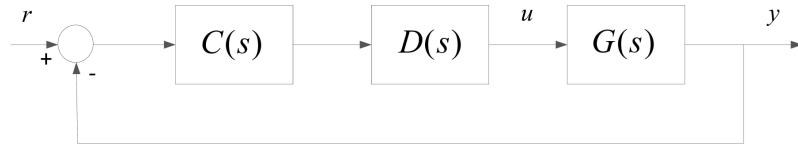
**Figure 4.1:** General Control structure of MIMO system

The decentralized controller is the simplest MIMO control strategy, where each input-output pair is managed independently by separate controllers, relying on a diagonal controller matrix  $\mathbf{C}(s) \in \mathbb{R}^{n \times n}$  defined as

$$\mathbf{C}(s) = \text{diag}\{C_{11}(s), C_{22}(s), \dots, C_{nn}(s)\}_{n \times n} \quad (4.3)$$

which simplifies the design and implementation but does not directly account for the interactions between different inputs and outputs, so it may struggle to manage strong couplings effectively.

For this reason, it is necessary to tackle explicitly the coupling issue, either realizing a decoupling compensator to keep a decentralized controller, being more efficient, or designing a centralized controller, with a higher computational cost. In this treatment, the focus is indeed studying the decoupling strategies, which offer a pre-compensation to cancel out cross-channel interactions, thus allowing the use of a decentralized controller, according to the scheme in Figure 4.2.



**Figure 4.2:** Decoupled Control structure of MIMO system

In order to explain the coupling problem theoretically, the simplest scenario consists of a system with two inputs and two outputs. Moreover, it lays the groundwork to develop the application of decoupling controllers to a robotic plant with two axes.

Let us consider a Two-Input Two-Output (TITO) defining the controller matrix  $\mathbf{C}(s) \in \mathbb{C}^{2 \times 2}$ , the decoupler matrix  $\mathbf{D}(s) \in \mathbb{C}^{2 \times 2}$ , and the system transfer function matrix  $\mathbf{G}(s) \in \mathbb{C}^{2 \times 2}$ , as follows

$$\mathbf{C}(s) = \begin{bmatrix} C_1(s) & 0 \\ 0 & C_2(s) \end{bmatrix} \quad \mathbf{D}(s) = \begin{bmatrix} D_{11}(s) & D_{12}(s) \\ D_{21}(s) & D_{22}(s) \end{bmatrix} \quad \mathbf{G}(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \quad (4.4)$$

and calling set-point signals  $r_i$ , control inputs  $c_i$ , input signals  $u_i$  and output signals  $y_i$ , the process output  $\mathbf{y} \in \mathbb{R}^2$  is related to the process input  $\mathbf{u} \in \mathbb{R}^2$  through linear combinations of each input with the cross-channel transfer functions, as shown in Figure 4.3.

$$Y_1(s) = G_{11}(s)U_1(s) + G_{12}(s)U_2(s) \quad Y_2(s) = G_{21}(s)U_1(s) + G_{22}(s)U_2(s) \quad (4.5)$$

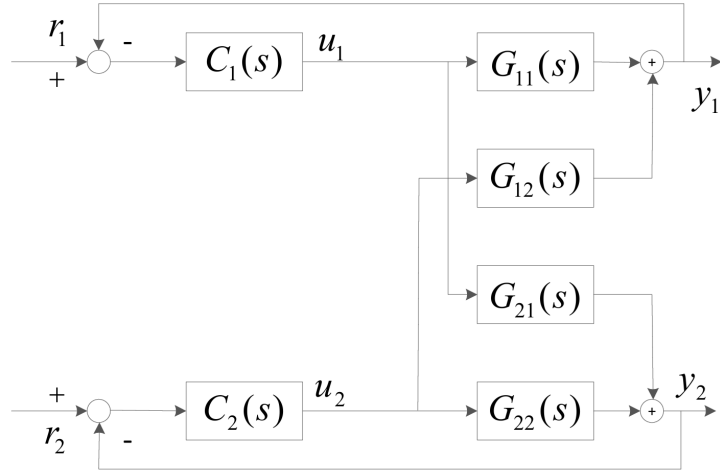
while the plant input vector  $\mathbf{u}$  in terms of the controller output  $\mathbf{c}$  is

$$U_1(s) = D_{11}(s)C_1(s) + D_{12}(s)C_2(s) \quad U_2(s) = D_{21}(s)C_1(s) + D_{22}(s)C_2(s) \quad (4.6)$$

Combining these two equations it is possible to derive the controller input  $\mathbf{c}$  to system output  $\mathbf{y}$  transfer functions

$$Y_1(s) = (G_{11}D_{11} + G_{12}D_{12})C_1 + (G_{11}D_{12} + G_{12}D_{22})C_2 \quad (4.7)$$

$$Y_2(s) = (G_{21}D_{11} + G_{22}D_{21})C_1 + (G_{21}D_{12} + G_{22}D_{22})C_2 \quad (4.8)$$



**Figure 4.3:** Decentralized Control structure of TITO system

## 4.2 Static Decoupling

Static decoupling approaches are the most computationally efficient techniques for reducing or eliminating interactions between multiple output variables in a linear MIMO system relying on the computation of a static gain matrix, calculated around a specified working frequency, that modifies the system's behavior to make it as diagonal as possible. This approach is particularly effective in systems where interactions remain constant over time, without significant dynamic variations, such as in steady-state industrial processes where interactions are well characterized and predictable [11].

The key advantage of static decoupling lies in its straightforward implementation, in fact being based solely on steady-state gains, it does not require extensive system modeling or real-time adaptation. This makes it particularly useful in applications where the system structure does not change significantly over time, and where simplicity and computational efficiency are prioritized. Many industrial processes, such as chemical reactors, distillation columns, and heat exchangers, employ static decoupling to improve control system performance by minimizing the coupling effects between control loops [11].

However, one of the major limitations of static decoupling is its inefficiency during transients. While it can effectively reduce steady-state interactions, it often fails to maintain decoupling dynamically with respect to different operational frequencies.

Furthermore, static decoupling can introduce undesired effects at high frequencies, potentially increasing interactions between control variables instead of eliminating them. So, this phenomenon is particularly problematic in control systems where fast responses are required, as the decoupling matrix may amplify high-frequency components, leading to oscillations or instability and thus static decoupling, when applied inappropriately, may exacerbate control system issues rather than resolve them [17].

The couplings are even more critical in systems modeled as Linear Parameter-Varying (LPV) systems, where the dynamic matrix varies significantly over time. Recent papers like [22] and [23] extended the use of decouplers to LPV systems, exploiting the static decoupling to devise a parameter-varying gain compensator that reduces cross-channel interactions at a frequency or even in the whole spectrum.

The following sections explore the applicability of two main static decoupling techniques. The first, known in the literature as static decoupling [12], is based on the realization of a pre-compensator while the second one, known as SVD decoupling [20], is more robust and involves both the design of a pre-compensator and a post-compensator.

### 4.2.1 Static Decoupling

The so-called static decoupling, as referred originally by papers [12], relies on the computation of a static gain matrix derived from the inversion of the system transfer function matrix, representing the most widely used pre-compensator for reducing the mutual input-output couplings in industrial MIMO plants. In detail, the static decoupling matrix is computed from the inverse of the system frequency transfer matrix, typically calculated at low-frequency as zero, or more in general at a specified value in the bandwidth like the resonant frequency, where a real approximation of the matrix is required [2].

Formally, a Linear Time-Invariant system of the form  $\mathbf{Y}(s) = \mathbf{G}(s)\mathbf{U}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{U}(s)$  is said to be statically decoupled in  $s = 0$  if it is stable and its static gain matrix  $\mathbf{G}(0)$  is diagonal and non-singular.

$$\mathbf{G}(0) = \lim_{s \rightarrow 0} \mathbf{G}(s) = \begin{bmatrix} g_{11} & 0 & \dots & 0 \\ 0 & g_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & g_{nn} \end{bmatrix} \quad (4.9)$$

To achieve static decoupling, a pre-compensator or static decoupling matrix  $\mathbf{D} \in \mathbb{R}^{n \times n}$  is introduced, which is chosen as the inverse of the steady-state gain matrix, evaluated at a certain angular frequency, typically  $\omega = 0$

$$\mathbf{D} \triangleq \mathbf{G}^{-1}(j\omega) \Big|_{\omega=0} = \mathbf{G}^{-1}(0) \quad (4.10)$$

Hence, the statically decoupled system is given by the transfer function matrix  $\mathbf{M}$

$$\mathbf{M} = \mathbf{G}(s)\mathbf{G}^{-1}(0) \quad (4.11)$$

If  $\mathbf{G}(s)$  behaves similarly to  $\mathbf{G}(0)$  in the desired frequency range, then it is approximately diagonal and with unitary magnitude

$$s \rightarrow 0 \implies \mathbf{G}(s)\mathbf{G}^{-1}(0) \approx \mathbf{I} \quad (4.12)$$

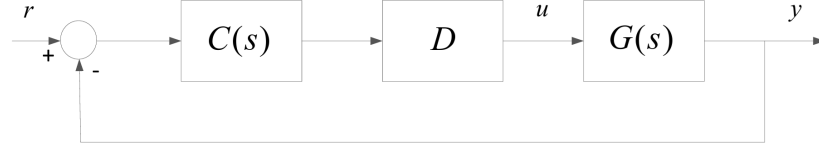
By applying this transformation and substituting  $\mathbf{D} = \mathbf{G}^{-1}(0)$ , the open-loop system dynamics from input to output is determined as

$$\mathbf{Y}(s) = \mathbf{M}(s)\mathbf{U}(s) = \mathbf{G}(s)\mathbf{D}\mathbf{U}(s) = \mathbf{G}(s)\mathbf{G}^{-1}(0)\mathbf{U}(s) \quad (4.13)$$

so that each input  $U_i(s)$  affects only the corresponding output  $Y_i(s)$ , achieving a diagonal transfer function matrix, only at low frequency.

The resulting closed-loop system, in Figure 4.4 obtained with a unitary feedback





**Figure 4.4:** Static Decoupling Control structure of MIMO system

and a controller  $\mathbf{C}(s)$  gets diagonalized at the chosen frequency, guaranteeing a decoupled control only in its neighborhood.

In a more general framework, it is possible to extend the inverting static decoupling, which is formulated for Linear Time-Invariant (LTI) models, to Linear Parameter-Varying (LPV) systems [22]. In cases of multivariable systems operating under variable conditions, where system parameters exhibit time-dependent variations, static decoupling can be formulated in a parameter-dependent manner to maintain effective interaction cancellation across different operating conditions. In fact, the transfer function matrix of the system,  $\mathbf{G}(s, \boldsymbol{\rho})$ , is even parameter-dependent, where the off-diagonal components represent the coupling effects that must be reduced. LPV static decoupling can be then achieved by introducing a parameter-dependent pre-compensator  $\mathbf{D}(\boldsymbol{\rho})$ , yielding a modified system

$$\mathbf{M}(s, \boldsymbol{\rho}) = \mathbf{G}(s, \boldsymbol{\rho})\mathbf{D}(\boldsymbol{\rho}) \quad (4.14)$$

where  $\mathbf{M}(s, \boldsymbol{\rho})$  is the transformed plant, expected to be diagonal or approximately diagonal. If  $\mathbf{G}(s, \boldsymbol{\rho})$  is invertible, a natural choice for the parameter-varying pre-compensator is, as before, the inverse itself which ensures perfect decoupling across all admissible parameter variations. However, full dynamic inversion is often impractical due to stability concerns and computational complexity, thus, the static inversion at a specified frequency, for example steady-state at  $\omega_0 = 0$  or cross-over frequency  $\omega_0 = \omega_c$ , is rather a more feasible approach

$$\mathbf{D}(\boldsymbol{\rho}) = \mathbf{G}^{-1}(j\omega_0, \boldsymbol{\rho}) \quad (4.15)$$

where a real approximation of the inverted matrix is required [2]. In detail, for non-square systems where  $\mathbf{G}(s, \boldsymbol{\rho})$  is not invertible, the Moore-Penrose pseudo-inverse can be employed, provided that the transfer matrix maintains full rank for all admissible values of  $\boldsymbol{\rho}$ . Moreover, by choosing an appropriate frequency  $\omega_0$ , the decoupling transformation remains effective in the relevant frequency range while maintaining stability, since only the DC gains are affected and not the poles of the transfer functions.

Finally, the input-output relation of the open-loop decoupled system is described by the following expression

$$\mathbf{Y}(s) = \mathbf{M}(s, \boldsymbol{\rho})\mathbf{U}(s) = \mathbf{G}(s, \boldsymbol{\rho})\mathbf{D}(\boldsymbol{\rho})\mathbf{U}(s) = \mathbf{G}(s, \boldsymbol{\rho})\mathbf{G}^{-1}(j\omega_0, \boldsymbol{\rho})\mathbf{U}(s) \quad (4.16)$$

The advantage of this LPV decoupling approach lies in its ability to adapt to system variations in real-time, unlike traditional static decoupling methods that assume a constant interaction structure.

However, static decoupling based on matrix inversion has serious limitations, particularly in systems with strong dynamic interactions. Since it is derived from steady-state conditions, its effectiveness diminishes in transient responses and at higher frequencies, where coupling effects change significantly, even in the extended LPV version, because of the fixed frequency dependence. In systems exhibiting strong resonances or right-half-plane poles, static decouplers can introduce instability or even exacerbate oscillations instead of mitigating them [17]. Nevertheless, their simplicity and ease of field tuning make them a practical solution for many industrial control applications [11].

### 4.2.2 SVD Decoupling

An alternative method for achieving static decoupling is based on the shaping of the open-loop system exploiting Singular Value Decomposition (SVD), relying on the factorization of the system's transfer matrix which allows for the reduction of cross-interactions between output variables [20]. The SVD is often used to evaluate interactions in a system with input-output cross couplings, and moreover can be exploited as a decoupling approach to improve the robustness of the control strategy adding a pre-compensator and a post-compensator to the open-loop system [2]. Given a LTI MIMO system with a transfer function matrix  $\mathbf{G}(s)$ , the SVD decomposition at a specific frequency  $\omega_0$  is expressed as

$$\mathbf{G}(j\omega_0) = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\dagger \quad (4.17)$$

where  $\mathbf{U} \in \mathbb{C}^{n \times n}$  and  $\mathbf{V} \in \mathbb{C}^{n \times n}$  are unitary matrices containing the left and right singular vectors, and  $\mathbf{\Sigma} = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$  is a diagonal matrix whose elements represent the system's singular values. Inverting this expression it is possible to make explicit  $\mathbf{\Sigma}$ , which represents a diagonalization of the transfer function matrix evaluated at angular frequency  $\omega_0$ .

$$\mathbf{\Sigma} = \mathbf{U}^{-1}\mathbf{G}(j\omega_0)(\mathbf{V}^\dagger)^{-1} \quad (4.18)$$

In order to achieve decoupling, it is possible to design pre- and post-compensators as the inverted right and left singular matrices, respectively

$$\mathbf{D}_1 \triangleq (\mathbf{V}^\dagger)^{-1} = \mathbf{V} \quad \mathbf{D}_2 \triangleq \mathbf{U}^{-1} = \mathbf{U}^\dagger \quad (4.19)$$

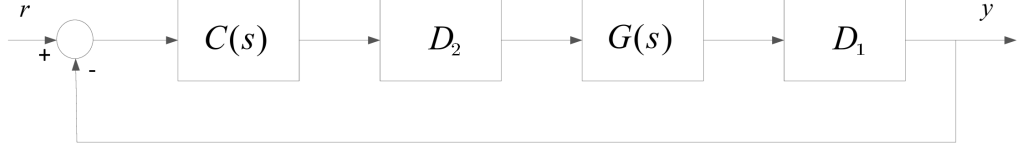
where, being unitary, the inverse of these matrices is equal to their transpose conjugate. In this way the open-loop system becomes

$$\mathbf{M}(s) = \mathbf{D}_2\mathbf{G}(s)\mathbf{D}_1 = \mathbf{U}^\dagger\mathbf{G}(s)\mathbf{V} \quad (4.20)$$

where the resulting decoupled system  $\mathbf{M}(s)$  exhibits a diagonal-like structure, effectively minimizing the coupling effects.

The SVD-based decoupling strategy results particularly effective for systems with strong couplings, as it systematically reshapes the system's interaction structure [2].

Thus it is possible to design a feedback control system with a diagonal controller, as shown in Figure 4.5, which affects only the specific output in a bandwidth around the working frequency. In the case of LPV systems, where the transfer function



**Figure 4.5:** SVD Decoupling Control structure of MIMO system

matrix depends on a time-varying parameter vector  $\boldsymbol{\rho}$ , the SVD decoupling is extended [22] as follows

$$\mathbf{G}(j\omega_0, \boldsymbol{\rho}) = \mathbf{U}(\boldsymbol{\rho})\boldsymbol{\Sigma}(\boldsymbol{\rho})\mathbf{V}^\dagger(\boldsymbol{\rho}) \quad (4.21)$$

with parameter-dependent compensators

$$\mathbf{D}_1(\boldsymbol{\rho}) = \mathbf{V}(\boldsymbol{\rho}) \quad \mathbf{D}_2(\boldsymbol{\rho}) = \mathbf{U}^\dagger(\boldsymbol{\rho}) \quad (4.22)$$

resulting in a transformed decoupled LPV system

$$\mathbf{M}(s, \boldsymbol{\rho}) = \mathbf{D}_2(\boldsymbol{\rho})\mathbf{G}(s, \boldsymbol{\rho})\mathbf{D}_1(\boldsymbol{\rho}) = \mathbf{U}^\dagger(\boldsymbol{\rho})\mathbf{G}(s, \boldsymbol{\rho})\mathbf{V}(\boldsymbol{\rho}) \quad (4.23)$$

Therefore, the output of the open-loop decoupled system can be expressed with respect to the input vector as

$$\mathbf{Y}(s) = \mathbf{M}(s, \boldsymbol{\rho})\mathbf{U}(s) = \mathbf{D}_2(\boldsymbol{\rho})\mathbf{G}(s, \boldsymbol{\rho})\mathbf{D}_1(\boldsymbol{\rho})\mathbf{U}(s) = \mathbf{U}^\dagger(\boldsymbol{\rho})\mathbf{G}(s, \boldsymbol{\rho})\mathbf{V}(\boldsymbol{\rho})\mathbf{U}(s) \quad (4.24)$$

However, for practical implementation, it is necessary to exploit real gain matrices [2], so a real approximation like the Align Algorithm [32] must be applied to the complex matrix before the computation of the Singular Value Decomposition, to remove the imaginary components while preserving the most orthogonality in directions [21].

Firstly, the transfer function matrix must be evaluated around a decoupling frequency  $\omega_0$

$$\mathbf{G}_0 = \mathbf{G}(j\omega) \Big|_{\omega=\omega_0} = \mathbf{G}(j\omega_0) \quad (4.25)$$

To compute a real approximation of  $\mathbf{G}_0$ , we first form the product of the complex matrix multiplied by itself  $\mathbf{G}_0^\dagger \mathbf{G}_0$ , where the crux represents the complex conjugate. Taking the real part of this product, we define the real matrix

$$\mathbf{D} = \left[ \text{Re}(\mathbf{G}_0^\dagger \mathbf{G}_0) \right]^{-1} \quad (4.26)$$

in which the inverse denotes, in general, the Moore-Penrose pseudo-inverse. Next, a phase correction is applied to the diagonal entries of  $\mathbf{G}_0$  by computing the diagonal matrix

$$\mathbf{\Lambda} = \text{diag} \left( e^{j \frac{1}{2} \angle \text{diag}(\mathbf{G}_0 \mathbf{D} \mathbf{G}_0^\top)} \right) \quad (4.27)$$

so that the updated real approximation of  $\mathbf{G}_0$  is obtained as

$$\mathbf{G}_0^{\text{real}} = \left[ \mathbf{D} \text{Re}(\mathbf{G}_0^\dagger \mathbf{\Lambda}) \right]^{-1} \quad (4.28)$$

Subsequently, the singular value decomposition of the real matrix  $\mathbf{G}_0^{\text{real}}$  is performed, resulting in a real matrix decomposition

$$\mathbf{G}_0^{\text{real}} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^\top \quad (4.29)$$

where  $\mathbf{U}$  and  $\mathbf{V}$  are orthogonal matrices containing the left and right singular vectors, respectively, and  $\mathbf{\Sigma}$  is the diagonal matrix of singular values.

As before, the obtained real matrices  $\mathbf{U}$  and  $\mathbf{V}$  are then employed to decouple the system. The decoupled LTI plant in this way is obtained defined by applying the real gain matrix  $\mathbf{V}$  as pre-compensator and the real static matrix  $\mathbf{U}^\top$  as post-compensator

$$\mathbf{M}(s) = \mathbf{U}^{-1} \mathbf{G}(s) (\mathbf{V}^{-1})^\top = \mathbf{U}^\top \mathbf{G}(s) \mathbf{V} \quad (4.30)$$

In the case of LPV system the SVD decoupler is computed by varying the parameter vector  $\boldsymbol{\rho}$ .

$$\mathbf{M}(s, \boldsymbol{\rho}) = \mathbf{U}(\boldsymbol{\rho})^{-1} \mathbf{G}(s, \boldsymbol{\rho}) [\mathbf{V}^{-1}(\boldsymbol{\rho})]^\top = \mathbf{U}(\boldsymbol{\rho})^\top \mathbf{G}(s, \boldsymbol{\rho}) \mathbf{V}(\boldsymbol{\rho}) \quad (4.31)$$

While SVD-based decoupling offers a systematic approach to reducing interaction effects, its practical implementation requires real-time computation of the singular value decomposition, especially in high-dimensional systems. However, when computational resources allow, this method provides a more robust approach for improving control performance in Multi-Input Multi-Output parameter-varying systems. Anyway, the static SVD Decoupling does not guarantee a cross input-output inhibition far away the chosen bandwidth, because of the limit imposed by the evaluation of the matrix transfer function on the specified frequency, resulting in inadequate decoupling performance when input signals at very different frequencies are provided.

## 4.3 Dynamic Decoupling

Dynamic decoupling approaches aim to overcome the restrictions of static decoupling, based on constant gain matrices, by designing a frequency-variant matrix which provides dynamic compensations that diagonalizes the system's dynamics, since they account for time-varying cross couplings beyond steady-state conditions, and therefore improving control accuracy [11].

However, it is important to notice that a dynamic approach requires an accurate process model, as the design of frequency-dependent compensators is sensitive to system dynamics [12]. Another main drawback of dynamic decoupling is its increased complexity with respect to the static approaches, leading to higher computational demands and implementation issues in real-time control.

Despite these challenges, dynamic decoupling remains a better approach for enhancing control precision in multivariable systems. In the following sections, Ideal, Simplified and Inverted dynamic decoupling techniques are explored, born in Linear Time-Variant framework and later extended, as purpose of this thesis, to Linear Parameter-Varying systems, by analyzing their specific properties and limitations.

The final aim is electing the best dynamic decoupling model for a robotic system, which is not a simple task since each technique has its own intrinsic characteristics: the Ideal Decoupling, rarely used in industrial plants, allows to keep the same tuning of the original controllers at the expense of robustness; the Simplified Decoupling, the most implemented type of decoupler in industry, which exploits a simplified compensation but altering the diagonal transfer function; the Inverted Decoupling, which represents the best trade-off among keeping the same diagonal dynamics, perfect diagonalization and ease of implementation.

For conceptual simplification the analysis are deepened for Two-Input Two-Output (TITO) systems, which is not the most general case but it covers most of actual industrial applications. In particular, this choice is related also to the experimental validations on a 2-axis robot manipulator model, reported in the final chapter.

### 4.3.1 Ideal Decoupling

Ideal Decoupling represents the generalization of the Static Decoupling approach, based on the design of a open-loop regulator by exploiting a complete inversion of the system's transfer matrix [12]. The goal is to make the original system behave as closely as possible to a completely diagonalized system for each possible operating frequency, so that each input affects only the corresponding output.

Formally, a MIMO LTI system  $\mathbf{Y}(s) = \mathbf{G}(s)\mathbf{U}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{U}(s)$  with  $n$  inputs  $u_i$  and  $n$  outputs  $y_i$ , is said to be dynamically decoupled if it is stable and its transfer function matrix  $\mathbf{G}(s)$  is diagonal and non-singular for each frequency.

Let us consider the TITO system, as defined in the previous section. If the system  $\mathbf{G}(s)$  is effectively decoupled, the transfer function matrix which represents the decoupled system must be diagonal.

$$\mathbf{M}(s) = \mathbf{G}(s)\mathbf{D}(s) \stackrel{!}{=} \text{diag}(M_{11}, M_{22}) \quad (4.32)$$

This condition implies, conversely, that the decoupling matrix  $\mathbf{D}(s)$  is given by

$$\mathbf{D}(s) = \mathbf{G}^{-1}(s)\mathbf{M}(s) \quad (4.33)$$

Consequently, making explicit the inverse of  $\mathbf{G}$  and by executing the matrix multiplication with the diagonal matrix  $\mathbf{M}$ , the decoupling matrix  $\mathbf{D}$  takes the following analytical form

$$\mathbf{D}(s) \triangleq \frac{1}{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)} \begin{bmatrix} G_{22}(s)M_{11}(s) & -G_{12}(s)M_{22}(s) \\ -G_{21}(s)M_{11}(s) & G_{11}(s)M_{22}(s) \end{bmatrix} \quad (4.34)$$

The first decoupling control design consists of choosing the diagonal transfer function to impose as behaviour. Selecting the diagonal dynamics of the system is the most natural choice: this is the case of Ideal Decoupling, where the diagonalized plant  $\mathbf{M}(s)$  is achieved by imposing the on-diagonal transfer functions of  $\mathbf{G}(s)$

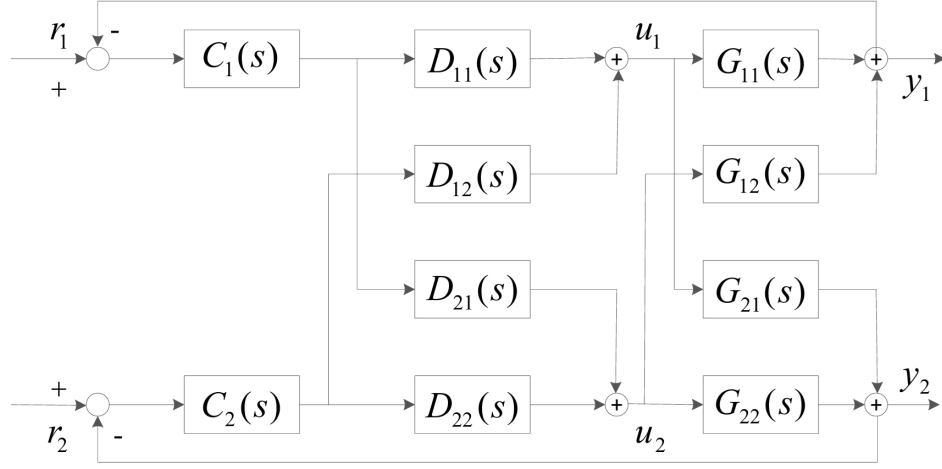
$$M_{11}(s) \stackrel{!}{=} G_{11}(s) \quad M_{22}(s) \stackrel{!}{=} G_{22}(s) \quad (4.35)$$

so that the Ideal Decoupler is analytically expressed by

$$\mathbf{D}(s) = \frac{1}{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)} \begin{bmatrix} G_{22}(s)G_{11}(s) & -G_{12}(s)G_{22}(s) \\ -G_{21}(s)G_{11}(s) & G_{11}(s)G_{22}(s) \end{bmatrix} \quad (4.36)$$

In this way, closing a unitary feedback as in Figure 4.6, the diagonal controller functions  $C_1(s)$  and  $C_2(s)$  can be designed and tuned independently with respect to  $M_{11}(s)$  and  $M_{22}(s)$ .

So the main advantage of ideal decoupling is that, once applied, the controller  $\mathbf{C}(s)$  does not need to be retuned, even if different loops operate in different modes [16]. However, ideal decoupling has several practical issues from theoretical and practical sides. The main challenge is the feasibility of inverting the transfer matrix, which often results in a sum of transfer functions, leading to complex high-order expressions, which can provide improper functions, thus not physically realizable, and even including possible unstable pole-zero cancellations, making the overall system unstable [16]. Moreover, since the decoupler is computed from the exact knowledge of the original system, entirely depending on the accuracy of  $\mathbf{G}(s)$ , ideal decoupling is highly sensitive to modeling errors and process uncertainties. Thus even small model uncertainties can degrade performance, making the approach unreliable in real-world applications, therefore it is rarely used in industrial applications [11].



**Figure 4.6:** Ideal Decoupling Control structure of TITO system

### 4.3.2 Simplified Decoupling

Simplified Decoupling is an alternative design of decoupling control which seeks to reduce the complexity of ideal decoupling by eliminating certain terms in the decoupling matrix. Specifically, it assumes that some interactions are negligible and thus removes complex dynamics from the compensation matrix, therefore it makes decoupling easier to implement, at the expense of modifications on the direct transfer functions.

A first Simplified Decoupling strategy [12] can be obtained by exploiting the interactions among the process dynamics and the decoupling system. The decoupling matrix transfer function  $\mathbf{D}(s)$  must be designed such that the overall system between the control signal and the output vector signal, said  $\mathbf{M}(s)$  is diagonal.

$$\mathbf{M}(s) = \mathbf{G}(s)\mathbf{D}(s) \quad (4.37)$$

Considering the Two-Input Two-Output system  $\mathbf{G}(s) \in \mathbb{C}^{2 \times 2}$  with a decoupler  $\mathbf{D}(s) \in \mathbb{C}^{2 \times 2}$  as described in the chapter introduction.

$$\begin{bmatrix} M_{11}(s) & M_{12}(s) \\ M_{21}(s) & M_{22}(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} D_{11}(s) & D_{12}(s) \\ D_{21}(s) & D_{22}(s) \end{bmatrix} \quad (4.38)$$

In order to make this product diagonal, so achieving decoupling, it is necessary to impose that the off-diagonal terms are zero.

$$\begin{cases} M_{12} \stackrel{!}{=} 0 \\ M_{21} \stackrel{!}{=} 0 \end{cases} \quad \begin{cases} G_{11}(s)D_{12}(s) + G_{12}(s)D_{22}(s) = 0 \\ G_{21}(s)D_{11}(s) + G_{22}(s)D_{21}(s) = 0 \end{cases} \quad (4.39)$$

The problem is an undetermined linear system, since there are two equations in four unknowns  $D_{ij}(s)$ . One simple solution is arbitrarily assuming the on-diagonal transfer functions as unitary gains

$$D_{11}(s) \stackrel{!}{=} 1, \quad D_{22}(s) \stackrel{!}{=} 1 \implies D_{12}(s) = -\frac{G_{11}(s)}{G_{12}(s)}, \quad D_{21}(s) = -\frac{G_{21}(s)}{G_{22}(s)} \quad (4.40)$$

Thus the Simplified Decoupling structure for a Two-Input Two-Output (TITO) system can be expressed with the following decoupling matrix

$$\mathbf{D}(s) = \begin{bmatrix} 1 & -G_{12}(s)/G_{11}(s) \\ -G_{21}(s)/G_{22}(s) & 1 \end{bmatrix} \quad (4.41)$$

With this choice, the controlled system becomes

$$\begin{aligned} \mathbf{M}(s) &= \mathbf{G}(s)\mathbf{D}(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} 1 & -G_{11}(s)/G_{12}(s) \\ -G_{21}(s)/G_{22}(s) & 1 \end{bmatrix} \\ &= \begin{bmatrix} G_{11}(s) - \frac{G_{12}(s)G_{21}(s)}{G_{22}(s)} & 0 \\ 0 & G_{22}(s) - \frac{G_{21}(s)G_{12}(s)}{G_{11}(s)} \end{bmatrix} \equiv \begin{bmatrix} \frac{\det(\mathbf{G}(s))}{G_{22}(s)} & 0 \\ 0 & \frac{\det(\mathbf{G}(s))}{G_{11}(s)} \end{bmatrix} \end{aligned} \quad (4.42)$$

which is finally a diagonal transfer matrix.

Unlike ideal decoupling, which requires computing and inverting the full transfer function matrix, simplified decoupling only requires two compensators, significantly reducing the implementation complexity. Additionally, alternative formulations of the Simplified decoupling matrix have been proposed [18], obtained by setting two elements in different columns of  $\mathbf{D}(s)$  into unitary gains, as follows

$$\mathbf{D}(s) = \begin{bmatrix} -G_{21}(s)/G_{22}(s) & 1 \\ 1 & -G_{11}(s)/G_{12}(s) \end{bmatrix} \quad (4.43)$$

$$\mathbf{D}(s) = \begin{bmatrix} -G_{22}(s)/G_{21}(s) & -G_{12}(s)/G_{11}(s) \\ 1 & 1 \end{bmatrix} \quad (4.44)$$

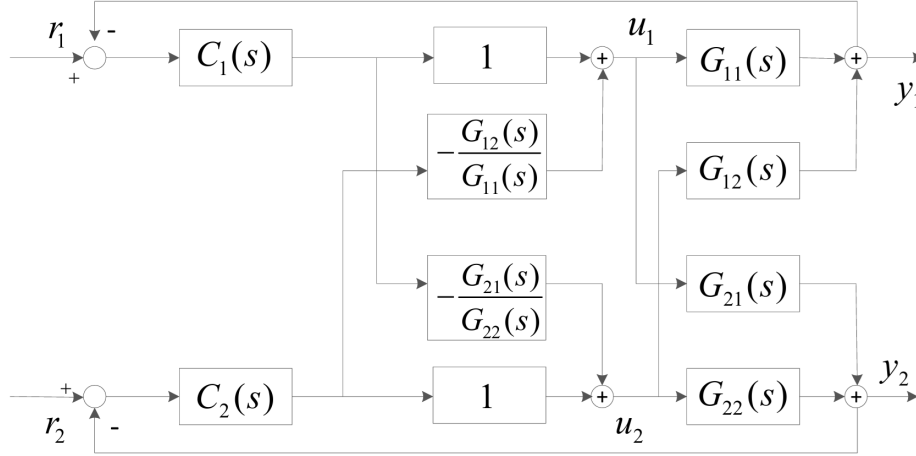
$$\mathbf{D}(s) = \begin{bmatrix} -G_{21}(s)/G_{22}(s) & -G_{11}(s)/G_{12}(s) \\ 1 & 1 \end{bmatrix} \quad (4.45)$$

These variations allow for practical flexibility in implementation, reducing the computational burden associated with ideal decoupling.

Designing the remaining part of the controller with a diagonal transfer function  $\mathbf{C}(s)$ , the open-loop transfer function

$$\mathbf{G}(s)\mathbf{D}(s)\mathbf{C}(s) \quad (4.46)$$





**Figure 4.7:** Simplified Decoupling Control structure of TITO system

becomes diagonal, and so the closed-loop system transfer function from  $\mathbf{r}$  to  $\mathbf{y}$ , whose diagram is shown in Figure 4.7, is also diagonal, thereby satisfying the previously stated decoupling conditions. In other words, the controller synthesis can be brought back to the design of  $n = 2$  independent regulators.

The main advantage of Simplified decoupling is its ease of implementation, in fact the resulting decoupler structure is straightforward, requiring fewer compensators and reducing computational complexity, making it particularly suitable for industrial applications where full ideal decoupling may not be feasible.

However, Simplified decoupling introduces some key limitations: since the decoupling matrix is not derived from a full system inversion, the apparent dynamics of the system changes, often requiring different controller tuning when switching between decoupled and non-decoupled modes (manual). Another challenge with simplified decoupling is the presence of summation terms in the compensators, resulting in higher order transfer functions. This may complicate controller tuning and, particularly when dealing with high-dimensional processes, simplified decoupling may still exhibit some of the same issues as ideal decoupling, such as sensitivity to modeling errors and the need for accurate transfer function estimation.

Despite its problems, simplified decoupling remains a widely used technique due to the fact that it generally provides an effective trade-off between ideal decoupling's theoretical performance and a realizable, lower-complexity solution.

### 4.3.3 Inverted Decoupling

Inverted Decoupling represents an alternative solution that combines the advantages of ideal, deriving the same decoupled process, and simplified decoupling, such that the decoupling functions are not so complex [14]. Instead of computing a conventional decoupling matrix through full matrix inversion, this method consists of designing a modified decoupling structure starting from the Simplified decoupler, which introduces a control strategy based on feedforward and feedback compensation aiming to decouple the multivariable system while maintaining a simple and practical implementation. The basic concept behind inverted decoupling is to construct the process input signals as a weighted combination of one controller output and the other process input signals, rather than using a fully inverted transfer function matrix. Each process input is viewed as a disturbance to the other process output, so a compensation for these internal disturbances can be designed exploiting a feedforward control approach [15].

In detail, the Inverted decoupler is theoretically derived from the Ideal Decoupling and it is implemented as a modified version of the Simplified Decoupling structure, where the compensation elements are kept the same, but the signal direction through the decoupling transfer functions is inverted and the location of the summing junction is moved to realize the compensation [15].

For a Two-Input Two-Output system, in order to obtain the Inverted Decoupler, it is possible to recall the Ideal Decoupler which is

$$\mathbf{D}(s) = \frac{1}{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)} \begin{bmatrix} G_{22}(s)G_{11}(s) & -G_{12}(s)G_{22}(s) \\ -G_{21}(s)G_{11}(s) & G_{11}(s)G_{22}(s) \end{bmatrix} \quad (4.47)$$

so that the control input process signals can be expressed as

$$U_1(s) = C_1(s) \left[ \frac{G_{11}(s)G_{22}(s)}{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)} \right] - C_2(s) \left[ \frac{G_{12}(s)G_{22}(s)}{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)} \right] \quad (4.48)$$

$$U_2(s) = -C_1(s) \left[ \frac{G_{21}(s)G_{11}(s)}{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)} \right] + C_2(s) \left[ \frac{G_{11}(s)G_{22}(s)}{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)} \right] \quad (4.49)$$

It is possible to demonstrate that these equations can be expressed also in a simplified form. Firstly, we can write equations (4.48) and (4.49) as

$$U_1(s) \left[ \frac{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right] = C_1(s) - C_2(s) \frac{G_{12}(s)}{G_{11}(s)} \quad (4.50)$$

$$U_2(s) \left[ \frac{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right] = C_2(s) - C_1(s) \frac{G_{21}(s)}{G_{22}(s)} \quad (4.51)$$

From each equation we obtain the control output signals, respectively

$$C_1(s) = U_1(s) \left[ \frac{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right] + C_2(s) \frac{G_{12}(s)}{G_{11}(s)} \quad (4.52)$$

$$C_2(s) = U_2(s) \left[ \frac{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right] + C_1(s) \frac{G_{21}(s)}{G_{22}(s)} \quad (4.53)$$

Substituting the expression of  $C_2$  (4.53) into the first equation (4.50)

$$\begin{aligned} U_1(s) \left[ G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s) \right] \frac{1}{G_{11}(s)G_{22}(s)} &= C_1(s) - \\ -U_2(s) \frac{G_{12}(s)}{G_{11}(s)} \left[ \frac{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right] &- C_1(s) \frac{G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \end{aligned} \quad (4.54)$$

And similarly, substituting  $C_1$  (4.52) into the second equation (4.51)

$$\begin{aligned} U_2(s) \left[ G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s) \right] \frac{1}{G_{11}(s)G_{22}(s)} &= C_2(s) - \\ -U_1(s) \frac{G_{21}(s)}{G_{22}(s)} \left[ \frac{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right] &- C_2(s) \frac{G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \end{aligned} \quad (4.55)$$

Therefore, factoring out  $C_1$ , the first equation (4.54) becomes

$$\begin{aligned} U_1(s) \left[ G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s) \right] \frac{1}{G_{11}(s)G_{22}(s)} &= \\ = C_1(s) \left[ G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s) \right] \frac{1}{G_{11}(s)G_{22}(s)} &- \\ -U_2(s) \frac{G_{12}(s)}{G_{11}(s)} \left[ \frac{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right] \end{aligned} \quad (4.56)$$

Factoring out  $C_2$  in the second equation (4.55) leads to

$$\begin{aligned} U_2(s) \left[ G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s) \right] \frac{1}{G_{11}(s)G_{22}(s)} &= \\ = C_2(s) \left[ G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s) \right] \frac{1}{G_{11}(s)G_{22}(s)} &- \\ -U_1(s) \frac{G_{21}(s)}{G_{22}(s)} \left[ \frac{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right] \end{aligned} \quad (4.57)$$

Finally, simplifying (4.56) and (4.57), the process input variables are

$$U_1(s) = C_1(s) - \frac{G_{12}(s)}{G_{11}(s)} U_2(s) \quad (4.58)$$

$$U_2(s) = C_2(s) - \frac{G_{21}(s)}{G_{22}(s)} U_1(s) \quad (4.59)$$

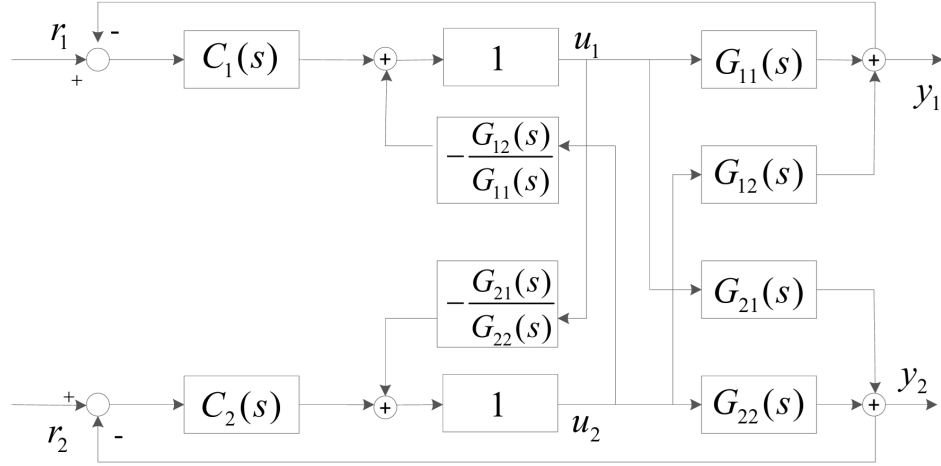
In other words, each process input is obtained as the relative control output corrected by a feedforward compensation coming from the opposite input channel, directly proportional to the off-diagonal transfer function and inversely proportional to the diagonal dynamics. These latter coefficients correspond exactly to the off-diagonal elements of the Simplified Decoupling matrix, thus the Inverted Decoupler is practically obtained from the Simplified Decoupling matrix where

$$D_{12}(s) \equiv -\frac{G_{12}(s)}{G_{11}(s)} \quad D_{21}(s) \equiv -\frac{G_{21}(s)}{G_{22}(s)} \quad (4.60)$$

by inverting the direction of input signals and summing the resulting compensation on the control outputs as

$$U_1(s) = C_1(s) + D_{12}U_2(s) \quad U_2(s) = C_2(s) + D_{21}U_1(s) \quad (4.61)$$

In this way, the feedforward-based formulation ensures that the control signals dynamically compensate for cross-channel interactions before being applied to the plant, according to the scheme in Figure 4.8.



**Figure 4.8:** Inverted Decoupling Control structure of TITO system

Furthermore, in order to derive analytically the Inverted Decoupler transfer function let us explicit the decoupling system in matrix form

$$\begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} = \begin{bmatrix} C_1(s) \\ C_2(s) \end{bmatrix} + \begin{bmatrix} 0 & -G_{12}(s)/G_{11}(s) \\ -G_{21}(s)/G_{22}(s) & 0 \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} \quad (4.62)$$

Rearranging the terms in the matrix equation

$$\begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} - \begin{bmatrix} 0 & -G_{12}(s)/G_{11}(s) \\ -G_{21}(s)/G_{22}(s) & 0 \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} = \begin{bmatrix} C_1(s) \\ C_2(s) \end{bmatrix} \quad (4.63)$$

Factoring out the input vector

$$\begin{bmatrix} 1 & G_{12}(s)/G_{11}(s) \\ G_{21}(s)/G_{22}(s) & 1 \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} = \begin{bmatrix} C_1(s) \\ C_2(s) \end{bmatrix} \quad (4.64)$$

let us note that the left matrix is similar to the Simplified Decoupler.

$$\begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} = \begin{bmatrix} 1 & G_{12}(s)/G_{11}(s) \\ G_{21}(s)/G_{22}(s) & 1 \end{bmatrix}^{-1} \begin{bmatrix} C_1(s) \\ C_2(s) \end{bmatrix} \quad (4.65)$$

This expression represents the dynamics of the input process given the control signals as input, through the transfer function matrix  $\mathbf{D}(s)$ , assuming that the argument matrix is full rank,

$$\mathbf{D}(s) = \begin{bmatrix} 1 & G_{12}(s)/G_{11}(s) \\ G_{21}(s)/G_{22}(s) & 1 \end{bmatrix}^{-1} \quad (4.66)$$

Therefore, the Inverted Decoupling matrix can be computed as the inverse of the Simplified Decoupling matrix with off-diagonal coefficients of opposite signs, from which the name of the approach itself is derived, and it may be further expressed as

$$\begin{aligned} \mathbf{D}(s) &= \frac{1}{1 - \frac{G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)}} \begin{bmatrix} 1 & -G_{12}(s)/G_{11}(s) \\ -G_{21}(s)/G_{22}(s) & 1 \end{bmatrix} = \\ &= \frac{1}{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)} \begin{bmatrix} G_{11}(s)G_{22}(s) & -G_{12}(s)G_{22}(s) \\ -G_{11}(s)G_{21}(s) & G_{11}(s)G_{22}(s) \end{bmatrix} \end{aligned} \quad (4.67)$$

which is exactly equivalent, from a theoretical point of view, to the Ideal Decoupler, as expected. While from a practical side, the Inverted decoupling system can be implemented in a more convenient form, rather than requiring a dynamic inversion, where the feedforward torque from the other channels through the "simplified" transfer functions allows the diagonalization of the overall plant.

In fact, it is straightforward to prove that the open-loop system  $\mathbf{M}(s)$  gets decoupled:

$$\begin{aligned} \mathbf{M}(s) &= \mathbf{G}(s) \mathbf{D}(s) = \\ &= \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \frac{1}{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)} \begin{bmatrix} G_{11}(s)G_{22}(s) & -G_{12}(s)G_{22}(s) \\ -G_{11}(s)G_{21}(s) & G_{11}(s)G_{22}(s) \end{bmatrix} \end{aligned} \quad (4.68)$$

$$\begin{aligned}
 &= \begin{bmatrix} \frac{G_{11}(s)G_{11}(s)G_{22}(s)-G_{12}(s)G_{11}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)-G_{12}(s)G_{21}(s)} & 0 \\ 0 & \frac{-G_{12}(s)G_{21}(s)G_{22}(s)+G_{11}(s)G_{22}(s)G_{22}(s)}{G_{11}(s)G_{22}(s)-G_{12}(s)G_{21}(s)} \end{bmatrix} \\
 &= \begin{bmatrix} G_{11}(s) \frac{G_{11}(s)G_{22}(s)-G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)-G_{12}(s)G_{21}(s)} & 0 \\ 0 & G_{22}(s) \frac{G_{11}(s)G_{22}(s)-G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)-G_{12}(s)G_{21}(s)} \end{bmatrix} = \begin{bmatrix} G_{11}(s) & 0 \\ 0 & G_{22}(s) \end{bmatrix} \quad (4.69)
 \end{aligned}$$

Inverted decoupling offers several benefits, including simplifying control design by maintaining the apparent process as in the absence of decoupling, similar to the Ideal case [15]. It also uses a straightforward decoupling structure, similarly to the Simplified technique, and is less affected by actuator saturation and disturbances, leading to more stable performance in practice [11].

However, it has some drawbacks similar to the Ideal strategy, as it cannot handle transfer matrices with right-half-plane zeros, because these introduce unstable poles in the decoupling network, and, additionally, it is still sensitive to modeling errors so that a small variation of the parameters can compromise performance [15].

Essentially, the Inverted Decoupling combines its ease of implementation, with its effectiveness in reducing process interactions, which makes it a preferred choice for many industrial applications, when the process is known with sufficient accuracy.

## Chapter 5

# Simulations and Results

In this chapter, we analyze the performance of the linear decoupling control strategies by evaluating their ability to suppress the mutual couplings among the robot links. Simulations are conducted by employing different mathematical systems derived from the Comau robot model, starting with the LTI approximation around a specified configuration, extending then to the LPV model, and finally generalizing to the nonlinear model.

Firstly, to assess the most effective decoupling strategies for the plant, the LTI open-loop transfer function matrix between the motor torque input  $\tau_m$  and the outputs  $\dot{\theta}$ ,  $\theta$ ,  $\dot{\mathbf{q}}$ , and  $\mathbf{q}$  are examined, with a particular focus on the joint position. The aim is to filter the best techniques, in terms of accuracy and robustness, between all the proposed decoupling compensators: Static, SVD, Ideal, Simplified and Inverted decouplers. The interaction analyses, on the basis of Bode plots, RGA, and Gershgorin Radii, quantify the degree of decoupling and highlight the conditions under which each method performs optimally.

In order to evaluate the output response of the proposed decoupling strategies, extensive simulations are performed, specifically designing a controller for each open-loop diagonal transfer function. The step response is primarily employed to verify the regulation performance, measured through metrics such as overshoot, settling time, and tracking accuracy. All these metrics are used to compare the approaches, with Simplified and Inverted Decoupling emerging as the best trade-offs between coupling reduction and implementation complexity.

After selecting the best LTI decoupling strategies, further analyses explore the application of decoupling strategies to the LPV model based on the variable transfer function matrix between  $\tau_m$  and  $\mathbf{q}$ , proving that LTI-based decoupling is effective only when the system dynamics remain nearly constant, while LPV decoupling techniques offer greater adaptability to the plant, ensuring better performance.

And finally, the two LPV decouplers are applied to the nonlinear robot, proving to be a surprisingly effective and efficient solution to solve the coupling problem.

## 5.1 Robot Dynamics Simulation

The simulative section begins with the introduction of the theoretical model employed by Comau for the industrial simulations, as proposed in the internal report [33]. Building on this foundation, we proceed to implement the model in MATLAB, enabling numerical simulations, setting the environment on which further analysis will be developed.

### 5.1.1 Comau Robot Manipulator Model

Comau employs the Elastic Joint Model (EJM) [7] to characterize the dynamics of robotic manipulators, following the formalism introduced by Spong [4], which can be traced back to a particular case of the elastic-dumped joint robot dynamics explained in Section 2.1.5. This approach considers a manipulator composed of  $n$  joints, either prismatic or revolute, connected to  $n$  rigid links, incorporating the effects of elasticity in the transmissions and connections. The links are driven by electric actuators through power transmission systems, each defined by a specific reduction ratio.

The torsional linear springs represent both the concentrated elasticity of the gear reducers and the distributed elasticity of real links, which causes bending in the plane of rotation of the links. As a result, the generalized coordinate  $q_i$  represents the position of the center of mass of the  $i$ -th link. In reality, due to the curvature of the non-rigid body caused by bending, it is not possible to define a single generalized coordinate that uniquely determines the position of the link in the plane of rotation. Therefore to simplify the analysis, the link is modeled as rigid, and its position is defined by the assumed position of its center of mass where the bending effects of each real flexible link are approximated using the joint springs, indeed.

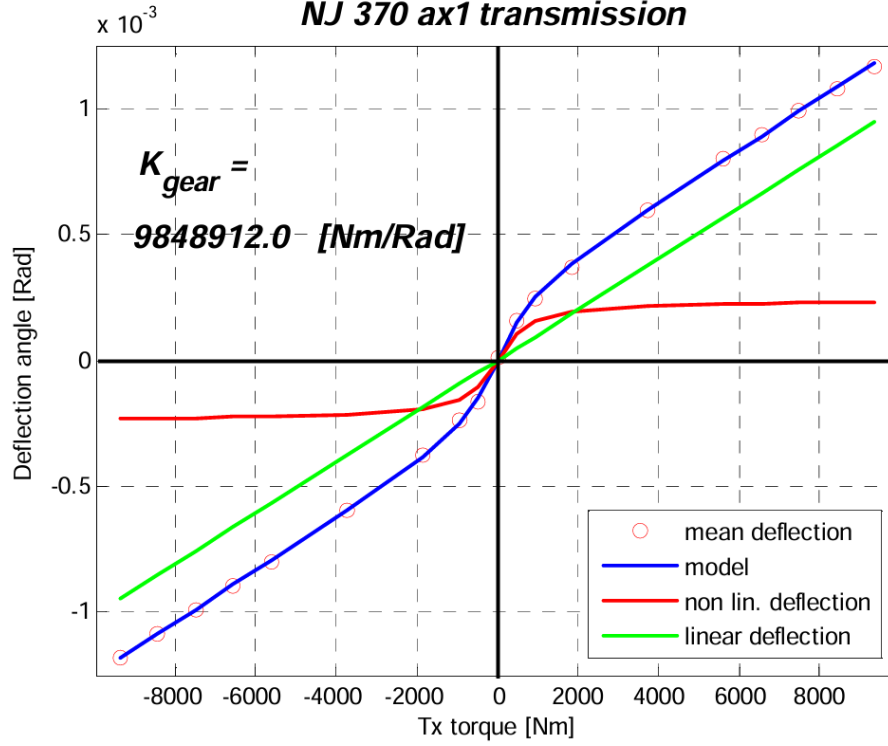
This abstraction allows for a first-order approximation of the real dynamics using a model of a rigid link manipulator with elastic joints. It consists of a set of three interconnected matrix equations that generalize, to the multivariable case, the dynamics of a single independent elastic joint model, where two inertias ( $\mathcal{I}_m$  for the motor and  $\mathcal{I}_l$  for the link) are connected by a non-rigid reducer. Notably, since the vector  $\tau_l$  represents the generalized forces applied to the links by the motor through the gearbox, it is common to all three equations,

$$\begin{cases} \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}_l(\dot{\mathbf{q}}) + \mathbf{D}_l\dot{\mathbf{q}} = \boldsymbol{\tau}_l \\ \boldsymbol{\tau}_l = \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) + \mathbf{D}_{ml}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) \\ \mathbf{B}\ddot{\boldsymbol{\theta}} + \mathbf{D}_m\dot{\boldsymbol{\theta}} + \mathbf{f}_m(\dot{\boldsymbol{\theta}}) + \mathbf{K}_r^{-1}\boldsymbol{\tau}_l = \mathbf{K}_t\mathbf{I}_m \\ \boldsymbol{\theta} = \mathbf{K}_r^{-1}\boldsymbol{\theta}_m \end{cases} \quad (5.1)$$



- The first equation represents the dynamic model of a rigid-body chain, including inertial terms  $\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}$ , Coriolis effects  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}$ , and gravitational forces  $\mathbf{g}(\mathbf{q})$ . Dissipative forces are explicitly modeled as linear  $\mathbf{D}_l\dot{\mathbf{q}}$  and nonlinear  $\mathbf{f}_l(\dot{\mathbf{q}})$  terms.
- The second equation represents the elastic gear transmission where generalized forces  $\boldsymbol{\tau}_l$  act on the rigid links via elastic deformation according to Hooke's Law: the elastic components are modeled by a diagonal stiffness matrix representing concentrated linear springs, aligned with the motion axes of the joints.
- The third equation representing the motor mechanical model, expresses the equilibrium between motor-generated forces and  $\boldsymbol{\tau}_l$ , transmitted through reducers, accounting for inertial, dissipative, and actuation forces. Notice that gyroscopic coupling between motors and links is neglected, a valid assumption for high transmission ratios typical of industrial manipulators, so we can impose  $\mathbf{S} = \mathbf{0}_{n \times n}$ . Diagonal matrices  $\mathbf{B}$ ,  $\mathbf{D}_m$ ,  $\mathbf{K}_t$ , and  $\mathbf{K}_r$  represent motor inertias  $\mathcal{I}_{m_i}$ , frictions  $D_{m_i}$ , torque constants  $K_{t_i}$ , and gear ratios  $K_{r_i}$ , respectively. The simplified torque-current relationship  $\tau_{m_i} = K_{t_i}I_{m_i}$  assumes negligible electrical dynamics and nonlinearity effects, as  $K_{t_i}$  in reality depends on the current reference  $I_{m_i}$ . This leads to a linear approximation obtained closing the internal feedback current loop inside the motor that is reasonable, because the closed-loop torque control bandwidth typically exceeds the frequency range of the mechanical dynamics.
- The fourth equation accounts for the rigid gear reduction ratio.

The torsion angle can be decomposed into a linear component, dependent on the applied torque, and a nonlinear component, derived from backlash and compliance, arising from gearbox elasticity, as shown in Figure 5.1. In the EJM model, linear springs  $K_i$ , represented by the diagonal matrix  $\mathbf{K}$ , capture the linear elastic deformations of the system, whose derivatives w.r.t. the transmitted torque contribute to define the elastic constants. Nonlinear deformations are primarily present in the gearboxes, while flexible link deformations behave predominantly as linear functions of the transmitted forces and also contribute to the definition of  $K_i$ . Neglecting the nonlinear components reduces the model's realism but does not significantly affect its ability to reproduce the manipulator's modal behavior [33]. For what regards friction, the damping is composed of both a linear component, i.e., the viscous friction, and a non-linear component with respect to velocity, which can be assumed as static/Coulomb friction, or Stribeck friction. In general, both links and motors are singularly affected by direct damping  $\mathbf{D}_x\mathbf{x} + \mathbf{f}_x(\dot{\mathbf{x}})$ , and moreover, a relative spring-damper torque is mutually considered in the transmission, where the friction is assumed just linear with respect to the relative deflection between each motor and the actuated link, with  $\mathbf{D}_{ml}$  collecting damping coefficients. In

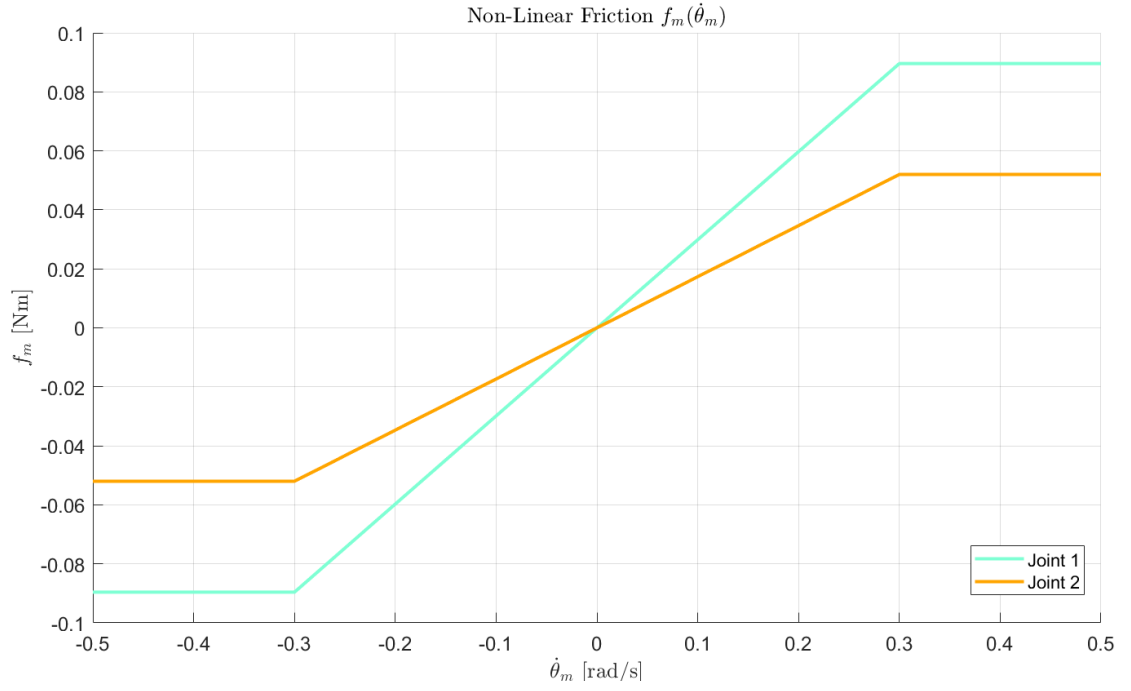


**Figure 5.1:** Elasticity behaviour in terms of deflection angle with respect to torque

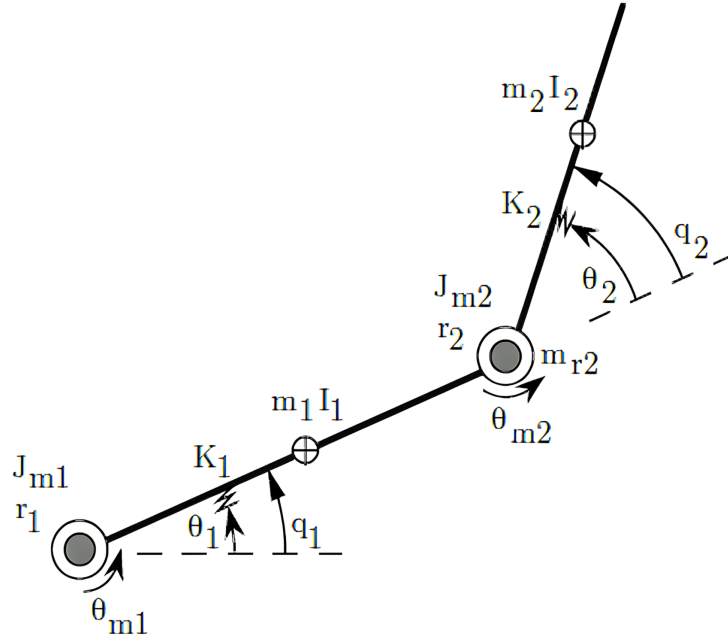
detail, to keep a balanced trade-off between simplicity and realism, it is possible to obtain a good approximation of the non-linear friction function, which is generally quite complex, by exploiting a piece-wise function as depicted in Figure 5.2. This type of friction, referred to as linear viscous friction with saturation, combines a linear increase with velocity  $\dot{\theta}_m$  at low speeds, typical of viscous friction, with a saturation effect at higher velocities, limiting the friction to a maximum static value, collected in the vector  $\mathbf{f}_m^{\max}$ .

$$f_{m_i}(\dot{\theta}_{m_i}) = \begin{cases} -f_{m_i}^{\max} & \dot{\theta}_{m_i} < -\dot{\theta}_{m_i}^{\max} \\ \frac{f_{m_i}^{\max}}{\dot{\theta}_{m_i}^{\max}} \dot{\theta}_{m_i} & -\dot{\theta}_{m_i}^{\max} \leq \dot{\theta}_{m_i} \leq \dot{\theta}_{m_i}^{\max} \\ f_{m_i}^{\max} & \dot{\theta}_{m_i} > \dot{\theta}_{m_i}^{\max} \end{cases} \quad (5.2)$$

In conclusion, for the purpose of this thesis, we take into consideration a real model of a Comau robot, in particular focusing on the first two axes of a SCARA manipulator, which is the chain most affected by inertial coupling effects, resulting in a relatively simplified and interpretable model. These axes correspond to a 2-DOF RR manipulator, which features two rotational flexible joints with parallel axes, namely corresponding to the double pendulum illustrated in Figure 5.3.



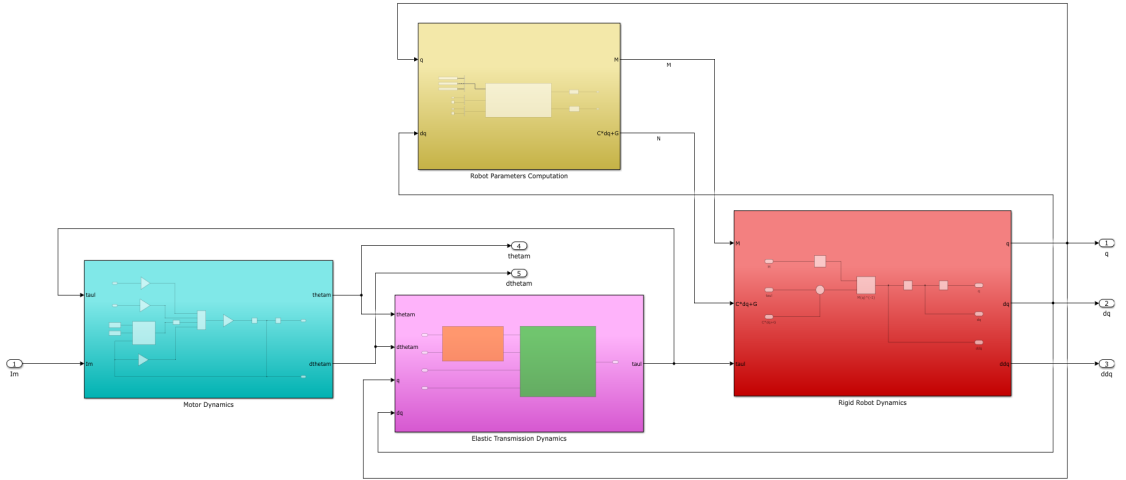
**Figure 5.2:** Non-linear friction torque with respect to motor speed for a RR robot



**Figure 5.3:** Illustration of a 2-DOF RR robot manipulator with flexible joints [33]

## Comau Robot Model Simulation

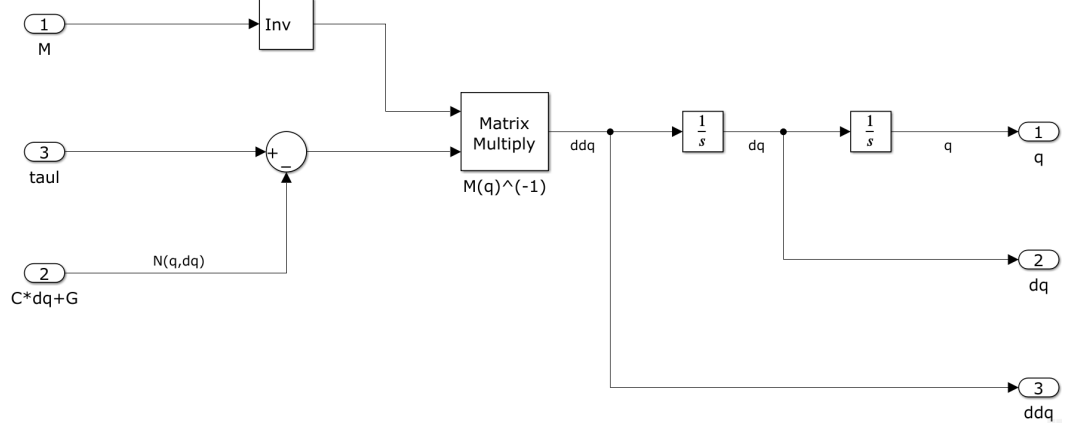
Based on the theoretical model detailed in the previous section, it is possible to devise numerical simulations, in particular on MATLAB by exploiting Simulink. The implementation of the block diagrams was deliberately designed to closely align with the physical formulation, ensuring the interactions between various physical subsystems are clearly visible, in such a way that priority was given to interpretability. While implementing a state-space formulation provides a compact and efficient model for analyzing and designing control systems, yet it lacks the intuitive physical clarity of mechanical models, making it harder to visualize the direct relationships between components. The flexible joint model used by Comau, expressed by the system of differential equations (5.1), is represented in Figure 5.4. Starting from the left, the current vector  $\mathbf{I}_m$ , corresponding to the input of the robot, enters the motor dynamics block (cyan), where is subject to the link torque vector  $\boldsymbol{\tau}_l$ , and it outputs the motor shaft angle position  $\boldsymbol{\theta}_m$  and velocity  $\dot{\boldsymbol{\theta}}_m$ . These then enter the elastic transmission dynamics block (purple), which also receives the position of the link joint variables  $\mathbf{q}$  and their velocity  $\dot{\mathbf{q}}$ , calculating the link torque value  $\boldsymbol{\tau}_l$  that acts, eventually, on the rigid robot dynamics block (red). This last block, receiving input from the inertia matrix calculation block and the nonlinear Centrifugal-Coriolis and gravity components (yellow), functions based on the pose, proceeds to calculate the actual joint positions  $\mathbf{q}$ , derived numerically from the acceleration  $\ddot{\mathbf{q}}$ .



**Figure 5.4:** Flexible-Joint Robot Model (EJM) block diagram

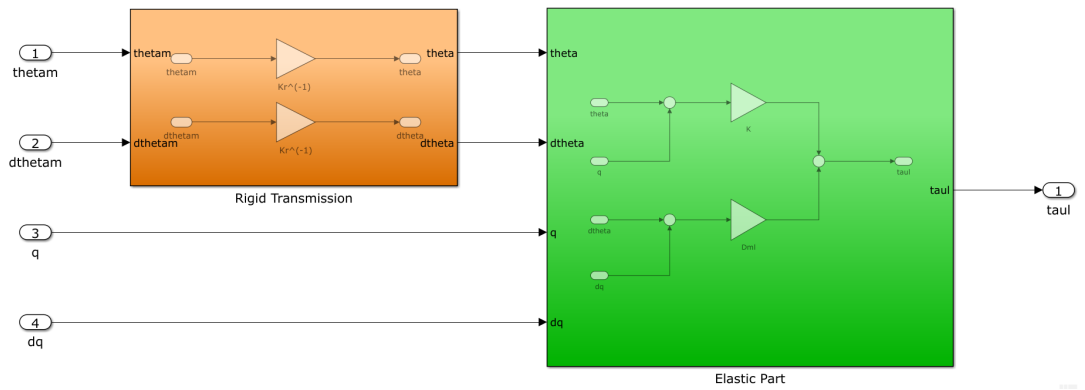
In detail we can analyze what each block implements, based on the mathematical model (5.1), noticing that the subsystems are all in mechanical Cauchy differential form, and not in a state-space representation.

- The rigid body robot equation (5.1.1) (red block) is implemented neglecting the direct friction effects  $\mathbf{f}_l(\dot{\mathbf{q}})$  and  $\mathbf{D}_l$ , which are not relevant for most of typical applications, as shown in Figure 5.5.



**Figure 5.5:** Rigid-robot model block diagram

- The parameter-varying coefficients, as well as the inertia tensor  $\mathbf{M}(\mathbf{q})$  and the Centrifugal and Coriolis vector  $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}})$ , are computed by a separated subsystem (yellow block), which exploits the Newton-Euler recursion, here not detailed. Specifically, the block derives from the gravity feedforward component of the industrial controller.
- The elastic transmission (purple block) represents the connection between the motor variables and the joint variables, thus collecting equations (5.1.2) and (5.1.4). It is designed to highlight, respectively, the part where the elasticity is ideally concentrated (green block) and the rigid section of the gearbox (orange block).



**Figure 5.6:** Elastic transmission block diagram

- The motor dynamics (cyan block) carries out the electromechanical behaviour of all motors described in (5.1.3), so by feeding an external input of current it provides the angular position and velocity of the motors as outputs, whose scheme is reported in Figure 5.7.

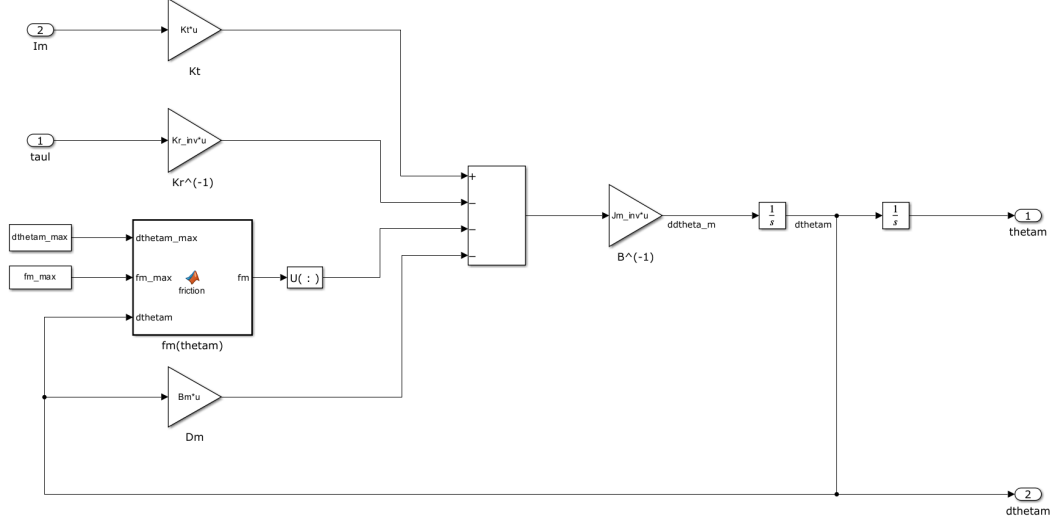


Figure 5.7: Motors' block diagram

### 5.1.2 Comau Robot Linearized Model

To obtain an analytical linearization of the flexible joint model (5.1) it is possible to consider the centrifugal-Coriolis and the gravitational terms plus the non-linear friction as disturbance on the links  $\mathbf{d}_l(t) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}_l(\dot{\mathbf{q}})$ ; similarly, the non-linear friction on the motor is treated as disturbance  $\mathbf{d}_m(t) = \mathbf{f}_m(\dot{\boldsymbol{\theta}})$ . The exogenous disturbances  $\mathbf{d}_l(t)$  and  $\mathbf{d}_m(t)$  represent the unmodeled dynamics of the manipulator bringing to a simplification that provides a linear approximation with respect to the joint and motor variables, allowing to obtain a Linear Parameter Varying (LPV) system, as  $\mathbf{M}$  is a pose-dependent matrix.

$$\begin{cases} \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{D}_l\dot{\mathbf{q}} + \mathbf{d}_l(t) = \boldsymbol{\tau}_l \\ \boldsymbol{\tau}_l = \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) + \mathbf{D}_{ml}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) \\ \mathbf{B}\ddot{\boldsymbol{\theta}} + \mathbf{D}_m\dot{\boldsymbol{\theta}} + \mathbf{K}_r^{-1}\boldsymbol{\tau}_l + \mathbf{d}_m(t) = \mathbf{K}_t\mathbf{I}_m \\ \boldsymbol{\theta} = \mathbf{K}_r^{-1}\boldsymbol{\theta}_m \end{cases} \quad (5.3)$$

Furthermore, to study the system behaviour in the frequency domain it is necessary to linearize the model around a reference configuration, i.e. a joint variables vector  $\mathbf{q}_0$ , resulting in a small-signal model, which corresponds to a Linear Time-Invariant system. In a gain scheduling control framework, each system represents the linearized or modal model of the manipulator in a neighborhood of  $\mathbf{q}_0$  in the

state-space, such that varying the operational point the LTI model switches. All matrices in the model are diagonal, except for the inertia matrix  $\bar{\mathbf{M}}(\mathbf{q}_0)$ , which represents the only term capturing the modal coupling of the system. Neglecting the off-diagonal terms of  $\mathbf{M}(\mathbf{q}_0)$ , treating them as additive external disturbances (included in  $\mathbf{d}_l(t)$ ), the model reduces to  $n$  formally decoupled linear differential equations. These equations correspond to a system of SISO elastic joint models (IEJM), where the parameters vary based on the configuration  $\mathbf{q}_0$ , thus forming an LPV system. Notice that even though these differential equations may be mathematically decoupled, in reality they are not from a physical point of view, since the dynamical coupling forces are strong enough to heavily affect the mutual dynamics, which is the target of this thesis.

Based on this linearized MIMO model, it is possible to construct a state-space representation of the robot dynamics, that can be used afterwards to derive the system's frequency response. Thus, let us define the state vector of the system as the concatenation of link and motor angular position and velocity as

$$\mathbf{x} \triangleq [\mathbf{q}^\top \quad \dot{\mathbf{q}}^\top \quad \boldsymbol{\theta}^\top \quad \dot{\boldsymbol{\theta}}^\top]^\top \quad (5.4)$$

where  $\mathbf{x}_1 = \mathbf{q}$ ,  $\mathbf{x}_2 = \dot{\mathbf{q}}$ ,  $\mathbf{x}_3 = \boldsymbol{\theta}$ ,  $\mathbf{x}_4 = \dot{\boldsymbol{\theta}}$  are vectors in  $\mathbb{R}^n$ .

Starting from the equation of the robot mechanics (5.3.1) and substituting the expression of  $\boldsymbol{\tau}_l$  from the torque link equation (5.3.2)

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{D}_l\dot{\mathbf{q}} + \mathbf{d}_l(t) = \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) + \mathbf{D}_{ml}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) \quad (5.5)$$

the joint acceleration vector is obtained

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1} \left[ \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) + \mathbf{D}_{ml}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) - \mathbf{D}_l\dot{\mathbf{q}} - \mathbf{d}_l(t) \right] \quad (5.6)$$

Similarly, applying the same substitution in the motor equation (5.3.3)

$$\mathbf{B}\ddot{\boldsymbol{\theta}} + \mathbf{D}_m\dot{\boldsymbol{\theta}} + \mathbf{K}_r^{-1} \left[ \mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) + \mathbf{D}_{ml}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) \right] + \mathbf{d}_m(t) = \mathbf{K}_t\mathbf{I}_m \quad (5.7)$$

the motor shaft accelerations are expressed as

$$\ddot{\boldsymbol{\theta}} = \mathbf{B}^{-1} \left[ \mathbf{K}_t\mathbf{I}_m - \mathbf{D}_m\dot{\boldsymbol{\theta}} - \mathbf{K}_r^{-1}\mathbf{K}(\boldsymbol{\theta} - \mathbf{q}) - \mathbf{K}_r^{-1}\mathbf{D}_{ml}(\dot{\boldsymbol{\theta}} - \dot{\mathbf{q}}) - \mathbf{d}_m(t) \right] \quad (5.8)$$

And by exploiting the definition of state, the system of differential equations can be cast into

$$\begin{cases} \dot{\mathbf{x}}_1 = \mathbf{x}_2 \\ \dot{\mathbf{x}}_2 = \mathbf{M}^{-1} \left[ \mathbf{K}(\mathbf{x}_3 - \mathbf{x}_1) + \mathbf{D}_{ml}(\mathbf{x}_4 - \mathbf{x}_2) - \mathbf{D}_l\mathbf{x}_2 - \mathbf{d}_l(t) \right] \\ \dot{\mathbf{x}}_3 = \mathbf{x}_4 \\ \dot{\mathbf{x}}_4 = \mathbf{B}^{-1} \left[ \mathbf{K}_t\mathbf{I}_m - \mathbf{D}_m\mathbf{x}_4 - \mathbf{K}_r^{-1}\mathbf{K}(\mathbf{x}_3 - \mathbf{x}_1) - \mathbf{K}_r^{-1}\mathbf{D}_{ml}(\mathbf{x}_4 - \mathbf{x}_2) - \mathbf{d}_m(t) \right] \end{cases} \quad (5.9)$$

Let us assume as input signal the motor currents

$$\mathbf{u}(t) = \mathbf{I}_m(t) \quad (5.10)$$

and let us consider four possible outputs in order to obtain different frequency responses, on which experimenting the application of sundry decoupling regulators to study as purpose of this thesis:

$$\mathbf{y}_1(t) = \mathbf{q}(t) \equiv \mathbf{x}_1 \quad \mathbf{y}_2(t) = \dot{\mathbf{q}}(t) \equiv \mathbf{x}_2 \quad \mathbf{y}_3(t) = \boldsymbol{\theta}(t) \equiv \mathbf{x}_3 \quad \mathbf{y}_4(t) = \dot{\boldsymbol{\theta}}(t) \equiv \mathbf{x}_4 \quad (5.11)$$

Finally, let us define as exogenous input the total disturbance vector

$$\mathbf{d}(t) = \begin{bmatrix} \mathbf{d}_l(t) \\ \mathbf{d}_m(t) \end{bmatrix} \quad (5.12)$$

Starting from the dynamics in state-space form it is possible to make explicit the linear relation of the derivatives with respect to the state vector, obtaining the dynamics matrix  $\mathbf{A} \in \mathbb{R}^{4n \times 4n}$

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}(\mathbf{D}_l + \mathbf{D}_{ml}) & \mathbf{M}^{-1}\mathbf{K} & \mathbf{M}^{-1}\mathbf{D}_{ml} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{B}^{-1}\mathbf{K}_r^{-1}\mathbf{K} & \mathbf{B}^{-1}\mathbf{K}_r^{-1}\mathbf{D}_{ml} & -\mathbf{B}^{-1}\mathbf{K}_r^{-1}\mathbf{K} & -\mathbf{B}^{-1}(\mathbf{D}_m + \mathbf{K}_r^{-1}\mathbf{D}_{ml}) \end{bmatrix} \quad (5.13)$$

Factoring out the chosen input it is possible to derive the input matrix  $\mathbf{B} \in \mathbb{R}^{4n \times n}$

$$\mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{B}^{-1}\mathbf{K}_t \end{bmatrix} \quad (5.14)$$

The disturbances are grouped in the total disturbance vector, which is pre-multiplied by the disturbance matrix  $\mathbf{E} \in \mathbb{R}^{4n \times 2n}$

$$\mathbf{E} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ -\mathbf{M}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{B}^{-1} \end{bmatrix} \quad (5.15)$$

Since each output is the subvector of the state-space vector, each output can be expressed linearly through the output matrix  $\mathbf{C} \in \mathbb{R}^{n \times 4n}$  with respect to it, where the  $i$ -th output matrix depends on which output is chosen

$$\mathbf{C}_1 = [\mathbf{I} \ \mathbf{0} \ \mathbf{0} \ \mathbf{0}] \quad \mathbf{C}_2 = [\mathbf{0} \ \mathbf{I} \ \mathbf{0} \ \mathbf{0}] \quad \mathbf{C}_3 = [\mathbf{0} \ \mathbf{0} \ \mathbf{I} \ \mathbf{0}] \quad \mathbf{C}_4 = [\mathbf{0} \ \mathbf{0} \ \mathbf{0} \ \mathbf{I}] \quad (5.16)$$



while the feedthrough matrix  $\mathbf{D} \in \mathbb{R}^{4n \times n}$  is clearly null

$$\mathbf{D} = \mathbf{0}_{4n \times n} \quad (5.17)$$

The Linear Parameter-Varying system can be expressed in state-space model, called also ABCD form, as

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}(\boldsymbol{\rho})\mathbf{x} + \mathbf{B}(\boldsymbol{\rho})\mathbf{u} + \mathbf{E}(\boldsymbol{\rho})\mathbf{d}(t) \\ \mathbf{y} = \mathbf{C}(\boldsymbol{\rho})\mathbf{x} + \mathbf{D}(\boldsymbol{\rho})\mathbf{u} \end{cases} \quad (5.18)$$

where  $\boldsymbol{\rho}$  is the exogenous variable vector, said parameter, which in this case corresponds to joint variables  $\mathbf{q} \in \mathbb{R}^n$ , so that  $\mathbf{A}(\boldsymbol{\rho}) \equiv \mathbf{A}(\mathbf{q})$ .

Now it is possible to derive the parametric matrix of transfer functions. Neglecting the disturbance inputs, i.e. imposing  $\mathbf{d} \stackrel{!}{=} \mathbf{0}$ , starting from the state-space model we take the Laplace transform assuming zero initial conditions, so yielding

$$s\mathbf{X}(s) = \mathbf{A}(\mathbf{q})\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s) \quad (5.19)$$

from which we solve for the state vector in the Laplace domain

$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A}(\mathbf{q}))^{-1} \mathbf{B}\mathbf{U}(s) \quad (5.20)$$

Substituting this result into the output equation, we obtain

$$\mathbf{Y}(s) = \mathbf{C}\mathbf{X}(s) + \mathbf{D}\mathbf{U}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A}(\mathbf{q}))^{-1} \mathbf{B}\mathbf{U}(s) + \mathbf{D}\mathbf{U}(s) \quad (5.21)$$

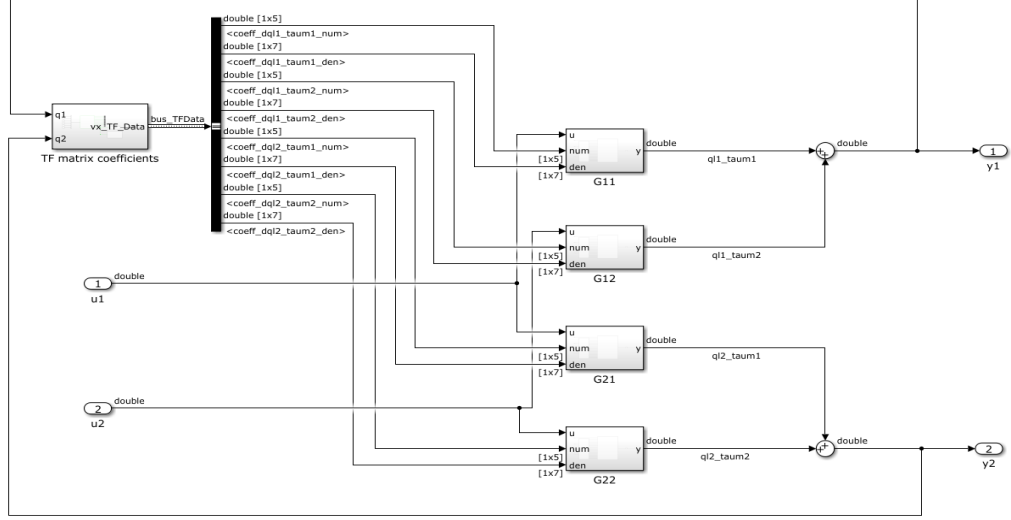
and factoring out the input we obtain the linear relation

$$\mathbf{Y}(s, \mathbf{q}) = \left[ \mathbf{C}(s\mathbf{I} - \mathbf{A}(\mathbf{q}))^{-1} \mathbf{B} + \mathbf{D} \right] \mathbf{U}(s) \quad (5.22)$$

Thus, the transfer function matrix from the input  $\mathbf{u}(t)$  to the output  $\mathbf{y}(t)$  is given by applying the Laplace transform to the differential system, to obtain  $\mathbf{Y}(s) = \mathbf{G}(s) \mathbf{U}(s)$ , where the transfer function matrix is calculated as

$$\mathbf{G}(s, \mathbf{q}) = \mathbf{C}(s\mathbf{I} - \mathbf{A}(\mathbf{q}))^{-1} \mathbf{B} + \mathbf{D} \quad (5.23)$$

in which  $\mathbf{I} \in \mathbb{R}^{4n \times 4n}$  denotes the identity matrix of the same dimension as  $\mathbf{A}$ , so that  $\mathbf{G}$  is the matrix of transfer functions of the MIMO system composed by  $n \times n$  functions of the Laplace variable  $s$  and the parameters represented by the joint variables collected in  $\mathbf{q}$ .



**Figure 5.8:** Simulink block diagram of the 2-axes LPV robot model

## 5.2 Decoupling Metrics

The analysis of the transfer function matrix in LTI MIMO systems helps to assess input-output coupling. By comparing Bode plots of diagonal and off-diagonal elements across frequencies, key coupling channels and their behaviors can be identified. RGA quantifies these interactions, aiding in decentralized control evaluation, while the Gershgorin Radius provides additional insights into decoupling. Finally, empirical metrics validate the effectiveness of these techniques in simulations.

### 5.2.1 Transfer Function Matrix

Given a LTI MIMO system, the qualitative analysis of the transfer function matrix  $\mathbf{G}(s)$  provides key insights into the degree of coupling between inputs and outputs. In fact a straightforward approach to evaluate directly the level of cross interactions is to study the off-diagonal transfer functions Bode plots, specifically focusing on the magnitude behaviour. Comparing the off-diagonal functions with respect to the relative diagonal dynamics over the whole spectrum allows to understand the most influent interacting channels as the frequency changes.

$$\mathbf{G}(s) = \left\{ G_{ij}(s), i = 1, \dots, n, j = 1, \dots, n \right\}_{n \times n} \quad (5.24)$$

### 5.2.2 Relative Gain Array

The Relative Gain array (RGA) [2] is one of most widely used mathematical tool in the analysis of Multi-Input Multi-Output (MIMO) control systems.

The RGA concept quantifies the level of interaction between the various inputs and outputs of a Multi-Input Multi-Output (MIMO) system being useful in determining whether a decentralized control configuration is appropriate.

Given a system described by a transfer function matrix  $\mathbf{G}(s)$ , its steady-state behavior is represented by the DC gain matrix

$$\mathbf{G} = \mathbf{G}(s) \Big|_{s=0} = \mathbf{G}(0) \quad (5.25)$$

This matrix describes the relationship between inputs and outputs at low frequencies, in particular under steady-state conditions. The fundamental intuition behind the RGA is to compare the sensitivity of an output to changes in an input when the other control loops are either open or closed. This leads to the definition of the inverse  $\mathbf{G}^{-1}$ , which represents the effect of each input on all outputs in a fully controlled system [34]. The Relative Gain Array is computed through the Hadamard (element-wise) product, indicated with  $\odot$  between  $\mathbf{G}$  and the transpose of its inverse like

$$\mathbf{\Lambda}(\mathbf{G}) = \mathbf{G} \odot (\mathbf{G}^{-1})^\top \quad (5.26)$$

so that each element is computed as

$$\lambda_{ij} = G_{ij} \cdot (G^{-1})_{ji} \quad (5.27)$$

This operation preserves the structure of the matrix and provides a direct measure of the relative effect of each input with respect to the outputs.

Thus each gain  $\lambda_{ij}$  provides crucial information about the interaction between inputs and outputs: if  $\lambda_{ii} \approx 1$  and  $\lambda_{ij} \approx 0$  for  $i \neq j$ , the system is nearly diagonally dominant, so this means that each input  $u_i$  heavily affects its corresponding output  $y_i$ , making a decentralized control strategy appropriate; while if  $\lambda_{ij}$  contains significant off-diagonal values, the system exhibits strong interactions between control loops, which may render a decentralized control strategy ineffective. Additionally, if the sum of each row equals one, that is:

$$\sum_{j=1}^m \lambda_{ij} = 1 \quad i = 1, \dots, n \quad (5.28)$$

then the system maintains relative gain conservation. Thus a useful criterion for evaluating the suitability of decentralized control is the RGA-number [34], defined as

$$\text{RGA-number} = \|\mathbf{\Lambda}(\mathbf{G}) - \mathbf{I}\|_{\text{sum}} = \sum_i \sum_{j \neq i} |\lambda_{ij}| \quad (5.29)$$

where  $\|\cdot\|_{\text{sum}}$  represents the sum of the absolute values of the off-diagonal elements. High values of this parameter indicate a high level of coupling between loops, making advanced compensation strategies necessary.

Generalizing this idea, which is based on the DC gains, to all the frequencies in the spectrum allows to obtain a Bode plot of the RGA in a more complete dynamical perspective [21].

$$\Lambda(\mathbf{G}(s)) = \mathbf{G}(s) \odot (\mathbf{G}(s)^{-1})^\top \quad (5.30)$$

However, this metrics is not perfect, since there are particular cases in which it fails to evaluate the couplings. To illustrate the limitations of the RGA-number, consider the DC gain

$$\mathbf{G} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0.1 & 1 & 1 \\ 1 & 1 & 0.1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

For this system,  $\Lambda(\mathbf{G}) = \mathbf{I}$ , and consequently the RGA-number is 0, which suggests a good level of decoupling. However, further other interaction analysis, as the Gershgorin Radius illustrated in the following section, indicates that the system is actually far from being decoupled, thus proving the need for additional robustness measures alongside RGA analysis.

### 5.2.3 Gershgorin Radius

The Gershgorin Circle theorem [24] provides a valuable tool for estimating the location of eigenvalues of a complex square matrix. Given a matrix  $\mathbf{Z} \in \mathbb{C}^{n \times n}$ , this theorem states that all its eigenvalues are contained within the union of Gershgorin discs. These discs are centered at the diagonal elements  $z_{ii}$ , with radii  $R_i$  determined by the sum of the absolute values of the off-diagonal elements in the corresponding row as follows

$$R_i = \sum_{j \neq i} |z_{ij}| \quad (5.31)$$

This property is particularly useful in control theory, where it helps to analyze the behavior of Multi-Input Multi-Output (MIMO) systems, in which the relationship between inputs and outputs is described by a transfer function matrix  $\mathbf{G}(s)$ , whose elements vary with the complex frequency variable  $s$ , indeed. The Gershgorin theorem can be extended to transfer matrices, where at each frequency  $s = j\omega$  a ball can be defined around each diagonal element with a radius

$$R_i(j\omega) = \sum_{j \neq i} |G_{ij}(j\omega)| \quad (5.32)$$

whose size provides insight into the dominance of the diagonal elements over the off-diagonal ones, which is crucial when evaluating system decoupling.

Formally, a MIMO system is said to be row diagonally dominant [2] when the magnitude of each diagonal element is greater than the sum of the magnitudes of the corresponding off-diagonal elements

$$|G_{ii}(j\omega)| > \sum_{j \neq i} |G_{ij}(j\omega)| \quad (5.33)$$

Thus a MIMO system is considered well-decoupled if its transfer function matrix exhibits diagonal dominance, so that each output is mainly influenced by its corresponding input, with minimal interference from other inputs. Finally, in order to quantify the degree of coupling, the Gershgorin Radius ratio [21] is introduced, defined as the radius itself normalized by the diagonal element

$$\zeta_i(j\omega) = \frac{\sum_{j \neq i} |G_{ij}(j\omega)|}{|G_{ii}(j\omega)|} = \frac{R_i(j\omega)}{|G_{ii}(j\omega)|} \quad i = 1, \dots, n \quad (5.34)$$

A low value of  $\zeta_i(j\omega)$  suggests strong decoupling, meaning that input  $i$  primarily affects output  $i$ , with little influence from other channels, and conversely, a high value indicates significant couplings among different inputs and outputs.

#### 5.2.4 Control Response

An empirical way to evaluate the level of couplings is studying the response behaviour when a decentralized controller is designed and applied to the system, in order to experimentally analyze the effects of the transient and the steady-state control performance, particularly when a channel is kept fixed to track a constant reference signal, while the other is moved. Thus, it is possible to firstly observe the quality of the direct control on the relative output, and moreover to study the effect of input-output couplings due to the mutual dynamics.

In order to evaluate the goodness of the degree of decoupling, the following response metrics are employed, separately for each channel  $i$ :

- Tracking accuracy, measured as the Root Mean Square Error between the reference signal and the system output signal;

$$\text{RMSE}_i = \sqrt{\frac{1}{N} \sum_{j=1}^N \left( y_i(t_j) - y_{r,i}(t_j) \right)^2} \quad (5.35)$$

- Overshoot, calculated as the maximum peak of the output response with respect to the steady-state asymptote, expressed in absolute value for evaluating even regulation to zero caused by coupling effects;

$$\hat{\delta}_i = \max \left\{ |y_i(t)| \right\} - y_{r,i}(t) \quad (5.36)$$

- Settling time, measured as the seconds from the step time  $\bar{t}_r$  required for the response to permanently land within the tolerance  $\delta_{tol}$  band around a constant reference signal.

$$\hat{t}_i = \max \{t_j : |y_i(t_j) - y_{r,i}(t_j)| > \delta_{tol}\} - \bar{t}_r \quad (5.37)$$

Eventually, three families of reference signals are taken into consideration: step, providing different steady-state values; ramp, imposing increasing slope values; and sine, changing amplitude and frequency. Not all the empirical metrics can be applied for each type of reference.

Furthermore, four different system outputs, and thereby transfer function matrices, are considered in this treatment, in order to study the coupling interactions and specifically the diagonalizing techniques at different states of the manipulator dynamics:  $\mathbf{q}$ ,  $\dot{\mathbf{q}}$ ,  $\boldsymbol{\theta}$  and  $\dot{\boldsymbol{\theta}}$ .

Finally, it is important to notice that the numerical results are fruit of the specific designed regulator mixed with the effect of the decoupling algorithms, so that the simulative outcomes must be interpreted in light of the relative difference, and not as general values to take with a major grain of salt, provided that each evaluated system is tested under the same conditions, i.e. designing diagonal regulators on the direct transfer functions with the same control specifications.

### 5.3 Decentralized and Decoupled LTI Dynamics

This exploratory phase aims to evaluate the effectiveness of linear decoupling techniques comparing the quality of the different systems' open-loop transfer functions in the case of the Linearized Time-Invariant Comau model (5.3) (computed in a specific working point), assessing the coupling metrics of Relative Gain Array and Gershgorin Radii for both the original decentralized system  $\mathbf{G}(s)$  and the decoupled systems  $\mathbf{M}(s) = \mathbf{G}(s)\mathbf{D}(s)$  exploiting Static, SVD, Ideal, Simplified and Inverted Decoupling strategies through  $\mathbf{D}(s)$  (or  $\mathbf{D}_1(s)$  and  $\mathbf{D}_2(s)$  in the case of SVD). In this way, it is possible to identify the best decoupling techniques for a robotic system, in order to develop the generalization to Linear Parameter Varying decouplers.

Let us take into account the transfer function matrix of the open-loop transfer matrices between the motor torque  $\boldsymbol{\tau}_m$  input with  $\dot{\boldsymbol{\theta}}$ ,  $\boldsymbol{\theta}$ ,  $\dot{\mathbf{q}}$ , and  $\mathbf{q}$  as output, in order to prove the extensive applicability of the decoupling techniques at different states of the linearized robot dynamics. In particular, the decoupling simulations have been conducted following the diagram of the robotic model from velocity of motors to joint angles, conversely to the way the results are presented and analyzed in the following sections.

The first figure for each analyzed method presents the Bode plots of the transfer function matrix before and after applying decoupling, where the original transfer functions are highlighted in blue while the overall decoupled system in red. These plots illustrate the frequency response of the system, showing changes in magnitude and phase for each one of the input-output pairs, in this case  $2 \times 2$ . A well-decoupled system should exhibit a strong magnitude attenuation on the off-diagonal functions indicating reduced cross-coupling effects, and preferably minimal differences between the original and decoupled diagonal transfer functions.

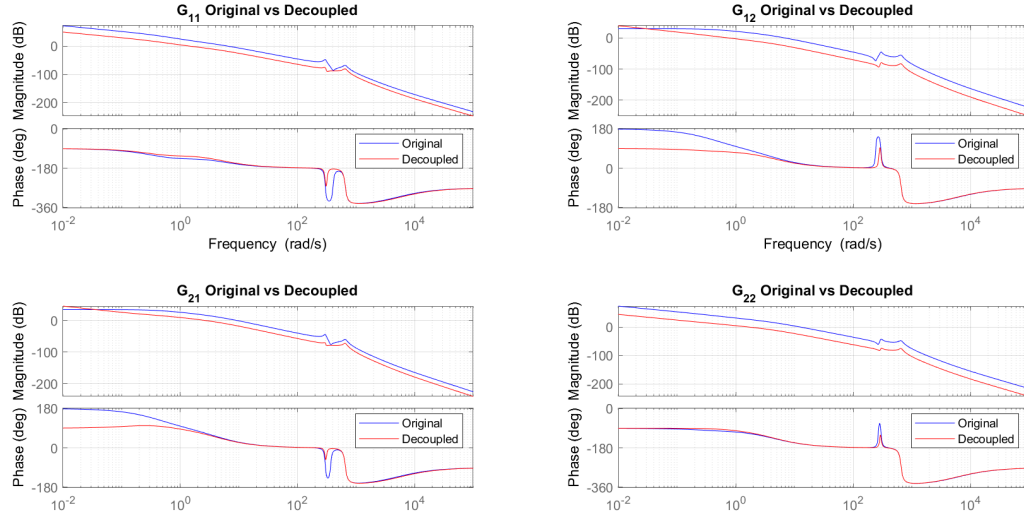
Following this, the second kind of figures displays the frequency-dependent RGA elements, showing how the relative gains evolve with frequency. Ideally the system is decoupled if the RGA diagonal values approximate 1 and the off-diagonal gains tend to 0, with respect to the frequency. So, the more the plot resembles an identity matrix, the better the system behaves. This type of plots shows the four components of the RGA with respect to frequency, where the original system is the blue dashed line and the decoupled system with the specific algorithm is highlighted in red.

The third type of plot in this section graphically represents Gershgorin radii of the transfer function matrix with respect to the frequency, that provides insight into the system's diagonal dominance. The objective of decoupling is to bring these radii closer as possible to zero, ensuring that the diagonal magnitude is sufficiently greater, hopefully for a large bandwidth. The plot condenses in one cartesian plane both the Gershgorin Radii for the original system, represented in dashed line, and the ones for the decoupled system in continuous line, where colors indicate the relative matrix row (red: 1, blue: 2).

### 5.3.1 Motor Torque $\tau_m$ to joint position $q$ dynamics

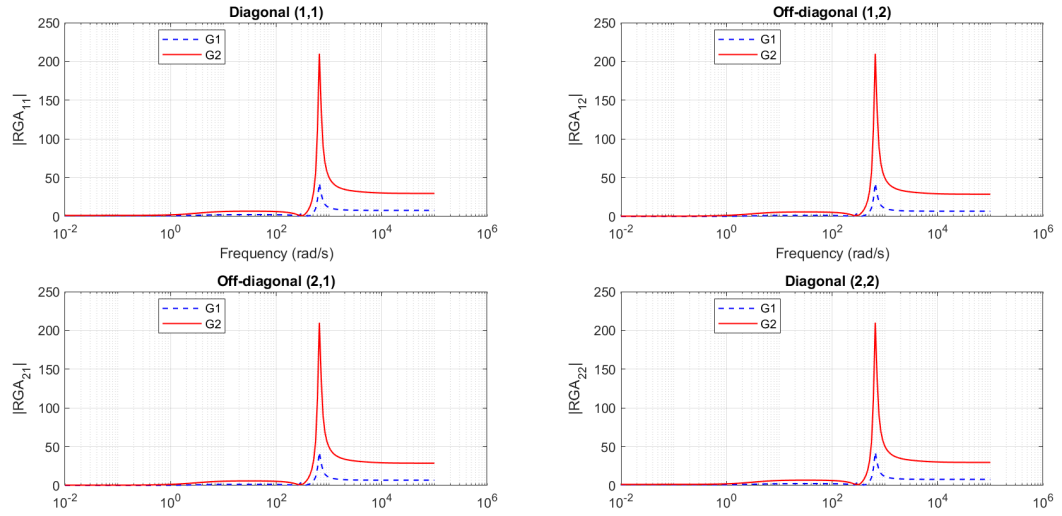
Let us take into consideration the most important transfer function matrix for the purpose of this thesis, i.e. the relationship between the input torque and the position of the joints as measured output.

Firstly, we discuss the application of the Static Decoupling technique to the robot plant, which is devised to make the system behaves diagonal with unitary gain on diagonals, at a low frequency of  $\omega = 1$  rad/s (typically is set to 0 but after different trials this working pulsation represents a good trade-off). The static decoupled transfer function matrix in Figure 5.9 shows that the the matrix is approximately identical at the specified frequency, however the static decoupler is not suitable for high-order systems with complex dynamics, since it may attenuate not only the off-diagonal components but also the direct transfer functions, as in this case. The RGA plots are more interesting, showing that the system at the designed frequency provides an attenuation on the four relative gains, showing the presence



**Figure 5.9:** Bode plot of the transfer function matrix  $\mathbf{q}/\tau_m$  with Static decoupling

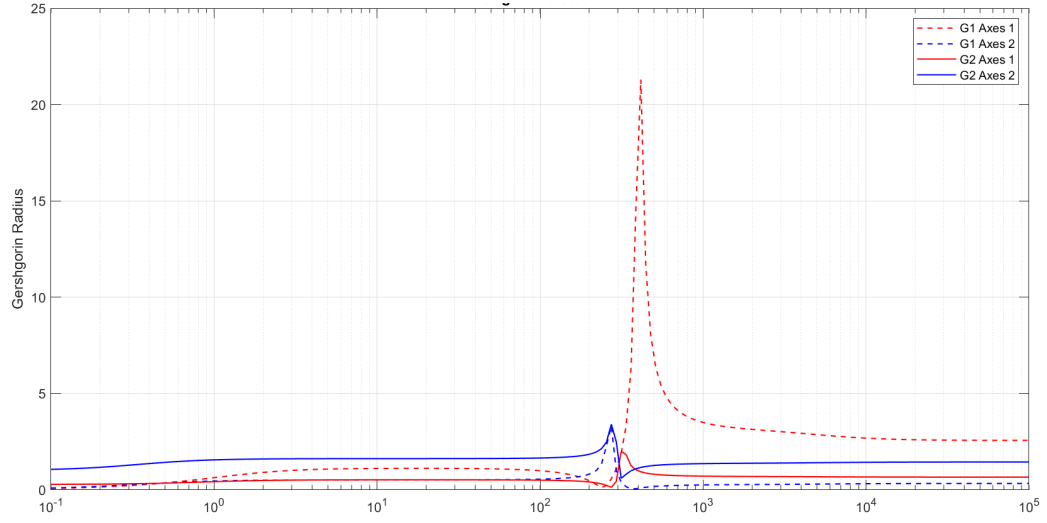
of couplings, and moreover highlighting a typical side effect of this approach, the amplification of the couplings in a high-frequency bandwidth ( $> 100$  rad/s), as reported in literature [11]. The Gershgorin Radii plot in Figure 5.11 provides also



**Figure 5.10:** RGA of the transfer function matrix  $\mathbf{q}/\tau_m$  with Static decoupling

other information, for example the fact that the first output is decoupled at low frequency, while the second one is even worsen in the whole spectrum, showing overall better performance mainly for decoupling the first axis. Eventually, the use of the Static Decoupler lowers the gains, requiring more aggressive controls.

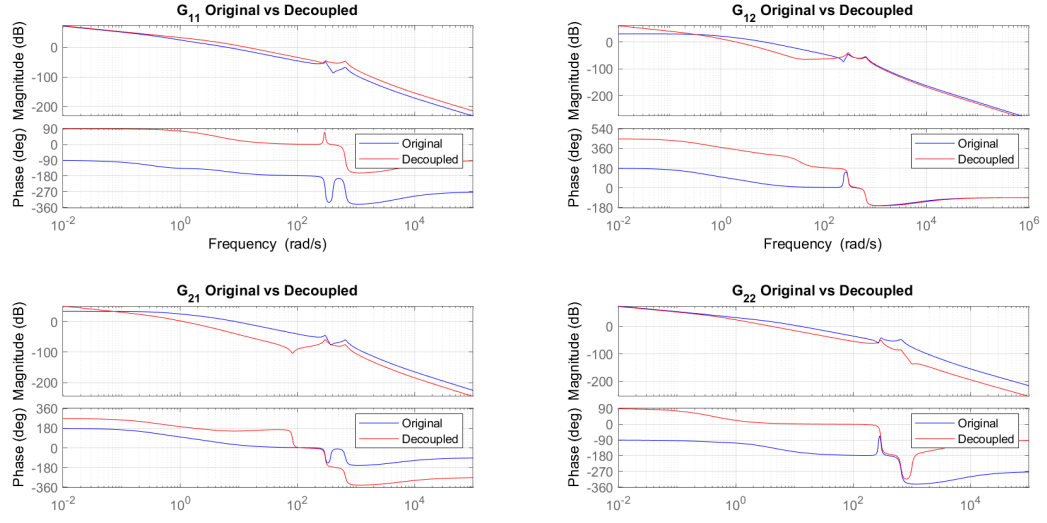




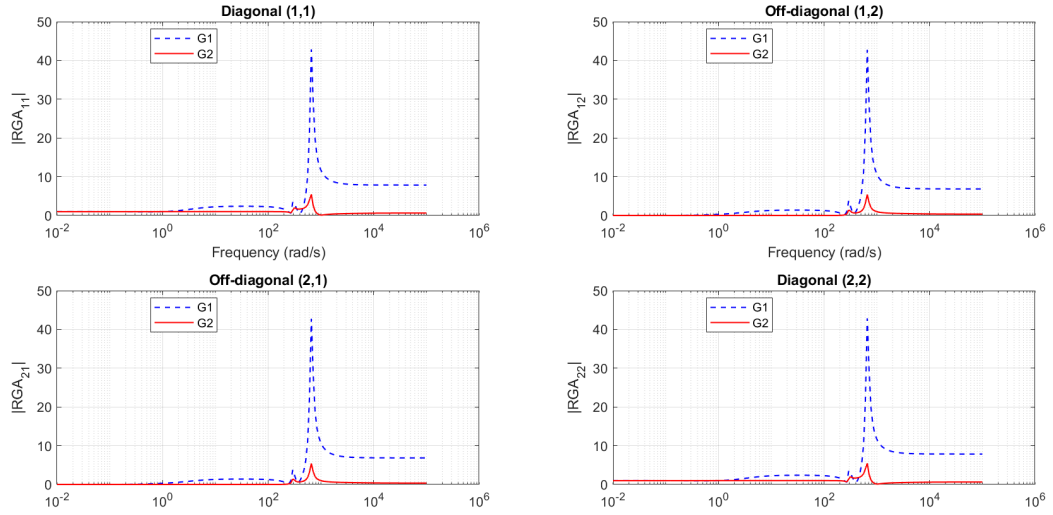
**Figure 5.11:** Gershgorin Radius of the transfer function matrix  $\mathbf{q}/\tau_m$  with Static decoupling

The family of static decouplers includes the SVD-based Decoupling, which consists of designing a pre- and post-compensator tuned around a specified frequency, in this case  $\omega = 10$  rad/s, that diagonalizes the system such that it behaves as its singular value matrix computed at steady-state. The Bode plots of the transfer function matrix for the SVD Decoupled system in Figure 5.12 shows a diagonalized behaviour around the chosen frequency, so that the off-diagonal elements are attenuated in magnitude, while the diagonal transfer functions are slightly mitigate due to the decomposition, in the  $[1, 100]$  rad/s spectrum bandwidth. The RGA plots of the SVD decoupled plant in Figure 5.13 better illustrates the couplings between each input-output pair where the off-diagonal ones result to be zero and similarly the diagonal gains are unitary, up to an angular frequency of 120 rad/s. And the SVD decoupling Gershgorin radii in Figure 5.14 offers a similar perspective, where the system actually shows a diagonal dominance until 110 rad/s, since the sum of off-diagonal per row tends to zero in the low-frequency, indeed. In other words, the SVD Decoupler ensures a far better diagonalization than the Static Decoupler, being less sensitive to the frequency.

In order to overcome the limitation of the static tuning, the dynamic decouplers family is introduced. The first examined dynamic technique gives the Ideal Decoupled system, whose transfer function matrix is presented in Figure 5.15. Here we can notice that the diagonal functions are almost the same, except for little numerical errors due to zero-pole cancellations, i.e. any modeling uncertainties could lead to instability. In addition, the cross transfer functions are basically reduced in magnitude to zero over all the spectrum apart around the anti-resonance

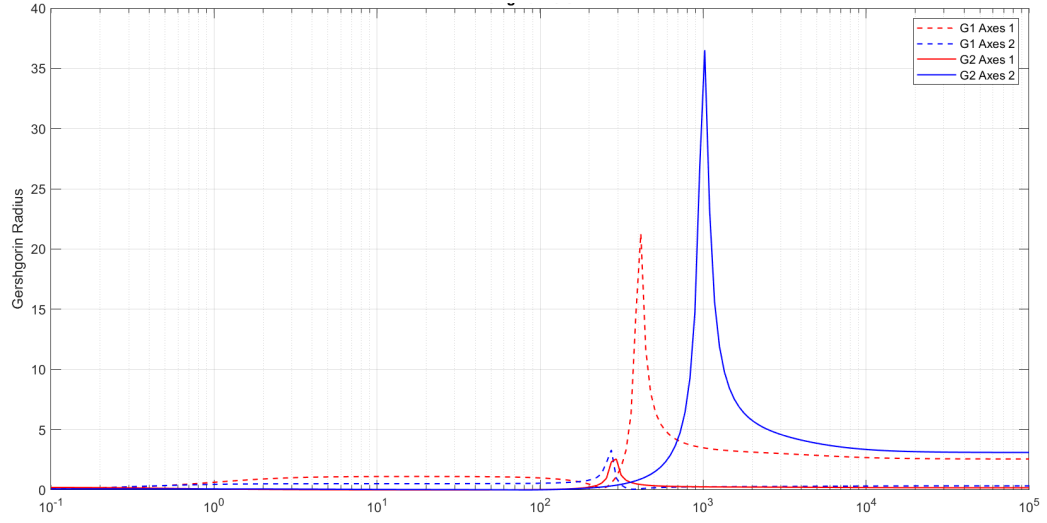


**Figure 5.12:** Bode plot of the transfer function matrix  $\mathbf{q}/\tau_m$  with SVD static decoupling

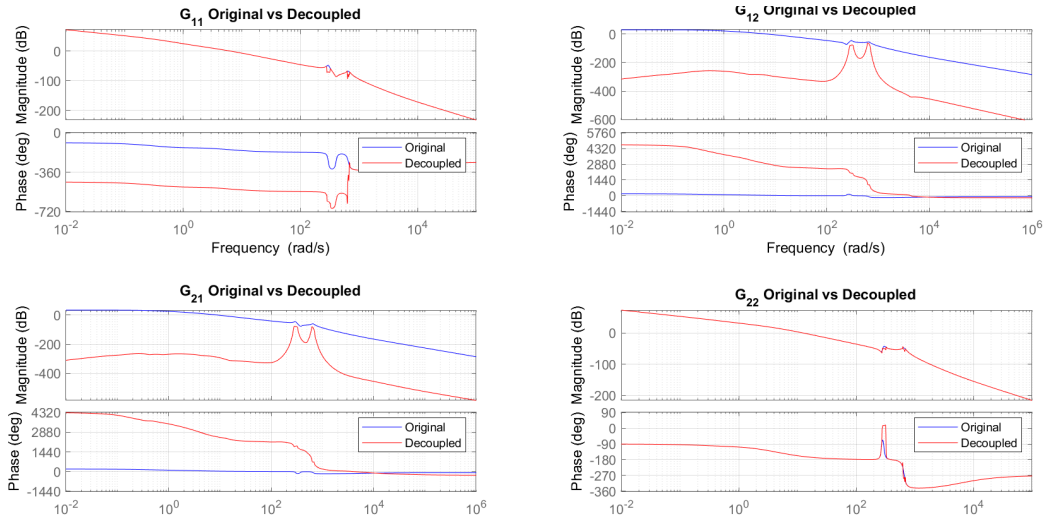


**Figure 5.13:** RGA of the transfer function matrix  $\mathbf{q}/\tau_m$  with SVD static decoupling

and resonance frequency, while about the phase plots note that the decoupler introduces phase rotations. It is crucial evaluating the open-loop stability of the system and the closed-loop stability by verifying the phase and gain margins. It turns out that the open-loop system is unstable due to the presence of poles with positive real parts. However, the positive phase and gain margins indicate that the system can be stabilized in closed-loop through proper feedback control.



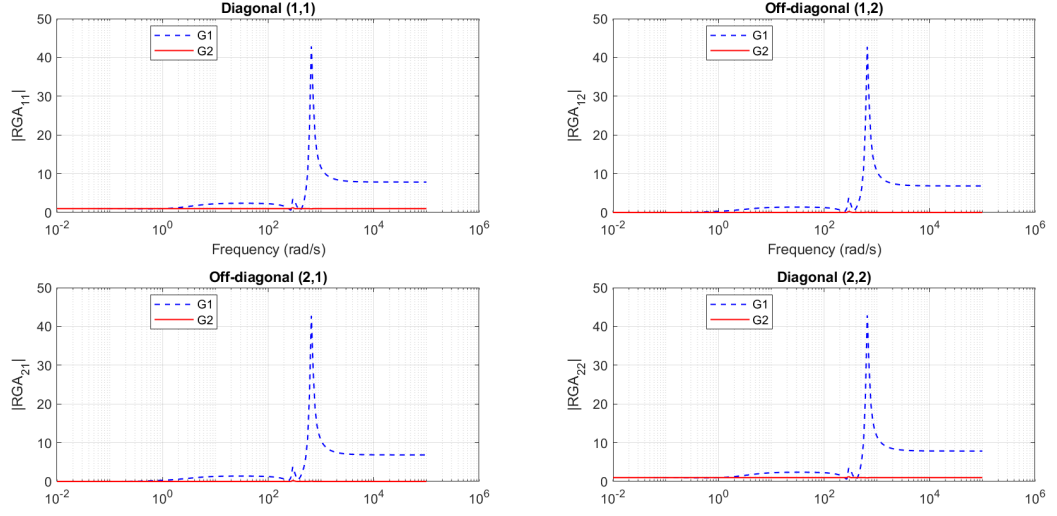
**Figure 5.14:** Gershgorin Radius of the transfer function matrix  $\mathbf{q}/\tau_m$  with SVD static decoupling



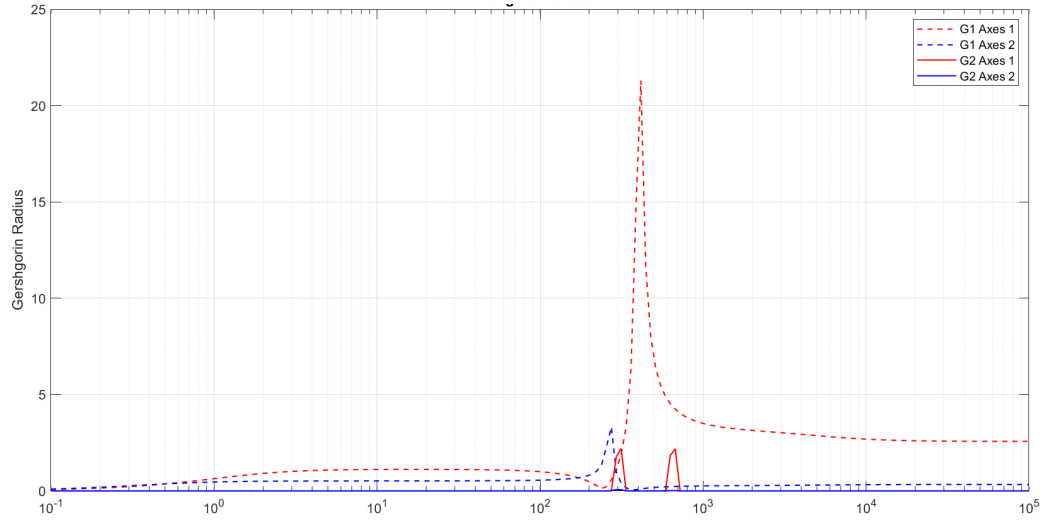
**Figure 5.15:** Bode plot of the transfer function matrix  $\mathbf{q}/\tau_m$  with Ideal decoupling

For what concerns the effect of Ideal decoupling the RGA, Figure 5.16 shows that the interaction among different channels are reduced to zero as the diagonal relative gains are kept unitary for each frequency of the spectrum, reaching a perfect diagonalization, assumed that an exact knowledge of the process is provided.

Even the Gershgorin Radii shown in Figure 5.17 proves the quality of the ideal decoupling, because it is possible to notice that each radius is reduced to zero.

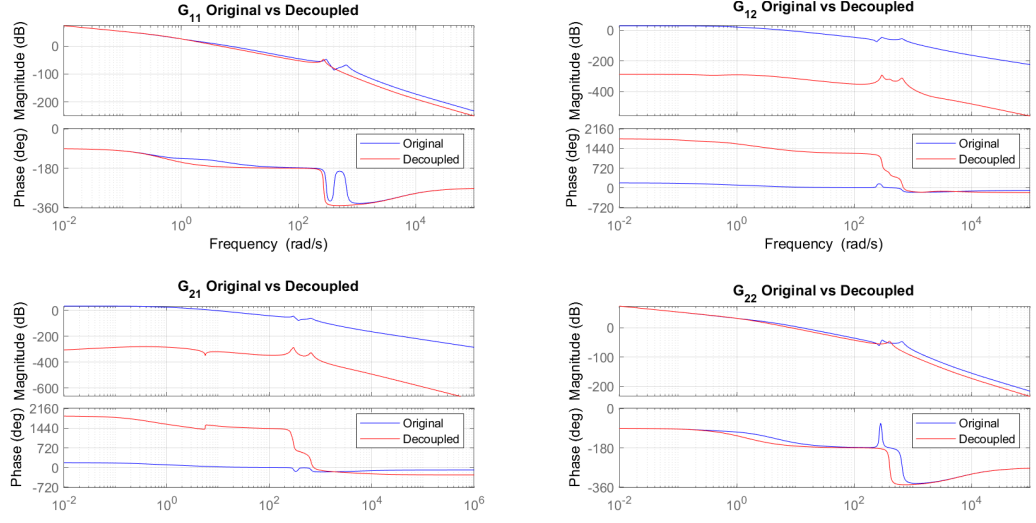


**Figure 5.16:** RGA of the transfer function matrix  $\mathbf{q}/\tau_m$  with Ideal decoupling



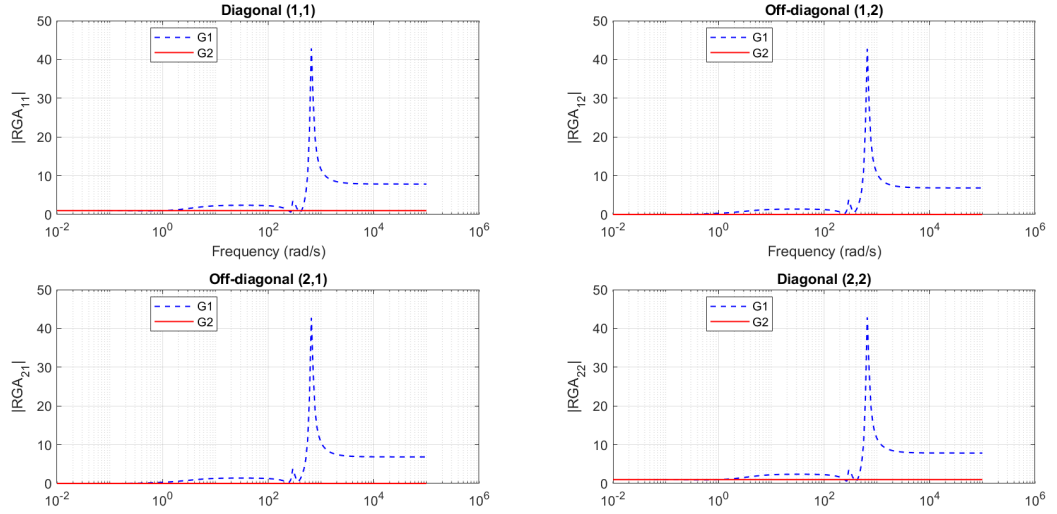
**Figure 5.17:** Gershgorin Radius of the transfer function matrix  $\mathbf{q}/\tau_m$  with Ideal decoupling

The dynamic Simplified decoupler is an approximation of the previous strategy, whose Bode plots are illustrated in Figure 5.18 demonstrating that the cross transfer functions have almost zero magnitude regardless the frequency, while the on-diagonal functions are modified with respect to the decentralized open-loop model, keeping a similar trend but having the resonance and anti-resonance frequencies moved to a lower bandwidth. About stability, it is possible to notice that the decoupled open-loop is marginally stable as the original one, so this simplified



**Figure 5.18:** Bode plot of the transfer function matrix  $\mathbf{q}/\tau_m$  with Simplified decoupling

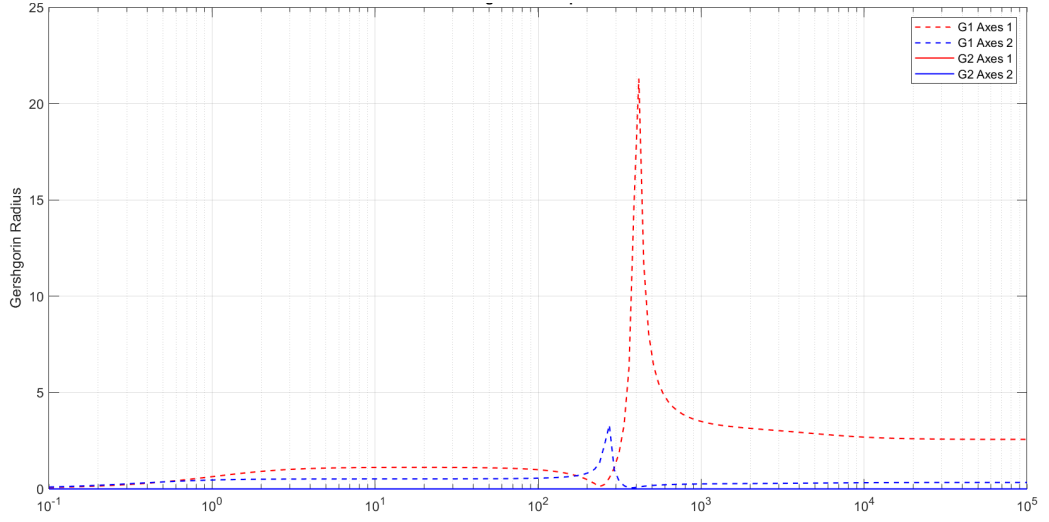
strategy ensures stability at the cost of a modification on the direct dynamics. In order to investigate further the effect of diagonalization, let us study the RGA through the plot in Figure 5.19, which proves that for each frequency the Relative



**Figure 5.19:** RGA of the transfer function matrix  $\mathbf{q}/\tau_m$  with Simplified decoupling

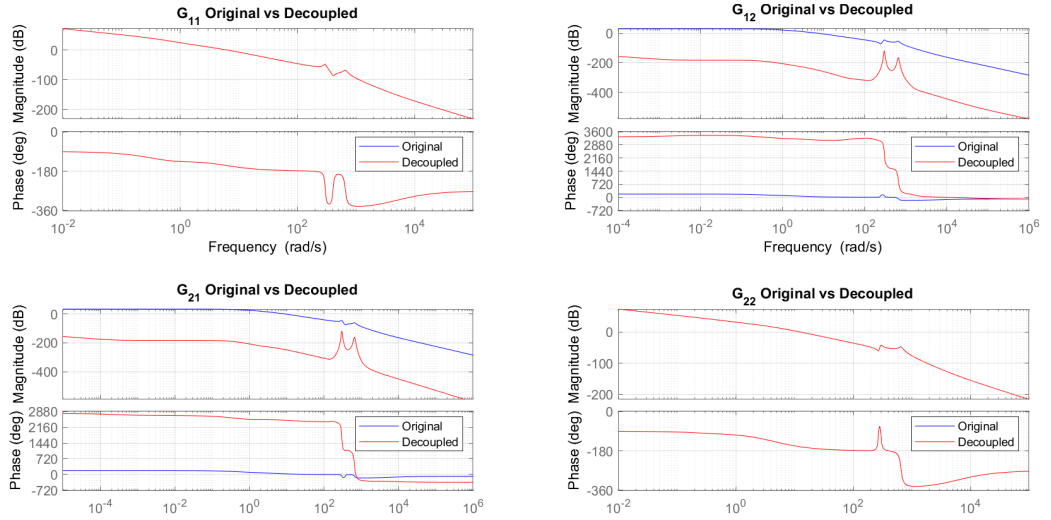
Gain Array is approximately an identity matrix for each frequency, so that the couplings are inhibited, leading to diagonal dominance, as illustrated in Figure 5.20 by the null Gershgorin Radii of the decoupled system as frequency varies,

eventually proving the effectiveness of this strategy with low computational cost.



**Figure 5.20:** Gershgorin Radius of the transfer function matrix  $\mathbf{q}/\tau_m$  with Simplified decoupling

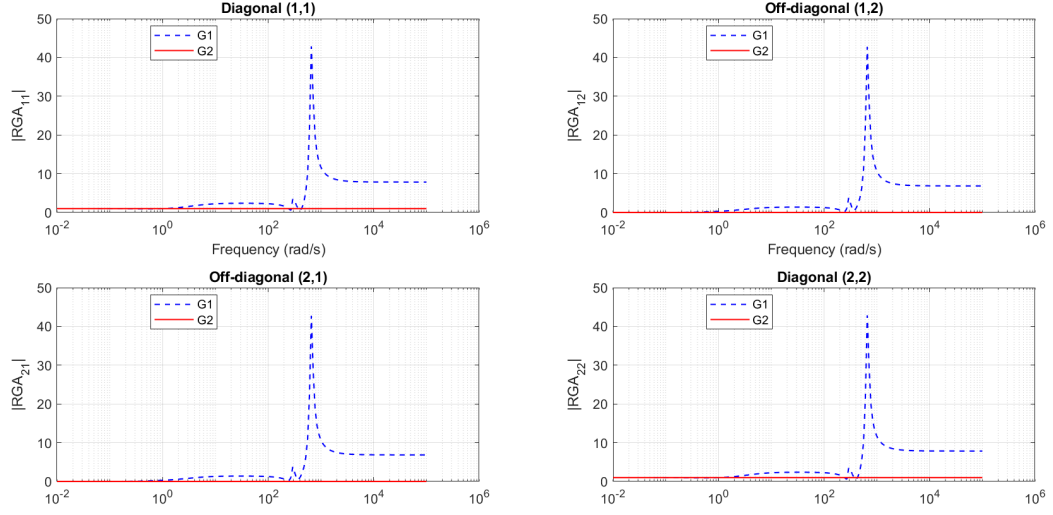
Finally, let us take into consideration the Inverted decoupling transfer function matrix, plotted in Figure 5.21, where the diagonal dynamics is kept exactly equal



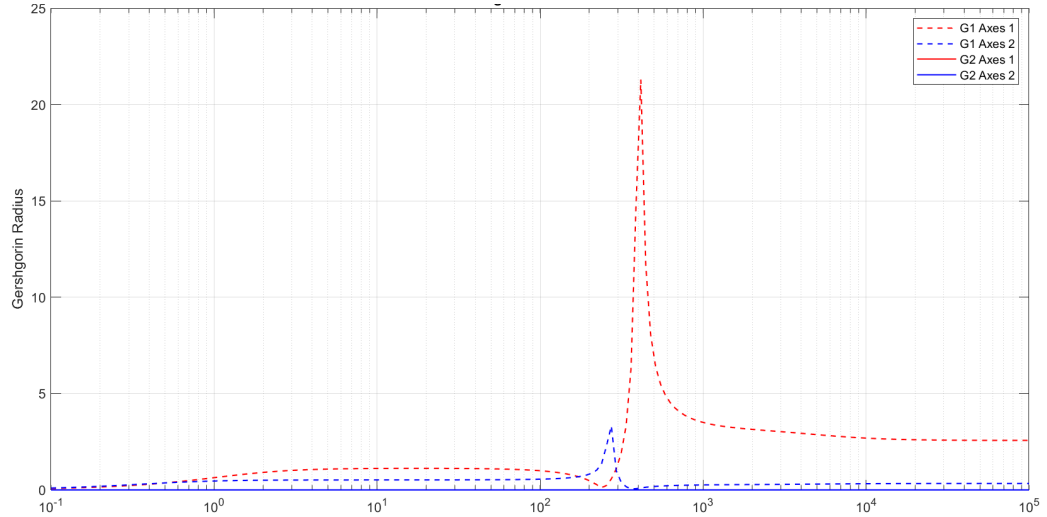
**Figure 5.21:** Bode plot of the transfer function matrix  $\mathbf{q}/\tau_m$  with Inverted decoupling

in magnitude and phase with respect to the original plant, while the cross functions are reduced in amplitude and rotated in phase, due to the higher-order functions,

and showing a double resonance which can be neglected for their low magnitude. About the open-loop stability, it is possible to claim that the Inverted decoupled system is marginally stable, as the original one, so that the closed-loop stability is achievable. The degree of coupling computed using this latter strategy is evaluated by computing the Relative Gain Arrays, illustrated in Figure 5.22, and the diagonal dominance is analyzed by studying the Gershgorin Radii in Figure 5.23, which



**Figure 5.22:** RGA of the transfer function matrix  $\mathbf{q}/\tau_m$  with Inverted decoupling



**Figure 5.23:** Gershgorin Radius of the transfer function matrix  $\mathbf{q}/\tau_m$  with Inverted decoupling

proves that the Inverted decoupled system achieves perfect diagonalization across the entire spectrum, as all the radii are reduced to zero, similarly to the Ideal case. In other words, the Inverted decoupler ensures an optimal diagonalization, supposing that the model is accurate, since the diagonal dynamics remains the same as the original one while the cross dynamics gets completely canceled. Eventually, it provides exactly the same response performance of the Ideal decoupler, but requiring a lower implementation complexity, comparable to that of the Simplified algorithm.

### 5.3.2 Motor Torque $\tau_m$ to Motor Angles $\theta$ dynamics

A similar frequency analysis has been conducted for the transfer function matrix relating the motor torque  $\tau_m$  to the motor position  $\theta$ . The derivation of the motor dynamics follows the same modeling process used for the joint position  $\mathbf{q}$ , with analogous results observed for the different decoupling techniques. Notably, these outcomes were established earlier in the modeling phase, confirming that the decoupling strategies exhibit comparable effectiveness when applied to  $\mathbf{q}$  and  $\theta$ .

The significance of  $\theta$  lies in the fact that it represents the motor-side variables, which are typically measured through rotary encoders for direct measurement. In contrast, the joint positions  $\mathbf{q}$  are often not directly measurable and require additional estimation techniques. This distinction makes the decoupling analysis for  $\theta$  particularly relevant, as it directly impacts the accuracy of motor control.

Regarding the decoupling techniques, static and dynamic strategies have been analyzed, yielding results similar to those obtained for  $\mathbf{q}$ . The Static Decoupling approach attempts to diagonalize the motor system at low frequencies, but may introduce attenuation in the diagonal elements while not often amplifying high-frequency couplings, being unsuitable for this plant. The SVD-based Decoupling improves upon this by designing two compensators around a specific frequency, ensuring better diagonalization over a wider range. On the other hand, Ideal Decoupling theoretically cancels cross-couplings but is highly sensitive to model uncertainties. More practical alternatives, such as Simplified and Inverted Decoupling, offer a trade-off between computational complexity and performance, effectively reducing interactions without significantly altering the motor system's direct dynamics. From a simulation perspective, it is important to note that in the block diagram representation of the system, the motor variables are positioned before the joint dynamics. Consequently, all decoupling tests have been conducted contrariwise starting from the motor torques and propagating through the system toward the joint positions. Thus, this approach ensures that decoupling strategies are validated at the motor level before analyzing their effects on the link dynamics, aligning with the physical structure of the robotic system.



Given that the transfer function behavior  $\mathbf{q}_m/\tau_m$ , Bode plots, Relative Gain Arrays, and Gershgorin Radii exhibit trends similar to those previously discussed for  $\mathbf{q}$ , these plots are omitted for brevity and because the results confirm that each decoupling method behaves consistently on both position variables  $\boldsymbol{\theta}$  and  $\mathbf{q}$ .

## 5.4 Control response of Decentralized and Decoupled LTI Dynamics

In this second experimental phase let us analyze the transient and steady-state frequency response of the open-loop transfer matrices with  $\tau_m$  input and  $\dot{\boldsymbol{\theta}}$ ,  $\boldsymbol{\theta}$ ,  $\dot{\mathbf{q}}$  and  $\mathbf{q}$  as output, when a diagonal regulator is applied to close a feedback loop on the original plant and on the different decoupled systems also. Specifically, the independent-joint decentralized control architecture is designed as a cascade two-loop controller, composed by a position and a velocity regulator, which may be overall interpreted as a PID controller, tuned on the corresponding diagonal transfer functions in order to impose for each experiment the same specifications given by Comau. In fact, under the same conditions, it is possible to validate the beneficial effects of the use of these diagonalization strategies applied at different states of the robot dynamics, providing the same reference signals to track. In detail, extensive tests are conducted by feeding as input three types of signals: step, ramp and sine. However, in the following treatment just the few most meaningful simulations are reported, mainly focusing on the regulation performance applying an input step holding one axis stationary, by analyzing the most interesting qualitative responses and by specifying the quantitative metrics like accuracy, coupling, overshoot and settling time.

Let us start from the experimental tests providing a step target with null initial conditions, in order to show the output responses of both the joints  $\mathbf{q}$  and motor shaft angles  $\boldsymbol{\theta}$ , to make a comparative study on the behaviour of the decoupled control strategies in action with respect to the decentralized controlled plant, in a qualitative way by observing the plots per axis (red: 1, blue: 2) of the six closed loop systems and also by the numerical results, which are collected in the following tables, showing the relative metrics per axis for each combination of steps (rows) and for each type of proposed systems (columns). Sundry simulations have been realized but only a subset of them are reported, especially the one where an axis is fixed into a position, e.g.  $y_{r1} = 0$ , and the other one gets moved to follow the given step,  $y_{r2} = r$  in this case. Even other combinations of combined steps have been tried but the results are similar to the simplified case, showing as expected a linear trend, so they are not included in tables.

### 5.4.1 Joint Step Response of $q/\tau_m$ Dynamics

The joints are controlled by a PID controller, being tuned on the diagonal transfer functions complying with the following specification: overshoot within 5% and settling time of 3 s.

Holding the first axis stationary and feeding a unitary step as reference for the second axis, to the decentralized control system and to the five decoupled systems, the output responses for both joint variables are presented in Figure 5.26 and the performance outcomes are listed in Table 5.1 for RMSE, Table 5.2 for overshoot and Table 5.3 for settling time.

**Table 5.1:** joint position  $\mathbf{q}$  response comparison: RMSE [rad] with different reference steps (target [rad]) for each output (n)

n	Target	Original	Static	SVD	Ideal	Simplified	Inverted
1	0	0.0141	0.0250	0.0041	3.2e-11	2.2e-16	6.2e-16
2	1	0.0549	0.0553	0.0709	0.0583	0.0605	0.0583
1	0	0.02824	0.0501	0.0083	1.0e-10	2.3e-16	1.5e-15
2	2	0.1104	0.1114	0.1424	0.1173	0.1217	0.1173
1	0	0.0423	0.0752	0.0125	1.0e-10	1.2e-15	3.0e-16
2	3	0.1657	0.1671	0.2137	0.1760	0.1826	0.1760
1	0	0.0565	0.1003	0.0167	2.3e-10	9.2e-16	5.7e-16
2	4	0.2203	0.2221	0.2845	0.2341	0.2429	0.2341
1	0	0.0706	0.1254	0.0209	3.4e-10	6.5e-16	2.7e-15
2	5	0.2762	0.2785	0.3561	0.2934	0.3044	0.2934
1	1	0.0396	0.0466	0.0350	0.0415	0.0571	0.0415
2	0	0.0525	0.0767	0.0004	1.2e-10	2.9e-16	2.9e-13
1	2	0.0793	0.0932	0.0701	0.0830	0.1142	0.0830
2	0	0.1050	0.1534	0.0008	2.5e-10	5.8e-16	5.9e-13
1	3	0.1189	0.1398	0.1050	0.1245	0.1713	0.1245
2	0	0.1575	0.2302	0.0012	2.3e-10	5.1e-16	2.6e-12
1	4	0.1586	0.1865	0.1401	0.1660	0.2284	0.1660
2	0	0.2101	0.3069	0.0016	5.1e-10	1.1e-15	1.1e-12
1	5	0.1982	0.2331	0.1751	0.2075	0.2855	0.2075
2	0	0.2625	0.3837	0.0021	3.2e-10	6.8e-16	4.2e-12

**Table 5.2:** joint position  $\mathbf{q}$  response comparison: overshoot [rad] with different reference steps (target [rad]) for each output (n)

n	Target	Original	Static	SVD	Ideal	Simplified	Inverted
1	0	0.0263	0.0516	0.0708	1.4e-10	8.8e-16	5.5e-16
2	1	0.0791	0.1027	0.0417	0.0407	0.0387	0.0407
1	0	0.0526	0.1032	0.1417	1.5e-10	1.7e-15	1.1e-14
2	2	0.1581	0.2055	0.0835	0.0814	0.0774	0.0814
1	0	0.0789	0.1548	0.2125	7.6e-10	5.6e-15	3.7e-15
2	3	0.2372	0.3083	0.1253	0.1222	0.1162	0.1222
1	0	0.1052	0.2064	0.2821	1.3e-09	1.0e-15	3.6e-15
2	4	0.3163	0.4111	0.1671	0.1629	0.1549	0.1629
1	0	0.1315	0.2580	0.3542	2.0e-10	2.6e-15	1.3e-14
2	5	0.3954	0.5139	0.2089	0.2036	0.1936	0.2036
1	1	0.0643	0.1411	0.0224	0.0328	0.0489	0.0328
2	0	0.1932	0.2732	0.0004	4.9e-10	8.8e-16	6.0e-13
1	2	0.1287	0.2823	0.0448	0.0656	0.0979	0.0656
2	0	0.3865	0.5464	0.0009	9.9e-10	1.7e-15	1.2e-12
1	3	0.1931	0.4235	0.0672	0.0985	0.1469	0.0985
2	0	0.5798	0.8196	0.0014	2.3e-09	6.2e-15	2.3e-11
1	4	0.2575	0.5647	0.0897	0.1313	0.1958	0.1313
2	0	0.7730	1.0928	0.0019	1.9e-09	3.5e-15	2.4e-12
1	5	0.3219	0.7059	0.1121	0.1642	0.2448	0.1642
2	0	0.9663	1.3661	0.0024	3.0e-09	1.0e-14	3.9e-11

It is possible to notice that in the original coupled system, applying the step to the second input, not only the second output reacts following the specified target, but also the first signal is affected by the other input, showing a strong coupling, so much as the first joint is moved in the opposite direction, causing a contrary response with "negative overshoot" of almost 0.18. This happens because the relative controller is not designed on the cross transfer function, and consequently also the second axis itself receives the effect of the impulse by the counter-reaction of the first link so that the design specifics are not met. For what concerns the Static Decoupler, it is proved that the coupling gets even exacerbated, as expected from the previous coupling analysis, worsening the control performance. The SVD Decoupled System behaves better since the control specifics, as shown in

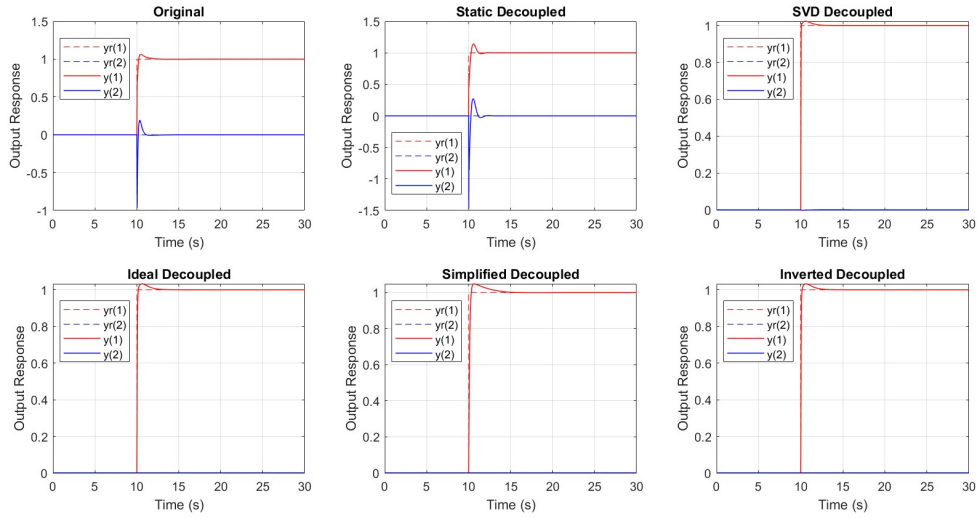
**Table 5.3:** joint position  $\mathbf{q}$  response comparison: settling time with different reference steps (target [rad]) for each output (n)

n	Target	Original	Static	SVD	Ideal	Simplified	Inverted
1	0	4.0515	1.9499	2.3273	-	-	-
2	1	0.8768	1.0448	1.9770	1.3540	1.5402	1.3540
1	0	5.1576	2.7302	2.7008	-	-	-
2	2	0.8764	1.0446	1.9767	1.3538	1.5404	1.3538
1	0	5.6676	2.8864	2.8999	-	-	-
2	3	0.8764	1.0446	1.9767	1.3538	1.5404	1.3538
1	0	5.9987	3.0001	3.0331	-	-	-
2	4	0.8765	1.0447	1.9768	1.3540	1.5404	1.3540
1	0	6.2423	9.2495	3.1314	-	-	-
2	5	0.8764	1.0446	1.9767	1.3538	1.5404	1.3538
1	1	1.2596	1.0812	0.8216	1.3040	2.4705	1.3040
2	0	4.3630	2.7994	0.8324	-	-	-
1	2	1.2596	1.0812	0.8216	1.3040	2.4705	1.3040
2	0	5.3456	2.9168	1.1894	-	-	-
1	3	1.2596	1.0812	0.8216	1.3040	2.4705	1.3040
2	0	5.9387	4.4376	5.0736	-	-	-
1	4	1.2596	1.0812	0.8216	1.3040	2.4705	1.3040
2	0	6.3666	5.7061	5.7325	-	-	-
1	5	1.2596	1.0812	0.8216	1.3040	2.4705	1.3040
2	0	6.7033	6.8986	6.1659	-	-	-

the reported tables, get respected: when the second link moves, the response of the first one shows a spike (greater overshoot over 0 w.r.t. the original) with a high harmonic component which settles almost immediately (lower settling time w.r.t. the coupled system), due to the fact that the two static compensators work well in the low frequency bandwidth. In the case of Ideal Decoupler applied to the LTI system, it is possible to appreciate the benefits of the theoretically perfect diagonalization, which works pretty well, since the first link is not affected at all by the second motor input, remaining fixed as testified by the null overshoot and zero tracking error: this validates the design specifications on the second SISO system. About the Simplified Decoupled system, it is possible to observe that the decoupling is achieved, since the tracking error on the first axis is practically zero.

Even though it is slightly worse than ideal, it is more robust in case of uncertainties. However, the diagonal response of the second axis is different from the one designed for the tuning of the original plant regulator, so that a new tuning was necessary in order to impose the right specifications. Finally, the controlled Inverted decoupled system shows exactly the same trends like the ideal case, as expected, so that the coupling from the second link to the first one is completely canceled out, proving the same performance in terms of Root Mean Square Error, overshoot and settling time, i.e. following exactly the design specifications and desired responses for the second axis.

Conversely, keeping the second axis fixed and providing as reference a unitary step as input only to the first motor controller, it is possible to make similar observations but with reversed roles, as demonstrated by the unit step response shown in Figure 5.24, where we notice similar coupling characteristics and tracking performance



**Figure 5.24:** Response plots of the transfer function matrix  $\mathbf{q}/\tau_m$  providing reference step values 1 rad for axis 1 and 0 rad for axis 2

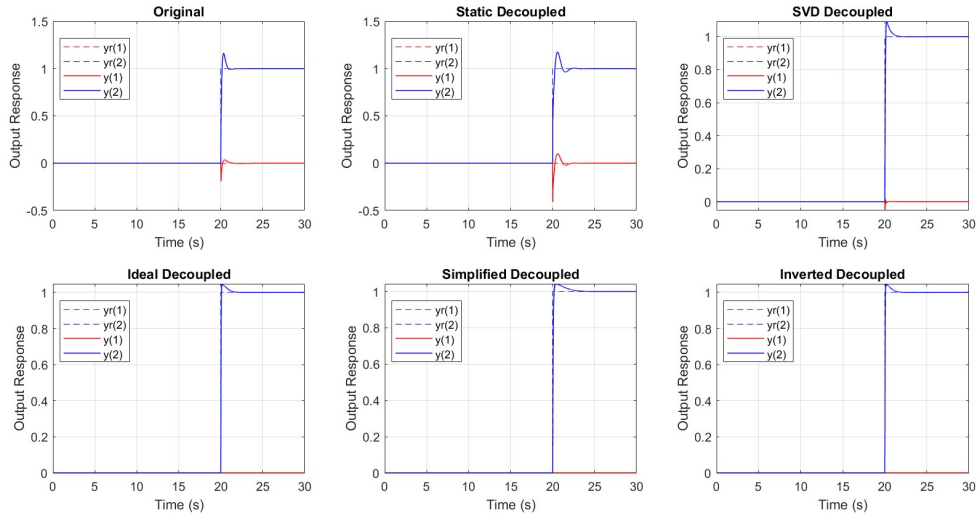
as proved by RMSE, overshoot and settling time values in the above cited tables, which encompass both scenarios where one remains stationary while the other is in motion. However, let us remark that two diagonal transfer functions exhibit really different dynamic behaviours, so that it is worth analyzing the study as axes vary.

#### 5.4.2 Motor Step Response of $\theta/\tau_m$ Dynamics

To control the motor response, a PID controller is fine-tuned on the diagonal motor transfer functions to fit the specified performance criteria: overshoot limited to 1.5% and settling time of 0.1 s.

Maintaining the first motor position at rest while applying a unitary step input to the second controller, the resulting shaft position responses are illustrated in Figure 5.25, while relevant performance indicators are presented in Table 5.4 for RMSE, Table 5.5 for overshoot, and Table 5.6 for settling time. The obtained results exhibit similarities with those of the joint dynamics, reaffirming the presence of coupling effects. In the original system, an input to the second motor produces a notable interaction with the first motor shaft, causing unintended displacement and deviation from the expected trajectory.

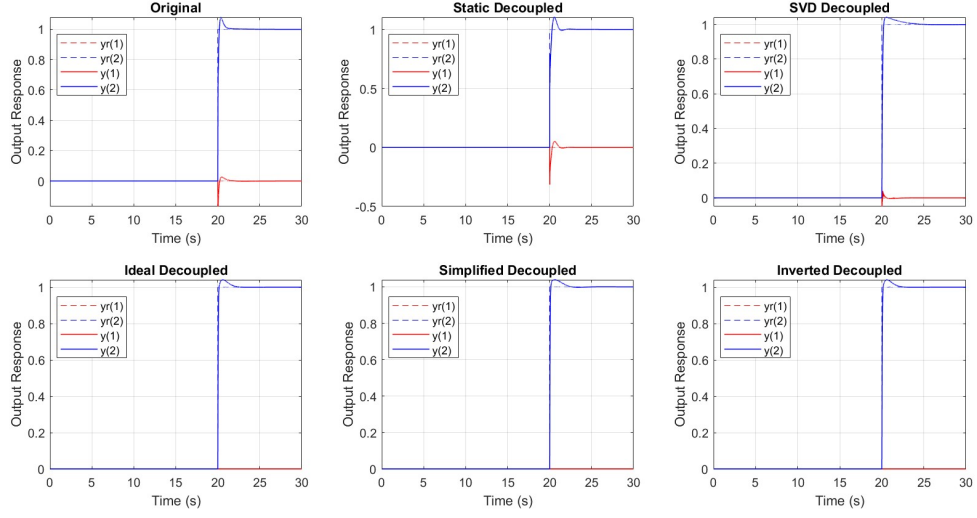
Keeping the first rotor fixed while applying a unitary step input to the second motor, gives the shaft position responses illustrated in Figure 5.25, whose performance are detailed in Tables 5.4 for RMSE, 5.5 for overshoot, and 5.6 for settling time.



**Figure 5.25:** Response plots of the transfer function matrix  $\theta/\tau_m$  providing reference step values 0 rad for motor 1 and 1 rad for motor 2

The observed trends closely resemble those seen in the joint dynamics, confirming the presence of coupling effects, where an input to the second motor not only affects its own response but also induces unintended displacement in the first motor shaft, deviating from the expected trajectory.

Applying the Static Decoupler further exacerbates these interactions, deteriorating control performance, whereas the SVD Decoupled System mitigates cross-coupling effects by significantly reducing unwanted interactions; however, the first motor still experiences a transient spike, though it quickly settles due to effective suppression of low-frequency disturbances. The Ideal Decoupler ensures complete decoupling, keeping the first motor entirely unaffected by the second motor's input, as confirmed by the null overshoot and zero tracking error, validating the effectiveness of the decoupling approach, while the Simplified Decoupler, achieving a similar effect,



**Figure 5.26:** Response plots of the transfer function matrix  $\mathbf{q}/\tau_m$  providing reference step values 0 rad for axis 1 and 1 rad for axis 2

introduces minor deviations that necessitate retuning of the controller, whereas the Inverted Decoupled System performs identically to the Ideal Decoupler, fully eliminating coupling and naturally meeting the specified performance criteria.

When applying a step input to the first motor while holding the second one stationary, an analogous situation happens, though with reversed roles. As shown in Figure 5.27, the responses exhibit similar coupling characteristics and tracking accuracy, as verified by the numerical results in the tables looking at the rows where the value for the second reference is zero, i.e. it should stay stationary. Specifically, it can be observed that the RMSE and overshoot measures of the decentralized controlled system are obtained as linear combinations of the inputs, while those of the decoupled systems grow linearly with respect to the reference value, highlighting the independent linear behavior of each output from the other inputs. Regarding the settling time, observe that it remains unchanged as the step changes.

Eventually, let us notice that the design of a SISO controller for each motor is more straightforward, allowing to achieve more easily better performance, managing to obtain lower overshoot and settling time with respect to the joints' regulators. And this rule of thumb ensures that the more the output is upstream of the input, the more straightforward the design becomes, so it is easier tuning velocity controllers with respect to position regulators and motor controllers than joints' ones.

**Table 5.4:** motor shaft position  $\theta$  response comparison: RMSE [rad] with different reference steps (target [rad]) for each output (n)

n	Target	Original	Static	SVD	Ideal	Simplified	Inverted
1	0	0.0231	0.0533	0.0047	3.2e-11	1.8e-16	3.9e-16
2	1	0.1064	0.1036	0.1019	0.0893	0.1101	0.0893
1	0	0.0463	0.1065	0.0094	5.4e-11	4.0e-16	2.5e-16
2	2	0.2139	0.2082	0.2048	0.1799	0.2213	0.1799
1	0	0.0696	0.1601	0.0141	5.8e-11	1.0e-15	3.0e-16
2	3	0.3209	0.3125	0.3081	0.2699	0.3325	0.2699
1	0	0.0927	0.2130	0.0187	9.7e-11	6.1e-16	5.7e-16
2	4	0.4275	0.4167	0.4099	0.3595	0.4429	0.3595
1	0	0.1161	0.2669	0.0235	7.1e-11	1.1e-15	7.7e-16
2	5	0.5361	0.5225	0.5142	0.4513	0.5556	0.4513
1	1	0.0546	0.0666	0.0572	0.0655	0.0768	0.0655
2	0	0.0718	0.0925	0.0019	3.2e-11	1.0e-16	2.4e-16
1	2	0.1095	0.1335	0.1145	0.1313	0.1541	0.1313
2	0	0.1440	0.1856	0.0039	8.1e-11	3.3e-16	7.1e-16
1	3	0.1658	0.2012	0.1734	0.1987	0.2327	0.1987
2	0	0.2164	0.2789	0.0059	8.9e-11	5.7e-16	1.1e-15
1	4	0.2210	0.2684	0.2310	0.2647	0.3100	0.2647
2	0	0.2884	0.3716	0.0078	1.4e-10	1.2e-15	9.3e-16
1	5	0.2766	0.3357	0.2892	0.3315	0.3884	0.3315
2	0	0.3612	0.4655	0.0098	1.0e-10	1.7e-15	1.2e-15

## 5.5 Decentralized and Decoupled LPV Dynamics

The linear decoupling strategies were born for Linear Time-Invariant system and have been recently extended for Linear Parameter-Varying systems, in particular for the static approaches. However these steady-state strategies are not suitable for a complex system as a robot, as empirically proved in the previous sections, thus requiring the necessity to rather realize linear dynamic decouplers where a parameter vector varies, which is the aim of this simulative thesis. The previous considerations on LTI decoupling can be generalized also to an LPV framework, requiring to extend the assumption of stability and realizability of the decoupler for each value of the parameter vector, in this case  $\mathbf{q}$ . In order to evaluate the systems'



**Table 5.5:** motor shaft position  $\theta$  response comparison: overshoot [rad] with different reference steps (target [rad]) for each output (n)

n	Target	Original	Static	SVD	Ideal	Simplified	Inverted
1	0	0.0354	0.1008	0.0689	2.9e-10	9.9e-16	2.1e-15
2	1	0.1628	0.1743	0.0929	0.0475	0.0424	0.0475
1	0	0.0709	0.2017	0.1380	3.6e-10	2.6e-15	1.5e-15
2	2	0.3257	0.3487	0.1859	0.0950	0.0850	0.0950
1	0	0.1064	0.3026	0.2070	3.0e-10	1.5e-15	2.2e-15
2	3	0.4885	0.5230	0.2789	0.1424	0.1274	0.1424
1	0	0.1419	0.4035	0.2759	6.4e-10	5.3e-15	1.9e-15
2	4	0.6514	0.6974	0.3718	0.1899	0.1700	0.1899
1	0	0.1773	0.5044	0.3446	7.7e-10	7.0e-15	6.6e-15
2	5	0.8143	0.8717	0.4647	0.2371	0.2122	0.2371
1	1	0.0555	0.1599	0.1144	0.0445	0.0438	0.0445
2	0	0.2170	0.2658	0.0167	1.5e-10	1.5e-15	9.4e-16
1	2	0.1110	0.3199	0.2287	0.0892	0.0876	0.0892
2	0	0.4341	0.5316	0.0335	3.0e-10	3.1e-15	6.2e-15
1	3	0.1666	0.4799	0.3432	0.1338	0.1315	0.1338
2	0	0.6513	0.7975	0.0504	9.9e-10	1.4e-15	1.3e-14
1	4	0.2221	0.6399	0.4576	0.1785	0.1754	0.1785
2	0	0.8685	1.0633	0.0672	6.5e-10	1.0e-14	3.9e-15
1	5	0.2777	0.7999	0.5720	0.2229	0.2192	0.2229
2	0	1.0854	1.3292	0.0840	1.3e-09	1.5e-14	5.5e-15

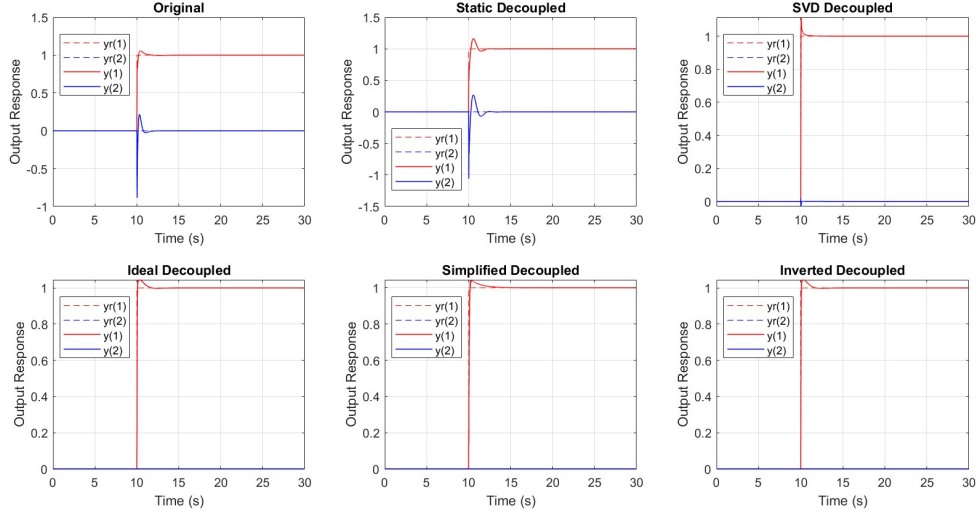
input-output behaviour a typical frequency representation used in literature for LPV systems is the Bode plot obtained by superimposing different dynamics as a system parameter function evolves, which in this case corresponds exactly to the trajectory in joint space. So specifying a function of  $\mathbf{q}$  it is possible to understand the behaviour of the system since the transfer matrix function varies with respect to the parameter also. Thus, as a possible example, the parameter-varying Bode plots of decoupled systems are illustrated in Figure 5.28 for the Static, in Figure 5.29 for the SVD-based, in Figure 5.30 for the Ideal, in Figure 5.31 for the Simplified and finally in Figure 5.32.

All these plots show a property which has not been investigated yet: the robustness. It is possible in fact to notice that all the theoretical considerations are verified

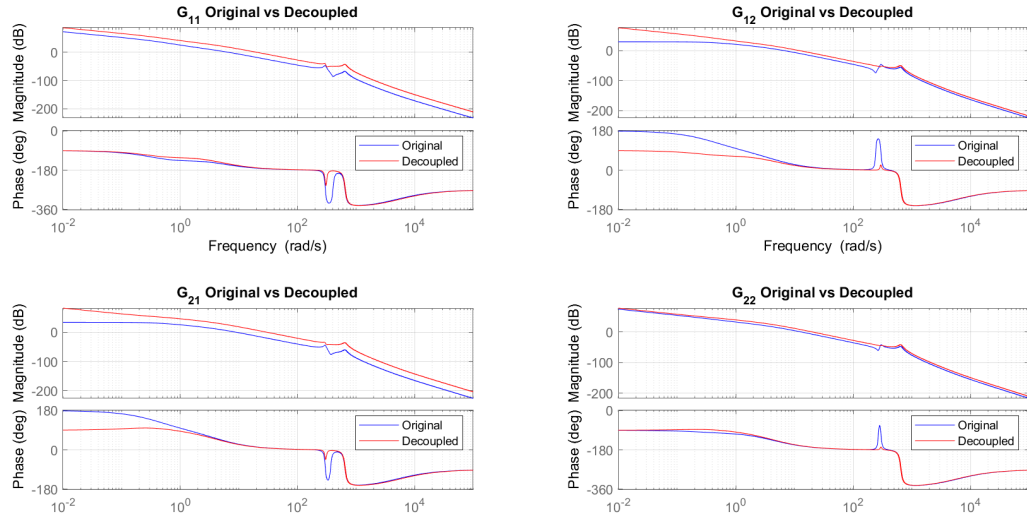
**Table 5.6:** motor shaft position  $\theta$  response comparison: settling time with different reference steps (target [rad]) for each output (n)

n	Target	Original	Static	SVD	Ideal	Simplified	Inverted
1	0	3.6502	2.8591	0.3315	-	-	-
2	1	0.7715	1.8658	0.9038	0.8009	1.5690	0.8009
1	0	4.0320	3.7382	1.2113	-	-	-
2	2	0.7724	1.8678	0.9043	0.8016	1.5683	0.8016
1	0	4.1980	3.8863	1.4886	-	-	-
2	3	0.7721	1.8680	0.9044	0.8014	1.5700	0.8014
1	0	4.2925	3.9678	1.6847	-	-	-
2	4	0.7722	1.8689	0.9040	0.8015	1.5704	0.8015
1	0	4.3595	4.0199	1.8394	-	-	-
2	5	0.7713	1.8682	0.9023	0.8038	1.5685	0.8038
1	1	0.9416	1.8776	0.2905	0.9674	1.2113	0.9674
2	0	2.4996	3.8181	0.1756	-	-	-
1	2	0.9399	1.8751	0.2901	0.9654	1.2112	0.9654
2	0	2.7506	3.9873	1.2183	-	-	-
1	3	0.9389	1.8750	0.2913	0.9647	1.2135	0.9647
2	0	2.8869	4.0588	2.0594	-	-	-
1	4	0.9394	1.8745	0.2916	0.9644	1.2111	0.9644
2	0	2.9707	4.0989	2.5409	-	-	-
1	5	0.9417	1.8748	0.2908	0.9675	1.2131	0.9675
2	0	3.0373	4.1267	2.9033	-	-	-

since the Ideal decoupler, which seems to be the best one, is actually bad because of the high sensitivity function, which can be appreciated from Figure 5.30 that shows high variability on dynamics and so on the related compensation. While the inverted exhibits a similar behaviour, even though it is not so accentuated, and moreover, thanks to the stability of the robotic plant its marginal stability is ensured, so that the diagonal transfer functions appear practically the same with respect to the original ones. In the end, the Simplified decoupler shows to be the best trade-off among performance and robustness on the diagonalizing compensation, showing a low sensitivity to the parameters, as expected from theory.

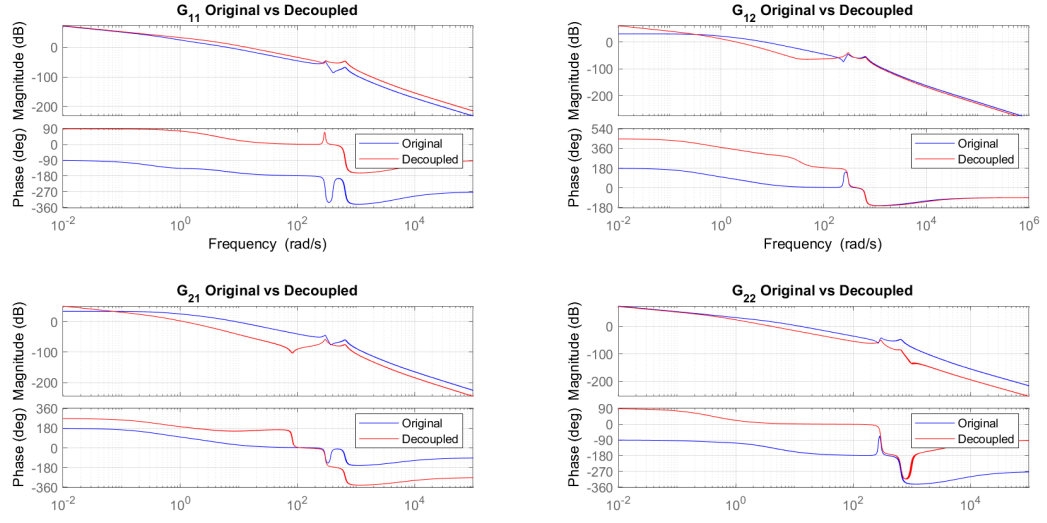


**Figure 5.27:** Response plots of the transfer function matrix  $\theta/\tau_m$  providing reference step values 1 rad for motor 1 and 0 rad for motor 2

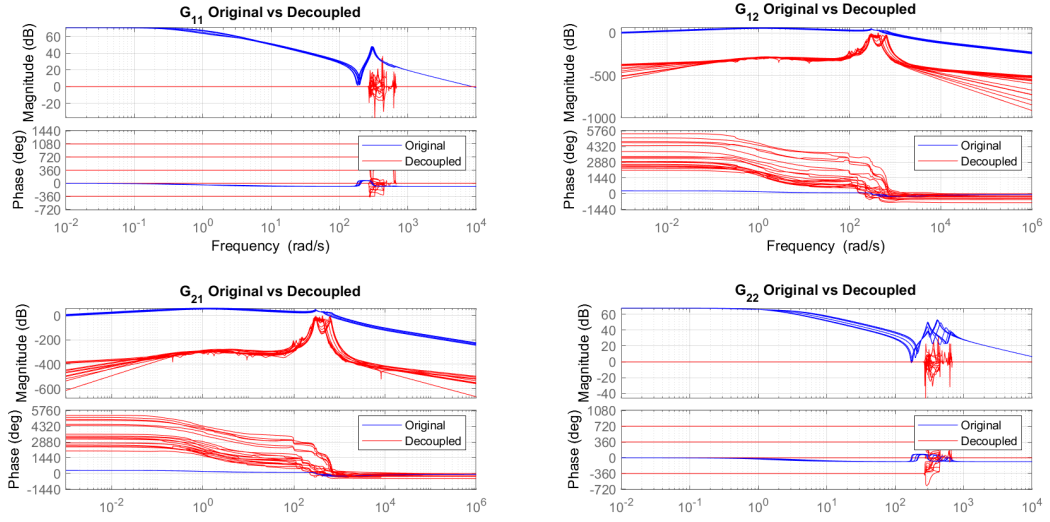


**Figure 5.28:** Bode plots of the LPV Static decoupled transfer function matrix  $q/\tau_m$  for a specified trajectory

Finally, regarding the decoupling performance and tracking benefits associated with the specific type of algorithm in the LPV case, the previously discussed LTI examination remains valid in this scenario by extension. In other words, the properties outlined within the LTI framework can be effectively found also in the LPV context.



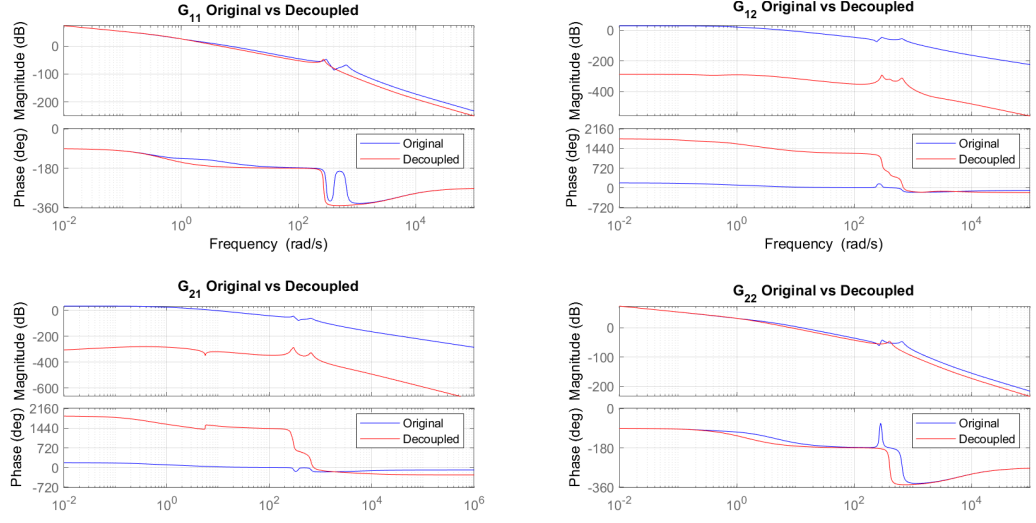
**Figure 5.29:** Bode plots of the LPV SVD decoupled transfer function matrix  $\mathbf{q}/\boldsymbol{\tau}_m$  for a specified trajectory



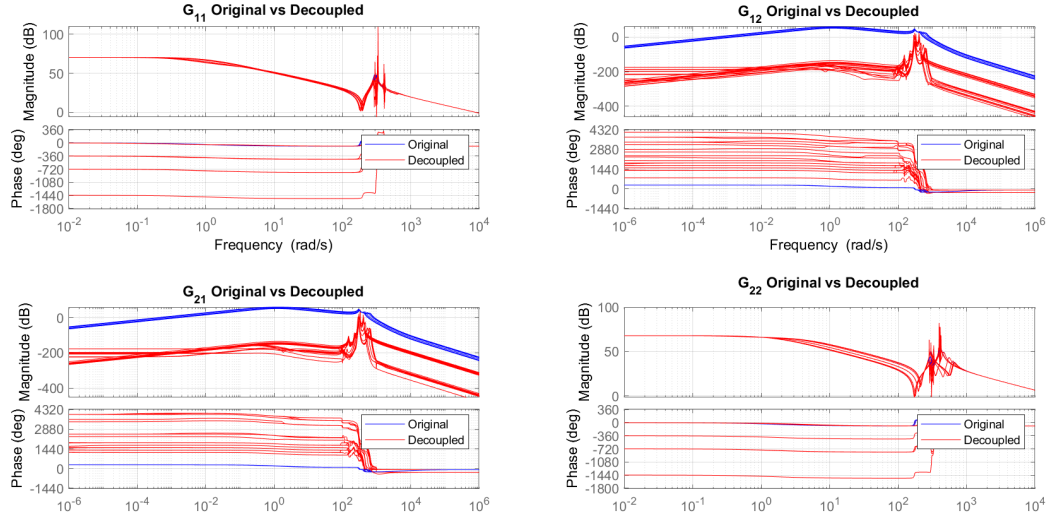
**Figure 5.30:** Bode plots of the LPV Ideal decoupled transfer function matrix  $\mathbf{q}/\boldsymbol{\tau}_m$  for a specified trajectory

## 5.6 Control response of Decentralized and Decoupled LPV Dynamics

We demonstrated that the linear decoupling techniques are quite valid to diagonalize the dynamics, especially the Simplified and the Inverted approaches, which represent the best trade-off, so that if the robot pose changes slowly in time, also the



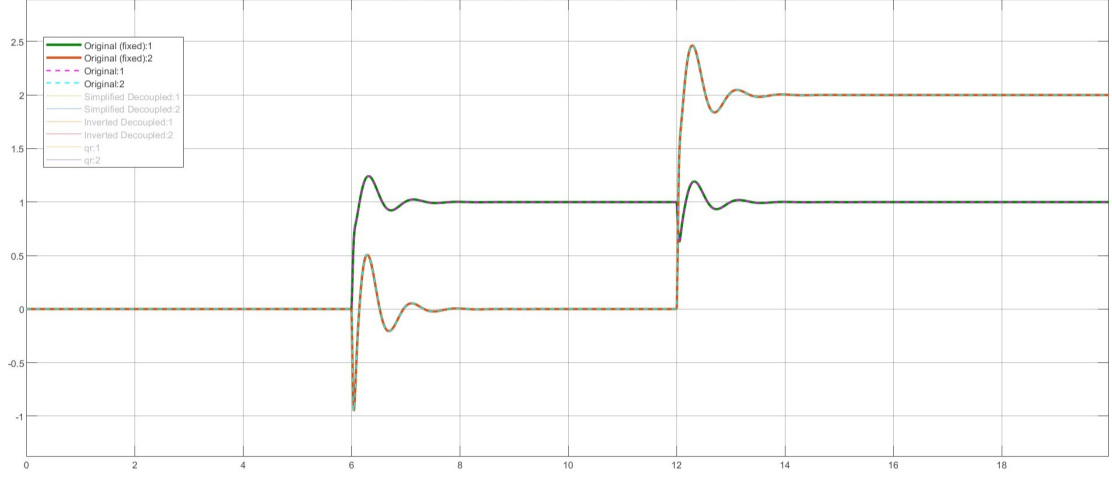
**Figure 5.31:** Bode plots of the LPV Simplified decoupled transfer function matrix  $\mathbf{q}/\boldsymbol{\tau}_m$  for a specified trajectory



**Figure 5.32:** Bode plots of the Inverted decoupled LPV transfer function matrix  $\mathbf{q}/\boldsymbol{\tau}_m$  for a specified trajectory

inertia matrix remains approximately constant in this temporal interval, which is experimentally verified by comparing the output response in closed-loop of the LTI and LPV systems with the same controllers. By imposing different trajectories it is possible to assess that the robot LTI model approximates the LPV model sufficiently well, thus the application of a linear decoupler designed on the robot dynamics evaluated on a specific pose provides sufficiently good performance if the inertia

matrix does not vary much, as visible in the response in figure 5.33 where the two models have been compared, showing only negligible differences. However this LTI



**Figure 5.33:** Step response of the LPV system vs the LTI system, around  $\mathbf{q}_0$

decoupling method shows its limit when the robot tracks trajectories for which the joint configuration is far from the configuration on which the decoupler is designed, which could cause a wrong compensation. So, an enhancement could be a gain scheduling decoupling approach. On this idea a first LPV generalization has been realized, computing offline the parameter-varying decoupling transfer functions supposing the trajectory is planned, leading to good results as far as the joints are close to trajectory. Anyway this is a too restrictive suboptimal approach, thus the efforts have been concentrated to develop an online linear parameter-varying decoupler. Supposing the knowledge of the LPV plant, it is possible to devise a really efficient online compensator by working through transfer functions in vector form, noticing that the so-called simplified decoupling functions  $D_{ij} = G_{ij}/G_{ii}$  can be obtained performing convolutions between the numerator and denominator vector of coefficients, which are known at each joint configuration. Specifically, these convolutions are expressed as

$$\text{num}_{D_{ij}} = \text{num}_{G_{ij}} * \text{den}_{G_{ii}} \quad (5.38)$$

$$\text{den}_{D_{ij}} = \text{den}_{G_{ij}} * \text{num}_{G_{ii}} \quad (5.39)$$

where  $\text{num}_{D_{ij}}$  and  $\text{den}_{D_{ij}}$  represent the numerator and denominator of  $D_{ij}$  in vector form, respectively, while the  $*$  is the convolution product, defined for two vectors  $\mathbf{f}$  and  $\mathbf{g}$  like

$$(f * g)[n] = \sum_{k=0}^{N-1} f[k] \cdot g[n - k] \quad (5.40)$$

Based on the latter intuition, the Linear Parameter-Varying Simplified and Inverted Decoupling are realized, moreover in a computationally efficient and easy implementation, as shown in Figures 5.34 and 5.35, which represent the block diagrams used for the simulation of the Simplified and Inverted decouplers, respectively, each one exhibiting the typical theoretical scheme with the two cross-compensators and even showing the parameter-varying dependence.

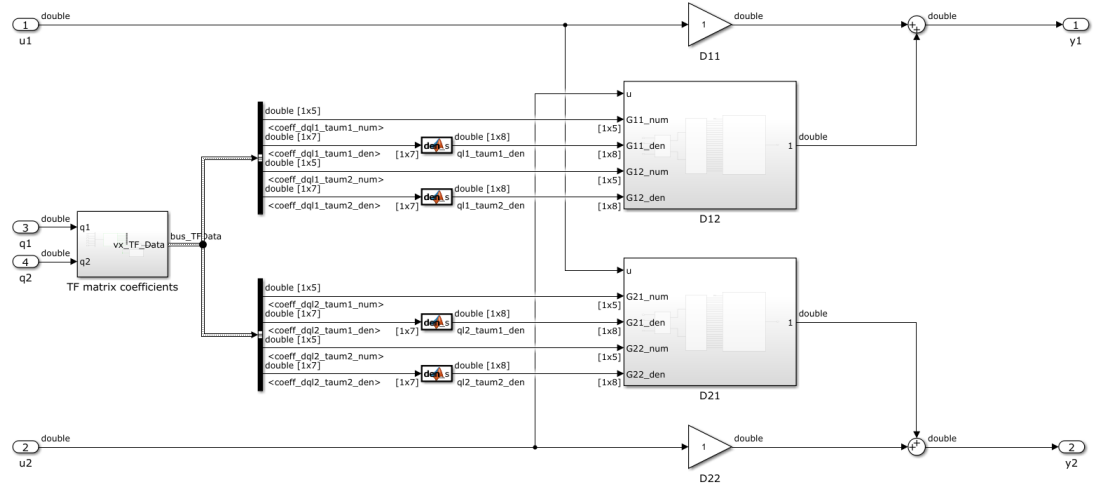


Figure 5.34: LPV Simplified decoupler

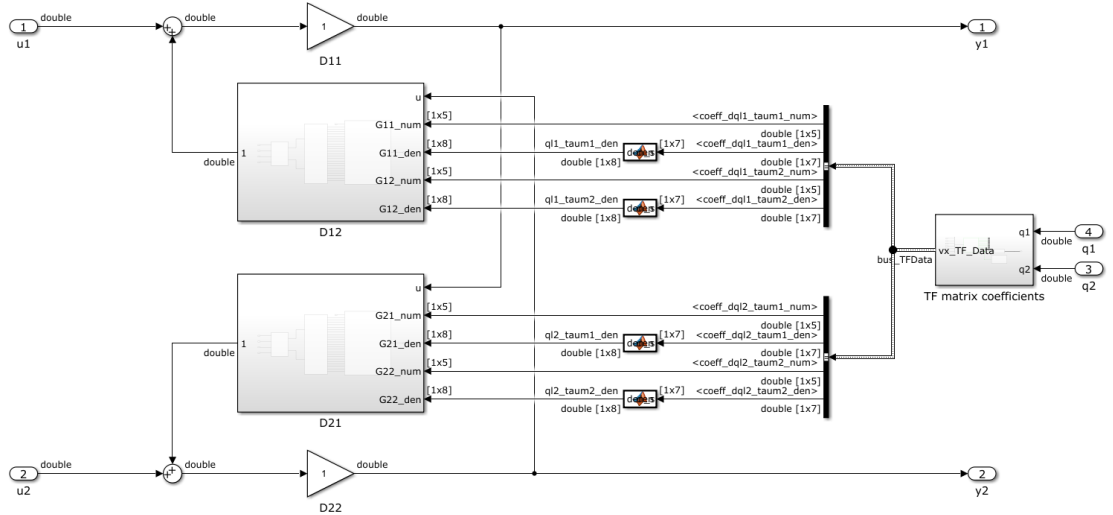
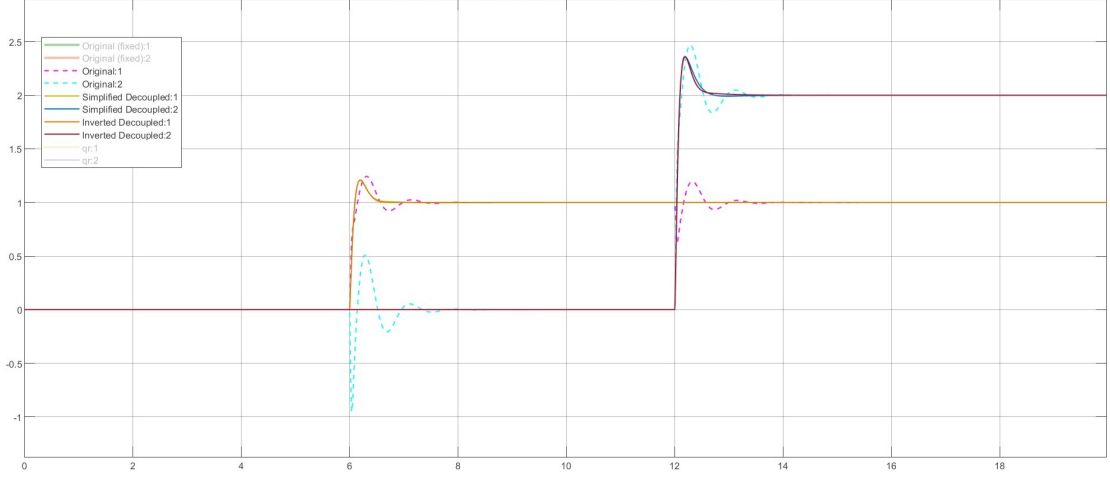


Figure 5.35: LPV Inverted decoupler

Different simulations with these LPV decouplers have been experimented, and again the step response is considered, being suggested as a fundamental test in

robotic industry. In Figure 5.36 two reference steps are provided separately for each motor controller at different time, in parallel with the plant, the Simplified decoupled system and the Inverted one, in order to evaluate the dynamic influence of axis 1 on axis 2 and vice versa.



**Figure 5.36:** Step response of the LPV, Simplified and Inverted decoupled systems

As a qualitative observation, it can be noted that the original system's response is significantly influenced by coupling effects. At  $t = 6$  s, the first axis is actuated, while the second axis, expected to remain stationary, moves in the opposite direction with a half-radian overshoot introducing oscillations that eventually stabilize after approximately 3 s. Similarly, at  $t = 12$  s, the second axis receives a control input to reach a step value of 2 radians, while, the first axis, despite reaching steady-state, is affected by the second joint and exhibits a counter-phase sinusoidal behavior, settling within the same timeframe as before. For further details, control performance metrics such as tracking error, overshoot, and settling time are summarized in Table 5.7, which support the latter observations and provide quantitative validation of the described dynamics.

As expected from the LPV transfer function matrices, the two decouplers, providing a diagonalizing torque for each link, manage to reduce the cross-link interactions and enhance the overall control performance on both diagonal and off-diagonal behaviours.

In conclusion, both the Simplified and Inverted LPV Decouplers guarantee a perfect decoupling of each input torque with respect to the joint position of the other axes, at least on the linear dynamics of the robot manipulator.



**Table 5.7:** LPV robot model  $\mathbf{q}$  response comparison: RMSE, overshoot and settling time with reference steps (t.) of 1 rad for axis 1 and 2 rad for axis 2

Metrics	T.	Original	Static	SVD	Ideal	Simplified	Inverted
RMSE	1	0.0444	0.0586	0.0355	0.0415	0.0571	0.04150
	2	0.0938	0.1096	0.1004	0.0826	0.0857	0.0826
overshoot	1	0.0643	0.1408	0.1416	0.0328	0.0489	0.0328
	2	0.1581	0.2056	0.0835	0.0814	0.0774	0.0814
settling time	1	1.0516	1.0821	0.3479	1.3040	1.4705	1.3040
	2	0.8763	1.0450	1.9764	1.3542	1.5402	1.3542

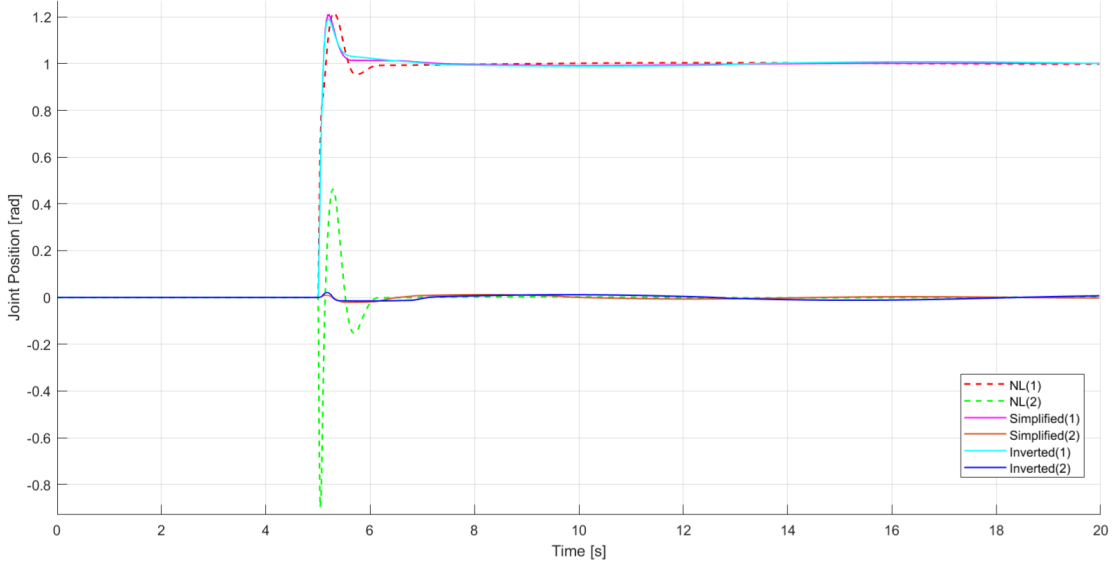
## 5.7 Control response of Decentralized and LPV Decoupled Nonlinear Dynamics

The previous analysis in Sections 5.3 and 5.4 studied Linear Time-Invariant Decouplers applied to the LTI model of the robotic system around a specific pose, while in Sections 5.5 and 5.6 we further examined Linear Parameter-Varying compensators to decouple the LPV model. However, let us recall that the dynamics of the robot manipulator is highly nonlinear (5.1). Therefore, the goal is to apply the linear parameter-varying decouplers to the nonlinear model in order to achieve a partial decoupling, only on the linear dynamics (5.3). In fact, the design and implementation of a linear decoupler is more feasible than relying on a computationally heavy nonlinear law, even though the compensation itself could not be as perfect as a more complex strategy.

The simulations are carried out by applying LPV Simplified Decoupling and LPV Inverted Decoupling as pre-compensators to the nonlinear plant, alongside the open-loop dynamics of the robot, and by closing a feedback loop for each one. In particular, for each scenario, a decentralized PID controller has been designed on the diagonal transfer functions computed around a specific pose, in this case  $q_i = 0 \forall i$ . This control approach is not optimal for a robotic manipulator, as the gains are fixed even though the inertia matrix changes. However, it is convenient and easier to design than a gain-scheduling PID controller, but above all it is sufficient to illustrate the effectiveness of linear decoupling. In order to analyze the effect of the LPV decouplers on the nonlinear robot plant, regulation at a specific step is considered, specifically providing a torque on the first axis while keeping the second axis fixed. To evaluate the decoupling performance, we analyze the Root Mean Square Error, the overshoot (in absolute value), and the settling time (with a 2% tolerance, measured from the step time), comparing the reference signal to

the output of the nonlinear system under Decentralized control, LPV Simplified Decoupling, and LPV Inverted Decoupling control.

Running a 20 s simulation where a unitary step is fed into the first controller at time  $t = 5$  s, it is possible to qualitatively assess the goodness of the response for each system, as shown in Figure 5.37.



**Figure 5.37:** Step response of the Decentralized, Simplified decoupled and Inverted decoupled control systems on the nonlinear robot plant

When the reference is applied to the first controller, both axes receive the control torque. Consequently, the first axis follows a response similar to the linear approximation on which the PID is tuned, while the second axis, ideally fixed, instead moves in a counter-response in the opposite direction. This initiates an oscillation around the zero reference, which eventually settles as the first axis also reaches its steady state.

From a quantitative point of view Table 5.8 reports the tracking error, Table 5.9 shows the overshoot in absolute value and Table 5.10 lists the relative settling times, for the three examined systems (columns) and separately for each channel, feeding different input values (rows). Firstly, under decentralized control, when the first input is applied, the relative axis exhibits an overshoot of about 21% and converges in approximately 1 s, with a transient tracking error of 0.0756 rad. However, the effect of coupling on the second axis, caused by the motion of the first axis, is clearly evident, producing an extremely large counter-response with a "negative overshoot" of nearly 90% relative to the step, and settling in approximately 1 s as originally designed, but resulting in a high average quadratic error of 0.0797 rad

**Table 5.8:** joint position  $\mathbf{q}$  nonlinear response comparison: RMSE [rad] with different reference steps (target [rad]) for each output (n)

n	Target	Decentralized	LPV Simplified	LPV Inverted
1	1	0.0756	0.0785	0.0781
2	0	0.0797	0.0091	0.0112
1	2	0.1512	0.1574	0.1568
2	0	0.1419	0.0066	0.0156
1	3	0.2297	0.2371	0.2361
2	0	0.1801	0.0043	0.0164
1	4	0.3097	0.3182	0.3168
2	0	0.1871	0.0040	0.0166
1	5	0.3856	0.3966	0.3947
2	0	0.1989	0.0054	0.0167

**Table 5.9:** joint position  $\mathbf{q}$  nonlinear response comparison: absolute overshoot [rad] with different reference steps (target [rad]) for each output (n)

n	Target	Decentralized	LPV Simplified	LPV Inverted
1	1	0.2171	0.2123	0.1919
2	0	0.8984	0.0218	0.0225
1	2	0.4478	0.4329	0.4014
2	0	1.5359	0.0188	0.0284
1	3	0.7331	0.6529	0.6106
2	0	1.8339	0.0178	0.0355
1	4	0.9554	0.8725	0.8197
2	0	1.9158	0.0169	0.0376
1	5	1.0598	1.0923	1.0283
2	0	1.9749	0.0161	0.0380

even comparable to that of the first axis. In other words, since the controllers are not tuned on the off-diagonal transfer functions, the mutual interactions between different axes are not explicitly constrained, allowing each input to interfere with the other outputs. Thus, the aim is to evaluate a strategy for mitigating these mutual couplings through the application of the above compensators.

**Table 5.10:** joint position  $\mathbf{q}$  nonlinear response comparison: settling time [s] with different reference steps (target [rad]) for each output (n)

n	Target	Decentralized	LPV Simplified	LPV Inverted
1	1	0.9770	0.5608	0.9888
2	0	1.0114	1.1754	0.2133
1	2	0.8960	0.5201	0.6646
2	0	0.9320	-	-
1	3	0.8420	0.5095	0.5946
2	0	0.8875	-	-
1	4	0.8076	0.5045	0.5699
2	0	0.8586	-	-
1	5	0.8038	0.5028	0.5578
2	0	0.8616	-	-

Applying the LPV Decouplers to the nonlinear plant partially compensates for the robot dynamics. In fact, the LPV Simplified Decoupler and LPV Inverted Decoupler manage to reduce the inertial couplings that heavily affect the movement of each axis with respect to the others. The first axis exhibits a transient error comparable to that of the decentralized control, with 0.0785 rad for the Simplified Decoupler and 0.0781 rad for the Inverted Decoupler, showing both an overshoot of about 20%, comparable with the performance of decentralized control. Regarding steady-state performance of the moving axis, the settling time is similar to the decentralized one, being less than 1 s. In other words, there are not particular benefits by employing decouplers for the forced output.

However, for the second axis, which should remain stationary, the effectiveness of the decoupling strategies is evident, as the RMSE is reduced from 0.0797 rad to 0.0091 rad and 0.0112 rad, thanks to Simplified and Inverted decoupling. These approaches almost completely eliminate even the overshoot, reducing it from 89.8% to 2.18% and 2.25%, respectively. Regarding the steady-state metrics, the settling time remains approximately the same, but if the step is sufficiently large, it may not even be defined, as the response stays within the relative tolerance band.

Increasing the step value, similar trends can be observed. It is important to remember that, since the system is nonlinear, maintaining linearity in the response with respect to the step value is not guaranteed, as clear from the variations in the results for the decentralized control, which therefore requires a deep analysis. Nonetheless, it can be noted that the RMSE and overshoot of the controlled channel

keep a certain linearity as the step is higher. On the other hand, for the second axis in the coupled case, the RMSE and overshoot increase due to centrifugal and Coriolis effects, related to higher velocities, whereas in the decoupled case, the metrics remain practically unchanged, symbolizing the inhibition of mutual interactions regardless of the reference value, or at least showing less sensitivity.

Regarding the type of decoupling, it can be observed that both the Simplified and Inverted decouplers provide good cross-compensations. However, the Simplified decoupler results to be slightly better as it is more robust to parameter variations, both dynamically during transients, ensuring a lower RMSE and overshoot, and at steady state, as it oscillates less. This is due to its less aggressive compensation compared to the Inverted decoupler, which often tends to be more sensitive to changes in robot pose, as expected from theory.

In conclusion, the proposed approach is suboptimal since only a part of the overall dynamics is compensated, in particular the acceleration couplings generated from the off-diagonal contributions of the inertia matrix  $\mathbf{M}(\mathbf{q})$ , while the remaining nonlinear dynamics, mainly composed of the centrifugal and Coriolis terms  $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}})$ , gravitational components  $\mathbf{g}(\mathbf{q})$ , and the nonlinear static friction  $\mathbf{f}_l(\dot{\mathbf{q}})$ , here treated as disturbances, affect the real system response. In this discussion, it is possible to ignore the gravity term because it is feedforward compensated in the industrial control system. This means that the main source of nonlinear coupling originates from the centrifugal-Coriolis terms, which are quadratic with respect to velocity, while the static damping influences the direct dynamics of each link. Therefore, two possible situations can arise, corresponding, respectively, to the regulation and tracking control problems: for steady-state convergence, if the nonlinear friction effects are strong, damping the robot in the slow dynamics, the linear decoupling may provide overcompensation; whereas for fast dynamics, if the mutual velocity contributions are too high, the linear decouplers may fail to provide sufficient decoupling torques, undercompensating the system. For this reason, employing a linear decoupler for a nonlinear system is always a suboptimal approach, moreover depending on the relative magnitude of the nonlinear effects within the overall robot dynamics. Eventually, the linear decouplers prove to be quite effective, even though not perfect, providing a decoupling torque that essentially transforms the system into  $n = 2$  independent SISO systems which can be controlled separately by designing diagonal regulators, all while maintaining a low implementation cost.

## Chapter 6

# Conclusions and Future Works

### 6.1 Conclusions

This work has analyzed various decoupling strategies for robotic manipulators, with a focus on the realization of pre-compensators in both the time and frequency domains, in order to propose a solution to the issue posed by Comau about the mutual dynamic effects among the links of a robot manipulator. In detail, in the time domain the nonlinear feedback linearization technique applied to different models have been studied, while in the frequency domain diagonalizing methods based on linear parameter-varying pre-filters are proposed.

The simulations conducted on the Comau manipulator model demonstrated that linear static and dynamic decoupling techniques, extended as linear parameter-varying compensators, achieve the diagonalization of the transfer function matrix, based either on motor shaft or joint positions as output, thus significantly enhancing the overall system response, both for the controlled output but also for the other channels, reducing inter-axis interactions and simplifying controller synthesis.

However, the limitation of linear strategies is intrinsic to their nature, as the plant to be controlled is highly nonlinear. These strategies cannot entirely cancel out the input-output coupling but can compensate for its linear component, primarily affected by inertia, which represents the most significant dynamic influence. Consequently, only a portion of the overall system is decoupled, leaving the nonlinear dynamics to be treated as an external disturbance.

In fact, it is well known that the most effective strategies for reducing coupling are nonlinear ones. However, they require higher computational costs that cannot be supported by the majority of hardware racks employed for industrial robots, including those used by Comau.

## 6.2 Future Works

Future developments of this work could focus on four main directions, based on different aspects of the thesis.

First, on the model side, it would be interesting to extend the decoupling analysis to  $n$ -DOF manipulators with even more complex dynamics, including advanced friction effects and a more detailed modeling of elastic transmission.

Second, about the linear decoupling performance, further investigation into the robustness of different decoupling strategies against parametric uncertainties and external disturbances could provide valuable insights for industrial applications.

Third, for what concerns the simulations, the application of the nonlinear feedback control as a partial decoupling strategy to the Comau robot manipulator would be an interesting experiment, in order to compare the decoupling performance between the frequency and time domain approaches, which in any case could not be implemented due to the need of more expensive circuit boards.

Finally, it would be highly interesting to evaluate the decoupling techniques on a real robotic manipulator, validating their performance and robustness in practical scenarios and exploring their potential for further optimization in real-world applications.

# Appendix A

## Supplementary Tables

In the following section of the appendix, additional insights into other interesting simulations are presented. These simulations were not directly addressed in the main experimental part due to their secondary importance.

### A.1 Step Response of Joint velocity $\dot{q}/\tau_m$ and Motor velocity $\dot{\theta}/\tau_m$ Dynamics

Comau's control architecture is an independent-joint cascade control system composed by closing three nested feedback loops based on gain scheduled robust regulators (position, velocity and current). The linear controller synthesis consist of a modified pole-placement algorithm, tuned on 6 different poses of the robot manipulator. The idea of the cascade control is to impose specific behaviours at different level of the linearized dynamics, e.g. position and velocity. This approach enables precise control of the nonlinear plant, even when using linear techniques indeed.

For this reason, this study not only aims to analyze decoupling at the level of position variables, both motor and link, but also delves into decoupling for velocity variables. This approach allows for improved design of the controllers for the respective loops, benefiting from the interactions' compensation provided by the decoupling strategies.



**Table A.1:** link velocity  $\dot{\mathbf{q}}$  response comparison: RMSE [rad] with different reference steps (target [rad]) for each output (n)

n	Target	Original	Static	SVD	Ideal	Simplified	Inverted
1	0	0.0170	0.0283	0.0111	1.0e-11	1.4e-16	1.3e-16
2	1	0.0493	0.1243	0.0883	0.0476	0.0830	0.0476
1	0	0.0340	0.0566	0.0223	3.1e-11	2.2e-16	2.5e-16
2	2	0.0991	0.2489	0.1769	0.0957	0.1663	0.0957
1	0	0.0510	0.0849	0.0335	3.8e-11	3.4e-16	3.3e-16
2	3	0.1490	0.3734	0.2655	0.1439	0.2496	0.1439
1	0	0.0680	0.1132	0.0446	6.1e-11	5.2e-16	4.4e-16
2	4	0.1988	0.4979	0.3540	0.1919	0.3329	0.1919
1	0	0.0850	0.1415	0.0558	9.3e-11	5.4e-16	7.2e-16
2	5	0.2485	0.6224	0.4425	0.2399	0.4161	0.2399
1	1	0.0452	0.0524	0.0565	0.0376	0.0649	0.0372
2	0	0.0411	0.1785	0.0011	1.7e-11	2.4e-16	8.6e-14
1	2	0.0907	0.1052	0.1133	0.0756	0.1301	0.0749
2	0	0.0822	0.3571	0.0022	3.7e-11	5.0e-16	2.8e-13
1	3	0.1362	0.1580	0.1701	0.1137	0.1953	0.1126
2	0	0.1234	0.5357	0.0033	5.0e-11	8.9e-16	3.1e-13
1	4	0.1816	0.2106	0.2268	0.1516	0.2604	0.1501
2	0	0.1644	0.7142	0.0045	5.5e-11	1.0e-15	2.2e-13
1	5	0.2270	0.2633	0.2835	0.1895	0.3255	0.1877
2	0	0.2055	0.8928	0.0056	9.5e-11	1.2e-15	6.7e-13

**Table A.2:** link velocity  $\dot{\mathbf{q}}$  response comparison: overshoot [rad] with different reference steps (target [rad]) for each output (n)

n	Target	Original	Static	SVD	Ideal	Simplified	Inverted
1	0	0.0233	0.0446	0.1039	8.1e-11	5.8e-16	2.7e-15
2	1	0.0375	0.1724	0.0002	0.0246	0	0.0246
1	0	0.0467	0.0893	0.2076	2.7e-10	1.1e-15	3.9e-15
2	2	0.0750	0.3448	0.0004	0.0493	0	0.0493
1	0	0.0701	0.1340	0.3110	3.4e-10	1.1e-15	6.8e-15
2	3	0.1125	0.5172	0.0007	0.0739	0	0.0739
1	0	0.0935	0.1786	0.4151	4.7e-10	2.1e-15	8.1e-15
2	4	0.1500	0.6896	0.0009	0.0986	0	0.0986
1	0	0.1169	0.2233	0.5193	7.4e-10	9.5e-16	9.9e-15
2	5	0.1876	0.8621	0.0012	0.1232	0	0.1232
1	1	0.0460	0.1242	3.3e-05	0.0522	0	0.0501
2	0	0.1194	0.4789	0.0007	3.2e-10	2.2e-15	1.4e-12
1	2	0.0920	0.2484	6.6e-05	0.1046	0	0.1001
2	0	0.2389	0.9579	0.0014	6.9e-10	3.7e-15	3.8e-12
1	3	0.1380	0.3726	9.9e-05	0.1568	0	0.1501
2	0	0.3584	1.4368	0.0020	6.7e-10	9.6e-15	1.1e-11
1	4	0.1840	0.4968	0.0001	0.2092	0	0.2003
2	0	0.4779	1.9158	0.0027	8.0e-10	1.1e-14	6.6e-12
1	5	0.2300	0.6211	0.0001	0.2614	0	0.2503
2	0	0.5974	2.3948	0.0035	1.6e-09	1.4e-14	1.0e-11

**Table A.3:** link velocity  $\dot{\mathbf{q}}$  response comparison: settling time [s] with different reference steps (target [rad/s]) for each output (n)

n	Target	Original	Static	SVD	Ideal	Simplified	Inverted
1	0	3.0010	5.5331	1.4986	-	-	-
2	1	0.8550	3.7572	0.5885	0.2849	0.5199	0.2849
1	0	3.5314	6.9335	1.6615	-	-	-
2	2	0.8547	3.7575	0.5889	0.2847	0.5202	0.2847
1	0	3.8418	7.1161	1.7560	-	-	-
2	3	0.8547	3.7575	0.5889	0.2847	0.5201	0.2847
1	0	4.0621	7.2076	1.8223	-	-	-
2	4	0.8553	3.7574	0.5889	0.2847	0.5202	0.2847
1	0	4.2325	7.2638	1.8736	-	-	-
2	5	0.8553	3.7574	0.5890	0.2847	0.5202	0.2847
1	1	1.2997	3.4750	0.4766	0.7533	0.6390	0.7533
2	0	1.7236	8.5764	4.0023	-	-	-
1	2	1.2999	3.4747	0.4763	0.7531	0.6390	0.7531
2	0	1.7937	9.7893	5.6998	-	-	-
1	3	1.2998	3.4750	0.4763	0.7533	0.6387	0.7529
2	0	2.6250	0.1236	6.6926	-	-	-
1	4	1.2998	3.4751	0.4763	0.7535	0.6387	0.7530
2	0	2.8622	0.2529	7.3970	-	-	-
1	5	1.2995	3.4745	0.4764	0.7534	0.6387	0.7529
2	0	3.0327	0.3301	7.9435	-	-	-

**Table A.4:** motor velocity  $\dot{\theta}$  response comparison: RMSE [rad] with different reference steps (target [rad/s]) for each output (n)

n	Target	Original	Static	SVD	Ideal	Simplified	Inverted
1	0	0.0193	0.0638	0.0085	7.0e-12	7.9e-17	6.1e-17
2	1	0.0595	0.0741	0.0558	0.0465	0.0550	0.0465
1	0	0.0393	0.1279	0.018	1.4e-11	1.3e-16	1.3e-16
2	2	0.1251	0.1520	0.1172	0.1005	0.1153	0.1005
1	0	0.059	0.1926	0.0273	2.3e-11	2.3e-16	1.9e-16
2	3	0.1891	0.2293	0.1765	0.1524	0.1737	0.1524
1	0	0.0796	0.2587	0.0371	2.5e-11	2.2e-16	2.8e-16
2	4	0.2558	0.3087	0.2396	0.2077	0.2342	0.2077
1	0	0.0986	0.3236	0.0467	3.8e-11	5.1e-16	2.9e-16
2	5	0.3196	0.3867	0.2979	0.2600	0.2923	0.2600
1	1	0.0374	0.0738	0.0427	0.0327	0.0422	0.0327
2	0	0.0438	0.0672	0.0011	7.5e-12	1.2e-16	1.2e-16
1	2	0.0757	0.1496	0.0869	0.0662	0.0850	0.0662
2	0	0.0879	0.1362	0.0022	1.6e-11	1.7e-16	2.4e-16
1	3	0.1157	0.2265	0.1343	0.1019	0.1285	0.1019
2	0	0.1323	0.2053	0.0033	2.9e-11	3.6e-16	5.0e-16
1	4	0.1539	0.3021	0.1790	0.1353	0.1714	0.1353
2	0	0.1758	0.2734	0.0045	3.1e-11	3.8e-16	4.5e-16
1	5	0.1941	0.3795	0.2265	0.1710	0.2154	0.1710
2	0	0.2209	0.3430	0.0057	3.5e-11	7.1e-16	6.3e-16

**Table A.5:** motor velocity  $\dot{\theta}$  response comparison: overshoot [rad] with different reference steps (target [rad]) for each output (n)

n	Target	Original	Static	SVD	Ideal	Simplified	Inverted
1	0	0.0298	0.0890	0.2215	5.9e-11	6.6e-16	6.5e-16
2	1	0.0536	0.0890	0.0055	0.0077	0	0.0077
1	0	0.0596	0.1781	0.4432	1.3e-10	1.1e-15	6.5e-16
2	2	0.1072	0.1781	0.0111	0.0154	0	0.0154
1	0	0.0894	0.2672	0.6648	3.4e-10	2.2e-15	2.6e-15
2	3	0.1608	0.2671	0.0166	0.0231	0	0.0231
1	0	0.1193	0.3563	0.8864	2.4e-10	2.6e-15	1.7e-15
2	4	0.2145	0.3561	0.0222	0.0298	0	0.0298
1	0	0.1491	0.4454	1.1080	3.6e-10	6.1e-15	1.5e-15
2	5	0.2681	0.4452	0.0278	0.0385	0	0.0385
1	1	0.0506	0.1524	0	0.0254	0.0042	0.0254
2	0	0.0922	0.1518	0.0209	1.1e-10	1.8e-15	1.7e-15
1	2	0.1012	0.3049	0	0.0509	0.0084	0.0509
2	0	0.1845	0.3036	0.0419	2.1e-10	2.7e-15	2.6e-15
1	3	0.1519	0.4574	0	0.0761	0.0126	0.0761
2	0	0.2767	0.4555	0.0629	4.4e-10	5.3e-15	3.5e-15
1	4	0.2025	0.6099	0	0.1025	0.0168	0.1025
2	0	0.3690	0.6074	0.0839	3.8e-10	5.3e-15	8.8e-15
1	5	0.2531	0.7624	0	0.1284	0.0210	0.1284
2	0	0.4612	0.7592	0.1047	5.1e-10	7.1e-15	1.0e-14

**Table A.6:** motor velocity  $\dot{\theta}$  response comparison: settling time [s] with different reference steps (target [rad/s]) for each output (n)

n	Target	Original	Static	SVD	Ideal	Simplified	Inverted
1	0	0.9084	2.8509	0.5930	-	-	-
2	1	0.4830	1.1343	0.0647	0.0607	0.0690	0.0607
1	0	1.0297	3.01499	0.7502	-	-	-
2	2	0.4831	1.1349	0.0661	0.0619	0.0689	0.0619
1	0	1.1028	3.0769	0.8436	-	-	-
2	3	0.4817	1.1322	0.0648	0.0609	0.0691	0.0609
1	0	1.1528	3.1077	0.9101	-	-	-
2	4	0.4831	1.1342	0.0654	0.0612	0.0695	0.0612
1	0	1.1928	3.6005	0.9596	-	-	-
2	5	0.4833	1.1335	0.0660	0.0619	0.0701	0.0619
1	1	0.4574	1.1795	0.0810	0.0502	0.0655	0.0502
2	0	1.0765	2.9244	0.1445	-	-	-
1	2	0.4575	1.1815	0.0803	0.0507	0.0644	0.0507
2	0	1.1928	3.0290	1.0408	-	-	-
1	3	0.4582	1.1786	0.0807	0.0506	0.0652	0.0506
2	0	1.2600	3.5861	1.6057	-	-	-
1	4	0.4579	1.1784	0.0806	0.0509	0.0652	0.0509
2	0	1.3044	3.7199	2.0081	-	-	-
1	5	0.4586	1.1787	0.0808	0.0498	0.0653	0.0498
2	0	1.3418	3.7869	2.3182	-	-	-

# Bibliography

- [1] Jean-Jacques E. Slotine and Weiping Li. *Applied Nonlinear Control*. Englewood Cliffs, New Jersey 07632: Prentice Hall, 1991. ISBN: 0-13-040890-5 (cit. on pp. 3, 4).
- [2] Sigurd Skogestad and Ian Postlethwaite. *Multivariable Feedback Control: Analysis and Design*. Second. Chichester, England: John Wiley & Sons Ltd, 2006. ISBN: 978-0-470-01167-6. URL: <http://www.wiley.com> (cit. on pp. 3, 4, 35–38, 62, 64).
- [3] Bruno Siciliano, Lorenzo Sciavicco, and Luigi Villani. *Robotics: Modelling, Planning and Control*. London, UK: Springer-Verlag London Limited, 2010. ISBN: 978-1-84628-641-4. DOI: 10.1007/978-1-84628-642-1. URL: <https://doi.org/10.1007/978-1-84628-642-1> (cit. on pp. 3, 7, 23).
- [4] Mark W. Spong, Seth Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. First. New York, Chichester, Weinheim, Brisbane, Singapore, Toronto: John Wiley & Sons, Inc., 2005. ISBN: 978-0-470-01167-6 (cit. on pp. 3, 51).
- [5] Alessandro De Luca. *Modeling and Control of Robots with Elastic Joints*. Lecture notes, Dottorato di Ricerca in Ingegneria dei Sistemi, Università di Roma La Sapienza. Corso: Modellistica e Controllo di Robot con Giunti Flessibili, Parte 1. Dipartimento di Informatica e Sistemistica (DIS), Roma, Italia, 2011 (cit. on p. 3).
- [6] Alessandro De Luca and P. Tomei. «Elastic Joints». In: *Theory of Robot Control*. Communications and Control Engineering. Springer, 2012, pp. 179–217. DOI: 10.1007/978-1-4471-1501-4\_5 (cit. on pp. 3, 13).
- [7] Stig Moberg. «Modeling and Control of Flexible Manipulators». Printed by LiU-Tryck, Linköping, Sweden. PhD thesis. Linköping, Sweden: Linköping University, Department of Electrical Engineering, 2010. ISBN: 978-91-7393-289-9. URL: <http://www.control.isy.liu.se> (cit. on pp. 3, 51).

- [8] Alessandro De Luca and Wayne J. Book. «Robots with Flexible Joints and Links». In: *Springer Handbook of Robotics*. Ed. by Bruno Siciliano and Oussama Khatib. Berlin, Heidelberg: Springer, 2016, pp. 243–279. ISBN: 978-3-540-23957-4. DOI: 10.1007/978-3-540-30301-5\_11 (cit. on pp. 3, 12, 15, 17, 22, 26, 30).
- [9] Alessandro De Luca. *Control of Soft and Articulated Elastic Robots: Robots with Flexible Joints - Modeling and Control*. Lecture notes and presentation material. Rome, Italy: EECI International Graduate School on Control (IGSC), Course M16, 2023. URL: <http://www.diag.uniroma1.it/deluca/Publications.php> (cit. on pp. 3, 12, 23, 26).
- [10] A. S. Boksenbom and R. Hood. *General algebraic method applied to control analysis of complex engine types*. Tech. rep. NCA-TR-980. Washington D.C.: National Advisory Committee for Aeronautics, 1950 (cit. on p. 3).
- [11] Lu Liu, Siyuan Tian, Dingyu Xue, Tao Zhang, YangQuan Chen, and Shuo Zhang. «A Review of Industrial MIMO Decoupling Control». In: *International Journal of Control, Automation and Systems* 17.5 (2019), pp. 1246–1254. ISSN: 1598-6446. DOI: 10.1007/s12555-018-0367-4. URL: <http://www.springer.com/12555> (cit. on pp. 3, 34, 37, 40, 41, 49, 67).
- [12] W. L. Luyben. «Distillation decoupling». In: *AIChE Journal* 16.2 (1970), pp. 198–203 (cit. on pp. 3, 4, 34, 35, 40, 42).
- [13] C. H. Liu. *General Decoupling Theory of Multivariable Process Control Systems*. Berlin-New York: Springer-Verlag, 1983 (cit. on pp. 3, 4).
- [14] F. G. Shinskey. *Process Control Systems: Application, Design, and Tuning*. McGraw-Hill, Inc., 1990 (cit. on pp. 3, 4, 45).
- [15] Harold L. Wade. «Inverted Decoupling: A Neglected Technique». In: *ISA Transactions* 36.1 (1997), pp. 3–10. ISSN: 0019-0578. DOI: 10.1016/S0019-0578(97)87647-8 (cit. on pp. 3, 4, 45, 49).
- [16] E. Gagnon, A. Pomerleau, and A. Desbiens. «Simplified, Ideal or Inverted Decoupling?» In: *ISA Transactions* 37.4 (1998), pp. 265–276. ISSN: 0019-0578. DOI: 10.1016/S0019-0578(98)00023-8 (cit. on pp. 3, 4, 41).
- [17] Jietae Lee, Dong Hyun Kim, and Thomas F. Edgar. «Static decouplers for control of multivariable processes». In: *AIChE Journal* 51.10 (2005), pp. 2712–2720. DOI: 10.1002/aic.10520 (cit. on pp. 3, 34, 37).
- [18] M. Waller, J. B. Waller, and K. V. Waller. «Decoupling revisited». In: *Industrial & Engineering Chemistry Research* 42.20 (2003), pp. 4575–4577 (cit. on pp. 4, 43).



- [19] Juan Garrido, Francisco Vázquez, and Fernando Morilla. «An Extended Approach of Inverted Decoupling». In: *Journal of Process Control* 21.1 (2011), pp. 55–68. ISSN: 0959-1524. DOI: 10.1016/j.jprocont.2010.10.004. URL: <http://www.sciencedirect.com/science/article/pii/S0959152410001903> (cit. on p. 4).
- [20] YS Hung and AGJ MacFarlane. «Multivariable feedback: A quasi-classical approach». In: *IEEE Transactions on Automatic Control* 26.1 (1982), pp. 70–74 (cit. on pp. 4, 34, 37).
- [21] Thomas Dehaeze. *Diagonal Control Using the SVD and the Jacobian Matrix*. Unpublished manuscript or technical report. Focus on gravimeter and Stewart platform models using SVD and Jacobian techniques. Apr. 2021 (cit. on pp. 4, 38, 63, 64).
- [22] Javad Mohammadpour, Karolos Grigoriadis, Matthew Franchek, and Yue-Yun Wang. «LPV Decoupling for Multivariable Control System Design». In: *Proceedings of the 2009 American Control Conference (ACC)*. Hyatt Regency Riverfront, St. Louis, MO, USA: American Automatic Control Council (AACC), June 2009, pp. 3112–3117. DOI: 10.23919/ACC.2009.5160376 (cit. on pp. 4, 34, 36, 38).
- [23] Tamás Baár, Péter Bauer, and Tamás Luspay. «Parameter Varying Mode Decoupling for LPV Systems». In: *IFAC PapersOnLine* 53.2 (2020), pp. 1219–1224. DOI: 10.1016/j.ifacol.2020.12.1338 (cit. on pp. 4, 34).
- [24] Qing-Guo Wang. *Decoupling Control*. Vol. 285. Lecture Notes in Control and Information Sciences. With 88 Figures. Berlin, Heidelberg, New York, Hong Kong, London, Milan, Paris, Tokyo: Springer-Verlag Berlin Heidelberg New York, 2003. ISBN: 3-540-44128-X (cit. on pp. 4, 63).
- [25] Peter Hippe. *Design of Observer-based Compensators: A Unified Frequency-domain Approach*. Springer Science & Business Media, 2006 (cit. on p. 4).
- [26] Ciprian Lupu, Dumitru Popescu, Andreea Udrea, and Catalin Dimon. «Design Procedure for Nonlinear Multivariable Processes Control». In: *9th WSEAS International Conference on Automation and Information (ICAI'08)*. WSEAS. Bucharest, Romania: WSEAS Press, 2008, pp. 362–367. ISBN: 978-960-6766-77-0. URL: <https://www.researchgate.net/publication/262168786> (cit. on p. 4).
- [27] M. W. Spong. «Modeling and Control of Elastic Joint Robots». In: *Journal of Dynamic Systems, Measurement, and Control* 109.4 (1987), pp. 310–318. DOI: 10.1115/1.3143860 (cit. on pp. 4, 12).

- [28] Alessandro De Luca. «Decoupling and Feedback Linearization of Robots with Mixed Rigid/Elastic Joints». In: *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*. Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza". Minneapolis, Minnesota, USA: IEEE, 1996 (cit. on p. 4).
- [29] Abhinandan Jain and Guillermo Rodriguez. «Diagonalized Lagrangian Robot Dynamics». In: *IEEE Transactions on Robotics and Automation* 11.4 (1995), pp. 571–583. DOI: 10.1109/70.406982 (cit. on p. 4).
- [30] Xiaogang Song, Yongjie Zhao, Lei Jin, Peng Zhang, and Chengwei Chen. «Dynamic feedforward control in decoupling space for a four-degree-of-freedom parallel robot». In: *International Journal of Advanced Robotic Systems* 16.1 (2019), pp. 1–10. DOI: 10.1177/1729881418820451. URL: <https://journals.sagepub.com/home/arx> (cit. on p. 4).
- [31] Vigen Arakelian, ed. *Dynamic Decoupling of Robot Manipulators*. Vol. 56. Mechanisms and Machine Science. ISSN: 2211-0984 (print), ISSN: 2211-0992 (electronic). Cham, Switzerland: Springer International Publishing, 2018. ISBN: 978-3-319-74362-2. DOI: 10.1007/978-3-319-74363-9 (cit. on p. 4).
- [32] J. Edmunds and B. Kouvaritakis. «Extensions of the Frame Alignment Technique and Their Use in the Characteristic Locus Design Method». In: *International Journal of Control* (1974), pp. 123–456 (cit. on p. 38).
- [33] Aldo Bottero. *Modelli per manipolatori industriali non rigidi: Strumenti e tecniche di linearizzazione dei modelli*. Versione 5.06. July 2011 (cit. on pp. 51, 52, 54).
- [34] P. Albertos Perez and Antonio Sala. *Multivariable Control Systems: An Engineering Approach*. Advanced Textbooks in Control and Signal Processing. London, UK: Springer-Verlag London Berlin Heidelberg, 2004. ISBN: 1-85233-738-9. URL: <http://www.springer.co.uk> (cit. on p. 62).
- [35] G. Fernández, J.C. Grieco, and M. Armada. «Decoupling Control of Flexible Manipulators». In: *Flexible Robot Manipulators*. Berlin, Heidelberg: Springer-Verlag, 2003. Chap. 13, pp. 244–343. ISBN: 978-3-540-12345-6.
- [36] Paolo Rocco. *Lezione 4: Schemi avanzati di controllo*. Dispense di Controlli Automatici, Politecnico di Milano. Modellistica e controllo avanzati, include compensazione, disaccoppiatori e regolatori anti-windup. 2011. URL: <https://rocco.faculty.polimi.it/controlli/LEZ4.pdf>.
- [37] V. Pavan Kumar. «A Comparative Study On Decoupling Methods For Time-Delay Systems». MA thesis. National Institute of Technology, Rourkela, 2015. URL: <http://ethesis.nitrkl.ac.in/6766/>.