University of Coimbra

Faculty of Science and Technology

Electrical and Computer Engineering Department

Fernanda de Madureira Coutinho

Contact Stiffness Estimation Techniques for Robotic Manipulation

Coimbra - Portugal

2013

University of Coimbra

Faculty of Science and Technology

Electrical and Computer Engineering Department

Contact Stiffness Estimation Techniques for Robotic Manipulation

by

Fernanda de Madureira Coutinho

Dissertation submitted to the University of Coimbra in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical Engineering under supervision of

Prof. Dr. Rui Pedro Duarte Cortesão

Coimbra - Portugal

2013

This work was partially supported by the Portuguese Science and Technology Foundation through the PhD grant SFRH/BD/36215/2007, and Projects PTDC/EEA-ACR/72253/2006 and PTDC/EEA-CRO/110008/2009.

To my husband Jorge,
and our children Diogo and Joana

Acknowledgements

I would like to thank everyone who has, directly or indirectly, helped me along this journey. Among these, I would especially like to acknowledge:

- *My PhD supervisor* - Professor Rui Pedro Duarte Cortesão, for his guidance, support, and friendship. With great admiration for his relentless pursuit of scientific accuracy and truth.
- *University of Coimbra* (UC) - where I carried out my PhD research.
- *Institute of Systems and Robotics of Coimbra* (ISR-Coimbra) - for welcoming and supporting me. Its excellent laboratory conditions were fundamental for the experimental component of this work.
- *Portuguese Foundation for Science and Technology* (FCT) - for my PhD grant and the financial support provided to my R&D Group.
- *Engineering Institute of Coimbra* (ISEC) - in particular to the Electrotechnical Engineering Department (DEE), where I have been a Professor since 1995. It has always offered me excellent work conditions, motivation and support to pursue my graduate degree.
- *Portuguese Association of Automatic Control* (APCA) - for the "Best APCA Student Paper Award", which has given me additional encouragement to advance my research studies.
- *Luso-American Foundation* (FLAD) - for supporting my participation in a scientific conference in the United States.
- *Conference and Journal Reviewers* - for constructive observations and comments.
- *My friends and colleagues from ISR* - particularly Cristóvão Silva, Pedro Queirós, Luis Santos, Michel Dominici, Gabriel Pires, Ana Lopes, and Pedro Fernandes, for their camaraderie.

- *My friends and colleagues from DEE/ISEC* - for their friendship.

Last but not least, I would like to thank my family for their unconditional love, support, and protection. Particularly my lovely children, Diogo and Joana, and my dearest husband, Jorge.

Fernanda Coutinho
Coimbra, May 2013

Abstract

This work addresses the problem of stiffness estimation for robotic tasks. Online stiffness estimation can be used to improve force tracking in explicit force control schemes being also important for realistic haptic feedback in telemanipulation tasks. It also enables accurate stiffness mapping and simulation of environment dynamics. Many applications involving contact can benefit from stiffness estimation. For instance, improved force tracking is useful to handle fragile organic tissues in robotic-assisted minimally invasive surgery. Enhanced haptic feedback allows the surgeon to have a better perception of contact forces, improving safety and allowing finer control. Environment stiffness estimation is useful in diagnosis, helping to detect pathologies through stiffness variations. Furthermore, accurate tissue simulation can be used in training and in task design. Applications of stiffness estimation, however, are not restricted to the medical area. These can be found in other contexts, such as industrial robotics. Examples include tasks involving contact, such as polishing or object assembly, that can benefit from improved force tracking. Also, inspection of object stiffness may be useful for quality control purposes. Many authors have described stiffness estimation as a complex problem. This complexity arises from multiple factors, where sensorial information with a poor signal-to-noise ratio is a major one. Uncertainty in the environment geometry is also relevant, as it obscures the relative positioning of the end-effector w.r.t. the environment, creating difficulties to detect free-space/contact transitions. This may be problematic, since most estimation techniques require accurate identification of the initial contact point.

In this thesis, three new online stiffness estimation algorithms are developed for robotic interaction tasks. These algorithms completely prevent the dependency on environment position-based data, relying instead on force-based data to obtain the stiffness estimation. Two different approaches have been adopted. The first relies on the availability of explicit models of the control system and environment while the second uses implicit models

trained with sensorial data. Two estimation algorithms, ASBA and COBA, were developed upon model-based techniques, while a third one, ANNE, relies on sensor-based techniques. ASBA is a stiffness estimator based on the active-state inspection of an active observer used in the control loop. The estimation is obtained by comparing the active state with theoretical predictions for different mismatches between nominal and real stiffnesses. COBA is a stiffness estimator based on the analysis of the output of two force observers, each tuned with a different nominal parameter. The estimation is obtained by considering the force prediction error of both observers. Finally, ANNE is an estimator supported by artificial neural networks (ANNs). It is composed of a layer of ANNs, whose outputs are combined by a fusion module that produces the final estimation. Theoretical analysis, simulation and experimental results are provided, demonstrating that these techniques can effectively achieve environment stiffness estimation. The algorithms described in this work present a contribution to online stiffness estimation in robotic tasks involving interactions with unstructured and unknown environments where contact geometrical data is unavailable or unreliable.

Resumo

Este trabalho aborda a estimação da rigidez do ambiente em tarefas robóticas. Estimações da rigidez obtidas em tempo-real podem ser utilizadas para obter controlo explícito de força de alta qualidade, bem como para melhorar a telepresença em sistemas de telemanipulação hápticos. Também tornam possível a inspeção e o mapeamento da rigidez do ambiente. Podem ser identificadas várias aplicações que beneficiam de estimação da rigidez. Por exemplo, no contexto de robótica médica, tarefas de cirurgia minimamente invasiva assistidas por telemanipulação robótica, beneficiam de controlo de força preciso quando envolvem o manuseamento de tecidos orgânicos frágeis. Por outro lado, melhorias a nível da resposta háptica permitem ao cirurgião ter uma melhor perceção da força aplicada, aumentando assim a segurança e permitindo um controlo mais fino da operação. O mapeamento da rigidez do ambiente pode ser útil para efeitos de diagnóstico, enquanto que simulações rigorosas de tecidos podem ser usadas para efeitos de treino ou de planeamento de tarefas. Porém, as aplicações de estimação da rigidez não são restritas à área médica. Podem ser encontradas noutros contextos, como na robótica industrial. Exemplos incluem tarefas que envolvem contacto e beneficiam de controlo de força preciso, tais como polimento ou montagem de objectos. A identificação da rigidez pode também ser útil para controlo de qualidade. Muitos autores descreveram a estimação da rigidez como um problema complexo. Esta complexidade resulta de diversos factores, dos quais se destaca a utilização de dados sensoriais com uma relação sinal-ruído fraca. Incerteza na geometria do ambiente é também um factor relevante, porque torna difícil a identificação da posição relativa entre o ambiente e a ferramenta acoplada à última junto do robô, complicando a detecção de transição entre operação em contacto e espaço livre. Isto pode ser problemático, dado que a maior parte das técnicas de estimação requer identificação precisa do ponto de contacto inicial.

Nesta tese, são propostos três algoritmos novos para estimação da rigidez em tarefas robóticas com contacto. Estes algoritmos evitam a dependência

de informação relativa à posição do ambiente, usando apenas informação de força para obter a estimação. Foram seguidas duas abordagens distintas, sendo que a primeira é baseada em modelos do ambiente e sistema de controlo e a segunda é em modelos implícitos treinados com dados sensoriais. Os dois primeiros algoritmos, ASBA e COBA, são baseados na primeira abordagem, enquanto o terceiro, ANNE, é baseado na segunda. ASBA é um estimador baseado na inspeção do estado adicional de um observador ativo usado na malha de controlo. A estimação é obtida comparando esse estado com previsões teóricas obtidas para diferentes cenários de desalinhamento entre a rigidez nominal, usada no projeto de controlo, e a rigidez real do ambiente. O algoritmo COBA é um estimador baseado na análise da saída de dois observadores de força calibrados com parametrizações distintas. A estimação é obtida a partir do erro de estimação de força de ambos os observadores. Por fim, o algoritmo ANNE é baseado em redes neuronais. É composto por uma camada de redes neuronais, cujas saídas são combinadas num módulo de fusão que produz a estimação final. São apresentadas análises teóricas e resultados de simulação e experimentais. Estes demonstram que os métodos propostos são capazes de estimar, de forma adequada, a rigidez do ambiente. Uma contribuição destes algoritmos é permitir a estimação da rigidez em tarefas robóticas que envolvem contacto com ambientes desconhecidos e pouco estruturados, para os quais a informação que descreve a geometria do contacto não está disponível ou é pouco fiável.

Symbols and Abbreviations

General abbreviations

ANN	Artificial Neural Network
ANNE	Artificial Neural Network Estimator
AOB	Active OBserver
ASBA	Active State Based Algorithm
COBA	Candidate Observer Based Algorithm
FOB	Force OBserver
LSE	Least Squared Error
MIMO	Multiple-Input and Multiple-Output system
MRAC	Model Reference Adaptive Control
MSE	Mean Square Error
RLS	Recursive Least Squares
RMIS	Robotic-assisted Minimally Invasive Surgery
SNR	Signal-to-Noise Ratio
WAM	Whole Arm Manipulator

Time

t	continuous time
$_k$	subscript k means discrete time index
T_s	sampling time
T_d	system time delay

General representation

$\hat{a}$	estimate of scalar, vector or matrix a
${}^i x$	upperscript i^{th} identifies the row of vector x
$_t$	subscript t refers to task space

Position data

X_0	robot and object initial contact task position
X_t	end-effector task position

Stiffness-based variables

K_s	real system stiffness
K_{sn}	nominal system stiffness (used by the control design)
ΔK_s	stiffness mismatch ($\Delta K_s = K_s - K_s$)
$\hat{K}_{s,k}$	system stiffness estimation at time instant k

Robot modeling

q	joint coordinate vector
$\dot{q}$	joint velocity vector
$\ddot{q}$	joint acceleration vector
$\Lambda_t(q)$	mass matrix in the task space
$M(q)$	mass matrix in the joint space
$V(\dot{q}, q)$	Coriolis and centrifugal vector in the joint space
$G(q)$	gravity vector in the joint space
$J_t(q)$	Jacobian matrix

System discrete state-space matrices, vectors and scalars

Φ	system state transition matrix
Γ	command matrix
C	output matrix
x_k	system state vector at iteration k
y_k	system output (force) at iteration k
d	number of extra states due to T_d

System continuous state-space matrices and vectors

A	system state transition matrix
B	output matrix
$x(t)$	system state vector at time instant t
$y(t)$	system output at time instant t

Control signals and variables

G(s)	linearized system plant, for each task dimension
K_D	damping loop gain
τ_f	time constant
τ	vector of joint input torques

τ_c	computed torque vector
f^*	system acceleration input
F_t	control force in task space
$F_{c,t}$	commanded force in the task space
u_k	system command input

State feedback

L_r	state feedback gain matrix
L_1	DC gain (1^{st} element of L_r)

Active observer

$\hat{x}_k$	system state estimation vector
$\hat{p}_k$	active (or extra) state of the AOB
Q_{noise}	system noise matrix
R_{noise}	measurement noise matrix
K_k	Kalman gains
P_k	state covariance matrix
ξ_k	stochastic input representing model uncertainties
η_k	stochastic input representing measurement uncertainties
Φ_{sn}	nominal system state transition matrix
Γ_{sn}	nominal command matrix
C_{sn}	nominal output matrix
Φ_n	augmented nominal system state transition matrix
Γ_n	augmented nominal command matrix
C_n	augmented nominal output matrix

Task specification

f_k	input force
r_k	reference force ($r_k = L_1 f_k$)
F_e	force applied by the robot on the environment

Disturbance signals

$\tilde{u}_k$	disturbed system command input
p_k	equivalent disturbance referred to system input
$p_k(o)$	disturbance due to factors other than $\Delta K_s \neq 0$
$p_k(\Delta K_s)$	disturbance due to stiffness mismatch

ASBA terms

Δr_k	reference difference ($\Delta r_k = r_k - r_{k-1}$)
K_{sc}	candidate stiffness value for K_s

S	set of candidate stiffnesses
$I^{K_{sn},K_{sc}}$	$\hat{p}_k$ curve for $K_s = K_{sc}$, computed offline, for an unitary input step
$\hat{p}_k^{K_{sn},K_{sc}}$	$\hat{p}_k$ curve for $K_s = K_{sc}$, computed online, for a custom input signal
T_{LSE}	time interval for LSE analysis
w	time interval for convolution computation
w_2	time interval for $I^{K_{sn},K_{sc}}$ computation
t_o	number of samples with stable estimations for validation purposes, in overestimated stiffness cenarios
t_u	number of samples with stable estimations for validation purposes, in underestimated stiffness cenarios
T_L	reference variation threshold
T_I	time interval to compute the $I^{K_{sn},K_{sc}}$ curve

COBA terms

$_c$	subscript $_c$ refers to variables of a candidate FOB
K_{sc}	FOB candidate stiffness (K_{sc_i} refers to FOB_i)
ΔK_{sc}	FOB stiffness mismatch ($\Delta K_{sc_i} = K_s - K_{sc_i}$ refers to FOB_i)
Φ_c	FOB state transition matrix (Φ_{c_i} refers to FOB_i)
$\Delta\Phi_c$	FOB_i state transition matrix mismatch
K_c	FOB gain vector (^1K_c and ^2K_c are for the first two FOB states)
$y_{c,k}$	FOB force estimation ($y_{c_i,k}$ refers to FOB_i
$x_{c,k}$	FOB state vector ($x_{c_i,k}$ refers to FOB_i)
$^1x_{c,k}$	1st state of $x_{c,k}$ FOB force estimation ($^1x_{c_i,k}$ refers to FOB_i)
$\Delta x_{c,k}$	FOB_i state mismatch ($\Delta x_{c_i,k}$ refers to FOB_i)
$^1\Delta x_{c,k}$	1st state of $\Delta x_{c,k}$ FOB force error ($^1\Delta x_{c_i,k}$ refers to FOB_i)
$u_{c,k}$	FOB command input
$\Delta u_{c,k}$	uncompensated additional disturbance at $u_{c,k}$
G_k	mismatch gradient
^1G_k	1st state of G_k force error gradient
$G_{\min}$	minimum threshold to assess the stiffness estimation
W_k	disturbance induced offset
^1W_k	1st state of W_k force error offset
λ_1,λ_2	eigenvalues of W_k transition matrix

ANNE terms

$K_T^{(j)}$	nominal stiffness value used for $\text{ANN}^{(j)}$ training
ANN $K_T^{(j)}$	ANN trained with $K_{sn} = K_T^{(j)}$
$\hat{K}_{s,k}^{(j)}$	stiffness estimation provided by ANN $K_T^{(j)}$

Contents

List of Figures

List of Tables

List of Algorithms

List of Theoretical Results

Chapter 1

Introduction

Contents

1.1 Motivation

The objective of this work is to develop techniques for estimating the stiffness of contact objects in robotic interaction tasks. Stiffness measures the resistance of materials to deformation in response to applied forces. Accurate knowledge of stiffness (and other contact parameters) can be important for the performance of robotic tasks involving explicit force control, although stiffness is the most important contact parameter for tasks with low dynamics. This is especially the case when model reference adaptive control (MRAC) architectures are considered. Mismatches between the real and nominal (that is, the stiffness value used in control design) stiffnesses can degrade force tracking and provoke bouncing and instability once contact is established between the manipulator and the environment. Accurate parameter knowledge is also relevant for improving telepresence in haptic feedback systems. Examples of tasks where these factors are relevant can be found in the area of telemanipulated robotic-assisted minimally invasive surgery (RMIS). In these tasks, accurate force tracking is essential to avoid

damage to fragile organic tissues, while high quality haptic feedback is of benefit to the surgeon controlling the manipulator. A stiffness mismatch scenario, i.e, one where the real stiffness value is different from the nominal stiffness used in control design, can be represented by

$$K_s = K_{sn} + \Delta K_s, \tag{1.1}$$

with K_s being the real system stiffness[1], K_{sn} the nominal stiffness and ΔK_{sn} the stiffness mismatch. The impact of contact parameter mismatch in force tracking behavior is explored in Section 6.3 and can be summarized in this way:

- Increased mismatch results in increased deviation from the expected behavior, decreasing dynamic consistency of the system response.
- The underestimated scenario is the critical situation for stability [20].

This motivates the development of stiffness estimation algorithms to cope with unknown system stiffness, guaranteeing dynamic consistency in the force response and system stability. However, some issues make stiffness identification a challenging task in practice. These are addressed in the next section.

1.2 Challenges in Stiffness Estimation

Many authors have described stiffness estimation as a relevant and complex problem [2, 106]. This complexity arises from multiple factors, such as sensorial information with poor signal-to-noise ratio (SNR) or limited accuracy of joint encoders. Uncertainty in the environment geometry is also relevant, as it obscures the relative positioning of the end-effector w.r.t. the environment, creating difficulties to detect free-space/contact transitions. This may be problematic, since most estimation techniques require accurate identification of the initial contact point. A position-based approach to obtain this information would be to contrast the end-effector position with known information describing the geometry of the environment. However, this may

[1]The *system stiffness* K_s is the combined stiffness of the manipulator/environment pair. It is the actual parameter being estimated and the one relevant for force control purposes. If the stiffness of the manipulator is significantly higher than the stiffness of the environment K_{env}, then $K_s \approx K_{env}$. Since rigid body dynamics are assumed, we commonly use the terms system stiffness, contact stiffness, object stiffness or environment stiffness interchangeably.

not be an option if the environment is poorly known or unknown, making it challenging to estimate the environment stiffness in these scenarios. Zero-crossing detection of force would allow such transitions to be detected, but force sensors are noisy, subjected to bias and may return non-null forces even in free-space operation, obscuring the exact instant of transition. This problem is exacerbated for low contact forces. These issues can be illustrated by considering a direct estimation approach using Hooke's Law along a single dimension. Hooke's Law is a linear model that relates applied force with deformation along an uncoupled dimension and stiffness coefficient. If F_e is the force applied by the robot on the environment, K_s the system stiffness coefficient, X_t is the current robot end-effector position and X_0 is the initial contact position, then Hooke's Law states that

$$F_e = K_s(X_t - X_0). \tag{1.2}$$

The estimation $\hat{K}_s$ can be given by

$$\hat{K}_s = \frac{\hat{F}_e}{(\hat{X}_t - \hat{X}_0)}, \tag{1.3}$$

where $\hat{F}_e$ is obtained by force sensing and $\hat{X}_t$ comes from robot forward kinematics. The estimation $\hat{X}_0$ could be obtained from geometric information if the relative positioning of the end-effector and object is well-known. However, if such information is not readily available, indirect contact position estimation approaches must be adopted. A viable approach entails the use of force information to detect the moment of free-space/contact transition and estimating X_0 with the end-effector position at that instant. This approach, however, can suffer from some drawbacks, namely:

- *Uncertainty in detection of contact* - Low SNR of the measured force makes zero-crossing detection unreliable. Contact episodes could be detected in free-space operation and, conversely, contact might not be properly detected, especially if low forces are applied. Delayed (or anticipated) detection of zero-crossing causes an erroneous initial contact position estimation $\hat{X}_0$. It will be maintained while the interaction is not broken, biasing the estimation throughout the entire contact episode. Subsequent contact with the same object may result in a different initial contact position estimation, resulting in inconsistent estimations for the same object.

- *Limited encoder resolution* - The resolution of joint encoders may be insufficient to properly account for minimal displacements that occur

in contacts with very stiff surfaces [2, 106]. Variations of the measured force $\hat{F}_e$ might occur without any discernible change in the estimated position $\hat{X}_t$ [20].

While the latter problem hardly occurs in soft environments, the former one still poses considerable practical difficulties. These can be overcome in tasks where the object geometry is static and well-known (as is the case of many industrial machining tasks) or if it is possible to start the task with the end-effector already placed at the initial contact position (as assumed in many estimation works). However, in some tasks (e.g., RMIS), these assumptions do not hold and another approach to stiffness estimation is warranted. It might be argued that more complex and presumably more accurate contact models could be considered instead of (1.2), however, two points should be considered:

1) High order contact terms can be ignored for tasks with low dynamics. Non-linear contact models bring additional precision and complexity, but (1.2) is in practice a useful approximation of the real behavior that is frequently adopted for the design of control systems. In this work, (1.2) matches the contact model used in the control design.

2) More importantly, these problems are not concerned with the specific formulation of the contact model itself, but with the data required by the estimator. For example, some approaches based on non-linear visco-elastic contact models [43, 35, 41] are more accurate than (1.2), but they require the availability of force and object position w.r.t. the robot and velocity data, being subject to the same type of practical difficulties.

Keeping this in mind, this work focuses in developing online stiffness estimation algorithms based on force data, not requiring object position information. In this way, less sensorial information is required and some of the difficulties described above can be avoided, broadening the range of applications where online stiffness estimation can be employed. Two approaches are successfully explored:

1) Using ***explicit*** mathematical models of the control system and contact interaction.

2) Using ***implicit*** mathematical models, trained with sensorial information.

An advantage of explicit models is that certain properties of the estimation can be mathematically derived and proven. Two algorithms are developed

with the first approach. One is based on contrasting the active state of the AOB with the predicted response for several mismatching stiffness scenarios, while the other is based on force observers. Estimation techniques based on implicit models do not require explicit modeling, making them less dependant on a specific control approach. An estimator based on artificial neural networks is developed with this approach.

1.3 Applications of Stiffness Estimation

Techniques for the estimation of stiffness and other contact parameters have found application in diverse areas:

- **Telepresence** - Environment estimation techniques are often used in teleoperation tasks, to achieve high quality haptic feedback and improved telepresence. Examples can be found in [37, 112, 68, 79, 122]. Dupont *et al.* [37] outline a general process of environment identification in teleoperation tasks, composed of task decomposition, data segmentation and property estimation, illustrating it with the estimation of the mass of a picked up object. Wang *et al.* [112] describe how the estimation techniques presented in [67] can be used to construct a discretized model of task space parameters during teleoperation tasks. A similar approach is presented in [68], including a method for detecting time dependant environment variations. Misra and Okamura [79] use the estimation techniques described in [94] to improve force tracking in a bilateral telemanipulation setup. Zarrad *et al.* [122] discuss the use of state-space based Kalman techniques to identify contact parameters and ensure stability and telepresence.

- **Force tracking** - A common motivation for adopting environment parameter estimation techniques is improving the performance of explicit force control schemes. These are usually designed and analyzed considering the model of the environment is known and available, typically resulting in the use of control laws dependent on nominal environment parameters. Force tracking performance and contact stability are therefore dependent on the accuracy of such parameters. Many examples of estimation approaches applied to force tracking problems can be found in the literature [60, 20, 113].

- **Simulation** - Simulation of robotic interactions with complex environments can be important for supporting the design of robotic tasks, especially when operation in unusual, unsafe or high-risk environments

is required. Examples include planning of space station operations [114] or virtual human tissue simulation, as proposed in [79].

- **Environment inspection** - The estimated parameters can provide useful insights about the environment the manipulator interacts with. For example, in the industrial domain, the stiffness estimation can be a relevant variable for controlling the quality of a manufactured object. In tasks involving contact with live animal tissue, the estimation can be used to detect anomalous and potentially pathological irregularities.

1.4 Key Contributions

This section highlights the main contributions of this work. These can be aggregated into the three new force-based stiffness estimation techniques herein proposed:

1) **The stiffness estimation algorithm ASBA**
The **A**ctive **S**tate **B**ased **A**lgorithm (ASBA) estimates the stiffness by analyzing the evolution of the active state of an Active OBserver (AOB) and comparing it with the expected behavior for several potential stiffness mismatch scenarios, computed analytically from known models. The main results and the ASBA description, including a theoretical discussion and simulation results, have been presented in [23, 25]. The latter work received the *Best APCA Student Paper Award* from Portuguese Association of Automatic Control - APCA (National Member Organization of IFAC – International Federation of Automatic Control).

2) **The stiffness estimation algorithm COBA**
The **C**andidate **O**bserver **B**ased **A**lgorithm (COBA) is the second algorithm for stiffness estimation. It uses two force observers with different candidate stiffnesses to infer the correct system stiffness. The theoretical grounds of this algorithm are presented, as well as experimental results. This algorithm addresses some limitations of earlier efforts, by having comparably negligible time and space-footprint, improving online operation and also addressing the issue of estimating under non-ideal (disturbed) conditions. The theoretical exposition of the algorithm, as well as simulation, validation and experimental results have been published in [21, 22, 24, 26, 27].

3) **The stiffness estimation algorithm ANNE**
The **A**rtificial **N**eural **N**etwork **E**stimator (ANNE) is the third estima-

tion technique, also independent of the object position. It is based on artificial neural networks and, unlike ASBA and COBA, it does not require explicit system or contact models. The estimator is trained to identify the stiffness using sensorial data. The estimator includes an input buffer with current and past force-based data, as well as a bank of feedforward ANNs and a fusion module. Simulation and experimental results demonstrate appropriate estimation behavior. One interesting aspect of ANNE is that, by not requiring explicit modeling, its dependency on a specific control architecture is reduced.

1.5 Thesis Structure

Chapter 2 provides background on related work such as contact modeling, force control and contact parameter estimation. In Chapter 3, the robotic control architecture targeted in all theoretical and experimental analysis is presented. It is based on feedback linearization and state-space Kalman techniques, including an AOB observer for feedforward disturbance compensation. Chapter 4 describes the model-based algorithms ASBA and COBA. ASBA is based on the least square error analysis of the extra state of the active observer and COBA is based on force observers. The algorithm descriptions can be found, as well as the relevant theoretical analysis and results. Chapter 5 presents the sensor-based estimation algorithm ANNE, based on multiple neural networks. It includes the description of the process of network input selection, as well as the overall architecture that fuses the outputs of multiple networks into a single estimation result. Simulation and experimental results are discussed in Chapter 6 and Chapter 7, respectively. Chapter 6 presents simulation tests for all algorithms. The interaction of a robot with the environment is simulated, based on the explicit force control architecture described in Chapter 3, with the Matlab Simulink tool. The impact of parameter mismatch in the force tracking dynamics is presented. Then, the behavior of the estimation algorithms in nearly ideal conditions is explored. Chapter 7 presents and discusses COBA and ANNE experimental results, including the validation and test of both algorithms. The validation is performed through WAM robot interactions with virtual objects and algorithm results are also analysed when the WAM robot interacts with real objects. Free-space/contact experiments are also presented. Chapter 8 concludes this thesis and discusses possible directions for future work. The Appendix collects proofs of relevant theoretical results.

Chapter 2

Background

Contents

2.1 Introduction

In this chapter, an overview of related work is presented. It is focused on the following three main topics:

1) **Contact modeling** - Hooke's Law, introduced in Chapter 1, is a simple linear approximation of a very complex physical phenomenon. Other

proposals in the literature provide alternative formulations, including viscous and nonlinear terms. This overview is necessary because many different models are assumed in works concerned with contact parameter estimation.

2) **Force control** - Robotic force control is an important area of application for contact parameter estimation techniques. The experimental results described in this work are obtained with a robotic platform with direct force control capabilities. This overview presents a broad view of this field, so that proper context of application can be better understood.

3) **Contact parameter estimation** - The practical relevance of contact parameter estimation has resulted in many publications in this field throughout the past decades. In this section, major approaches to estimation are presented and the work in this thesis is related to open issues.

Section 2.2 presents a survey on contact modelling, while an overview of force control and contact parameter estimation techniques is provided in sections 2.3 and 2.4, respectively.

2.2 Contact Dynamics Modeling

Impact and contact are complex physical phenomenon involving two or more bodies that come into touch. The term impact is commonly used to refer to episodes of very short duration, typically involving significant accelerations and forces of impulse-like nature, while contact is more commonly used to describe a state of continuous collision over an extended time period. Correspondingly, two modelling approaches can be found in the literature. The first category includes the impact-momentum or discrete techniques [61, 41, 102, 118, 89, 101, 65]. These are based on the analysis of energy transfer and restitution. They are used to model the impact of rigid bodies, by describing the relation between their pre-impact and post-impact physical states. Secondary phases such as sticking and sliding can also be considered. The second category includes the continuous approaches [61, 41, 6, 9] (also known as compliant or force-based approaches) and is based on the analysis of the force behaviour during the interval of contact. While discrete techniques presume near instantaneous duration of contact episodes, continuous approaches do not require such an assumption, and are correspondingly more capable of representing prolonged contact situations.

2.2.1 Continuous Contact Modeling

This work is concerned with robotic tasks involving prolonged contact with the environment. Continuous contact models are therefore more suited for the purposes of this work. Unlike discrete models, in these the deformation of bodies must be explicitly accounted for, as the normal contact force is related to the local indentation and its ratio of change [6, 9]. The simplest of continuous models, known as Hooke's Law, is described by

$$F_n = K\delta, \tag{2.1}$$

where F_n is the normal contact force, K is the stiffness and δ is the local indentation. It represents a linear spring, with perfectly elastic contact and no energy dissipation. While it is a simplified model of reality, Hooke's Law is nevertheless useful in practice in robotic system control design, as it is amenable to analytical treatment and, more importantly, includes the stiffness parameter, that dominates force response for prolonged contacts with low indentation velocity. Low contact velocity is typical in most robotic interaction scenarios, to avoid damage to the manipulator or environment. Hooke's Law can be generalized to account for energy dissipation by including a linear damping component, obtaining the *Kelvin-Voigt* (or *spring-dashpot*) model [42]

$$F_n = \mathtt{B}\dot{\delta} + K\delta, \tag{2.2}$$

where $\mathtt{B}$ is the damping coefficient, and $\dot{\delta}$ is the time derivative of the indentation. However, the linear damping term introduces physically unrealistic force behavior:

- The damping term generates a discontinuous force at the beginning of impact, due to non-null contact velocity.
- As the force due to the elastic term gradually decreases just before contact is broken at the end of a contact episode, the result will be dominated by a negative force holding the objects together, generated by the damping term.

Additionally, it has been demonstrated experimentally that the coefficient of restitution, used in discrete techniques to model energy loss during impact, should be dependent on velocity [41]. However, this is not the case if the equivalent coefficient of restitution is computed assuming a contact described by a spring-dashpot model [42]. Nevertheless, despite these shortcomings, it is a popular modelling option [50, 62, 111, 78], since it is a

simple linear model that allows energy dissipation in a perfectly elastic contact to be accounted for. Other linear contact models can be constructed by alternative combinations of linear spring and damper elements. These include the so-called *Standard Linear Solid* model, a parallel composition of a Hookean spring and a single spring-dashpot element, and its generalization, the *Maxwell-Wiechert* model (or *generalized Maxwell* model), which includes an arbitrary number of the latter elements [13]. A nonlinear generalization of Hooke's Law, also ignoring energy dissipation, is *Hertz's* model (also known as the *power law* model) [46]

$$F_n = K\delta^m. \tag{2.3}$$

Unlike Hooke's Law, it describes a nonlinear relation between local indentation and normal force, with K and m depending on geometrical and material properties of the colliding bodies. These parameters can be computed analytically for specific conditions of axisymmetric contact of certain solids, such as sphere-sphere central collision [63], however no solution exists in the general case. By adding a damping term to Hertz's Law, energy dissipation can be modelled. A simple extension is the *impact-pair* model, described by Dubowski and Freudenstein [36], where a linear damping term is added to a nonlinear Hertzian spring. However, just like in the case of the Kelvin-Voigt model, the linear dampers introduce unrealistic force discontinuities. To address these problems, Hunt and Crossley proposed in [51] a nonlinear damping term, obtaining the model in (2.4),

$$F_n = \mathtt{B}\delta^p\dot{\delta}^n + K\delta^m. \tag{2.4}$$

This model addresses limitations of the Kelvin-Voigt and impact-pair models by making damping depend on indentation, ensuring continuous force, beginning and ending at zero in every contact episode, although negative sticking forces may still be generated. Applications of Hunt-Crossley's model can be found in [75, 72, 62, 111]. An alternative model, based on similar principles but different formulation of the nonlinear damping term was described by Lee and Wang in [64]. All the above models consider elastic contact. Adaptation to plastic contact can be achieved by differentiating the models of the compression and restitution phases of contact, accounting for persistent deformation of the material [65, 1]. In practice, these models are less commonly used, not only because of parameterization difficulties but also because persistent deformation is not expected nor relevant in most applications [41]. Analytical parameterization of Hertz's model in generic contact scenarios is not possible. An alternative approach for representing

nonlinear elastic behaviour is the empiric model described in [73, 71]. This model attempts to relate the stiffness coefficient to the area of contact $\mathtt{A}$,

$$F_n = K(\mathtt{A})\delta, \tag{2.5}$$

where $K(\mathtt{A})$ is proportional to the square root of $\mathtt{A}$ and dependent of contact geometry and material properties of the colliding objects.

2.2.2 Friction Models

Regardless of the specific continuous model chosen to represent the contact dynamics, several alternatives are available for representing the effect of frictional forces. Friction models provide only approximations of the outcome of a complex physical phenomenon. The simplest and more commonly used friction model is provided by the well known Coulomb's Law [41, 62, 111, 78]. *Static friction* (or *dry*, or *traction*) represents friction force between two sticking objects (i.e: in contact with no relative motion), while *kinetic friction* represents frictional force during sliding movement. Both kinetic and maximum static friction are proportional to the normal force and oppose tangential movement. If v^t is the tangential velocity and μ_k and μ_s are the dimensionless kinetic and static friction coefficients, respectively, then the kinetic friction F_{fk} is described by

$$F_{fk} = -sgn(v^t)(\mu_k F_n), \tag{2.6}$$

while the maximum threshold for static friction is represented by

$$F_{fs} \leq \mu_s F_n. \tag{2.7}$$

Coulomb's friction model generates unrealistic force discontinuities in the transition from sticking to sliding motion. Alternative nonlinear models have been proposed to address these issues, such as the *Karnopp* model [54, 87], which makes friction also dependent on overall non-frictional force, or the *bristle* model, which models friction as being dependent on the integral of tangential velocity up to a saturation limit [44, 33, 73, 114, 41]. Both models ensure continuity in the vicinity of $v^t = 0$, while the latter also allows the friction force to be defined explicitly during the sticking phase.

2.3 Control Approaches for Constrained Robotic Tasks

Earlier research efforts on the control of robotic manipulators were concerned with motion control, that is, the tracking or regulation of end-effector or joint

trajectories [95]. However, manipulator motion may be constrained by the physical environment. While the trajectory may be planned to navigate the manipulator in such a way that unintended collisions with the environment are avoided, this may not be sufficient in practice: the environment may not be well known in advance and some tasks may require operation in significantly constrained spaces, making occasional fortuitous contact with the environment a possibility due to imperfect motion tracking. Furthermore, some tasks, such as polishing, cutting or pushing an object, inherently require interaction of the manipulator with the environment. A pure motion-based controller may perform poorly if the manipulator enters contact with the environment, generating random forces and torques that may damage the environment or manipulator. This problem is exacerbated as the (stiff) manipulator enters contact with stiff environments, as the generated forces can be of high magnitude.

2.3.1 Passive Compliance and Indirect Force Control

One approach for handling contact between the environment and the manipulator is to fit the latter with passive mechanical interaction devices providing the required compliance with the environment, effectively reducing the equivalent contact stiffness (other contact parameters may also be modulated in this way) [31]. Passive compliance is a simple strategy, which does not impair motion tracking performance and addresses the primary concern of conditioning the contact forces. However, this approach may not be sufficiently flexible for handling different tasks with diversified compliance parameter requirements. To overcome this, several control techniques, referred to collectively as compliance control or indirect force control, were devised. In these, compliant behaviour is obtained by active control action rather than specialized physical interfaces. In stiffness control [90], the manipulator behaves as a linear spring in contact situations, according to a control parameter, the active stiffness. Motion tracking performance degrades with lower active stiffness, so a trade-off between compliance and tracking performance is required. Impedance control generalizes this approach, by allowing the contact behaviour to follow that of a mass-spring-damper system with customizable parameters [47, 56, 48, 55]. Tuning these parameters, however, can be challenging, because good motion tracking and disturbance rejection in free-space are obtained at the cost of compliant contact behaviour, and *vice-versa*. Although adaptive impedance techniques have been proposed [57, 15], the main improvements are related to robustness to manipulator model uncertainty rather than enhanced free-space tracking and contact

compliance. Admittance control addresses the latter problem by separating motion control from impedance control. Rather than tracking reference position and orientation, a high-bandwidth inner loop tracks the position resulting from an outer impedance control loop. The inner loop uses a stiff parameterization so that good disturbance rejection and position tracking is achieved in free-space, while the outer loop ensures that the required dynamic response is obtained in contact.

2.3.2 Direct Force Control

While passive compliance and indirect force control techniques allow the contact dynamics to be specified, forces are not explicitly controlled and are therefore potentially unbounded. Additionally, indirect approaches are unsuitable for interaction tasks that require fine control over applied forces. In these cases, direct force control, offering explicit force tracking, is the preferable alternative [117, 76, 86]. Contact is a significantly complex physical phenomenon, so design of explicit force control techniques is usually based on some simplifying assumptions. Typically, the robot is assumed to be perfectly rigid and contact is approximated by a linear model [97], such as Hooke's Law or Kelvin-Voigt's model [42]. Plastic deformation effects are also ignored, assuming a scenario of perfectly elastic contact. Hybrid force/motion control approaches attempt to decouple the motion control from force control [86]. In force/velocity control, interaction tasks are defined in the force and velocity-controlled subspaces, typically in a reference frame that is convenient to the task at hand [76, 32, 10]. Using a nominal model of the manipulator dynamics, an inner loop decouples and linearizes the robot dynamics at the acceleration level [121, 77, 100]. An outer loop is responsible for ensuring disturbance rejection and force/velocity tracking, by means of appropriate proportional, derivative and integral action. If the force derivative is available, asymptotic tracking of a desired force trajectory is possible in interactions with compliant environments. However, force sensors are noisy so direct measurements are often eschewed in favour of model-based computation of the derivative using joint velocities and an estimate of the environment stiffness along the constrained direction. In this case, the dynamics of force tracking are affected by the quality of the stiffness parameter estimation. Force/position control is also possible, by using the direct kinematic model to obtain the required position feedback from joint position measurements. Some approaches are based on parallel, rather than nested, force/position control loops [11, 12]. In this case, force control is made dominant over position control by the use of integral action on the

force error [115]. Typically, force control designs use proportional-integral strategies to achieve zero steady state error. However, configuration of the feedback gains is dependent on estimations of the environmental parameters. In this way, the dynamic response can be adversely impacted by estimation errors, leading to poor tracking performance or even instability. The significance of this problem increases with the stiffness of the environment, as the control action may be insufficient to deal with the faster force dynamics and instability may arise. Passivity-based approaches, based on Lyapunov techniques, have also been applied to design hybrid force/motion control of robotic manipulators [96]. By not requiring inverse-dynamics linearization, these techniques tend to be more robust to manipulator modelling errors. MRAC techniques have also been used for achieving direct force control [108, 69, 116, 120]. In these techniques, control is based on a nominal model of manipulator and environment dynamics. Robustness can be achieved by active control action, using state-space Kalman techniques [8], to ensure that system behaviour closely follows the specified model, ensuring the designed closed-loop behaviour is achieved [20]. However, this compensation may be insufficient for achieving high quality force tracking or even stable behaviour, in the presence of large parameter mismatches between the real and nominal model parameters. Stiffness mismatches are the primary concern. If stiffness is underestimated, the dynamics of the contact force are much faster than anticipated, leading to high transient forces, bouncing and even contact instability in more severe cases. The previous discussion highlights a problem prevalent with direct force control approaches: instability may arise during contact episodes, due to uncertainty in the contact parameters [38, 3, 109]. Strategies based on online adaptation of the control law can be used to address this problem, if techniques are available for providing estimations of the required parameters during online operation.

2.4 Contact Parameter Estimation

Many works concerned with estimation of the environment parameters have been described in the literature. These works can be differentiated according to the contact models being assumed, the estimation approach and interaction control system used to obtain the data.

2.4.1 Recursive Least Squares

A large number of works focuses on the use of recursive least squares techniques (RLS) [66] to identify the contact parameters, considering different

contact models and control approaches. RLS techniques are powerful tools for parameter identification tasks. Depending on the specific approach, convergence can often be proved, although assumptions such as unbiased data and persistent excitation may be required.

Linear Models

Most works are based on linear contact models. In [67], Love and Book use a multiple-input and multiple-output (MIMO) RLS approach to address the problem of adapting the parameters of an impedance controller (a linear mass-spring-damper model) to variations in the environment impedance. Wang *et al.* proposes adaptive control techniques for improving force tracking performance, using estimations obtained by RLS techniques with time-varying forgetting factors, in order to improve estimation convergence speed [113]. A similar technique is described in [59], where a speed-dependent forgetting factor is complemented with a discontinuity detection mechanism, allowing past data to be quickly discarded in contact transitions, further improving estimation responsiveness.

Nonlinear Models

Nonlinear contact models have also been addressed. Diolaiti *et al.* proposed the use of dual recursive RLS estimators for estimating the parameters of Hunt Crossley's contact model [35]. Haddadi *et al.* expanded this work by reducing dependency on initial conditions, improving estimation consistency [43]. In [106], Verscheure *et al.* employ a total least squares estimator to identify the contact parameters of an environment modelled by an Hertzian spring.

2.4.2 Adaptive Control

Another class of estimators are based on parameter adjustment in adaptive control approaches. Kim *et al.* [60] extend the active stiffness control strategy by including an adaptive variable stiffness coefficient based on force data, while a similar approach, with a different formulation requiring force and position data, is proposed for a general impedance control architecture by Seraji and Colbaugh [94]. A sliding force controller using a gradient-based adaptation of the stiffness parameter is described by Tsaprounis and Aspragathos [104]. Convergence of estimation in adaptive control techniques is usually supported by analysis based on Lyapunov's theory. This may entail strong assumptions, such as sustained small velocities and large force

variations, as observed in [60]. Alternatively, as in [104], estimation stability may be demonstrated, but not convergence.

2.4.3 Artificial Neural Networks

Artificial neural networks have also been applied to stiffness estimation and related problems. A systematic approach to modelling environment variables with ANNs is discussed in [74]. In [80], a force controller includes a stiffness estimation module, which requires the use of object position and force data and is based on three interconnected neural models. The controller receives the desired grasping force and computes control signals accordingly, using stiffness estimation. ANNs are adopted in [81] to account for nonlinear environment stiffness in light-emitting diode lens polishing tasks with numerical control machines. Two ANNs with two hidden layers are trained: one for pressing and the other one for unpressing actions. These networks compute the effective stiffness at each time instant, which is then used to adapt control laws. These networks receive only the contact force: the reference force is assumed to be known and constant. Free-space/contact transitions are not considered, since the tool is always in contact with the environment.

2.4.4 Other Approaches

Other estimation approaches can also be found. Erickson *et al.* presents an offline signal processing technique, where the stiffness and damping parameters are identified from the system natural frequency and damping ratio, determined from the system step response [39]. Yamamoto *et al.* presents a multi-estimator technique [119]. Rather than identifying the model parameters, in this approach several candidate Kelvin-Voigt models with different fixed parameterizations are considered. The best estimator is selected by integrating the estimation error of each model with an exponentially decaying factor. Most estimation approaches address the problem of identifying contact parameters in single-point contact scenarios. Weber *et al.* present an identification framework to address the problem of identifying intervals of single-point contact in complex contact scenarios, potentially involving multiple contact points, allowing standard single point identification techniques to be used [114].

2.4.5 Geometric Uncertainty in Estimation Tasks

With a few exceptions noted bellow, many approaches presented so far focus on contact model parameter identification presuming the availability of geometrical information, such as the relative position of the environment w.r.t. the manipulator end-effector. Results are typically collected with simulations or via experimental setups where the manipulator begins at rest in unstretched contact, so no impact occurs and geometrical uncertainty is mitigated. These conditions are not representative of practical scenarios, however, and geometrical uncertainty is a significant practical obstacle to high-quality estimation. Some works attempt to address this issue by combining contact parameter estimation with estimation of the geometrical parameters. Examples of this approach can be found in [94, 107, 122]. The Lyupanov-based technique described in [94] includes estimation of stiffness and geometrical information. In [107], a dual RLS estimator is used for similar purposes, while [122] achieves the same objective using state-space Kalman techniques. However, the estimation of geometrical parameters in a contact situation is in itself a difficult, ill-posed problem, with observability conditions that may not be attained in contact situations [107, 45]. This motivates approaches that completely avoid reliance on geometrical parameters. One of the earliest example of such approach can be found in Love and Book [67], where differentiation is used to eliminate position dependence. However, this creates dependency on the force derivative signal, which is also a significant practical obstacle, as it is known to be an extremely noisy signal. To mitigate ill-conditioning of measured geometric data when in contact with very stiff surfaces, Kikuuwe *et al.* [59] considers a virtual robot end-effector position, computed by adding to the end-effector position a virtual displacement, based on a parameterized virtual impedance model. However, results can be sensitive to the parameterization of this model, and no clear tuning guidelines are provided. The signal processing algorithm described in [39] also avoids geometric information by requiring only force data. However, it is an offline technique, making it unsuitable for control adaptation purposes. The adaptive stiffness control strategy described by Kim *et al.* [60] is based on force data only, but parameter convergence assumes sustained small velocities and large force variations. Another approach can be found in [20], where a sigmoid-like function, dependant on force data only, is tuned experimentally to provide an estimation of the system stiffness leading to good control results. Marked improvements in force response are obtained, indicating good estimation performance, but a significant number of parameters must be empirically tuned to obtain these

results.

2.5 Contributions

Although many estimation approaches independent of geometric information have been proposed, it remains challenging to simultaneously address a significant number of practical concerns: independence of noisy data (e.g., force derivative), suitability for online implementation, having simple tuning guidelines and minimizing assumptions on contact dynamics. This work seeks to develop novel online stiffness estimation techniques addressing these issues, enabling the use of estimation techniques in practical settings where it would be challenging to do so with current techniques.

Chapter 3

Robotic Control Architecture

Contents

3.1 Introduction

Some robotic manipulation tasks such as tele-echography and robotic-assisted minimally invasive surgery, greatly benefit from stiffness estimation algorithms, not only to boost control performance but also to potentiate the haptic telepresence needed by the surgeon. A robotic platform installed at the Medical Robotics and Assistive Technologies R&D Group, from Institute of Systems and Robotics of Coimbra (ISR-Coimbra), has been used in telemanipulation tasks, such as tele-ecography [91], but without online estimation of contact parameters. The work presented in this thesis aims

to endow the existing robotic platform with this important capability. Accordingly, the corresponding control setup has been targeted in all theoretical and experimental analysis, even though the proposed techniques may be portable to other control architectures. This chapter details the target control architecture. This chapter is organized as follows. The control architecture description is given in Section 3.2 and Section 3.3 presents the system plant considering the presence of disturbances.

3.2 Control Architecture

This section describes the control architecture in the task space of the robotic platform used in this work: a WAM™ robot, shown in Fig. 3.1. Section 3.2.1 presents the system plant in the task space and Section 3.2.2 shows the control design, which takes into account discrete state space control. Section 3.2.3 explains how to conduct the nominal stiffness adaptation.

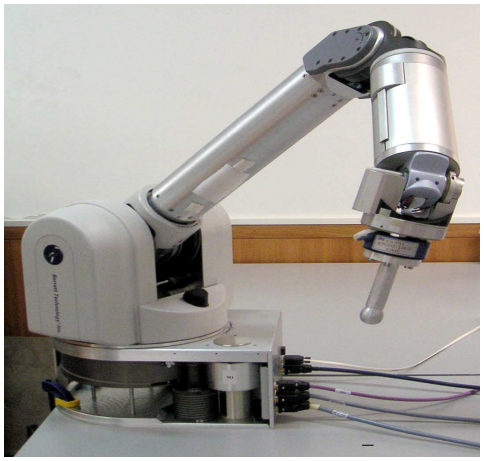


Figure 3.1: WAM™ arm developed by Barrett Tecnology®, Inc..

3.2.1 System Plant in the Task Space

Fig. 3.2 presents our robotic manipulator with computed torque control in the task space [20, 58]. The robot dynamic model is well known and provided by the manufacturer (Barret Technology®, Inc.). It is used for feedback linearization to achieve a decoupled double integrator behavior in the task space [52, 98, 29, 99, 53].

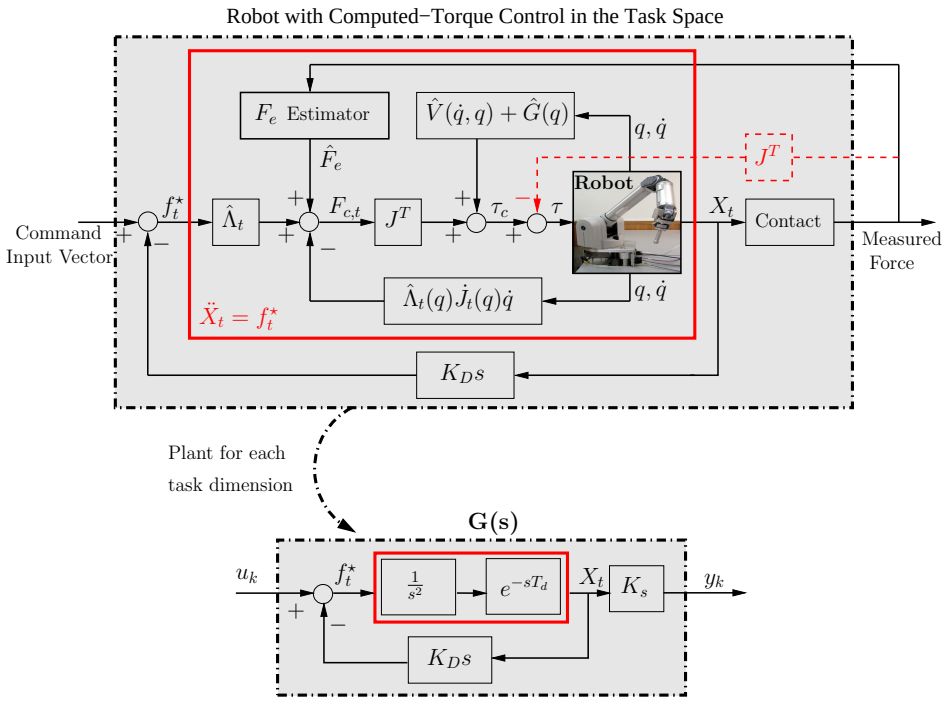


Figure 3.2: Robot with computed torque control in the task space (top) for each task dimension (bottom). The robot plant is linearized and decoupled to achieve a double integrator behavior. The commanded force $F_{c,t}$ includes the compensation of the environment force $(-F_e)$. The computed-torque τ_c compensates for Coriolis, centripetal and gravity forces ($\hat{V}(\dot{q},q)$ and $\hat{G}(q)$). q is a set of joint coordinates describing robot position. $\hat{\Lambda}_t$, J_t and X_t are the mass matrix in the task space, the Jacobian matrix and the end-effector Cartesian position, respectively. G(s) is the system plant for each task dimension and it includes a stiffness contact model with stiffness K_s, a damping loop with gain K_D, the system time delay T_d and the decoupled linearized plant.

Robot Dynamics in the Task Space

In free space, the dynamic equations of a rigid robotic manipulator[1] can be described by Euler-Lagrange's equation (frictions and disturbance inputs are not represented) [40, 93, 110, 34, 28, 92]

$$M(q)\ddot{q} + V(\dot{q},q) + G(q) = \tau, \tag{3.1}$$

[1]Rigid body dynamics has been considered, since the manipulator is very stiff compared to the workspace.

where q is the vector of joint angles, $M(q)$ is the mass matrix, the vector $V(q, \dot{q})$ represents Coriolis and centrifugal forces, and $G(q)$ represents gravity forces. τ is the vector of joint control input torques to be designed. When the robot end-effector is in contact with the environment, an external force F_e is applied to the end-effector and it is included in (3.1) as

$$M(q)\ddot{q} + V(\dot{q}, q) + G(q) + J_t^T(q)F_e = \tau, \tag{3.2}$$

where $J_t(q)$ is the Jacobian matrix corresponding to the end-effector [99]. In our robotic setup, we pre-compensate the Coriolis and gravity forces with the corresponding estimated terms $\hat{V}(\dot{q}, q)$ and $\hat{G}(q)$. We also compensate the environment reaction force with $\hat{F}_e$. $\hat{F}_e$ corresponds to the AOB first state (it can also be given by a filtered version of the measured force). Taking into account these compensations, and representing the term $\hat{M}(q)\ddot{q}$ by $\tau^{'}$ as

$$\tau^{'} = \hat{M}(q)\ddot{q}, \tag{3.3}$$

we obtain [93, 98],

$$\tau_c = \tau^{'} + \hat{V}(\dot{q}, q) + \hat{G}(q) + J_t^T(q)\hat{F}_e. \tag{3.4}$$

To control the contact force at the robot end-effector, while compensating the dynamic effects of the robot, a task space description of the dynamics is required [58, 70]. Therefore, projecting (3.3) into the task space, the corresponding task space control force F_t becomes

$$F_t = \hat{\Lambda}_t(q)\ddot{X}_t - \hat{\Lambda}(q)\dot{J}_t(q)\dot{q}, \tag{3.5}$$

where $\hat{\Lambda}_t(q)$ and X_t are, respectively, the estimated mass matrix and the end-effector position in the task space. From (3.4) and (3.5), the computed torque command τ_c results in

$$\tau_c = J_t^T(q)(F_t + \hat{F}_e) + \hat{V}(\dot{q}, q) + \hat{G}(q). \tag{3.6}$$

Defining $F_{c,t}$ as the commanded force in the task space as

$$F_{c,t} = F_t + \hat{F}_e, \tag{3.7}$$

then (3.6) is equivalent to (as represented in Fig. 3.2)

$$\tau_c = J_t^T(q)F_{c,t} + \hat{V}(\dot{q}, q) + \hat{G}(q). \tag{3.8}$$

Decoupled Control

If the desired plant is

$$\ddot{X}_t = f_t^*, \tag{3.9}$$

i.e., a decoupled system with unitary mass for each Cartesian dimension, then, the commanded torque τ_c should be computed as ((3.5),(3.6),(3.8))

$$\tau_c = J_t^T(q)\left(\hat{\Lambda}_t(q)f_t^* - \hat{\Lambda}_t(q)\dot{J}_t(q)\dot{q} + \hat{F}_e\right) + \hat{V}(\dot{q}, q) + \hat{G}(q). \tag{3.10}$$

In this way, we obtain a decoupled linearized plant (with a double integrator behavior) for each Cartesian dimension, as represented in Fig. 3.2.

3.2.2 Control Design

The system plant G(s) is controlled with explicit force control in the task space. State feedback is implemented and the system state is estimated by a stochastic disturbance observer - the AOB, as represented in Fig. 3.3.

Plant Discretization

The contact is modelled by a spring with stiffness K_s. Assuming perfect feedback linearization and taking into account the system time delay T_d and the damping loop with gain K_D, the system plant transfer function G(s) for each Cartesian dimension becomes

$$G(s) = \frac{Y(s)}{U(s)} = \frac{K_s e^{-sT_d}}{s(s + K_D e^{-sT_d})}. \tag{3.11}$$

$Y(s)$ and $U(s)$ are the output and input of G(s), respectively. The damping loop allows velocity feedback, important when the robot end-effector moves in free-space. For low values of T_d, which is the case of our robotic platform, (3.11) can be approximately given by [20]

$$G(s) \approx \frac{K_s e^{-sT_d}}{s(s + K_D)}. \tag{3.12}$$

The equivalent temporal representation of (3.12) is

$$\ddot{y} + K_D\dot{y} = K_s u(t - T_d), \tag{3.13}$$

where u and y are respectively the plant input (command input) and output (output force). Defining the state variables x_1 as the output force y and x_2

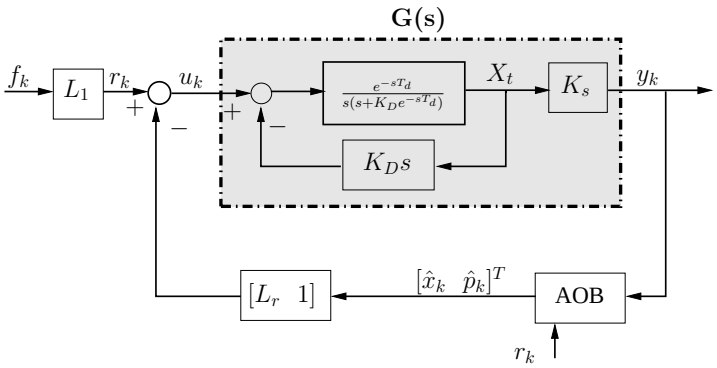


Figure 3.3: G(s) with state feedback from an AOB observer. f_k and y_k are the input and output forces, respectively, and u_k is the command input. $\hat{p}_k$ is the AOB extra state used for disturbance compensation. L_r is the state feedback gain and L_1 is the first element of L_r.

as its derivative $\dot{y}$, (3.13) can be rewritten as

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t - T_d) \\ y(t) = x_1(t) \end{cases}, \tag{3.14}$$

where

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}, A = \begin{bmatrix} 0 & 1 \\ 0 & -K_D \end{bmatrix} \text{ and } B = \begin{bmatrix} 0 \\ K_s \end{bmatrix}. \tag{3.15}$$

Discretizing with sampling time T_s [4], (3.14) becomes

$$\begin{cases} x_k = \Phi x_{k-1} + \Gamma u_{k-1} \\ y_k = Cx_k \end{cases}, \tag{3.16}$$

with

$$x_k = \begin{bmatrix} x_k^* \\ u_{k-d} \\ \vdots \\ u_{k-1} \end{bmatrix}, \; \Gamma = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \text{ and } C = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}^T. \tag{3.17}$$

The vector x_k^* represents the system *core state*. It includes ${}^1x_k$ and ${}^2x_k$, the first and second state, respectively, representing the force and force derivative. The other d extra states u_{k-i} $(i = \{1, \cdots, d\})$ are due to T_d, and y_k is the measured force. The term d is computed from

$$T_d = (d-1)T_s + \gamma, \quad 0 < \gamma \leq T_s. \tag{3.18}$$

Φ, Γ and C represent the state transition, command and measurement discrete state-space matrices. Φ is given by

$$\Phi = \begin{bmatrix} \Phi_1 & \Gamma_1 & \Gamma_0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}, \tag{3.19}$$

where Φ_1, Γ_1 and Γ_0 are defined by

$$\Phi_1 = e^{AT_s} = \phi(T_s), \tag{3.20}$$

$$\Gamma_1 = \phi(T_s - \gamma) \int_0^{\gamma} \phi(\lambda)\ d\lambda\ B, \tag{3.21}$$

and

$$\Gamma_0 = \int_0^{T_s - \gamma} \phi(\lambda)\ d\lambda\ B. \tag{3.22}$$

The matrix $\phi(t)$ is given by

$$\phi(t) = \begin{bmatrix} 1 & (1 - e^{-K_D t})/K_D \\ 0 & e^{-K_D t} \end{bmatrix}. \tag{3.23}$$

Our robotic platform has $T_d = T_s$ resulting, from (3.18),

$$d = 1. \tag{3.24}$$

Consequently, Φ, Γ and C become

$$\Phi = \begin{bmatrix} 1 & (1 - e^{-K_D T_s})/K_D & K_s(T_s + e^{-K_D T_s} - 1)/K_D \\ 0 & e^{-K_D T_s} & K_s(1 - e^{-K_D T_s}) \\ 0 & 0 & 0 \end{bmatrix}, \tag{3.25}$$

$$\Gamma = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \text{ and } C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}. \tag{3.26}$$

State Feedback with an Active Observer

Our linearized plant, represented in discrete state-space by (3.16), (3.25) and (3.26), is controlled with state feedback from an observer. To estimate the system state and to deal with disturbances, such as higher order dynamics, applied external forces, unmodeled terms and parameter mismatches, we use the AOB observer (see Fig. 3.3).

- **State Feedback**
  The state feedback gain matrix L_r is tuned by Ackermann's formula [4] to achieve a critically damped response with time constant τ_f. L_1 represents the first element of L_r and is also used to compensate the DC term.

- **Active Observer**
  The AOB is a MRAC technique based on extended state estimation, requiring knowledge of contact parameters for force control [20, 16, 17]. Besides the stochastic estimation of x_k, the AOB has an extra state $\hat{p}_k$ that estimates and compensates an equivalent disturbance referred to the system input. The ideal performance is obtained if the stiffness used in the nominal plant K_{sn} matches the system stiffness, i.e., $K_{sn} = K_s$. Otherwise, the control performance is degraded to the point where instability can arise for severe parameter mismatch [20]. The stochastic design of the AOB involves the design of Kalman matrices Q_k (model uncertainty) and R_k (measure uncertainty) for the AOB context. Some robotic applications based on the AOB can be found in [14, 18, 19, 5, 83, 20, 85, 84, 82].

  Consider the system plant G(s), represented by (3.17), plus stochastic inputs ξ_k and η_k

$$\begin{cases} x_k = \Phi x_{k-1} + \Gamma u_{k-1} + \xi_k \\ y_k = C x_k + \eta_k \end{cases} . \tag{3.27}$$

  The stochastic inputs ξ_k and η_k represent model and measurement uncertainties, respectively. Controlling this system through state feedback from an AOB, the state estimate of (3.27) is given by

$$\begin{bmatrix} \hat{x}_k \\ \hat{p}_k \end{bmatrix} = \Phi_n \begin{bmatrix} \hat{x}_{k-1} \\ \hat{p}_{k-1} \end{bmatrix} + \Gamma_n r_{k-1} + K_k(y_k - \hat{y}_k), \tag{3.28}$$

  and the output force $\hat{y}_k$ is given by

$$\hat{y}_k = C_n \left(\Phi_n \begin{bmatrix} \hat{x}_{k-1} \\ \hat{p}_{k-1} \end{bmatrix} + \Gamma_n r_{k-1} \right), \tag{3.29}$$

with

$$\Phi_n = \begin{bmatrix} \Phi_{sn} - \Gamma_{sn} L_r & 0 \\ 0 & 1 \end{bmatrix}, \tag{3.30}$$

$$\Gamma_n = \begin{bmatrix} \Gamma_{sn} \\ 0 \end{bmatrix} \quad \text{and} \quad C_n = \begin{bmatrix} C_{sn} & 0 \end{bmatrix}. \tag{3.31}$$

The real matrices Φ, Γ and C represented in (3.27) are now replaced by the homologous nominal matrices Φ_{sn}, Γ_{sn} and C_{sn}, respectively. Γ_{sn} and C_{sn} are equal to Γ and C. Φ_{sn} differs from Φ by replacing the unknown real stiffness value K_s by a nominal stiffness value K_{sn}. K_k are the Kalman gains and reflect the uncertainty associated to each state, which is a function of ξ_k and η_k [8]. It is computed by

$$K_k = P_{1k} C^T [C P_{1k} C^T + R_{\text{noise}}]^{-1}, \tag{3.32}$$

where

$$P_{1k} = \Phi_n P_{k-1} \Phi_n^T + Q_{\text{noise}}, \tag{3.33}$$

and

$$P_k = P_{1k} - K_k C P_{1k}. \tag{3.34}$$

Q_{noise} and R_{noise} are, respectively, the system noise matrix and the measurement noise matrix [16].

In our robotic platform, Φ_{sn} is given by (from (3.25))

$$\Phi_{sn} = \begin{bmatrix} 1 & (1 - e^{-K_D T_s})/K_D & K_{sn}(T_s + e^{-K_D T_s} - 1)/K_D \\ 0 & e^{-K_D T_s} & K_{sn}(1 - e^{-K_D T_s}) \\ 0 & 0 & 0 \end{bmatrix}. \tag{3.35}$$

Knowing (3.24), the system state estimated by the AOB of our system plant has dimension four, as represented by (3.36),

$$\hat{x}_k = \begin{bmatrix} {}^1\hat{x}_k \\ {}^2\hat{x}_k \\ u_{k-1} \\ \hat{p}_k \end{bmatrix}. \tag{3.36}$$

The first two states represent the estimated end-effector force and its derivative. The third state appears due do the system time delay and the fourth state is the AOB extra state.

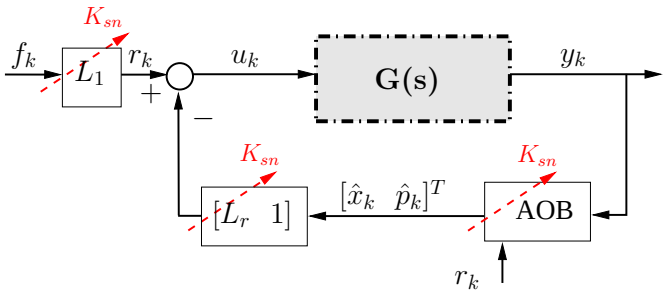


Figure 3.4: Nominal terms involved in the nominal stiffness update process.

3.2.3 Stiffness Adaptation

Stiffness adaptation (i.e., update of the nominal stiffness value K_{sn} by the current estimation of K_s) can be achieved by updating the control laws accordingly [17, 20]. In Fig. 3.4, the dotted arrows represent the terms that are dependent on the nominal stiffness value and should be updated. The update involves the recalculation of the feedback gain matrix L_r, the adaptation of Φ_{sn} (and consequently Φ_n) and, inherently, the Kalman gains since they depend on Φ_n [17].

3.3 System Plant with Disturbances

Many different disturbances, such as measurement noise or modeling errors, can cause the system to deviate from the ideal behavior. The lumped effect of disturbances can be represented by an equivalent disturbance signal referred to the system input. In subsequent chapters, it will be useful to distinguish between disturbance components due to stiffness mismatch and other sources. This section provides the required notation and modeling background.

3.3.1 Stiffness Mismatch Disturbance

A stiffness mismatch ΔK_s corresponds to a mismatch between the nominal stiffness value K_{sn} and the real value K_s. This mismatch, given by (1.1), originates a *stiffness mismatch disturbance*, which can be modeled by $p_k(\Delta K_s)$ referred to the system input. An equivalent representation of the system plant G(s) is shown in Fig. 3.5, where G(s) is modeled by a nominal plant (with K_{sn}) plus the disturbance $p_k(\Delta K_s)$.

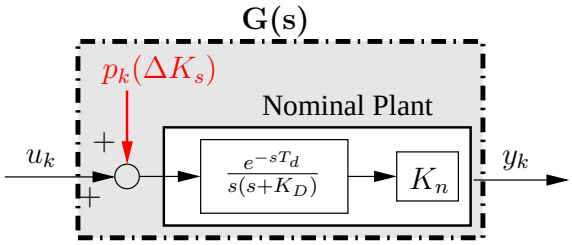


Figure 3.5: G(s) representation with stiffness mismatch disturbance. G(s) can be modeled by a plant with nominal stiffness K_{sn} plus a disturbance $p_k(\Delta K_s)$ due to the stiffness mismatch, referred to the system input. $p_k(\Delta K_s)$ is due to the stiffness mismatch $\Delta K_s = K_s - K_{sn}$. u_k is the command input and y_k is the output force.

3.3.2 Additional Disturbances

Other disturbances besides stiffness mismatch can exist (e.g., frictions, higher order unmodeled terms, etc). Those *additional disturbances* can be represented by $p_k(o)$, an equivalent disturbance referred to the system input, as shown in Fig. 3.6. So, the overall disturbance p_k referred to the system input, which includes $p_k(\Delta K_s)$ and $p_k(o)$, can be represented by

$$p_k = p_k(\Delta K_s) + p_k(o). \tag{3.37}$$

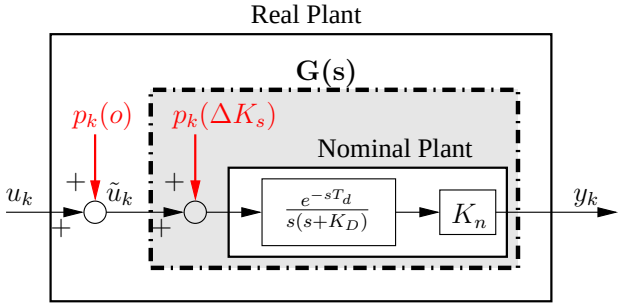


Figure 3.6: Real plant with stiffness mismatch and other disturbances. The unknown disturbance $p_k(o)$ referred to the system input, lumps all disturbance sources other than stiffness mismatch. G(s) can be modeled by a nominal plant with K_{sn} plus a disturbance due to stiffness mismatch $p_k(\Delta K_s)$.

Chapter 4

Model-Based Estimation

Contents

4.1 Introduction

This chapter presents and discusses two system stiffness estimation approaches, based on explicit models of the system. The Active State Based Algorithm (ASBA) is based on the least-squared-error (LSE) comparison between the theoretical prediction of the AOB extra state, based on models

of the manipulator and contact for different scenarios of stiffness mismatch, with its actual value. The Candidate Observer Based Algorithm (COBA) introduces two additional force observers, designed using different parameterizations of the contact model. The prediction error of these observers will be further processed to obtain the stiffness estimation. ASBA is presented in Section 4.2, while COBA is discussed in Section 4.3.

4.2 ASBA: Active State Based Algorithm

4.2.1 Overview

ASBA is a non-recursive algorithm that estimates the system stiffness by contrasting the actual AOB active state with theoretically expected results. ASBA is based on the premise that the active state embeds information about unmodeled disturbances. Assuming that the stiffness mismatch is the only source of disturbance (as described in Section 3.3.1), it is possible to infer the stiffness mismatch from active state analysis. To this purpose, active state curves are computed for several mismatch scenarios. These are referred to as *candidate curves*. An estimation is obtained by noting that the candidate curve that better matches the actual AOB active state can be assumed to have been computed with parameters that more closely correspond to the real stiffness mismatch. ASBA is described in Section 4.2.2. Section 4.2.3 discusses the computation of the candidate active state curves and Section 4.2.4 presents the parametrization of ASBA. Section 4.2.5 shows an illustrative example and in Section 4.2.6 some considerations about ASBA are made. The demonstrations of the supporting theorem and lemma are collected in the Appendix (Section A.2).

4.2.2 Description

ASBA computes several candidate curves $\hat{p}_k^{K_{sn},K_{sc}}$ for the AOB active state, for several stiffness mismatch scenarios. Each scenario considers the current nominal value K_{sn} and a candidate value K_{sc} for the system stiffness K_s. The $\hat{p}_k^{K_{sn},K_{sc}}$ curve closest to the actual active state gives information about the current stiffness mismatch. A detailed description of ASBA is presented in Algorithm 1. The parameters of ASBA are the predetermined candidate stiffnesses S and the time windows w, T_{LSE}, t_o and t_u. These allow establishing compromises between precision, efficiency, responsiveness and consistency. Other parameters, required for step 1, are detailed in the discussion of Algorithm 2. A higher values of T_{LSE}, w and a high number

of predetermined candidate stiffnesses improve precision at the cost of efficiency. High values of t_o and t_u increase response consistency at the cost of responsiveness.

ASBA Algorithm

At each instant k, for each Cartesian dimension, do:

1. Compute the candidate $\hat{p}_k$ curves ($\hat{p}_k^{K_{sn},K_{sc}}$) for a set S of specified stiffnesses over a specified time window w, considering the reference force r_k and the nominal stiffness value K_{sn}. If valid candidate curves cannot be computed (see Algorithm 2 for computing candidate $\hat{p}_k$ curves), preserve the previous estimation and terminate the current iteration.

2. Using a LSE analysis over a specified time window T_{LSE}, identify the candidate curve $\hat{p}_k^{K_{sn},K_{sc}}$ closest to the actual $\hat{p}_k$. A proposed estimation of K_s, $\hat{K}_{s,k}^{-}$, is given by the candidate value K_{sc} corresponding to the best match $\hat{p}_k^{K_{sn},K_{sc}}$,

$$\hat{K}_{s,k}^{-} = \mathrm{Z} : \left(\sum_{k-T_{LSE}}^{k} (\hat{p}_k - \hat{p}_k^{K_{sn},Z})^2 = \min_{s \in S} \sum_{k-T_{LSE}}^{k} (\hat{p}_k - \hat{p}_k^{K_{sn},s})^2 \right).$$

3. If the proposed estimations of K_s have remained constant over a specified time window (t_o if the estimated stiffness is decreasing, t_u if the estimated stiffness is increasing), then the proposed estimation is accepted as good and made definitive, otherwise, preserve the previous estimation,

$$\hat{K}_{s,k} = \begin{cases} \hat{K}_{s,k}^{-} & \text{if } \hat{K}_{s,k}^{-} > K_{sn} \text{ and } \hat{K}_{s,i}^{-} = \hat{K}_{s,k}^{-},\ i = k - t_u, ..., k \\ \hat{K}_{s,k}^{-} & \text{if } \hat{K}_{s,k}^{-} < K_{sn} \text{ and } \hat{K}_{s,i}^{-} = \hat{K}_{s,k}^{-},\ i = k - t_o, ..., k \\ \hat{K}_{s,k-1} & \text{otherwise} \end{cases}.$$

Algorithm 1: ASBA Algorithm. This algorithm is based on the LSE analysis of theoretical predictions of the AOB extra state for several scenarios of stiffness mismatch.

Fig. 4.1 represents the ASBA Algorithm embedded in the control loop.

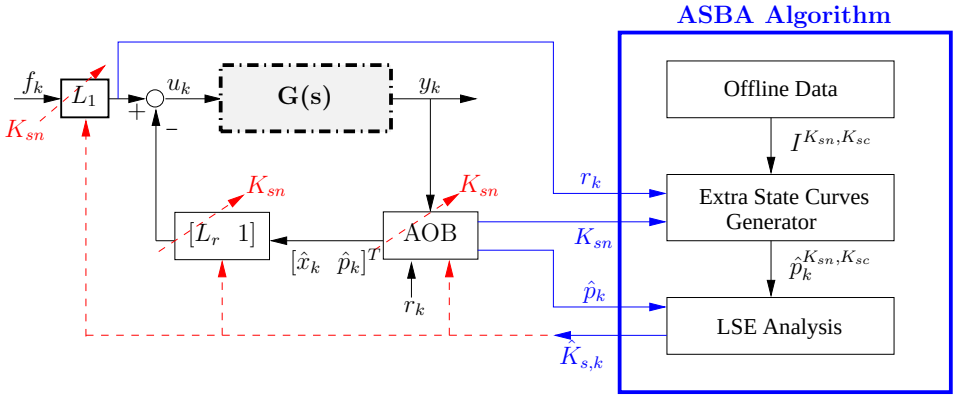


Figure 4.1: ASBA Algorithm embedded in the control loop. ASBA receives the reference force r_k, the current nominal stiffness value K_{sn} and the AOB active state $\hat{p}_k$. It computes an estimation $\hat{K}_{s,k}$ of the system stiffness value. Online stiffness adaptation can be done by adjusting the DC gain L_1, the feedback gains L_r and the nominal AOB model to conform to the estimated stiffness $\hat{K}_s$.

The reference force and nominal stiffness value are used to compute candidate active state curves, which are then compared to the actual active state. No object position data is required. The estimation $\hat{K}_s$ computed by ASBA can be used for online stiffness adaptation by adjusting the DC gain, the feedback gains and the nominal AOB model, as described in Section 3.2.3.

4.2.3 Active State Computation

ASBA requires the computation of several candidate $\hat{p}_k$ curves ($\hat{p}_k^{K_{sn},K_{sc}}$) for different mismatch scenarios (see Step 1 of Algorithm 1). These can be computed with Algorithm 2. The parameters of Algorithm 2 perform tradeoffs between efficiency, precision, responsiveness and consistency. Higher values of w, w_2, T_I, as well as higher number of 2-permutations of S, improve consistency and precision at the cost of efficiency. Higher values of T_L improve consistency at the cost of responsiveness. Relevant theoretical results for ASBA can be found in Theoretical Results 1. Lemma 1 establishes the relation between the active state and the stiffness mismatch, while Theorem 1 supports Algorithm 2, by proving that, given K_{sn} and K_s, $\hat{p}_k^{K_{sn},K_s}$ ($\hat{p}_k$ for short) can be computed using the active state response I^{K_{sn},K_s} (for a step input) and the online reference difference Δr_k. This method is used in Step 2 of Algorithm 2 for computing $\hat{p}_k^{K_{sn},K_{sc}}$, by replacing K_s by K_{sc}.

Algorithm for computing candidate $\hat{p}_k$ curves

Offline: Compute the step response of the active state for all 2-permutations (K_{sn}, K_{sc}) of a set S of specified stiffnesses. These step responses $(I^{K_{sn},K_{sc}})$ are truncated to a finite time interval T_I.
Online:

1. Verify if a large variation of the reference r_k is present at the w^{th} previous sample. A large variation is defined as a variation of the absolute value of the reference value that is at least T_L times greater than average value of the reference in the last w_2 samples (see Fig. 4.2). If a large variation is not present, return without computing valid candidate curves. Otherwise, continue in Step 2.

2. For all $I^{K_{sn},K_{sc}}$ computed offline where K_{sn} is the current nominal stiffness, compute the corresponding candidate curve in the interval of the previous w samples, using the convolution $\hat{p}_k^{K_{sn},K_{sc}} = (I^{K_{sn},K_{sc}} * \Delta r)_k$.

Algorithm 2: Algorithm for computing candidate $\hat{p}_k$ curves for different stiffness mismatch scenarios.

4.2.4 Parameters

The parameters of ASBA reflect some necessary tradeoffs and practical implementation considerations:

S The set of specified candidate stiffnesses must be chosen in advance. The estimations provided by ASBA will always correspond to one of these values. To achieve a desired precision, the candidates should be uniformly distributed across the expected range of stiffness values. Higher densities of candidate values increase precision, with the drawback of deteriorating runtime performance as additional candidates must be considered. Memory resource requirements also increase due to the higher volume of offline data that must be stored.

w When computing $\hat{p}_k^{K_{sn},K_{sc}}$, for practical reasons (speed and memory limitations), a limited interval (of duration w) must be considered

Lemma 1. *Assuming null initial conditions, the active state $\hat{p}_k$ of the active observer can be given by*

$$\hat{p}_k = KC \sum_{i=1}^{k} \sum_{j=0}^{i-1} Z_{j,i} \Gamma r_{i-1-j},$$

where r, K, C and Γ are the reference force, the Kalman gains, the output and the command matrices, respectively. Z is represented by

$$Z_{j,k} = (\Phi_{n,k-1} + \Delta\Phi_{k-1} - \Gamma L)^j - (\Phi_{n,k-1} - \Gamma L)^j,$$

where L is the state feedback matrix, Φ_n is the nominal state transition matrix and $\Delta\Phi$ is the mismatch matrix given by the difference between the real and the nominal state transition matrices.

Theorem 1. *If I^{K_{sn},K_s} is the active state response for K_{sn} and K_s when the system is excited with a step input, then $\hat{p}_k$ can be computed, for any arbitrary reference input, through the convolution:*

$$\hat{p}_k = (I^{K_{sn},K_s} * \Delta r)_k,$$

where $\Delta r_k = r_k - r_{k-1}$ is the reference input difference.

Theoretical Results 1: Results relevant for analysis of ASBA operation. Proofs are in the Appendix (Section A.2).

when computing the convolution, resulting in:

$$\hat{p}_k^{K_{sn},K_{sc}} = \sum_{i=k-w}^{k} I_{k-i}^{K_{sn},K_{sc}} \Delta r_i. \tag{4.1}$$

The parameter w embodies a tradeoff between accurate computation of $\hat{p}_k^{K_{sn},K_{sc}}$ and performance.

T_{LSE} The LSE analysis must be restricted to a finite time interval T_{LSE}. If a smaller interval is considered, then performance is improved. The algorithm will also be more responsive as tendencies expressed in older data are quickly disregarded in favor of more recent data. However, fo-

cusing the LSE analysis on a more restricted interval increases susceptibility to high frequency disturbances, resulting in poorer estimation consistency.

$\mathbf{T_I}$ The curve $I^{K_{sn},K_{sc}}$ must be computed for a finite time interval T_I. While larger values of T_I are potentially related with increased precision in the computation of $p_k^{K_{sn},K_{sc}}$, in practice for stable configurations $I^{K_{sn},K_{sc}} \to 0$. Therefore, when performing the offline generation of $I^{K_{sn},K_{sc}}$, it is possible to determine a value for T_I that binds the error of $I^{K_{sn},K_{sc}}$ below a desired error margin. Higher values of T_I have negative performance impact as additional data must be considered for computing $p_k^{K_{sn},K_{sc}}$ in each iteration.

$\mathbf{T_L}, \mathbf{w_2}$ Estimations are computed in the presence of a large variation of r_k at the w^{th} previous sample (Fig. 4.2). A large variation was defined to be a variation of the absolute value of the reference that is T_L times the average value of reference in the last w_2 samples. The values of w, w_2, T_I and T_{LSE} should be articulated so that $w_2 > w$, $T_I \geqslant w$ and $T_{LSE} \leqslant w$, minimizing the adverse impact of the restrictions on these parameters.

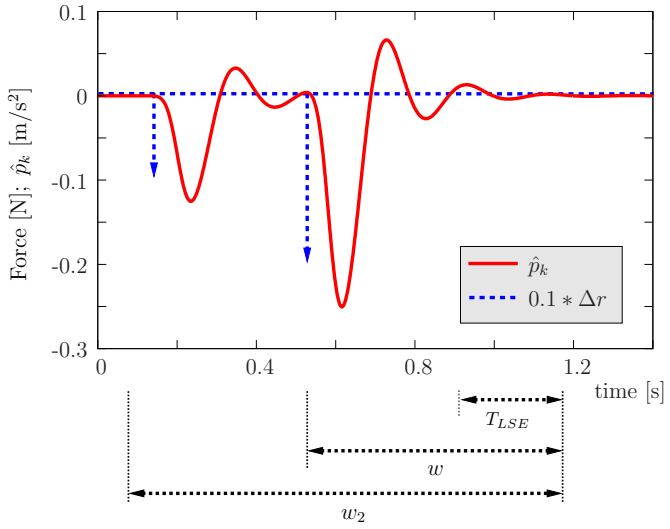


Figure 4.2: Illustration of the time intervals associated with ASBA operation. Underestimated scenario: $K_{sn} = 500$ [N/m] and $K_s = 700$ [N/m]. System excited with two step inputs: -10 [N] at $t = 0.12$ [s] and -30 [N] $t = 0.52$ [s].

t_u, t_o To improve response consistency, the actual nominal system stiffness is only updated when the estimated stiffness is stabilized for a parameterizable number of samples. This parameter is defined differently for both overestimated (t_o) and underestimated (t_u) scenarios. This means that nominal stiffness will be updated to a lower (higher) value only if estimation has stabilized for t_o (t_u) samples. As illustrated in Fig. 4.3, the active state $\hat{p}_k$ has higher dynamics in underestimated scenarios. This led us to define different values for t_o and t_u. Specifically, t_o will typically be higher than t_u, to compensate for the slower variation of $\hat{p}_k$ in overestimated scenarios.

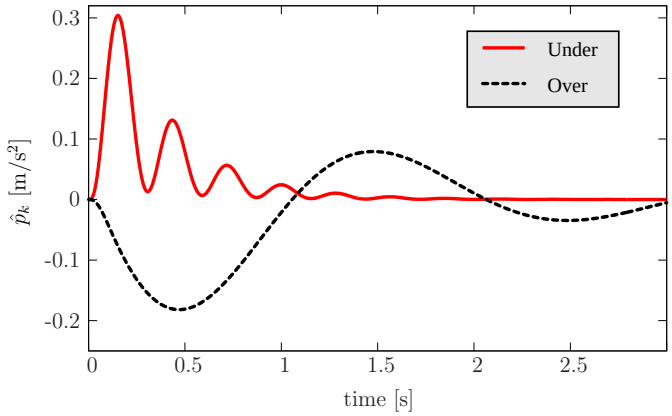


Figure 4.3: Active state behavior with two different scenarios of stiffness mismatch: underestimated and overestimated. In the underestimated scenario, $K_{sn} = 200$ [N/m] and $K_s = 1000$ [N/m]. In the overestimated scenario, $K_{sn} = 1000$ [N/m] and $K_s = 200$ [N/m]. A step input is applied at $t = 0$ [s].

4.2.5 Illustrative Example

In this section an illustrative example of execution of ASBA is presented. A system plant is considered where the stiffness mismatch is the only unmodeled disturbance, with $K_{sn} = 500$ [N/m] and $K_s = 630$ [N/m]. The candidate stiffnesses are defined as $S = \{300, 400, 500, 600, 700\}$ [N/m]. The $I^{K_{sn},K_{sc}}$ curves are computed offline for every 2-permutation $\{K_{sn}, K_{sc}\}$ of S. Fig. 4.4 presents $I^{K_{sn},K_{sc}}$ curves for $K_{sn} = 500$ [N/m] (the curve for $K_{sc} = 500$ [N/m] is zero and is omitted for clarity). When the input force represented in Fig. 4.5(a) is applied to the system, the stiffness mismatch disturbance induces the activity of the AOB extra state seen in Fig. 4.5(b). ASBA estimates the system stiffness by contrasting this activity

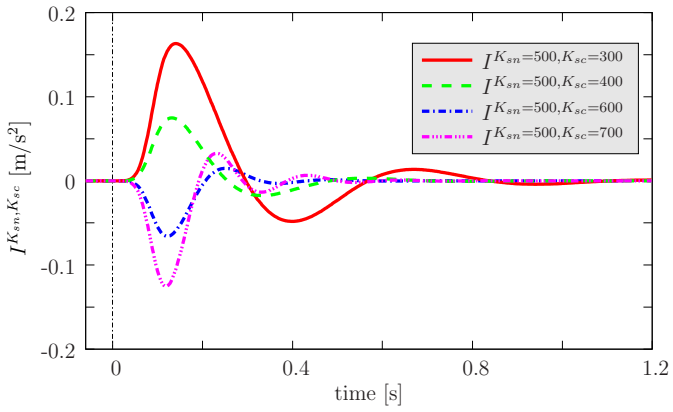


Figure 4.4: $\mathbf{I^{K_{sn},K_{sc}}}$ curves for $\mathbf{K_{sn} = 500}$ [N/m]. The duration of these curves is in practice defined by the parameter T_I ($T_I = 1000$ in this example).

with expected results for candidate scenarios based on nominal and system stiffnesses derived from 2-permutations of S. The expected active state for a candidate scenario $\{K_{sn}, K_{sc}\}$ ($\hat{p}_k^{K_{sn},K_{sc}}$) is computed as the convolution of $I^{K_{sn},K_{sc}}$ with Δr_k, for the current nominal stiffness value, yielding the results represented in Fig. 4.6(a). Estimations are obtained by finding which $\hat{p}_k^{K_{sn},K_{sc}}$ is the closest match to the actual $\hat{p}_k$. This is achieved by computing the LSE over a specified time window. Fig. 4.6(b) represents the error between $\hat{p}_k^{K_{sn},K_{sc}}$ and $\hat{p}_k$. Fig. 4.7 illustrates this process for a time instant just after the second step of the input force. Fig. 4.7(a) shows the mean squared-error computed at iteration $k = 566$ ($t = 0.67$ [s]), over the previous 12 samples ($T_{LSE} = 12$). A proposed estimation $\hat{K}_{s,k}^{-}$ is found by determining that the closest match corresponds to the curve with $K_{sc} = 600$ [N/m]. The proposed estimation is accepted as good if $\hat{K}_{s,k}^{-}$ has remained constant over the last $t_u = 6$ iterations (if the estimation was decreasing, the complementary parameter t_o would be considered instead). In Fig. 4.7(b), it can be observed that the MSE of 600 [N/m] has remained the lowest during that interval, so the proposed estimation is accepted as good and $\hat{K}_{s,k} = \hat{K}_{s,k}^{-}$, otherwise the earlier estimation would be preserved ($\hat{K}_{s,k} = \hat{K}_{s,k-1}$).

4.2.6 Discussion

The goal of this work is to present a model-based technique for stiffness estimation that does not require object position data and is amenable to theoretical validation, as discussed in the introductory chapter (see Chapter

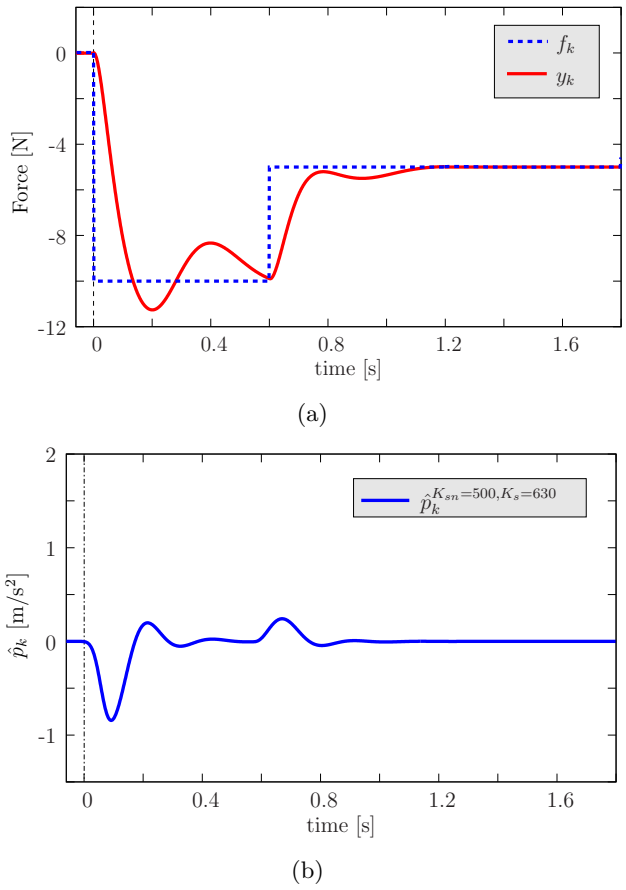


Figure 4.5: Forces and AOB active state. (a) Input f_k and output y_k forces. (b) Active state $\hat{p}_k$ for the mismatch $K_{sn} = 500$ [N/m] and $K_{sc} = 630$ [N/m].

1). ASBA fits this role, as it is a model-based algorithm capable of successfully estimating system stiffness without using object position data. The computation of the candidate curves is grounded on the theoretical results presented on Theorem 1. Nevertheless, some shortcomings of this approach were identified and addressed in subsequent work (see Section 4.3).

1. **Runtime Efficiency**: Although computation time is bounded, estimation time increases proportionally with the number of candidate stiffnesses and also with the duration of the time intervals (T_I, w, w_2 and T_{LSE}). This may cause computation time to be excessive for runtime operation if high precision and accurate estimations are de-

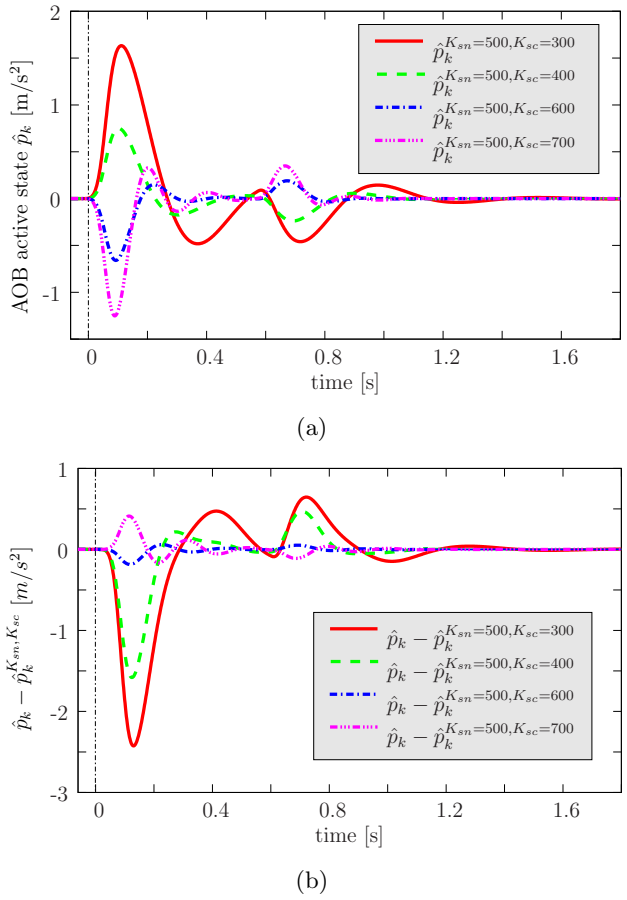
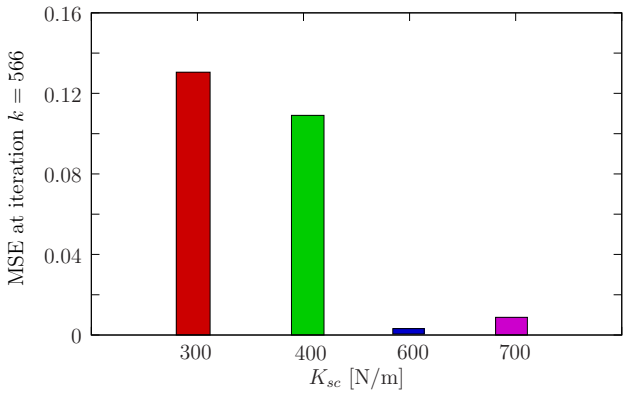


Figure 4.6: Candidate active states and their difference w.r.t. the actual active state. (a) Theoretical prediction of active state for each candidate scenario $\hat{p}_k^{K_{sn},K_{sc}}$ for $K_{sn} = 500$ [N/m]. These curves were computed with the parameters $w_2 = w = 835$ and $T_L = 0.1$. (b) Difference between the theoretical prediction for each candidate scenario $\hat{p}_k^{K_{sn},K_{sc}}$ and the actual active state $\hat{p}_k$.

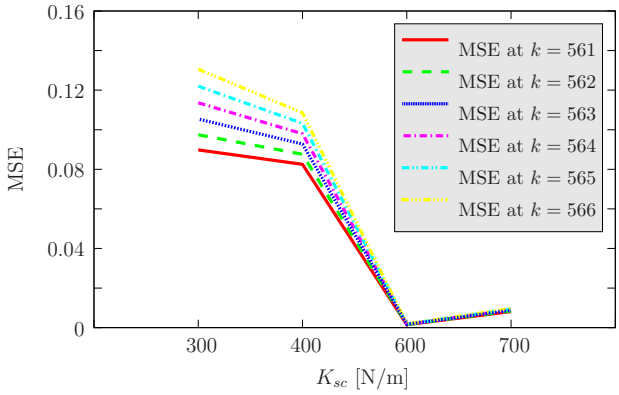
sired. Offline data and temporary candidate curves must be stored, so memory requirements are also fairly high.

2. **Adaptive Behavior**: The LSE approach hampers algorithm response to dynamic environment conditions, making it more difficult to adjust the estimation accordingly.

3. **Implementation Complexity**: Practical implementation considerations impose some deviations from the theoretically ideal formulas



(a)



(b)

Figure 4.7: Error analysis. (a) Mean squared value of the error $\hat{p}_k - \hat{p}_k^{K_{sn}-K_{sc}}$ over an interval of the last $T_{LSE} = 12$ samples, measured at instant $k = 566$. (b) Mean squared-error over an interval of the last $T_{LSE} = 12$ samples, measured from instants $t = 0.6108$ [s] until $t = 0.6168$ [s] (corresponding, respectively, to the iterations from $k = 561$ until $k = 566$).

(e.g.: finite summations). These impact negatively on the algorithm behavior, although the effect may be hard to quantify.

4. **Additional Disturbances**: The operation principle of ASBA assumes that the only source of disturbance is the stiffness mismatch (or that it is the dominant source of disturbance, allowing other to be ignored). In practice, operation will be degraded if these assumptions fail to hold. Some of the more arbitrary aspects of the algorithm (e.g:

stable estimations required for acceptance of results, disregarding results for low variations of the reference) attempt to address these issues and improve consistency, but impair responsiveness and analyzability.

4.3 COBA: Candidate Observer Based Algorithm

4.3.1 Overview

COBA is an online system stiffness estimation algorithm, based on force data [27]. It uses two Force OBservers (FOBs) tuned with two different nominal values of system stiffness. The differentiated tuning makes each observer provide an alternative estimation of the output force. By contrasting these estimations with the actual measured force, the system stiffness estimation $\hat{K}_{s,k}$ is obtained. COBA is described in Section 4.3.2 and an illustrative example follows in Section 4.3.3. The analysis of COBA under ideal and non-ideal operating conditions is done in Section 4.3.4. The relevant parameters for COBA operation are presented in Section 4.3.5. Section 4.3.6 discusses COBA. The demonstrations of the supporting theorems, lemmas and corollaries are collected in the Appendix (Section A.3).

4.3.2 Description

COBA contrasts the measured force with the estimated forces of two FOBs. Each FOB is a force observer that uses a model based on a candidate value for the system stiffness. The FOBs are designed in such a way that the force errors, computed as the error between the output of each FOB and the measured force, are proportional to the correspondent candidate stiffness mismatch. Therefore, the stiffness estimation can be obtained by computing, from the known FOB force errors, the stiffness value that would correspond to a candidate with null force error (and correspondingly no stiffness mismatch).

Force Observers

Each FOB is defined by

$$\begin{cases} x_{c,k} = \Phi_c x_{c,k-1} + \Gamma u_{c,k-1} + K_c(y_k - y_{c,k}^-) \\ y_{c,k} = C x_{c,k} \end{cases}, \tag{4.2}$$

where $x_{c,k}$ is the FOB state and is of form

$$x_{c,k} = \begin{bmatrix} {}^1x_{c,k} & {}^2x_{c,k} & u_{c,k-1} \end{bmatrix}^T. \tag{4.3}$$

$u_{c,k}$ is the FOB command input and $y_{c,k}$ is the FOB force estimation. $y^-_{c,k}$ is the *a-priori* estimation of $y_{c,k}$,

$$y^-_{c,k} = C(\Phi_c x_{c,k-1} + \Gamma u_{c,k-1}). \tag{4.4}$$

Φ_c is obtained by replacing K_s by a candidate value K_{sc} in (3.12) and discretizing with $T_s = T_d$ (see (3.24)). K_c is the FOB gain, which is equal in both FOBs. It is represented by

$$K_c = \begin{bmatrix} {}^1K_c & {}^2K_c & 0 \end{bmatrix}^T. \tag{4.5}$$

${}^1K_c$ and ${}^2K_c$ should satisfy certain conditions to guarantee stability (see Theoretical Results 4). K_{sc} is the only design parameter that is specific to each FOB.

Force Observers Mismatches

For each FOB, ΔK_{sc}, $\Delta x_{c,k}$ and $\Delta\Phi_c$ represent the stiffness, state estimation and state transition matrix mismatches, respectively, i.e.,

$$\Delta K_{sc} = K_s - K_{sc}, \tag{4.6}$$

$$\Delta x_{c,k} = x_k - x_{c,k}, \tag{4.7}$$

and

$$\Delta\Phi_c = \Phi - \Phi_c. \tag{4.8}$$

$\Delta\Phi_c$ can be written as

$$\Delta\Phi_c = \Delta K_{sc} T, \tag{4.9}$$

where

$$T = \begin{bmatrix} 0 & 0 & (T_s + e^{-K_D T_s} - 1)/K_D \\ 0 & 0 & 1 - e^{-K_D T_s} \\ 0 & 0 & 0 \end{bmatrix}. \tag{4.10}$$

$\Delta\Phi_c$ is proportional to ΔK_{sc} since T is independent of ΔK_{sc}. This linear relation is the fundamental result behind the COBA Algorithm.

COBA Algorithm Description

A detailed description of COBA is presented in Algorithm 3. The parameters of COBA are K_{sc_1}, K_{sc_2}, ${}^1K_c$, ${}^2K_c$ and $G_{\min}$. The specific values of the candidate stiffnesses (K_{sc_1} and K_{sc_2}) are unimportant as long as they are sufficiently distinct to avoid numerical problems computing (4.12). The

values of ${}^1K_c$ and ${}^2K_c$ determine the dynamic response of the FOBs and should satisfy certain conditions detailed in Theorem 4. Higher values of $G_{\min}$ improve conditioning at the cost of responsiveness. These parameters are further discussed in Section 4.3.6. Fig. 4.8 presents COBA embedded in the control loop. The input command and measured force values are used to compute the FOBs state. No object position data is required. The estimation $\hat{K}_s$ computed by COBA can be used for online K_{sn} stiffness adaptation.

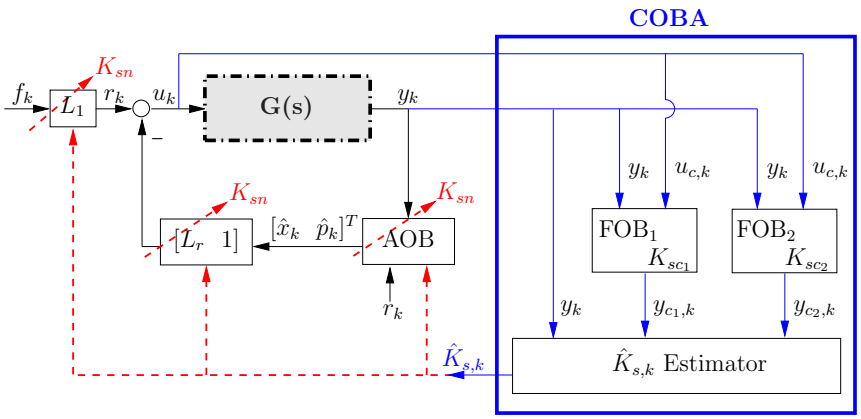


Figure 4.8: COBA architecture embedded in the control loop. The estimation $\hat{K}_{s,k}$ can be used to adapt the state feedback gains L_r, DC compensation gain L_1 and AOB matrices. K_{sc_1} and K_{sc_2} are the stiffnesses of each candidate observer FOB$_1$ and FOB$_2$.

COBA Algorithm

1. Set to FOB_1 a soft stiffness K_{sc_1} and to FOB_2 a stiff stiffness K_{sc_2} (e.g., $K_{sc_1} = 100$ [N/m] and $K_{sc_2} = 6000$ [N/m].
2. Set the initial state of both FOBs with the same value $\hat{x}_0$: $x_{c_1,0} = x_{c_2,0} = \hat{x}_0$.
3. For each iteration k do:
   (a) From (4.2), compute $y_{c_1,k}$ and $y_{c_1,k}$.
   (b) Compute the force errors ${}^1\Delta x_{c_1,k} = y_k - y_{c_1,k}$ and ${}^1\Delta x_{c_2,k} = y_k - y_{c_2,k}$.
   (c) Compute the 2D points $(K_{sc_1}, {}^1\Delta x_{c_1,k})$ and $(K_{sc_2}, {}^1\Delta x_{c_2,k})$.
   (d) Compute the line that contains the points of c). It is given by
   $$ {}^1\Delta x_{c,k} = {}^1\hat{G}_k K_{sc} + \hat{\varphi}_k, \tag{4.11}$$
   where the slope ${}^1\hat{G}_k$ and the offset $\hat{\varphi}_k$ are given by
   $$ {}^1\hat{G}_k = \frac{{}^1\Delta x_{c_2,k} - {}^1\Delta x_{c_1,k}}{K_{sc_2} - K_{sc_1}} = \frac{y_{c_1,k} - y_{c_2,k}}{K_{sc_2} - K_{sc_1}} \tag{4.12}$$
   and
   $$ \hat{\varphi}_k = {}^1\Delta x_{c_2,k} - {}^1\hat{G}_k K_{sc_2}. \tag{4.13}$$
   (e) Compute $\hat{K}^-_{s,k}$ given by
   $$ \hat{K}^-_{s,k} = -\frac{\hat{\varphi}_k}{{}^1\hat{G}_k}, \tag{4.14}$$
   which corresponds to the intersection point of line (4.11) with the stiffness axis.
   (f) The estimation is given by
   $$ \hat{K}_{s,k} = \begin{cases} \hat{K}^-_{s,k} & \text{if } |{}^1\hat{G}_k| \geq G_{\min} \\ \hat{K}_{s,k-1} & \text{otherwise} \end{cases}. \tag{4.15}$$

Algorithm 3: COBA Algorithm. This algorithm is based on the analysis of the output of two force observes and the current measured force.

4.3.3 Illustrative Example

In this section an illustrative example of the execution of COBA is presented. A system plant is considered where the stiffness mismatch is the only unmodeled disturbance, with $K_{sn} = 500$ [N/m] and $K_s = 630$ [N/m]. Two candidate observers with candidate stiffnesses of 100 and 6000 [N/m] are used. In Fig. 4.9(a), the measured force obtained by applying a 2-step input can be observed. The corresponding force estimations from both FOBs are also presented. The discrepancies between the measured and estimated forces, seen in Fig. 4.9(b), are due to mismatches between the system stiffness, nominal and candidate stiffnesses. COBA takes advantage of these

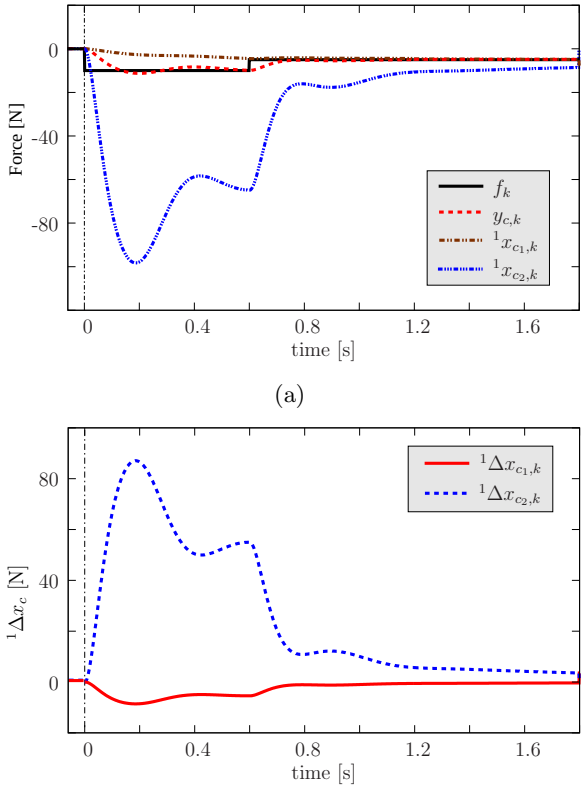


Figure 4.9: Forces and FOB force errors. (a) System input, system output and FOB estimated forces. (b) Errors between the output force y_k and the FOBs forces (${}^1x_{c_1,k}$ and ${}^1x_{c_2,k}$). ${}^1\Delta x_{c_1,k}$ represents the force error between y_k and ${}^1x_{c_1,k}$ and ${}^1\Delta x_{c_2,k}$ represents the analogous error but with respect to ${}^1x_{c_2,k}$.

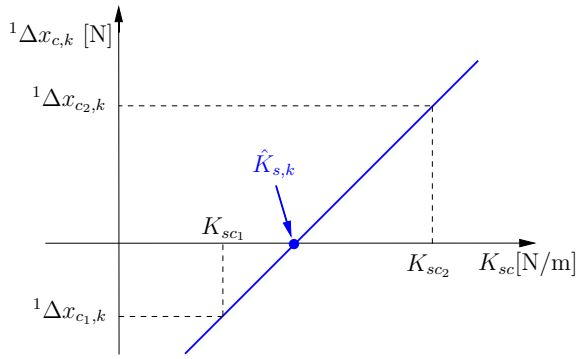


Figure 4.10: Graphical representation of COBA. The bold line contains the points $(K_{sc_1}, {}^1\Delta x_{c_1,k})$ and $(K_{sc_2}, {}^1\Delta x_{c_2,k})$. The intersection point with the stiffness axis corresponds to $\hat{K}_{s,k}$.

differences to estimate the system stiffness, as illustrated in Fig. 4.10. At each instant, the estimation is obtained by computing the intersection point between the line connecting the force estimation errors of the candidates and the stiffness axis. The results of applying this process to the data of Fig. 4.9(b) can be seen in Fig. 4.11. Fig. 4.11(a) zooms in the results for iterations k={2,3,4,5}. Fig. 4.11(b) presents the system stiffness estimation obtained for the entire interval of 1.8 [s] represented in Fig. 4.9(b).

4.3.4 Theoretical Analysis

A disturbed real plant can be modeled by an undisturbed nominal plant plus an equivalent disturbance p_k referred to the system input that lumps all the disturbances. p_k can be decomposed in two contributions:

$$p_k = p_k(\Delta K_s) + p_k(o), \tag{4.16}$$

where $p_k(\Delta K_s)$ is the disturbance due to the stiffness mismatch ΔK_s (i.e., due to the mismatch between the real system stiffness K_s and the nominal stiffness value K_{sn} used in the control design) and $p_k(o)$ includes all the additional disturbances other than $p_k(\Delta K_s)$ (e.g., due to high-order unmodeled terms, frictions and measurement noise), both referred to the system input. Fig. 3.6 represents the real plant corresponding to the linearized plant G(s) (modeled as a nominal plant with $p_k(\Delta K_s)$) and additional disturbances $p_k(o)$. The FOB command input $u_{c,k}$ is obtained from the command input of the real plant, that is,

$$u_{c,k} = u_k. \tag{4.17}$$

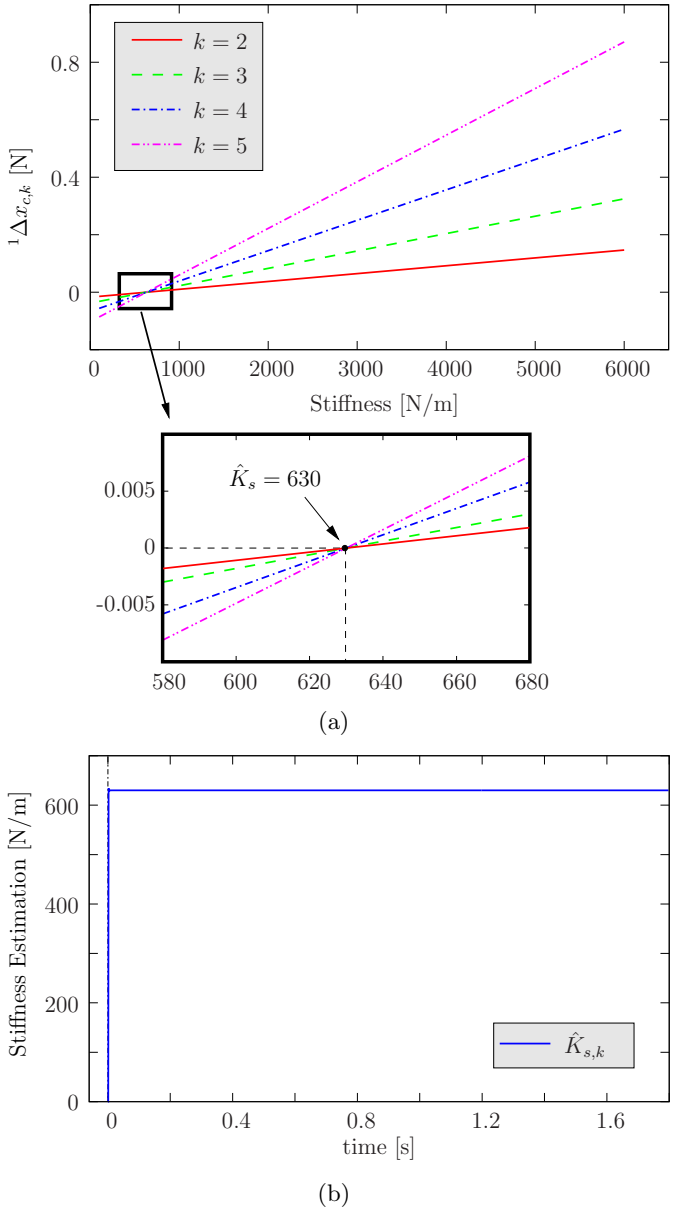


Figure 4.11: COBA stiffness estimations. (a) $\hat{K}_{s,k}$ evolution. (b) Each line contains the points ($K_{sc_1} = 100$ [N/m], $^1\Delta x_{c_1,k}$) and ($K_{sc_2} = 6000$ [N/m], $^1\Delta x_{c_2,k}$) at iterations ranging from $k = 2$ until $k = 5$. $\hat{K}_{s,k}$ corresponds to the point of intersection of the force error line with the stiffness axis.

From Fig. 3.6, it can be seen that

$$\tilde{u}_k = u_k + p_k(o). \tag{4.18}$$

From (4.17) and (4.18),

$$u_{c,k} = \tilde{u}_k - p_k(o). \tag{4.19}$$

A scenario is designated as having *ideal conditions* if $p_k(o) = 0$, i.e., if no additional disturbances exist besides the one due to stiffness mismatch. In this case, by (4.19), $u_{c,k} = \tilde{u}_k$. *Non-ideal conditions* reflect the existence of additional disturbances, i.e., $p_k(o) \neq 0$. In the sequel,the theoretical analysis of COBA under ideal and non-ideal conditions is presented.The proofs of relevant theorems, lemmas and corollaries can be found in the Appendix.

Ideal Conditions

In ideal conditions ($p_k(o) = 0$) the FOB command input $u_{c,k}$ is equal to the command input of G(s). Fig. 4.12 shows COBA inputs under ideal operating conditions where

$$u_{c,k} = \tilde{u}_k = u_k. \tag{4.20}$$

Lemma 2, Corolary 1 and Theorem 2 are the relevant theoretical results for analysing COBA operation in ideal conditions (see Theoretical Results 2). According to Lemma 2, the force estimation error of each FOB is given by

$${}^1\Delta x_{c,k} = -\Delta K_{sc}\ {}^1G_k = -(K_s - K_{sc})\ {}^1G_k, \tag{4.21}$$

where ${}^1G_k$ is independent of ΔK_{sc}. Corollary 1 demonstrates that the unknown term ${}^1G_k$ is equal to the well known ${}^1\hat{G}_k$, computed by (4.12). Therefore solving (4.21) for K_s and replacing data from one FOB (FOB_2 in this case) results in

$$K_s = -\frac{{}^1\Delta x_{c_2,k}}{{}^1\hat{G}_k} + K_{sc_2}. \tag{4.22}$$

The right hand sides of (4.22) and (4.14) are equivalent, showing that $\hat{K}_{s,k} = K_s$, as specified by Theorem 2.

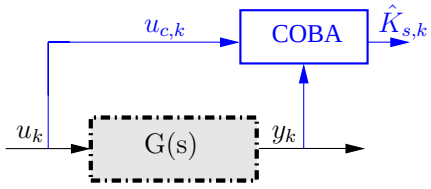


Figure 4.12: COBA inputs under ideal operating conditions.

Lemma 2. *Under ideal operating conditions, there exists a value G_k, independent of ΔK_{sc} and equal for both FOBs, that verifies:*

$$\Delta x_{c,k} = -\Delta K_{sc}\, G_k,$$

where $\Delta x_{c,k}$ is the FOB state estimation error and ΔK_{sc} is the FOB stiffness mismatch.

Corollary 1. *Under ideal operating conditions,*

$$^1\hat{G}_k = {}^1G_k.$$

Theorem 2. *Under ideal operating conditions, COBA estimates the correct object stiffness,*

$$\hat{K}_{s,k} = K_s.$$

Theoretical Results 2: Results relevant to analysis of COBA operation: Ideal conditions. Proofs are in the Appendix (Section A.3.1).

Non-ideal Conditions

In this section, the analysis of COBA is extended to contemplate additional disturbances besides the one due to the stiffness mismatch, which are lumped into $p_k(o)$. This analysis is based on Lemma 3, Corollary 2 and Theorem 3, shown in Theoretical Results 3. If $p_k(o) \neq 0$, $u_{c,k}$ is different from the G(s) command input $\tilde{u}_k$. This difference, represented by $\Delta u_{c,k}$, can be defined, according to (4.19), by

$$\Delta u_{c,k} \equiv \tilde{u}_k - u_{c,k} = p_k(o). \tag{4.23}$$

If $\Delta u_{c,k} = 0$, then $\tilde{u}_k = u_{c,k}$ and the analysis is similar to that of the ideal scenario. If $\Delta u_{c,k} \neq 0$, the relation between $^1\Delta x_{c,k}$ and ΔK_{sc}, represented in (4.21), is affected by an offset ^1W_k, as demonstrated in Lemma 3,

$$^1\Delta x_{c,k} = -(K_s - K_{sc})\ {}^1G_k + {}^1\ W_k. \tag{4.24}$$

Theorem 3 shows that under non-ideal conditions, the additional disturbances negatively impact the estimation performance. Corollary 2 demon-

Lemma 3. *Under non-ideal operating conditions, there exists a value G_k and W_k, independent of ΔK_{sc} and equal for both FOBs, that verifies:*

$$\Delta x_{c,k} = -\Delta K_{sc} G_k - W_k,$$

where $\Delta x_{c,k}$ is the FOB state estimation error and ΔK_{sc} is the FOB stiffness mismatch.

Corollary 2. *Under non-ideal operating conditions,*

$$^1\hat{G}_k = {}^1G_k.$$

Theorem 3. *Under non-ideal operating conditions, the estimation computed by COBA is given by*

$$\hat{K}_{s,k} = K_s + \Delta\hat{K}_{s,k},$$

with

$$\Delta\hat{K}_{s,k} = -\frac{^1W_k}{^1G_k},$$

if $^1G_k \neq 0$. $\Delta\hat{K}_{s,k}$ is the stiffness estimation error.

Theoretical Results 3: Results relevant for analysis of COBA operation: Non-ideal conditions. Proofs are in the Appendix (Section A.3.2).

strates that, even when disturbances are present, $^1\hat{G}_k = {}^1G_k$, so by replacing (4.24) and $^1\hat{G}_k$ in (4.13), the stiffness estimation computed by (4.14) will suffer an error $\Delta\hat{K}_{s,k}$ given by

$$\Delta\hat{K}_{s,k} = -\frac{^1W_k}{^1G_k}. \tag{4.25}$$

^1W_k cannot be computed since it depends of the unknown uncompensated disturbance $\Delta u_{c,k}$ (A.69). To reduce the magnitude of the error, $\Delta u_{c,k}$ should be as small as possible. This can be achieved by compensating the FOBs command input with an estimation $\hat{p}_k(o)$, as represented in Fig. 4.13.

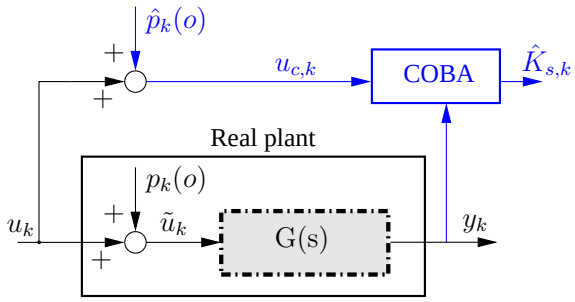


Figure 4.13: COBA inputs under non-ideal operating conditions.

In this case, instead of (4.17), $u_{c,k}$ is computed by

$$u_{c,k} = \hat{p}_k(o) + u_k. \tag{4.26}$$

Consequently,

$$\Delta u_{c,k} = p_k(o) - \hat{p}_k(o). \tag{4.27}$$

4.3.5 Parameters

The relevant parameters for COBA operation are ${}^1K_c$, ${}^2K_c$ and $G_{\min}$. Proper tuning of these parameters can improve the estimation error $\Delta\hat{K}_{s,k}$. Equation (4.25) depends on the unknown value of ${}^1W_k$ and is poorly conditioned for low magnitudes of ${}^1G_k$. ${}^1W_k$ is bounded if the parameters ${}^1K_c$ and ${}^2K_c$ are tuned according to the specific conditions described in Theoretical Results 4. Therefore, the absolute value of ${}^1G_k$ can be used as an indicator of the magnitude of $\Delta\hat{K}_{s,k}$. An online quality assessment based on $|{}^1G_k|$ can be implemented, allowing continuous evaluation of the estimation quality. Low magnitudes of ${}^1G_k$ can potentially correspond to poorly conditioned estimations, therefore they should be discarded. ${}^1G_k$ is not dependent of the y_k magnitude (see (4.12)), hence high magnitudes of ${}^1G_k$ can be found even for low magnitudes of y_k. This means that high quality estimations can be obtained for low contact forces.

4.3.6 Discussion

COBA is an algorithm that preserves the same desirable properties of ASBA and avoids its problematic issues described in Section 4.2.6. The operation principle of COBA is based on theoretically demonstrated properties of the candidate observers. In accordance to the discussion in the introductory

Theorem 4. *W_k is bounded if $\Delta u_{c,k-1}$ is bounded and the following conditions hold:*

$$|\lambda_1| < 1 \text{ and } |\lambda_2| < 1, \tag{4.28}$$

where

$$\lambda_1 = \alpha/2 - \sqrt{\alpha^2/4 + \beta} \tag{4.29}$$

and

$$\lambda_2 = \alpha/2 + \sqrt{\alpha^2/4 + \beta}, \tag{4.30}$$

with

$$\alpha = {}^1K_c, \tag{4.31}$$

$$\beta = -2\,{}^1K_c{}^2K_c + {}^1K_c\,c + 2\,{}^2K_c a - c, \tag{4.32}$$

$$a = (1 - e^{-K_D T_s})/K_D \tag{4.33}$$

and

$$c = e^{-K_D T_s}. \tag{4.34}$$

Theoretical Results 4: Results relevant for analysis of COBA operation: FOB gain tuning. Proof is in the Appendix (Section A.3.2).

chapter (see Chapter 1), it avoids the practical difficulties of relying on contact point and deflection information by not requiring object position data. One of the desirable properties of an estimation algorithm is runtime efficiency. COBA is very efficient, as the execution time and memory overhead is limited to the computation of two candidate observers. Memory overhead is limited to the state of the FOBs, while execution time overhead is limited to the operations required to update the FOB states and compute the line intersecting the estimation errors (see Section 4.3.2). Contrary to ASBA, the implementation of COBA is simple and does not require any simplification of the relevant formulas. This has the benefit of ensuring results are consistent with the theoretical analysis, as no compromise is being made for practical considerations. By avoiding the need to conduct LSE analysis, COBA is well suited to obtaining estimations in dynamic environments, as the candidate observers adjust their state estimations to changing environmental conditions. The effects of additional disturbances has also been analyzed, allowing for the design of compensation strategies described in Section 4.3.4.

Chapter 5

Sensor-Based Estimation

Contents

5.1 Introduction

This chapter explores the stiffness estimation problem using sensor-based approaches. These are based on techniques known for their ability to identify complex and nonlinear relations between inputs and outputs of a system, without recourse to an explicit model of the system under consideration. In this work one of the most important heuristics commonly used for solving identification problems - Artificial Neural Networks - was adopted. ANNs were first discussed in the early forties and have since been the object of much research and development. They have been successfully applied in several areas, such as prediction, system identification, classification and pattern recognition. The main objective of the research presented in this chapter is to explore the application of ANNs to the stiffness estimation problem, without recourse to object position data, in robotic interaction tasks. One advantage of this approach over model-based techniques is that explicit models are not required.

5.2 Artificial Neural Networks

Biological neural networks are composed of interconnected neurons. Neurons generate electrochemical stimulus that induce activity in other connected neural cells. An artificial neural network mimics its biological counterpart. It is a set of interconnected artificial neurons, whose basic model is founded upon the functionality of the biological neuron. Synapses, the interconnections between neurons, are emulated by weighted interconnections. A negative weight reflects an inhibitory connection, while positive values designate excitatory connections. All weighted inputs to an artificial neuron $\mathtt{k}$ are combined into an aggregate activity level $\beta_{\mathtt{k}}$. This is typically achieved by (5.1),

$$\beta_{\mathtt{k}} = \sum_{j=1}^{n}(w_{\mathtt{k},j}a_j) - \theta_{\mathtt{k}} = \mathcal{W}_{\mathtt{k}}^T\mathcal{A}, \tag{5.1}$$

where n is the number of inputs, $w_{\mathtt{k},j}$ is the weight of the connection feeding the output of neuron j into the input of neuron $\mathtt{k}$, a_j is the output of neuron j, $\theta_{\mathtt{k}}$ is the offset of neuron $\mathtt{k}$, $\mathcal{W}_{\mathtt{k}} = [w_{\mathtt{k},1}\ w_{\mathtt{k},2}\ \cdots\ w_{\mathtt{k},n} - 1]$ and $\mathcal{A} = [a_1\ a_2\ \cdots\ a_n\ \theta_{\mathtt{k}}]$. An activation function calculates the amplitude of the output based on the activity level, typically restricting the output to a bounded range. Popular choices include the *threshold* (5.2) and *sigmoid* (5.3) functions.

$$a_{\mathtt{k}}(\beta_{\mathtt{k}}) = \begin{cases} 0 & \text{if } \beta_{\mathtt{k}} \geq 0 \\ 1 & \text{if } \beta_{\mathtt{k}} < 0 \end{cases}. \tag{5.2}$$

$$a_{\mathtt{k}}(\beta_{\mathtt{k}}) = \frac{1}{1 + e^{-\beta_{\mathtt{k}}}}. \tag{5.3}$$

The latter can be interpreted as a continuous version of the former, allowing continuous outputs to be produced. The pattern of connection between artificial neurons leads to different topologies. A main distinction can be made between *feedforward* (most common) and *recurrent* topologies. The latter contain feedback connections, while the former do not. Classical examples of feedforward and recurrent ANN are the *Perceptron* and *Hopfield* networks, respectively [88, 49]. The behavior of the ANN is dependant on the topology, activation function and weights. Usually, the topology and activation functions are chosen by the designer according to the problem being considered, and remain constant during the entire lifecycle of the ANN. Accordingly, the desired behavior must be obtained, for a specific choice of topology and activation functions, by adjusting the synaptic weights. In most cases, it

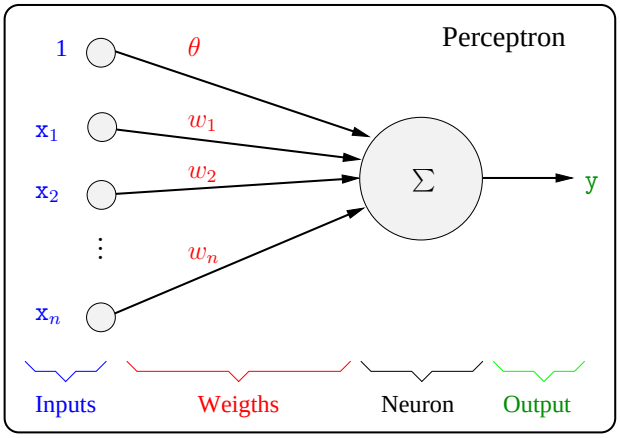


Figure 5.1: Perceptron. The output value $\mathtt{y}$ is computed by weighting the input vector $\mathcal{X} = [1\ \mathtt{x}_1\ \mathtt{x}_2\ ...\ \mathtt{x}_n]^T$ with $\mathcal{W} = [\theta\ w_1\ w_2\ ...\ w_n]$. The output may subsequently be further modified by application of an activation function.

is not feasible to precompute the desired weight configuration using *a priori* domain knowledge. In this way, a training process is conducted, using sample data, where the weights are dynamically adapted until the desired behavior is observed. Training is achieved by applying a learning algorithm that adapts the weights according to provided data samples. A plethora of learning algorithms can be found in the literature. The specific details depend on factors such as the network topology, activation functions, the type of problem addressed and data availability. For example, the network in Fig. 5.1, using a threshold activation function, can be trained to correctly classify an input vector $\mathcal{X}$ into one of two linearly separable classes. If the activation function is smooth and continuous (as is the case if a sigmoid activation function is used), then backpropagation techniques can be used. These "propagate" the output error from the network output back towards the inputs, updating the weights along the way to minimize the squared classification error. The weights can be adjusted proportionally to the amount of error, the local gradient of the error and the input activation. Specific details may differ, but all variants of this process are essentially a gradient-based search for the weight vector that minimizes the squared error of prediction. Convergence is ensured, but only to an local, not global, optimum. Backpropagation algorithms are an example of *supervised* learning algorithms that use training data composed of input-output pairs, indicating the desired network output for the given input vector. Typical applications include curve fitting or classification. Conversely, unsupervised learning al-

gorithms exist that require only input vector sample data for training (but no desired output information). These are mainly used for purposes such as clustering or pattern recognition. Another approach is *reinforcement* learning, where a feedback response to the ANN output is taken into account by the learning rule [103]. Although no strict generic rules for selecting the size and parameters of an ANN exist, some guidelines can be found. The dimension of the sample set required to properly train the network increases with the dimension of the network. Increasing the dimension and number of hidden layers in feedforward networks may improve fitting to training data, but actually decrease generalization capabilities, a phenomenon called overfitting. Conversely, generalization capabilities may be improved by reducing the dimension of hidden layers, but underfitting (poor training performance) may then become an issue. Both designer experience and experimental trials play an important role in finding the best ANN parameterization for a given problem.

5.3 ANNE: Artificial Neural Network Estimator

5.3.1 Overview

The Artificial Neural Network Estimator (ANNE) is composed by three modules: a delayed input buffer, a layer of ANNs and a fusion module. No input signals specific to the control approach are used, promoting portability to other platforms and control strategies. Each ANN computes a stiffness estimation and is trained for a specific nominal stiffness used in control design. The fusion module merges ANNs outputs into a single improved estimation. Section 5.3.2 discusses the most relevant aspects concerning ANNE architecture. Section 5.3.3 presents ANN and ANNE training and testing procedures.

5.3.2 Architecture

Fig. 5.2 represents the ANNE estimator embedded in the control loop and the ANNE architecture is shown in Fig. 5.3. A buffer feeds the delayed input signals y_k and u_k to multiple ANNs. Each ANN is trained with force-based data from interactions with a fixed nominal stiffness K_{sn} equal to $K_T^{(j)}$, outputting an independent estimation of K_s, $\hat{K}_{s,k}^{(j)}$. The fusion module fuses the information of the ANNs, providing the $\hat{K}_{s,k}$ output.

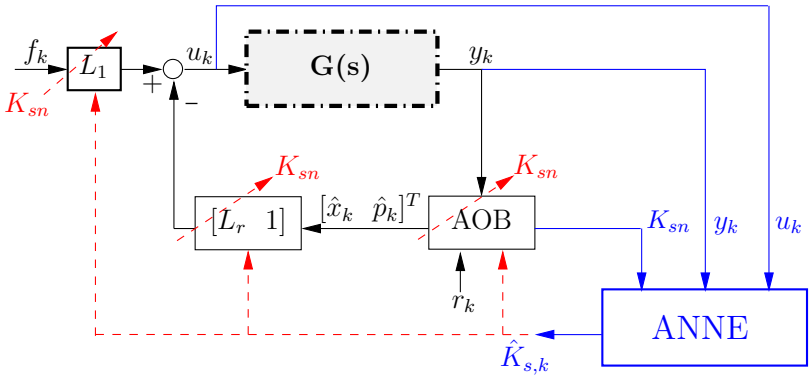


Figure 5.2: ANNE embedded in the control loop. ANNE performs the stiffness estimation which can be used to adapt control laws online.

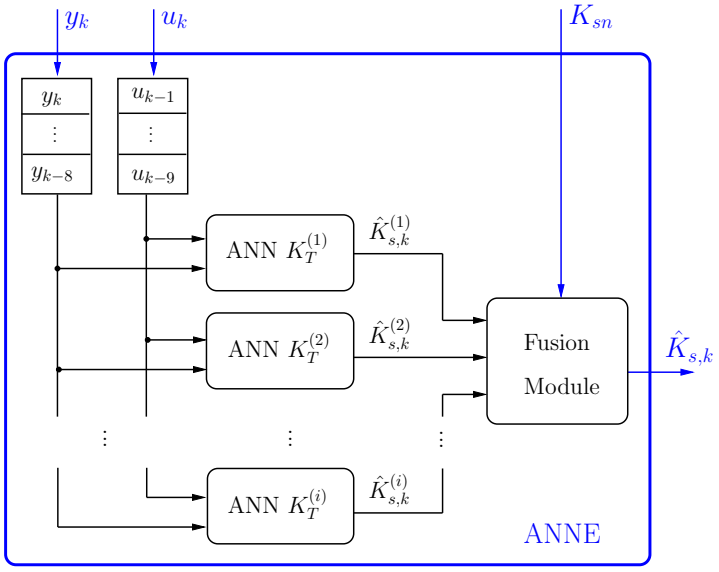


Figure 5.3: ANNE architecture. It receives the measured force y_k, the command input u_k and current K_{sn}, outputting the stiffness estimation $\hat{K}_{s,k}$. A buffer is constructed to feed multiple ANNs with force-based data, only. A fusion module computes $\hat{K}_{s,k}$ from ANN outputs.

Input Vector Definition

An analysis was conducted to identify suitable force-based inputs for the ANNs.

- ***Candidate Variables*** - The following variables were considered potential candidates for inclusion in the input vector:
  - f_k (reference force)
  - y_k (measured force)
  - u_k (command input)
  - ${}^0\hat{x}_k$ (AOB first state)
  - $\hat{p}_k$ (AOB active state)

  since they include information that is measurable and relevant for stiffness estimation.

- ***Variable Selection*** - Although candidate variables can all be included in the ANN design, too much redundant information can jeopardize the training process. With this in mind, correlation analysis was conducted to reduce the number of ANNE inputs. High correlation between two variables can be interpreted as a sign of redundancy, allowing to remove one variable from the input vector. Several robotic telemanipulation tasks have been performed, for multiple contact scenarios and Table 5.1 shows the results.

Table 5.1: Average correlation results of force-based variables.

	y_k	f_k	u_k	${}^0\hat{x}_k$	$\hat{p}_k$
y_k	1	0.9837	-0.7695	0.9992	0.7640
f_k	0.9837	1	-0.7004	0.9897	0.7662
u_k	-0.7695	-0.7004	1	-0.7558	-0.9124
${}^0\hat{x}_k$	0.9992	0.9897	-0.7558	1	0.7637
$\hat{p}_k$	0.7640	0.76624	-0.9124	0.7637	1

  It can be seen that y_k, ${}^0\hat{x}_k$ and f_k exhibit strong correlation, while u_k and $\hat{p}_k$ are also highly correlated. Therefore, only one variable among y_k, ${}^0\hat{x}_k$ and f_k is required to be present in the input set, as well as one of u_k and $\hat{p}_k$. y_k was chosen since it reacts faster to environment changes providing richer transient information. u_k was also chosen since it is not specific to our control approach, allowing easier adaptation to other control techniques.

- ***Time-lag Selection*** - Having selected the variables to include in the input vector, the dimension of the buffer that feeds y_k and u_k into the

ANNE must be determined. The objective is to reduce input dimension as much as possible, for training and computational efficiency. A set of experiments has been conducted to this effect, using training data from interaction tasks between a WAM robot and objects with different stiffnesses. Good results were obtained by including the most recent nine samples of each variable in the input vector. Since the experimental platform has one sampling time delay (see (3.24)), y_k is dependent on u_{k-1}. Therefore, the desired inputs are $y_k, y_{k-1} \ldots, y_{k-8}$ which combined with $u_{k-1}, u_{k-2} \ldots, u_{k-9}$ form the ANN input vector:

$$\Upsilon = \left[\; y_k, \quad y_{k-1}, \quad \ldots, \quad y_{k-8}, \quad u_{k-1}, \quad u_{k-2}, \quad \ldots, \quad u_{k-9} \; \right]. \tag{5.4}$$

ANN Topology

Fig. 5.4 represents the ANN topology of ANNE. A typical feedforward network with a single hidden layer is used. A balance between fitting and generalization must be sought. Higher numbers of hidden neurons improve fitting to training data, but may result in overfitting, resulting in poorer generalization capabilities.

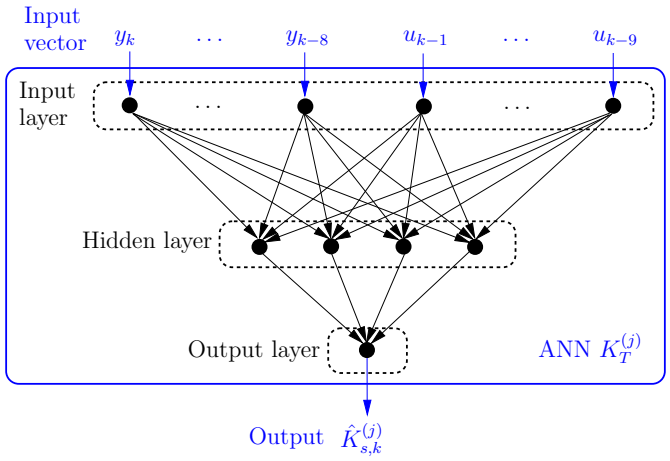


Figure 5.4: ANN $\mathbf{K_T^{(j)}}$ topology. The ANN $K_T^{(j)}$ is trained with telemanipulation data where the nominal stiffness used in the control architecture is equal to $K_T^{(j)}$. It has an input vector with force-based data, with dimension 18, and one hidden layer. The output corresponds to the stiffness estimation $\hat{K}_{s,k}^{(j)}$.

Fusion Module

If during online operation $K_{sn} \neq K_T$, where

$$K_T \in \{K_T^{(1)}, \ldots, K_T^{(i)}\}, \tag{5.5}$$

the ***training stiffness mismatch*** $\Delta K_T = K_T - K_{sn}$ is non null and ANN K_T response degradation is expected. It has been consistently observed that slightly negative ΔK_T and slightly positive ΔK_T result in estimation errors with opposite signs. Therefore, if K_L and K_H are the nearest training stiffnesses to K_{sn}, so that $K_L \leq K_{sn} \leq K_H$, the correct estimation lies in the range provided by ANN K_L and ANN K_H ($[\hat{K}_{s,k}^{(L)}, \hat{K}_{s,k}^{(H)}]$). Its value is obtained by fusing the results of multiple ANNs with polynomial regression [30]. If $K_{s,k}$ is represented by a general m-order polynominal regressor

$$K_{s,k} = a_{0,k} + a_{1,k}K_{sn} + a_{2,k}K_{sn}^2 + a_{3,k}K_{sn}^3 + \cdots + a_{m,k}K_{sn}^m + \varepsilon_k, \tag{5.6}$$

where ε_k is the regression error, then the $\hat{K}_{s,k}$ can be obtained by

$$\hat{K}_{s,k} = \hat{a}_{0,k} + \hat{a}_{1,k}K_{sn} + \hat{a}_{2,k}K_{sn}^2 + \hat{a}_{3,k}K_{sn}^3 + \cdots + \hat{a}_{m,k}K_{sn}^m. \tag{5.7}$$

The $a_{i,k}$ coefficients can be estimated in each iteration by using a regression set including n points $(K_T^{(j)}, \hat{K}_{s,k}^{(j)})$, with $n > m$. Replacing each point of the regression set into (5.6) and conjoining resulting expressions, the following system is obtained,

$$\mathbb{K}_k = \mathbb{X}\mathbb{A}_k + E_k, \tag{5.8}$$

where

$$\mathbb{K}_k = \begin{bmatrix} \hat{K}_{s,k}^{(0)} \\ \hat{K}_{s,k}^{(1)} \\ \vdots \\ \hat{K}_{s,k}^{(n)} \end{bmatrix}, \quad \mathbb{A}_k = \begin{bmatrix} a_{0,k} \\ a_{1,k} \\ \vdots \\ a_{m,k} \end{bmatrix}, \quad E_k = \begin{bmatrix} \epsilon_{0,k} \\ \epsilon_{1,k} \\ \vdots \\ \epsilon_{n,k} \end{bmatrix}, \tag{5.9}$$

and $\mathbb{X}$ is given by

$$\mathbb{X} = \begin{bmatrix} 1 & K_T^{(0)} & {K_T^{(0)}}^2 & \cdots & {K_T^{(0)}}^m \\ 1 & K_T^{(1)} & {K_T^{(1)}}^2 & \cdots & {K_T^{(1)}}^m \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & K_T^{(n)} & {K_T^{(n)}}^2 & \cdots & {K_T^{(n)}}^m \end{bmatrix}. \tag{5.10}$$

Using ordinary least squares estimation, $\hat{\mathbb{A}}_k$ is obtained by

$$\hat{\mathbb{A}}_k = (\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T\mathbb{K}_k. \tag{5.11}$$

Some guidelines relate the choice of regressor order, the composition of the regression set and the distribution of $K_T^{(j)}$:

a) Concerning selection of the points $(K_T^{(j)}, \hat{K}_{s,k}^{(j)})$ to integrate the regression set, preference should be given to those closer to K_{sn}, so that relevant information in the neighborhood of the point of interest is not discarded. This makes $(K_L, \hat{K}_{s,k}^{(L)})$ and $(K_H, \hat{K}_{s,k}^{(H)})$ major candidates for inclusion.

b) High order polynomials should generally be avoided if the regression set is composed of equidistant points. In this case, the degree of the polynomial regressor m should be bounded by $m < 2\sqrt{n}$. Respecting this upper limit ensures that the approximation is well conditioned [30].

c) If a high order regressor is used, the regression set should be more densely populated nearer the edges of the regression interval, to reduce oscillations due to Runge's phenomenon [7].

5.3.3 Training and Testing

ANN Training

Each ANN K_T is trained by performing the following steps:

1. ***Training Setup***: The control system is designed by setting $K_{sn} = K_T$. A set of objects with different stiffnesses is then chosen.

2. ***Data Collection***: For each object, the human operator performs several telemanipulation experiments involving free-space/contact transitions. Each data sample is obtained by:

   - Choosing a random time instant k.
   - Obtaining the input vector from (5.4) and setting the target estimation, required by the supervised learning process, to $K_{s,k}$. This value is known for virtual objects. For real objects, an external source is used to provide target stiffnesses.

3. ***Training***: Samples are divided into a *training set* and a *validation set*. The training set provides the samples necessary for the Levenberg-Marquardt backpropagation algorithm, while the validation set assesses the training phase. Training runs for a maximum number of epochs bounded by the validation set.

ANN and ANNE Testing

Testing is conducted by following these steps:

1. ***Test Setup***: The nominal stiffness K_{sn} is set for a given telemanipulation session. Testing of ANN K_T includes both scenarios where $K_{sn} = K_T$ and $K_{sn} \neq K_T$. ANNE testing entails setting K_{sn} to an arbitrary value, different from the K_T, while interacting with different objects.

2. ***Data Collection***: For each object, a telemanipulated experiment is performed. These experiments are used solely for testing purposes (i.e., testing data is never used for training). Although similar setups may be used for collecting both training and testing data, human telemanipulation ensures, in practice, that reproduction of the reference and control signals across both data sets is unlikely. This allows generalization capabilities of the ANN to be assessed, since during testing the network is being stimulated with signals that were not used in training.

3. ***Result Analysis***: Comparison of $\hat{K}_{s,k}$ with the expected outcome.

5.3.4 Discussion

Like the other techniques presented in this thesis, ANNE is an estimation technique that does not rely on object position data to compute estimation of the environment stiffness. Unlike those techniques, however, the ANNE estimator does not rely on a specific model of environment or control system, but on a implicit model that is tailored using a training process. The input data required is less specific to the control architecture described in Chapter 3 than COBA or ASBA, increasing potential for application in other control settings. One important point to observe is that ANNE does require an initial data set that is seeded with initial stiffness estimations. As an example of application, those initial estimations can be obtained in a structured setting, where geometry uncertainty is minimal and a conventional estimation algorithm produces good results. Then, after training the corresponding ANNE estimator, position dependency is eliminated and the same control architecture can be used for tasks in less structured environments (where the original estimator fails to produced adequate estimations), while maintaining stiffness estimation capabilities with ANNE. Alternatively, as illustrated in Chapter 7, ANNE can be seeded with the results of other object position-independent estimator, such as COBA.

Chapter 6

Simulation Results

Contents

6.1 Introduction

This chapter presents and discusses the results of simulation tests. These were conducted with the following objectives:

1. Assessing the impact of parameter mismatch in the force tracking dynamics (Section 6.3).
2. Assessing the behavior of the estimation algorithms (sections 6.4, 6.5 and 6.6), in nearly ideal operating conditions:
   - Perfect linearization of manipulator dynamics, resulting in double integrator behavior.
   - Contact dynamics are perfectly modeled by a linear spring.
   - The manipulator is perfectly rigid.

In all tests, the interaction of a robot with the environment was simulated, based on the explicit force control architecture described in Chapter 3, using the Matlab Simulink tool [105]. The simplified graphical representation of the simulation platform, including the interaction between the robot and the environment, can be seen in Fig. 6.1 (in Section 6.3.2 damping is also considered in the contact model). A more detailed description of the simulation setup is given in Section 6.2. The effect of parameter mismatch in force tracking dynamics is explored in 6.3. Simulation results for ASBA, COBA and ANNE can be found in sections 6.4, 6.5, and 6.6, respectively. A discussion is presented in Section 6.7. All quantities are based on SI units and only the Z Cartesian dimension is considered.

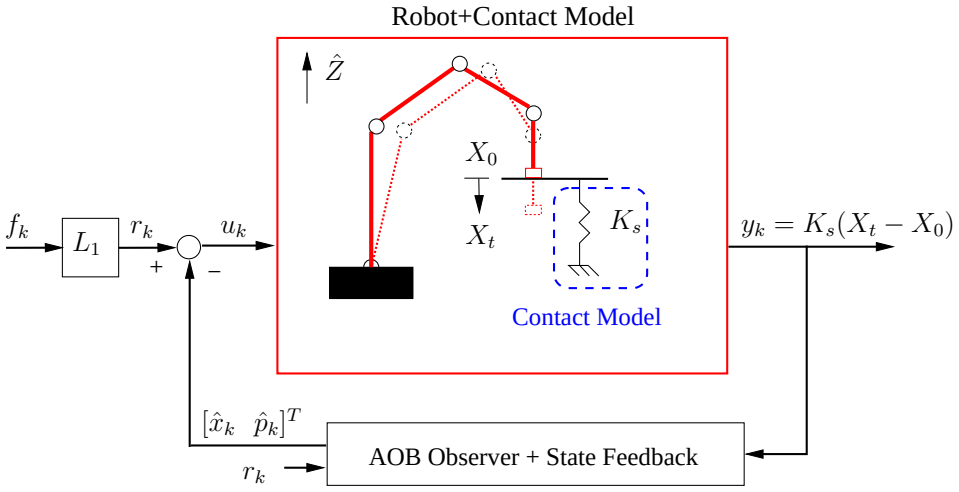


Figure 6.1: Simulation platform. General representation of the simulation platform, including the linearized robotic manipulator, control loop and contact model.

6.2 Simulation Setup

The simulation platform is represented in Fig. 6.2. It includes a perfectly linearized plant, modelled with a double integrator behavior, with system time delay T_d and a spring with stiffness K_s as the contact model. The system plant is controlled with discrete state-space techniques in the task space (state feedback and an AOB observer). A critically damped control design with time constant τ_f is implemented for force tracking. To reflect the force measurement noise, Gaussian noise with approximately zero mean and variance σ^2 was added to the system output y_k. The noise variance was obtained experimentally from the JR3 force sensor. The simulation setup considers that initially the manipulator is in contact with the environment, except in Section 6.5.3 where free-space/contact transitions are simulated. The control design and system parameters common to all tests are presented in Table 6.1.

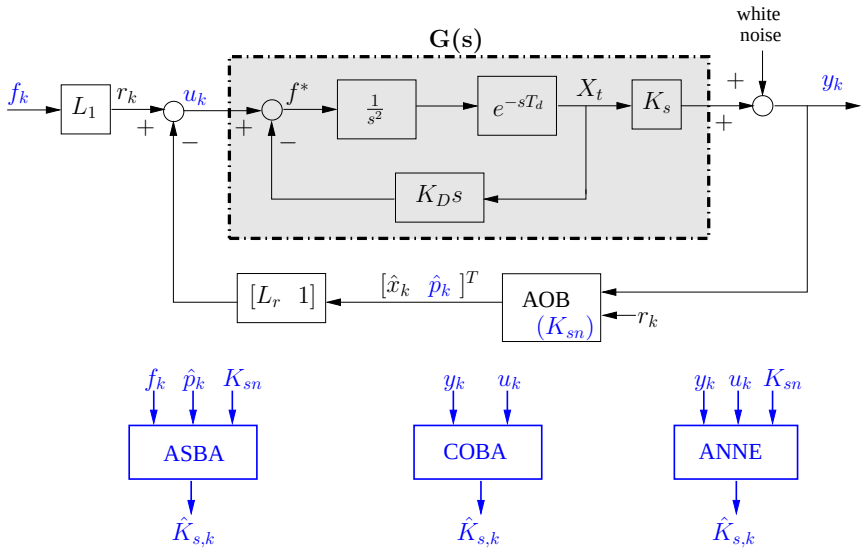


Figure 6.2: Simulation platform used to assess ASBA, COBA and ANNE. G(s) represents the robotic manipulator (with feedback linearization and velocity feedback) and the contact model. f_k is the input force, y_k is the output force, u_k is the command input, L_r is the feedback gain matrix and L_1 is the DC gain compensation term (first element of L_r). $\hat{p}_k$ is the extra state and $\hat{x}_k$ is the system state estimation, both computed by the AOB observer. K_s is the real system stiffness and $\hat{K}_{s,k}$ is the corresponding estimation.

Table 6.1: Simulation results - common control design and system parameters.

Control Parameters

Description	**Parameter**	**Value (ASBA)**	**Value (COBA, ANNE)**
Sampling time	T_s	0.002	0.0012
System time delay	T_d	0.002	0.0012
Damping gain	K_D	15	10
Force loop time constant	τ_f	0.045	0.045
Force loop damping coefficient	ξ	1	1

AOB Parameters

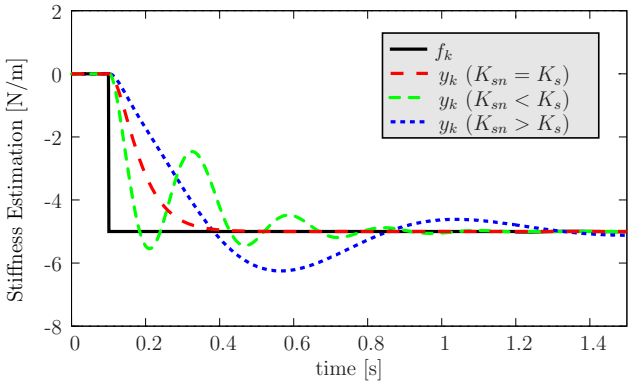
Description	**Parameter**	**Value (ASBA)**	**Value (COBA, ANNE)**
Measur. noise matrix	R_{noise}	1	1
System noise matrix	Q_{noise}	$diag = \{10^{-5}, 10^{-12}, 10^{-12}, 10^{-5}\}$	$diag = \{10^{-6}, 10^{-12}, 10^{-12}, 10^{-5}\}$

6.3 Effect of Parameter Mismatch in Dynamic Response

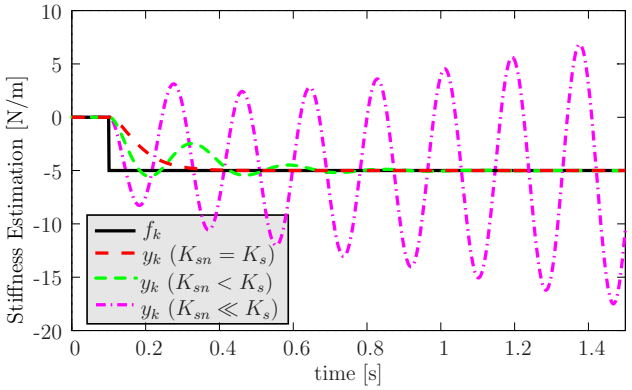
This section illustrates the effect of mismatches between nominal and real contact parameters on the dynamic response. Analysis are conducted using two different contact models. For assessing the effects of stiffness mismatch, the contact is first modelled by a linear spring with stiffness K_s. Then, a Kelvin-Voigt contact model (2.2) is adopted, including a damping component with coefficient B_s, so that the effect of mismatching viscosity can also be assessed. In each case, the AOB nominal matrices are designed accordingly. The control design and AOB parameters are the same of ASBA (Table 6.1).

6.3.1 Stiffness Mismatch Effect

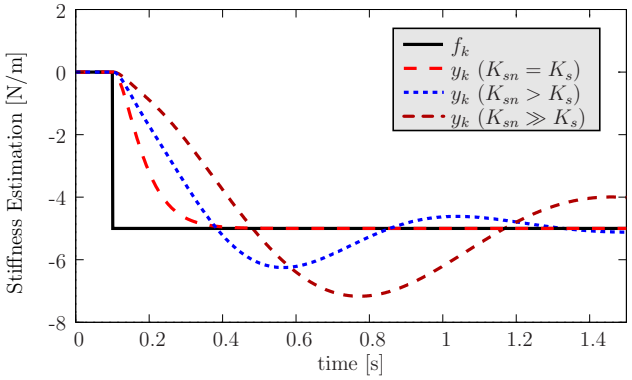
Fig. 6.3 illustrates the effect of different stiffness mismatch scenarios in force tracking dynamics. A step input force of -5 [N] is applied at $t = 0.1$ [s]. It is assumed that the robot is initially in contact with the environment. Four different scenarios are analysed. Fig. 6.3(a) presents the input and output forces for a matching ($K_{sn} = K_s$), under ($K_{sn} < K_s$) and over ($K_{sn} > K_s$) stiffness scenarios. In the matching stiffness case, the measured force follows the input force with the designed critically damped response. However, in both under and overestimated cases, force tracking degradation can be



(a) Matching, under ($K_{sn} = 1/2K_s$) and overestimated ($K_{sn} = 2K_s$) nominal stiffness.



(b) Matching, moderate ($K_{sn} = 1/2K_s$) and severe ($K_{sn} = 1/4K_s$) underestimation of nominal stiffness.



(c) Matching, moderate ($K_{sn} = 2K_s$) and severe ($K_{sn} = 4K_s$) overestimation of nominal stiffness.

Figure 6.3: Stiffness mismatch effects in the dynamic response: an illustrative example. In all scenarios, $K_{sn} = 1000$ [N/m] and $K_s \in \{250, 500, 1000, 2000, 4000\}$ [N/m]. (a) Match, under and overestimated stiffness scenarios. (b) Dynamic response degradation for increasing underestimation. (c) Dynamic response degradation for increasing overestimation.

observed. For the underestimated case, the response has faster dynamics and high ripple effects. For the overestimated case, the response has slower dynamics and small bouncing effects. Figs. 6.3(b) and 6.3(c) show the degradation evolution for increasing under and overestimated mismatches, respectively. Instability arises for the severe underestimation scenario ($K_{sn} \ll K_s$). These results illustrate two points:

(i) Increased mismatch results in increased deviation from the expected behavior, decreasing dynamic consistency of the system response.

(ii) The underestimated scenario is the critical situation for stability.

6.3.2 Combined Damping and Stiffness Mismatch Effect

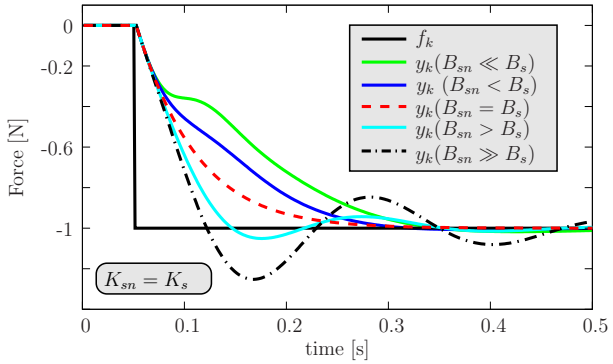
This section discusses how the dynamic response can be affected by damping and stiffness mismatches. Three scenarios of stiffness mismatch are contemplated and, for each one, several damping mismatch scenarios were applied. Input and output forces are presented in Fig. 6.4. The reference is a step of -1 [N] at $t = 0.05$ [s]. From Figs. 6.4(a), 6.4(b) and 6.4(c), it is notorious that $\Delta B_s \neq 0$ degrades the system response when compared with the one with $\Delta B_s = 0$, as expected. From these results, it can be inferred that:

(i) Increased damping mismatch results in increased deviation and degradation, decreasing dynamic performance of the system response.

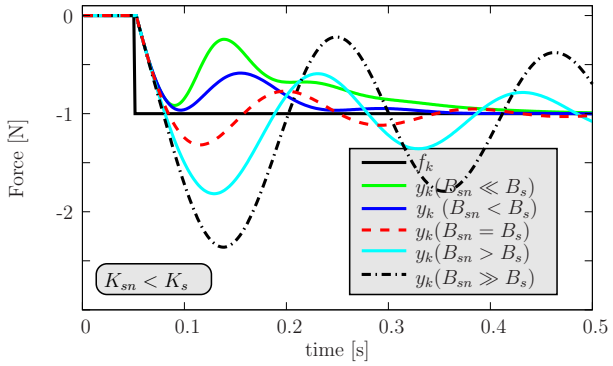
(ii) Underestimated damping decreases the system response dynamics and has less bouncing effects.

(iii) Overestimated damping increases the system response dynamics but causes high ripple effects.

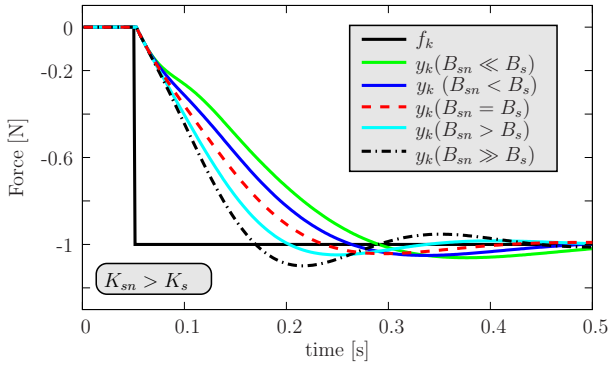
(iv) The underestimated stiffness scenario is the critical situation for stability and that risk is increased when combined with an overestimated damping scenario.



(a) Stiffness match: $K_{sn} = K_s$.



(b) Underestimated stiffness: $K_{sn} = \frac{1}{4}K_s$.



(c) Overestimated stiffness: $K_{sn} = \frac{5}{3}K_s$.

Figure 6.4: Damping and stiffness mismatch effects in the dynamic response: an illustrative example. Each figure represents a specific stiffness mismatch scenario and shows the input and output forces for different values of damping mismatches, with $K_s = 3000$ [N/m], $Bs = 100$ [Ns/m] and $B_{sn} \in \{0, 50, 100, 150, 200\}$ [Ns/m]. (a) Matching stiffness scenario($\Delta K_s = 0$). (b) Underestimated stiffness scenario ($\Delta K_s > 0$). (c) Overestimated stiffness scenario ($\Delta K_s < 0$). $\Delta K_s = K_s - K_{sn}$ and $\Delta B_s = B_s - B_{sn}$.

6.3.3 Discussion

These results demonstrate that mismatches between the nominal and real contact parameters may degrade the designed force tracking dynamics, significantly so for mismatches of higher magnitude. Accurate contact parameter estimation and subsequent adaptation of the control laws can mitigate this effect. In the control architecture described in Chapter 3, no damping component is included in the contact model, so damping will be typically underestimated. While this may cause slower dynamics (see Fig. 6.4(a)), it is not critical with respect to response stability. On the other hand, underestimated stiffness may originate unstable behavior (see Fig. 6.3(b)), justifying the focus on stiffness estimation.

6.4 ASBA

ASBA results are analysed for two different scenarios of stiffness mismatch and for different ASBA parameters. The results with and without K_{sn} adaptation are shown as well. The parameters of ASBA can be found in Table 6.2. The I^{K_{sn},K_s} curves were computed for stiffnesses $\in \mathbf{S}$, i.e., ranging from 100 [N/m] up to 6000 [N/m], with fixed increments of 100 [N/m]. All combinations of K_{sn} and K_s for these values were considered, giving a total of 3600 curves. The I^{K_{sn},K_s} curves were computed for an interval with duration T_I and stored with a sampling time T_s giving 1001 samples.

Table 6.2: ASBA simulation results - design parameters.

AOB Parameters

Description	**Parameter**	**Value**
Nominal stiffness	K_{sn}	{200,3000,5000}

ASBA Parameters

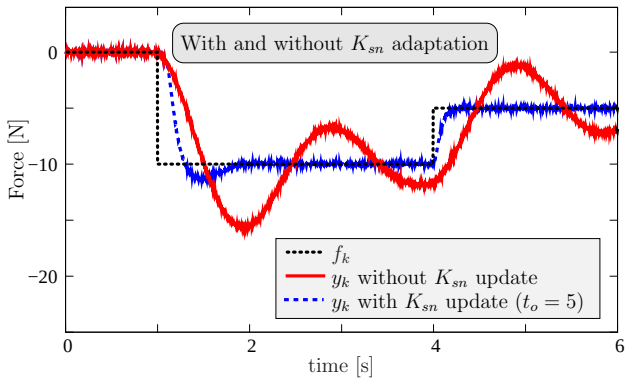
Description	**Parameter**	**Value**
Set of candidate stiffnesses	S	$\{100, 200, 6000\}$
Time window for convolution computation	w	4
Time window for LSE analysis	T_{LSE}	4
Time window for I^{K_{sn},K_s} computation	T_I	2
Times factor of average value of r_k	T_L	2
Time window for detect T_L r_k occurrence	w_2	2
N.º of stable estimations for under scenarios	t_u	$\{2, 5\}$
N.º of stable estimations for over scenarios	t_o	$\{5, 20\}$

6.4.1 Overestimated Stiffness Scenario

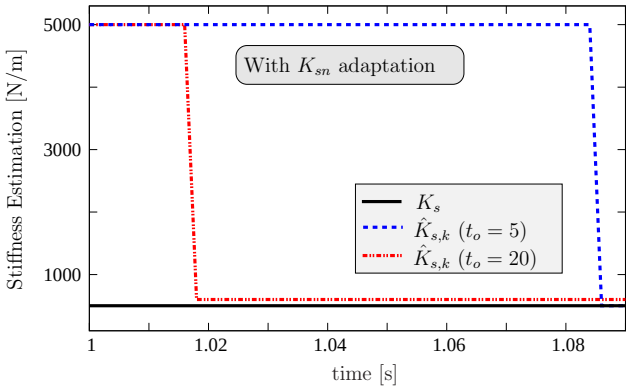
Fig. 6.5 presents the results for a severely overestimated scenario. The measured forces for two opposite situations are represented in Fig. 6.5(a): without and with stiffness adaptation. Without stiffness adaptation, the deterioration of dynamic behavior due to stiffness mismatch is evident. Improved dynamic accuracy is achieved with online stiffness adaptation. After the input force change at $t = 1$ [s], ASBA estimates a new value for K_s and the corresponding control matrices updates are made. The positive effect of this adaptation can be observed, namely at $t = 4$ [s], where no bouncing is present at the measured force, in contrast with the response of the non-adaptive system. Fig. 6.5(b) shows how ASBA behaves for two different values of the ASBA parameter t_o. This parameter is a trade-off between accuracy and estimation speed. Lower values of t_o increase estimation speed at the cost of increased noise sensitivity. For $t_o = 5$, ASBA returns a stiffness estimation in just 7 samples although is has a 100 [N/m] error. For $t_o = 20$ the estimation $\hat{K}_{s,k}$ is computed, matching the real value after 40 samples. For each case, the transient behavior of the measured force, while the estimation and subsequent adaptation is carried out, can be seen in Fig. 6.5(c).

6.4.2 Underestimated Stiffness Scenarios

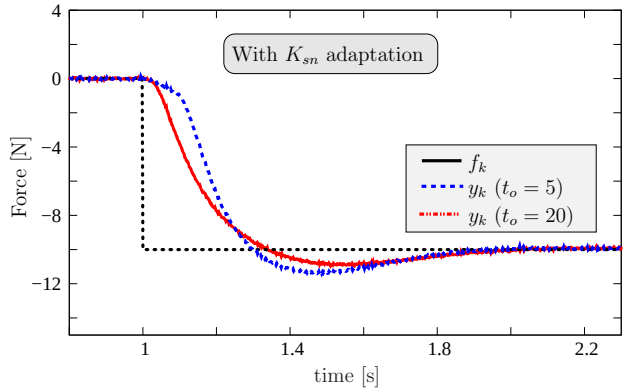
Figs. 6.6 and 6.7 present the results for two underestimated scenarios. Both scenarios differ from each other by the relative stiffness mismatch and by the stiffness values involved. Fig. 6.6 presents a severely underestimated scenario where $K_{sn} \approx 0.15K_s$ ($K_{sn} = 200$ [N/m] and $K_s = 1300$ [N/m]) and Fig. 6.7 presents an underestimated scenario $K_{sn} = 0.5K_s$ for a stiffer environment with $K_{sn} = 3000$ [N/m] and $K_s = 6000$ [N/m]. In Fig. 6.6(a), besides the input force, the measured forces with and without stiffness estimation and adaptation are represented. Without stiffness adaptation, undesirable bouncing on the measured force is present. This bouncing is reduced or disappears when stiffness adaptation is made. Fig. 6.6(b) shows the behavior of ASBA for two values of the parameter t_u. Higher values of t_u degrades estimation speed at the expense of improved robustness. When $t_u = 2$, the estimation is achieved under 4 samples with an error of 100 [N/m]. When $t_u = 5$, the estimation matches the real value in 23 samples. Looking at Fig. 6.6(c), it can be inferred that the fastest estimation, obtained with $t_u = 2$, results in a better overall system response.



(a)



(b)



(c)

Figure 6.5: ASBA - scenario with severely overestimated stiffness, with $\mathbf{K_{sn} = 5000}$ [N/m] and $\mathbf{K_s = 500}$ [N/m]. (a) Input and output forces with and without K_{sn} adaptation. (b) K_s and K_{sn} evolution for two different values of parameter t_o. (c) Zoom of the contact transient at $t = 1$ [s] for scenarios with $t_o = 20$ and $t_o = 5$.

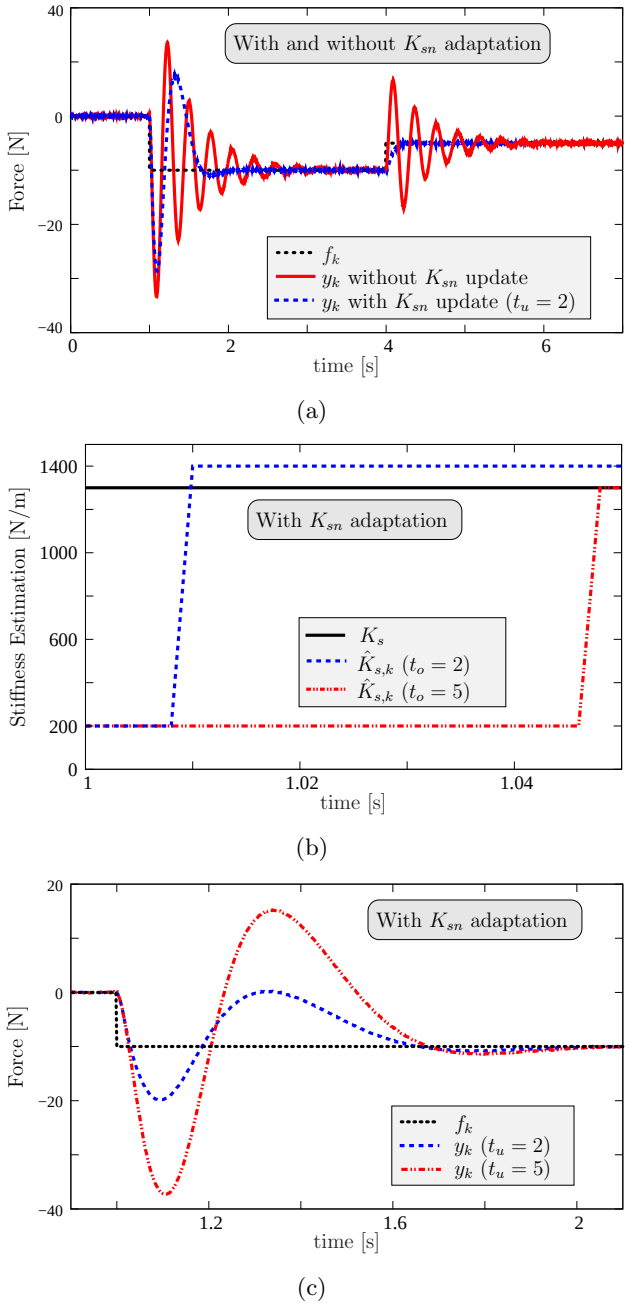


Figure 6.6: ASBA - scenario with severely underestimated stiffness, with $\mathbf{K_{sn} = 200}$ [N/m] and $\mathbf{K_s = 1300}$ [N/m]. (a) System response with and without stiffness estimation. (b) Evolution of the stiffness estimation value for different t_u. (c) Zoom of the contact transient at $t = 1$ [s] for scenarios with $t_u = 2$ and $t_u = 5$.

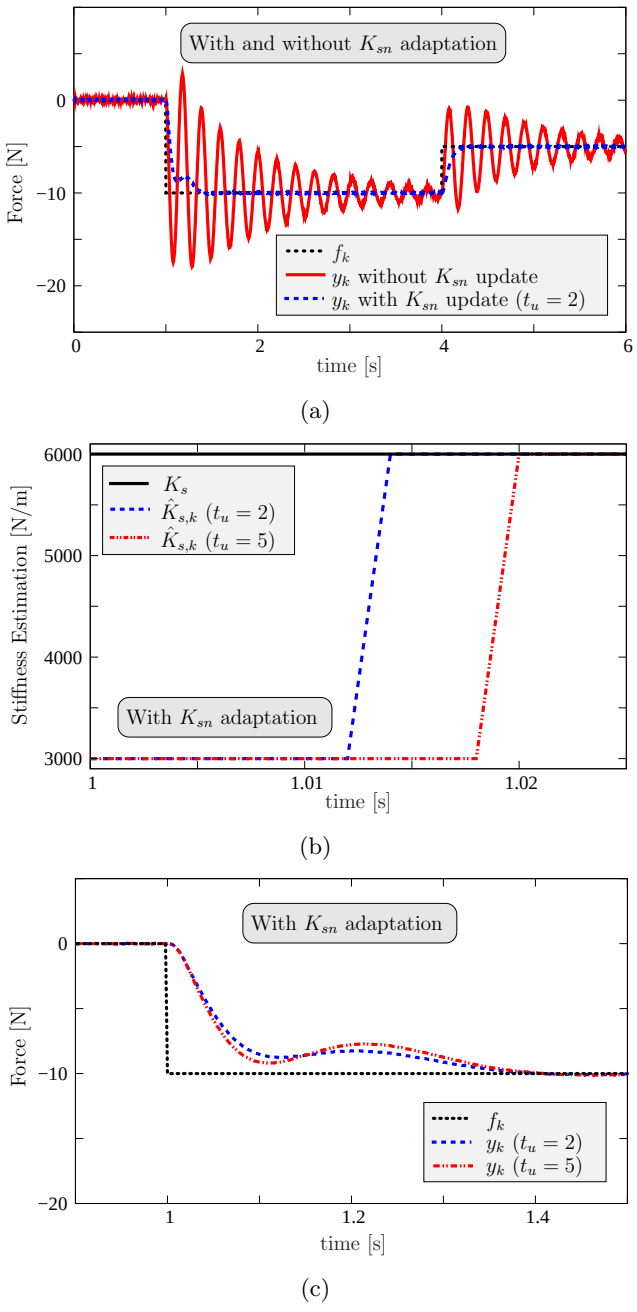


Figure 6.7: ASBA - scenario with severely underestimated stiffness, with $\mathbf{K_{sn} = 3000}$ [N/m] and $\mathbf{K_s = 6000}$ [N/m]. (a) System response with and without stiffness estimation. (b) Evolution of the stiffness estimation value for different t_u. (c) Zoom of the contact transient at $t = 1$ [s] for scenarios with $t_u = 2$ and $t_u = 5$.

Fig. 6.7 shows information similar to the one presented in Fig. 6.6 for a scenario with stiffer system stiffness, where $K_s = 6000$ [N/m] and $K_{sn} = 3000$ [N/m]. Although the stiffness mismatch error is 100%, ASBA behaves properly, achieving an exact estimation under 6 samples when $t_u = 2$ and 9 samples when $t_u = 5$.

6.5 COBA

In this section, COBA simulation results are presented. Different scenarios of stiffness mismatch are explored. Tests were conducted in both homogenous (Section 6.5.1) and heterogenous (Section 6.5.2) environments, that is, including constant *versus* dynamic stiffnesses. Free-space/contact transitions are also considered (Section 6.5.3). Table 6.3 presents the parameters of COBA used in each simulation.

Table 6.3: COBA simulation results - design parameters.

AOB Parameters

Description	**Parameter**	**Value**
Nominal stiffness	K_{sn}	$\{500, 1000, 6000\}$

COBA Parameters

Description	**Parameter**	**Value**
Stiffness of FOB_1	K_{sc_1}	100
Stiffness of FOB_2	K_{sc_2}	6000
FOB gains	K_c	$\{0.001, 0, 0\}$
FOB initial state	$x_{c,0}$	$\{0, 0, 0\}$

6.5.1 Contact with Homogenous Environments

The operation principle of COBA is presented under ideal operating conditions, i.e., in scenarios where there is no other disturbance besides the stiffness mismatch. Diverse stiffness mismatch scenarios are analyzed with constant K_s and without nominal stiffness update. All the tests share the same K_{sn} (1000 [N/m]) and f_k (-5 [N] step at $t = 1$ [s]). It is assumed that the robot is always in contact with an object with stiffness K_s. Three different analysis are presented.

- ***Underestimated Stiffness*** - Fig. 6.8 represents scenarios with moderate ($K_{sn} < K_s$) and a severe ($K_{sn} \ll K_s$) underestimation of stiffness. These are the most critical w.r.t. to instability due to parameter

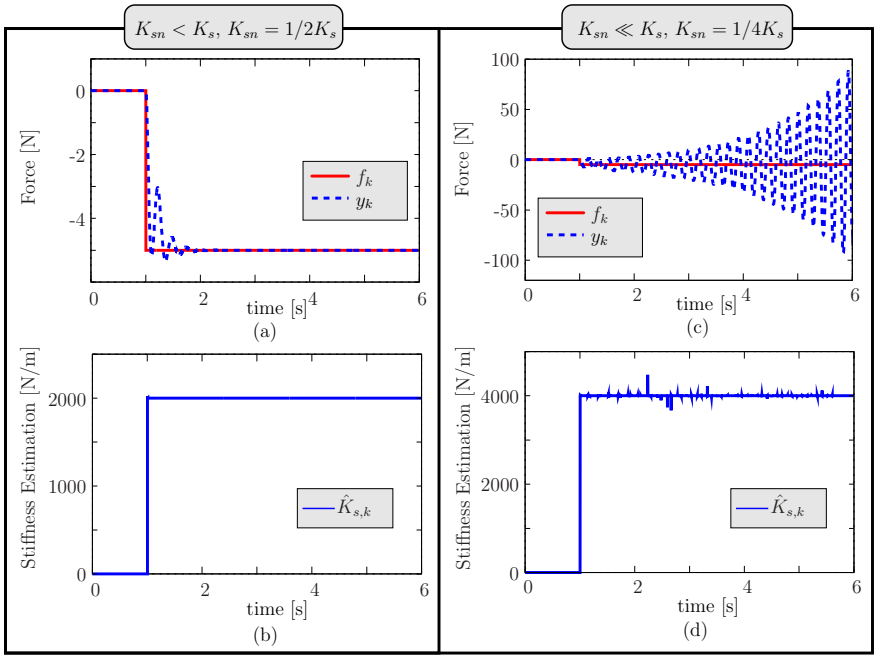


Figure 6.8: COBA - two scenarios with underestimation of stiffness. $K_{sn} = 1000$ [N/m]. (a-b) moderate scenario with $K_{sn} = 1/2K_s$ and (c-d) severe scenario with $K_{sn} = 1/4K_s$. (a)-(c) show the input and measured forces f_k and y_k, respectively. (b)-(d) represent the stiffness estimation evolution. In both scenarios, $\hat{K}_{s,k}$ converges, in just one sampling time, to the expected value, respectively $K_s = 2000$ [N/m] and $K_s = 4000$ [N/m]. No control law adaptation is performed.

mismatches. As expected, force tracking degradation occurs, with pronounced bouncing increasing with the mismatch. COBA converges to the correct value in just one time-iteration, even in the severely underestimated stiffness scenario.

- ***Matching Stiffness*** - Fig. 6.9 shows a scenario with matching stiffness ($K_{sn} = K_s$). The output force tracks the input force, as can be seen in Fig. 6.9(a) and COBA estimations converge to K_s just in one sampling time after a non-null input force is applied, as shown in Fig. 6.9(b).

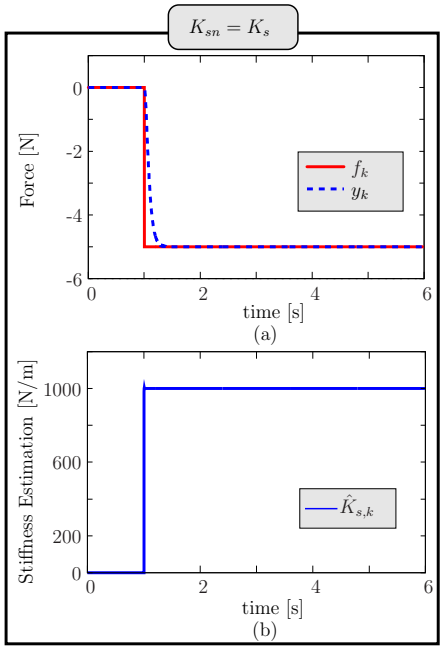


Figure 6.9: COBA - matching stiffness scenario. $K_{sn} = K_s = 1000$ [N/m]. (a) f_k and y_k are, respectively, the input and measured forces. (b) Stiffness estimation evolution. $\hat{K}_{s,k}$ converges, in just one sampling time, to the expected value $K_s = 4000$ [N/m]. No control law adaptation is performed.

- ***Overestimated Stiffness*** - Fig. 6.10 represents two scenarios with moderate ($K_{sn} > K_s$) and severe ($K_{sn} \gg K_s$) overestimations of stiffness, which are characterized with a lower dynamic response (see Fig. 6.10(a) and Fig. 6.10(c)). In either case, COBA converges to the correct value just in one sampling time, as shown in Fig. 6.10(b) and Fig. 6.10(d).

In all tests, COBA converges to the correct value in just one time-iteration, even in the severely underestimated stiffness scenario. This conforms to the theoretical analysis, since under ideal operation conditions ${}^1W_k$ is null (see Theorem 2) and consequently ΔK_s is null. When ${}^1G_k$ is zero, the previous estimation is kept.

6.5.2 Contact with Heterogenous Environments

This section analyses COBA results in an heterogenous stiffness scenario, i.e., where K_s dynamically changes along the experience. Two situations are analyzed: with and without K_{sn} adaptation. In the latter case, estimation is conducted but no adaptation of K_{sn} is made. Fig. 6.11 shows the results. An initial step occurs in the force input at $t = 0.2$ [s], while a second step is applied at $t = 1.8$ [s]. It is assumed that the robot is always in contact with

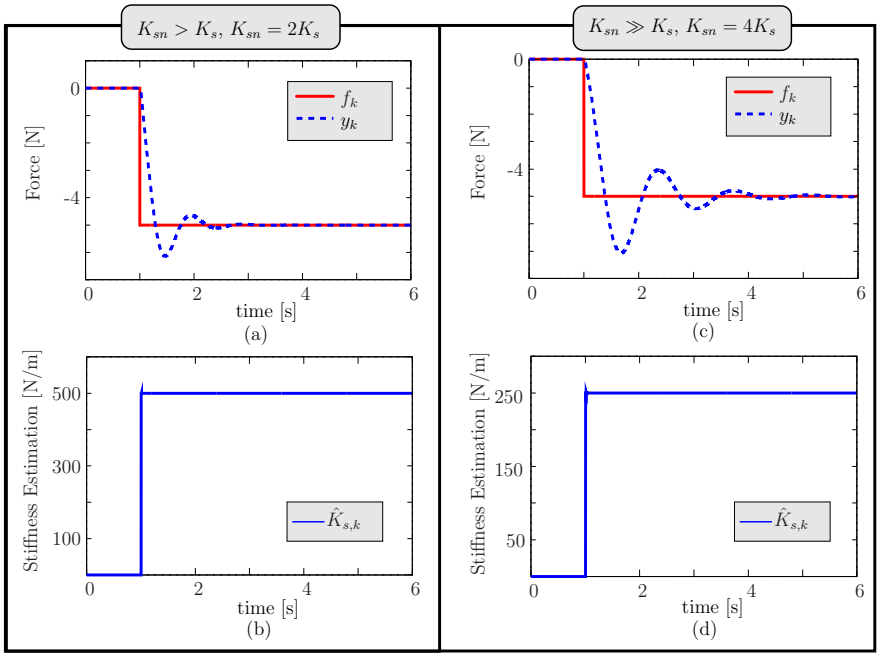


Figure 6.10: COBA - two overestimated stiffness scenarios. $K_{sn} = 1000$ [N/m]. (a-b) moderate scenario with $K_{sn} = 2K_s$ and (c-d) severe scenario with $K_{sn} = 4K_s$. (a)-(c) show the input and measured forces f_k and y_k, respectively. (b)-(d) represent the stiffness estimation evolution. In both scenarios, $\hat{K}_{s,k}$ converges, in just one sampling time, to the expected value, respectively $K_s = 500$ [N/m] and $K_s = 250$ [N/m]. No control law adaptation is performed.

the object with stiffness K_s. Gaussian noise with null mean and variance 1.6×10^{-3} was added to the measured force. Initially, the robot end-effector is at rest position, without any input force stimulus. The initial value of $\hat{K}_{s,k}$ is the K_{sn} value, i.e., 500 [N/m]. K_s has the value 3000 [N/m] (very stiff) between both steps and 300 [N/m] (moderated soft) after the second step. Fig. 6.11(a) shows the input and output forces for both cases (with and without adaptation) and in Fig. 6.11(b) the stiffness estimation evolution is shown. Strong improvements can be observed when stiffness adaptation is used, as the resulting force is closer to the desired critically damped behavior.

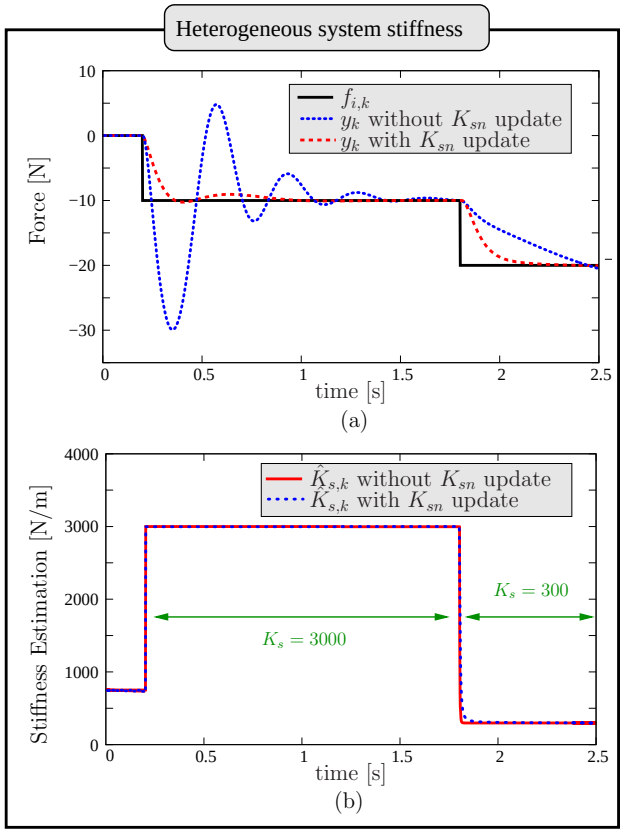


Figure 6.11: COBA - heterogeneous system stiffness. (a) Input and measured forces with and without stiffness adaptation; (b) Stiffness estimation evolution. Updating K_{sn} (blue plot) with the current value $\hat{K}_{s,k}$ returned by COBA, the system response is improved when compared with the non-update case, approximating to the designed critically damped response.

6.5.3 Free-Space/Contact Transitions

In this section, a simulated experience involving free-space/contact transitions was conducted. The free-space is modeled by $K_s = 0$ [N/m] and contact by $K_s = 6000$ [N/m]. White noise with variance 1.6×10^{-3} was added to the output force. Fig. 6.12 shows the control and COBA Algorithm results. The end-effector position X_t, input and output forces (f_k and y_k, respectively) are represented in Fig. 6.12(a). Initially, the robot end-effector is in free-space. A step input of -10 [N] is applied and the robot starts moving down until it enters in contact with the constrained surface. While the robot is moving in free-space, the measured force is approximately zero.

The contact happens in the neighborhood of $t = 0.32$ [s]. From the impact, an undershoot force of about -40 [N] occurs. This behavior is due to the robot kinetic energy and to abrupt variation of stiffness (K_s changes from 0 [N/m] to 6000 [N/m]). COBA results are represented in Fig. 6.12(b). The COBA estimation converges to the correct value, without any stationary error or estimation spikes. The transition between free-space and contact is handled successfully.

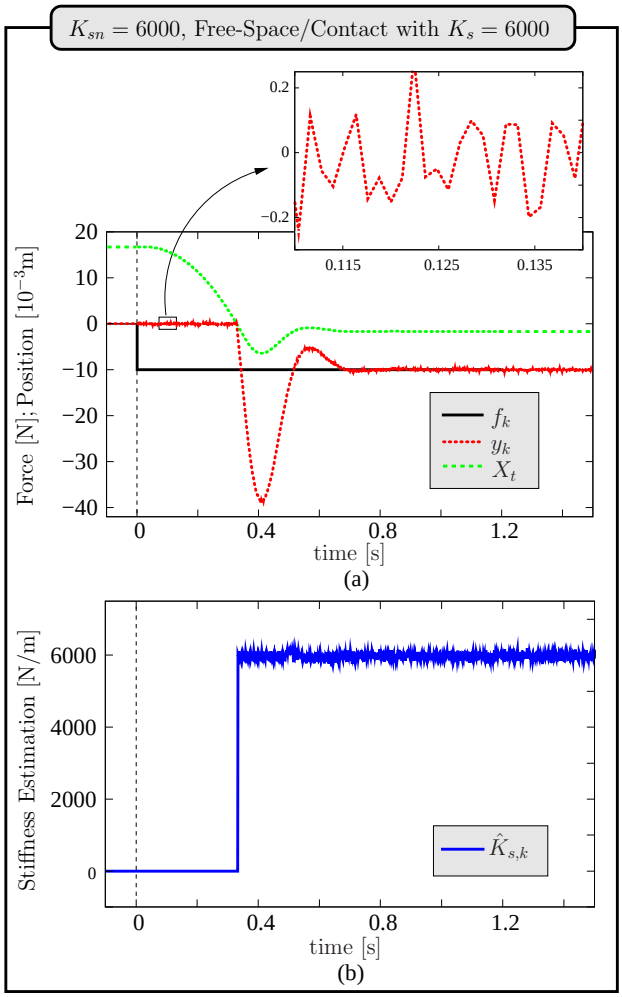


Figure 6.12: COBA - Free-space/contact transition. (a) Input and output forces (with force sensor bias). (b) COBA estimation $\hat{K}_{s,k}$.

6.6 ANNE

To assess the efectiveness and applicability of ANNs to the stiffness estimation problem, data from an ideal and controlled environment, obtained by simulation of the system plant, was used. Several aspects of the ANNE Algorithm, such as the input set configuration and network topology, were decided by running several preliminary trial test runs. Table 6.4 presents the setup configuration parameters and the ANN and ANNE design parameters used in the simulation results presented in the current section.

Table 6.4: ANNE simulation results - design parameters.

AOB Design Parameters

Description	Parameter	Value
Nominal stiffness	K_{sn}	{400, 500, 800, 900, 1200, 1300}

Training and Testing Parameters

Description	Value
Experiment duration	1 [s]
Samples/experiment	2000 random samples
Training set	75% of all samples
Validation set	25% of all samples
Max. training epochs	1000
Convergence criteria	5 epochs w/o improv.
Nr. of neurons on input layer	18
Nr. of neurons on hidden layer	10
Regressor order	1
Regression set dimension	2

Training Stiffness

Description	Parameter	Value
Training Stiffnesses	K_T	{400,800,1200}
System Stiffness	K_s	{100,150,...,1600}

6.6.1 Training

The ANNE estimator considered in this section is composed of three ANNs: ANN400, ANN800 and ANN1200. These were trained with $K_{sn} = 400$, 800 and 1200[N/m], respectively, using a similar procedure:

- The ANN input data (y_k and u_k) was collected from several simulations of scenarios with matching and mismatching stiffnesses and diverse system input forces f_k.

- The input forces adopted in training were obtained from experimental Phantom telemanipulation experiments. These provide the rich frequency content fundamental for success of the training process.

6.6.2 Testing

Several tests have been conducted under different conditions with the goal of analysing the generalization capabilities of ANNE. Testing was conducted as described in Section 5.3.3. The input data (y_k and u_k) was collected from experiences with system input signals f_k that were not used in the training process. Stiffness mismatch scenarios not specifically included in the training set were used, i.e., different ΔK_s values ($\Delta K_s = K_s - K_{sn}$). Fig. 6.13 illustrates the test results. Fig. 6.13(a) represents the input and output forces of the system plant. Fig. 6.13(b) shows that the stiffness estimation evolution computed by ANNE converges to approximately 1020 [N/m]. Results show that ANNE stiffness estimation $\hat{K}_{s,k}$ successfully converges to K_s. Since the parameters of the simulated environment model are well known, the estimations provided by ANNE can be compared against expected results. These are satisfactory and indicative that ANNE is, in fact, successful at estimating the known environment stiffness, using the provided input data.

6.7 Discussion

Previous results show that ASBA, COBA and ANNE estimate successfully the system stiffness in simulated experiments involving the interaction between the manipulator and a known environment. The algorithms are very distinct from each other, but share the fact that they do not require object position data. ASBA requires processing a higher volume of data, making it computationally heavier that COBA and ANNE. COBA is very adequate for online operation and estimates the contact stiffness in just one time iteration even for severe stiffness mismatch scenarios. ANNE also reveals potential for online implementation and good estimation behavior. These simulation results demonstrate that, according to conceptual and theoretical analysis, the algorithms effectively estimate the system stiffness. However, certain strong assumptions are made. In these simulations, perfect linearization is assumed and consequently the manipulator is represented by a double integrator. Also, contact is modeled by a perfect Hookean spring and the manipulator is perfectly rigid. These assumptions describe ideal conditions that may not hold in experimental settings:

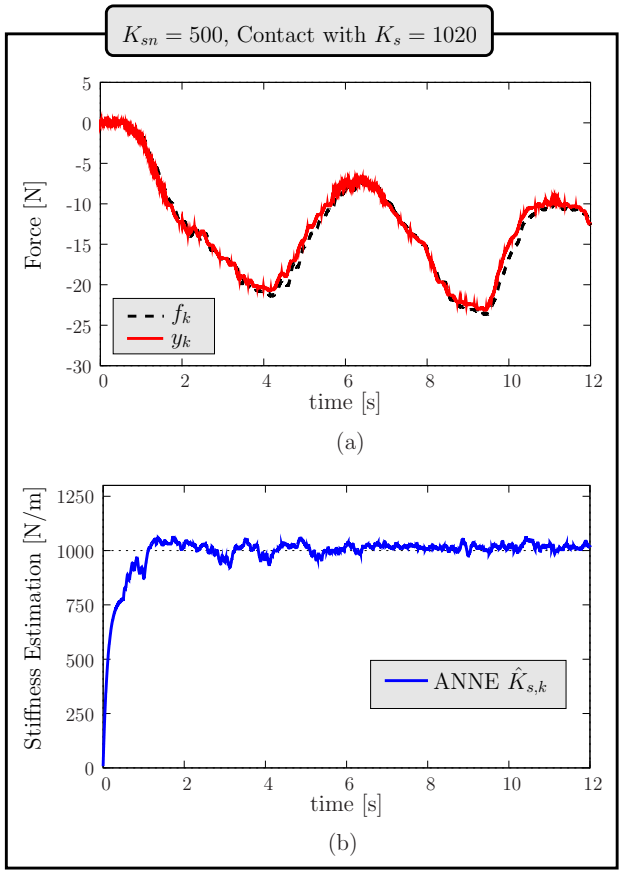


Figure 6.13: ANNE - simulation results for untrained data. $K_{sn} = 500$ [N/m] and $K_s = 1020$ [N/m]. (a) Input f_k and output y_k forces. (b) ANNE stiffness estimation evolution.

- Feedback linearization may not perfectly compensate the manipulator dynamics.
- The linear Hooke model is but an approximation of the real contact dynamics. Unmodeled viscous and non-linear terms are not accounted for.
- Although the manipulator is significantly stiffer than the environment, its stiffness is in fact finite.

Chapter 7 presents experimental results, obtained with a robotic manipulator interacting with virtual and real objects. These illustrate the behavior

of the algorithms when ideal simulation conditions are not observed.

Chapter 7

Experimental Results

Contents

7.1 Introduction

This chapter presents and discusses COBA and ANNE experimental results, including the validation and test of both algorithms. The validation is performed using WAM robot interactions with virtual objects and the algorithms results are also analysed when the WAM robot interacts with real objects. A free-space/contact experiment is also presented. The experiments are focused on contact stiffness estimation, so no adaptation of the

control laws is performed. All quantities are based on SI units and only the Z Cartesian dimension is considered. The experiments were conducted at the Institute of Systems and Robotics of the University of Coimbra. This section is organized as follows. The experimental setup is described in Section 7.2. COBA and ANNE results are presented in sections 7.3 and 7.4, respectively. Results discussion can be found in Section 7.5.

7.2 Experimental Setup

The experimental setup consists of a WAM robot with a JR3 force sensor, controlled as described in Chapter 3, teleoperated through a Phantom haptic device. A critically damped control design with time constant τ_f is implemented for force tracking. The control design and system parameters are presented in Table 7.1. COBA, ANNE and AOB parameters specific to each experiment are presented in the respective section. The manipulator can be teleoperated by the human operator to interact with real objects or, alternatively, virtual objects. The contact properties of the latter can be specified by the operator, allowing the estimations provided by the algorithms to be contrasted against known values.

Table 7.1: Experimental results - common control design and system parameters.

Control Parameters

Description	**Parameter**	**Value**
Sampling time	T_s	0.0012
System time delay	T_d	0.0012
Damping gain	K_D	10
Time constant of the force loop	τ_f	0.045
yhu Damping coeff. of the force loop	ξ	1

AOB Common Parameters

Description	**Parameter**	*Value*
Meas. noise matrix	R_{noise}	1
System noise matrix	Q_{noise}	$diag\{10^{-4}, 10^{-12}, 10^{-12}, 10^{-4}\}$

Teleoperation Parameters

Description	**Parameter**	**Value**
Input force gain	K_v	800
Position gain factor	β_p	1
Force gain factor	β_f	0.5

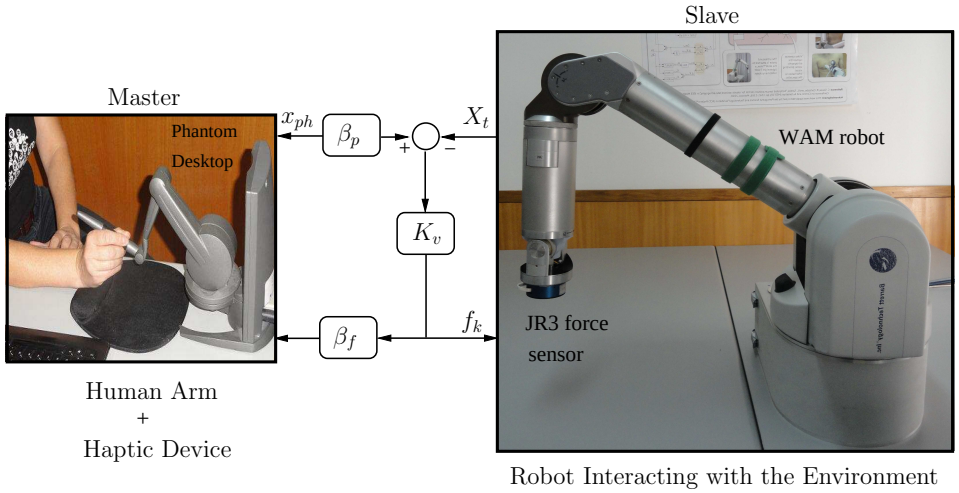


Figure 7.1: Teleoperation architecture of an existing robotic platform at ISR-Coimbra. The human arm manipulates the haptic device generating position data x_{ph}. The slave station includes the robot with feedback linearization controlled by state feedback. The position error between the robot end-effector X_t and the Phantom x_{ph}, scaled by a virtual coupling K_v, generates the reference force f_k. β_f scales back f_k to the master station and and β_p scales the phantom position x_{ph}. $\hat{Z}$ is the Z coordinate versor for both position and force.

7.2.1 Teleoperation Architecture

The teleoperation architecture is represented in Fig. 7.1. The human operator interacts with the environment using the haptic device, generating force signals f_k in Z Cartesian dimension. The task space position error between the haptic device and the WAM robot end-effector is scaled by a virtual coupling K_v to compute the reference force f_k,

$$f_k = K_v(\beta_p x_{ph} - X_t). \tag{7.1}$$

This force is tracked by the control architecture. The haptic feedback is a scaled version of f_k.

- ***Master Station*** - The master station includes the Phantom Desktop haptic device, controlled by the human arm. This haptic device generates 3-DOF positioning and orientation data, and provides 3-DOF force-feedback. In the experiments, only 1-DOF positioning (the Z Cartesian dimension) and 1-DOF force feedback was considered.

- ***Slave Station*** - The slave station consists on a computed torque controlled WAM robot, equipped with a JR3 force sensor.

The master and slave stations are connected to the same server. The position-based information X_t and x_{ph}, required by this teleoperation architecture, is not dependant on the contact object: it can be determined from the well-known geometric model of the manipulator and joint encoder information. This means that, although the teleoperation scheme is based on position data, it is not affected by the same issues that can negatively affect stiffness estimation algorithms relying on accurate object position data.

7.2.2 Virtual and Real Environments

Two different types of environments are used in the experiments: virtual and real objects (see Fig. 7.2). A virtual object is modeled by the perfect spring K_{plane} with known rest position. In this case, the measured force is computed as $y_k = K_{\text{plane}}(X_t - X_0)$, where X_t is the current end-effector position, X_0 is the rest position and K_{plane} is the virtual object stiffness, specified by the user. It is assumed that initially the robot end-effector is in contact with the virtual object. The experiments with virtual objects can be used for validation purposes, since the contact parameters are well known in this case. Other experiments involve contact with real objects: a pillow, a sponge and a book. Fig. 7.2 highlights the slave station workspace, showing the virtual and real objects used in the experiments. Blocks representing the COBA and ANNE algorithms are also included, highlighting the input signals required by each one.

7.3 COBA

Two different types of experiments were conducted. In the first one, the WAM robot interacts with virtual objects of known stiffness, with the purpose of validating the COBA Algorithm. In the second one, COBA behavior is analysed when robot interacts with real objects of unknown stiffness. The AOB and COBA design parameters are shown in Table 7.2.

7.3.1 WAM Robot Interaction with Virtual Objects

The purpose of conducting experiments involving interactions with a virtual plane is twofold:

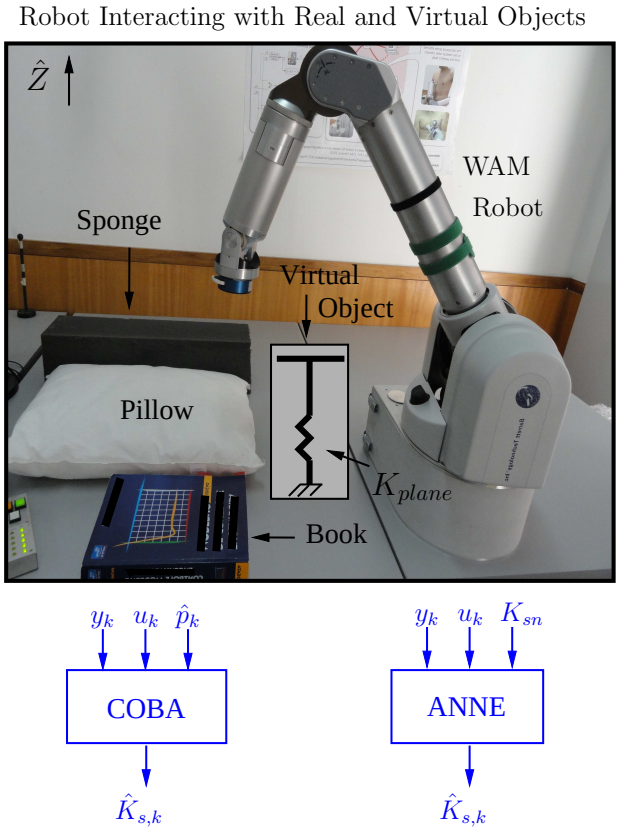


Figure 7.2: Environment used in COBA and ANNE experiments. WAM robot interacts with virtual objects (modelled as springs with stiffness K_{plane}) and with real objects (pillow, sponge and book). y_k and u_k are inputted to both algorithms, while COBA also receives $\hat{p}_k$, additionally.

1. **Characterizing and compensating the dominant disturbance**: As described in Chapter 4, compensating the additional unknown disturbance improves the estimation performance. By contrasting the results of simulation tests with those of manipulator/virtual object experiments, using an identical parametrization of the environment, it is possible to accurately identify the additional disturbance. This makes it possible to observe estimation behavior when perfect compensation is employed and also identifying the dominant disturbance pattern so that appropriate compensation strategies can be devised.

2. **Validating the operating principle of COBA**: Simulation test

Table 7.2: COBA experimental results - design parameters.

AOB Parameters

Description	Parameter	Value
Nominal stiffness	K_{sn}	{300,800,1500}

COBA Parameters

Description	Parameter	Value
Stiffness of FOB_1	K_{sc_1}	100
Stiffness of FOB_2	K_{sc_2}	6000
FOB gains	K_c	$\{0.001, 0, 0\}$
FOB initial state	$x_{c,0}$	$\{0, 0, 0\}$

results, presented in Chapter 6, allow the operation principle of the algorithms to be assessed under ideal assumptions. One of these is perfect feedback linearization, resulting in ideal double integrator behavior. However, no such assumption is involved in interaction experiments involving manipulator/virtual plane contact. This allows the results of the algorithm to be compared to expected values, while removing one significant assumption, offering increased validation of COBA results.

Additional Disturbance Compensation

As discussed in Section 4.3.4, COBA is negatively affected by additional disturbances lumped in $p_k(o)$. However, if compensation of this disturbance is perfect, the correct stiffness estimation is obtained (see Theorem 2). The experiment described in this section aims to provide the corresponding empirical confirmation. Taking advantage of the known environment, a (nearly) ideal compensation of $p_k(o)$ can be computed, allowing the algorithm operation principle to be demonstrated. The estimation $\hat{p}_k$, computed by the AOB, includes the components $\hat{p}_k(\Delta K_s)$ and $\hat{p}_k(o)$ (see (3.37)). An estimation of $p_k(o)$ can be obtained by

$$\hat{p}_k(o) = \hat{p}_k - \hat{p}_k(\Delta K_s). \tag{7.2}$$

The term $\hat{p}_k(\Delta K_s)$ is dependant on the unknown stiffness mismatch ΔK_s and cannot be directly computed in practice. In these experiments, however, the environment is known and so $\hat{p}_k(\Delta K_s)$ can be obtained analitically for a given input force. In this way, a precise estimation $\hat{p}_k(o)$ can be obtained by (7.2), knowing the AOB active state $\hat{p}_k$ and the estimation $\hat{p}_k(\Delta K_s)$.

With this in mind, the following experiment was conducted, with the purpose of observing estimation behavior with nearly ideal compensation of the additional disturbance, as discussed in Section 4.3.4 and represented in Fig. 4.13:

- A sinusoidal input force f_k is used to compute analytically $\hat{p}_k(\Delta K_s)$ for given K_{sn} and K_s, corresponding to the nominal stiffness of the controller and the programmed virtual plane stiffness.
- The same reference force is inputted to the robotic controller, and ideal compensation is computed and applied according to (7.2).

Figs. 7.3 and 7.4 present the results for all four combinations of nominal and virtual plane stiffnesses of 300 [N/m] and 800 [N/m]. Input and measured forces present in Figs. 7.3(a), 7.3(d), 7.4(a), and 7.4(d) show good tracking capabilities of the control architecture. In Figs. 7.3(b), 7.3(e), 7.4(b) and 7.4(e), COBA estimations can be seen to closely match the virtual plane stiffness, although poorly conditioned results can be observed at specific time instants. In accordance to the discussion in Section 4.3.5, these correspond to low values of $|^1G_k|$. In Figs. 7.3(c), 7.3(f), 7.4(c) and 7.4(f), an assessment technique has been implemented by discarding estimations corresponding to

$$|^1G_k| < 0.002. \tag{7.3}$$

These results validate the operation principle of COBA, ensuring that, with adequate compensation of additional disturbances, correct estimations can be obtained.

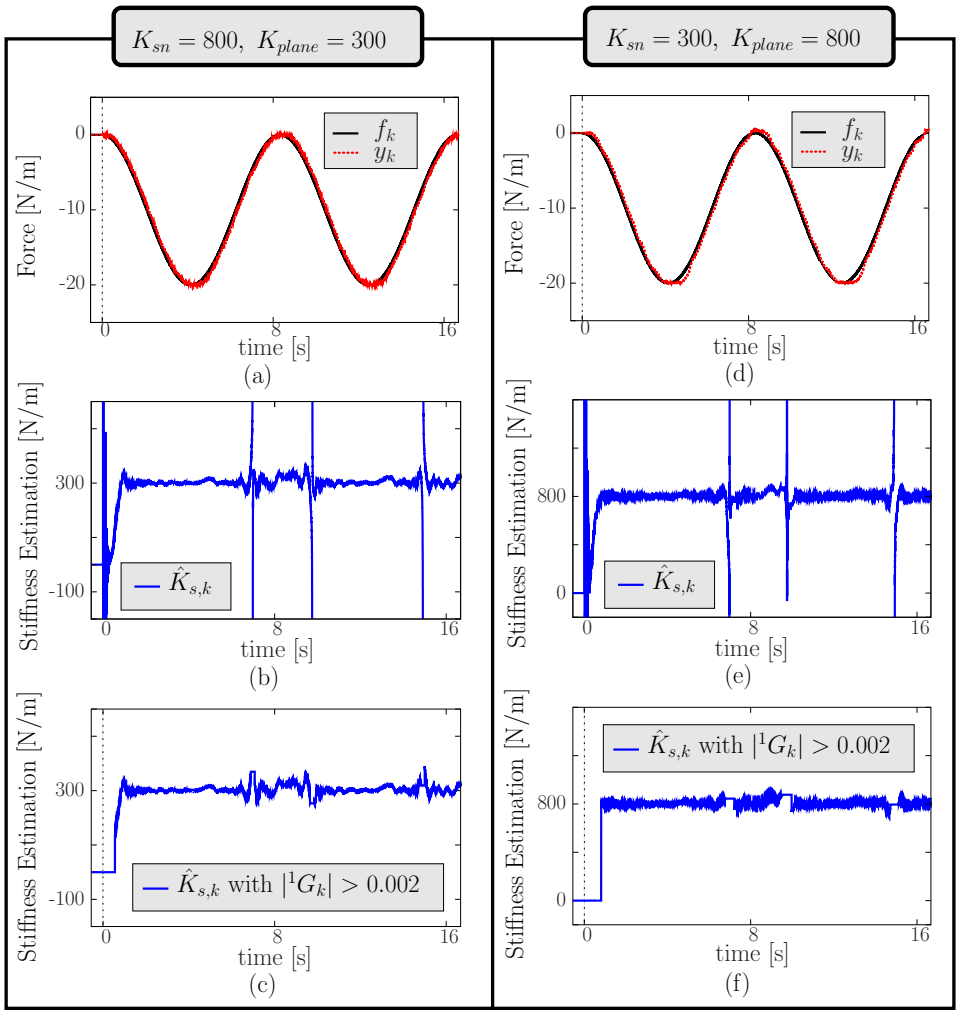


Figure 7.3: COBA - WAM robot interacting with virtual planes in mismatching stiffness scenarios with a sinusoidal input force. f_k and y_k are, respectively, the input and measured forces. $\hat{K}_{s,k}$ is the stiffness estimation. (a)-(c) Overestimated stiffness scenario with $K_{sn} = 800$ [N/m] and $K_{\text{plane}} = 300$ [N/m]. (d)-(f) Underestimated stiffness scenario with $K_{sn} = 300$ [N/m] and $K_{\text{plane}} = 800$ [N/m]. The vertical dotted line represents the starting point of each experiment.

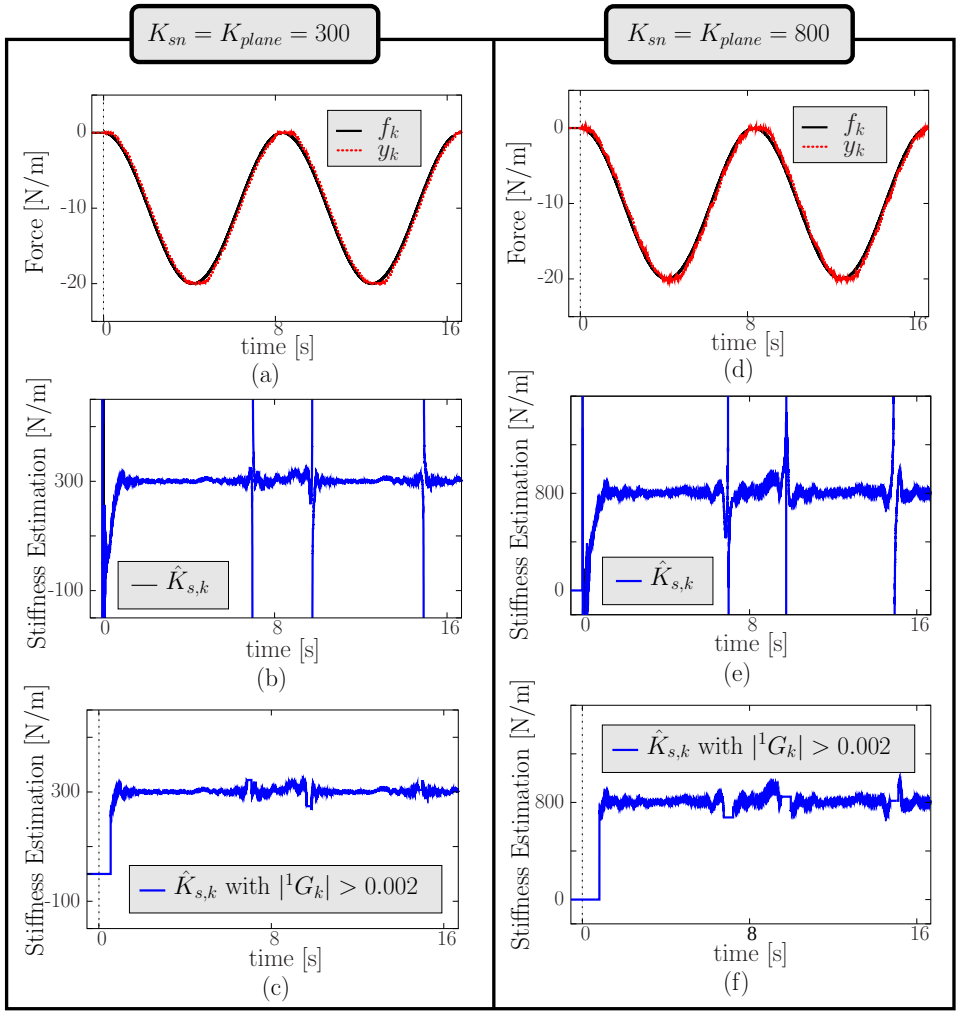


Figure 7.4: COBA - WAM robot interacting with virtual planes in matching stiffness scenarios with a sinusoidal input force. f_k and y_k are, respectively, the input and measured forces. $\hat{K}_{s,k}$ is the stiffness estimation.(a)-(c) $K_{sn} = K_{\text{plane}} = 300$ [N/m]. (d)-(f) $K_{sn} = K_{\text{plane}} = 800$ [N/m]. The vertical dotted line represents the starting point of each experiment.

Analysis of the Dominant Additional Disturbance

Although the approach described in the previous experiments successfully compensates the additional disturbances entailing a correct virtual plane stiffness estimation, it cannot be applied to real objects since the stiffness mismatch is not known, precluding the analytical computation of $\hat{p}_k(\Delta K_s)$. An alternative strategy to obtain an estimation of $p_k(o)$ is to identify the dominant additional disturbance. To this end, results have been collected for several matching stiffness scenarios where $\Delta K_s = 0$ (and therefore $\hat{p}_k(\Delta K_s) = 0$), making $\hat{p}_k = \hat{p}_k(o)$ by (3.37). Fig. 7.5 presents the results for one such experiment. By inspecting $\hat{p}_k$ from the AOB, the disturbance be-

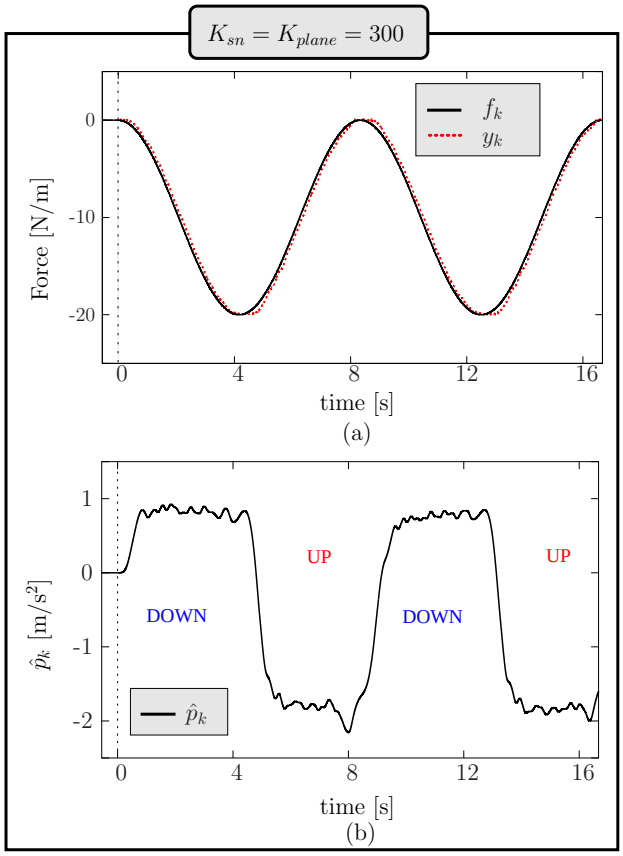


Figure 7.5: Analysis of dominant disturbance when a WAM robot interacts with a virtual plane with up and down sinusoidal movements. $K_{sn} = K_{\text{plane}} = 300$ [N/m]. (a) Reference and output forces. (b) AOB active state $\hat{p}_k$. The vertical dotted line represents the starting point of the experiment.

havior can be observed. Results demonstrate that $\hat{p}_k$ can be approximately described as alternating between two different constant levels, depending on whether the end-effector is penetrating or leaving the virtual plane. Knowing this, an appropriate compensation strategy can be devised for generating the compensation $\hat{p}_k(o)$. In the experiments described in Section 7.3.2, $\hat{p}_k(o)$ is obtained by applying a low-pass filter to $\hat{p}_k$. From (4.27), the dominant additional disturbance is compensated, approximating ideal operating conditions.

7.3.2 WAM Robot Interaction with Real Objects

This section describes experiments where the manipulator, teleoperated through an haptic device, interacts with real objects with different stiffnesses: a pillow, a sponge and a book. The purpose of these experiments is to allow the behavior of the algorithm to be observed in practical experimental conditions. Additionally, a comparative experiment with COBA and a position-based estimator is also presented. The compensation of the additional disturbance follows the process described in Section 7.3.1.

Interactions with a Pillow, a Sponge and a Book

Figs. 7.6, 7.7 and 7.8 present force control results for interactions with pillow, sponge and book, respectively. Initially, the end-effector is in free-space, very close to the contact position. After entering contact, the measured force tracks the input force. A constant nominal stiffness of $K_{sn} = 800$ [N/m] is used in all experiments, entailing different stiffness mismatches and distinct dynamic behaviors. The stiffness estimation provided by COBA converges to approximately 30 [N/m], 500 [N/m] and 1400 [N/m] for pillow, sponge and book, respectively (see Figs. 7.6(b), 7.7(b) and 7.8(b)), indicating an underestimated scenario for book and overestimated for pillow and sponge. Force tracking behavior is consistent with these results: underestimated stiffness exhibits slower dynamics (see Fig. 7.6(a)), while overestimated stiffness corresponds to faster dynamics (see Figs. 7.7(a) and 7.8(a)). The assessment criteria validates estimations for low contact forces (e.g., -0.15 [N] for pillow, -0.3 [N] for sponge and -0.45 [N] for book). As the magnitude of contact force increases, the stiffness estimation increases until it saturates around a constant value, reflecting nonlinearities in the contact interaction.

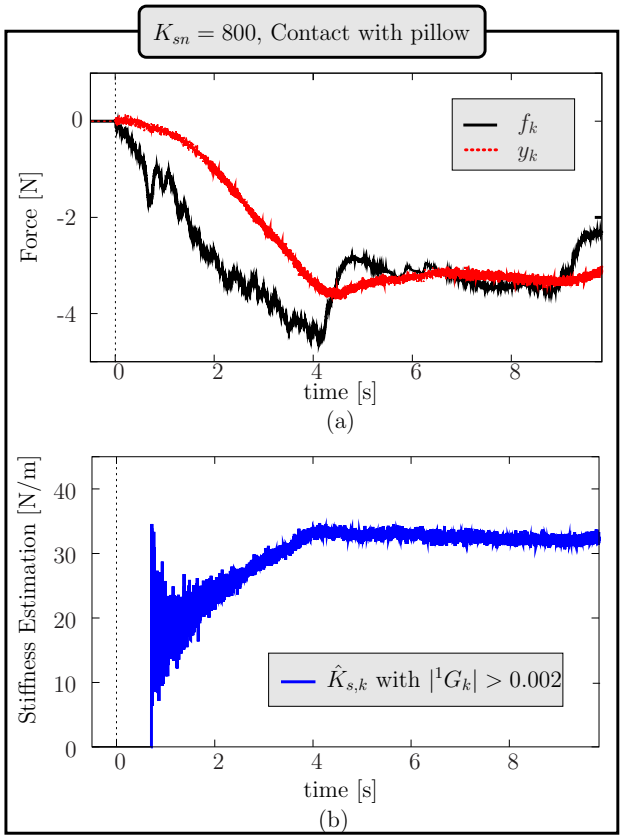


Figure 7.6: COBA - WAM robot interacting with a pillow. f_k and y_k are, respectively, the input and measured forces. $\hat{K}_{s,k}$ is the stiffness estimation. The vertical dotted line represents the starting point of each experiment.

7.3.3 Free-Space/Contact Transitions

This section presents a teleoperated experiment involving transitions from free-space to contact with a sponge (and *vice-versa*). These transitions can be challenging for estimation algorithms, as the contact parameters change abruptly and past system behavior no longer conveys information about the current contact environment[1]. The experiment aims to assess the performance of COBA in these conditions. The control architecture has a constant nominal stiffness of $K_{sn} = 1500$ [N/m]. The WAM robot begins operation in free-space and establishes contact with a sponge in two occasions. Fig. 7.9(a)

[1]Environment is used here in a broad sense, including free-space operation, which can be modeled as an interaction with an environment with null contact parameters.

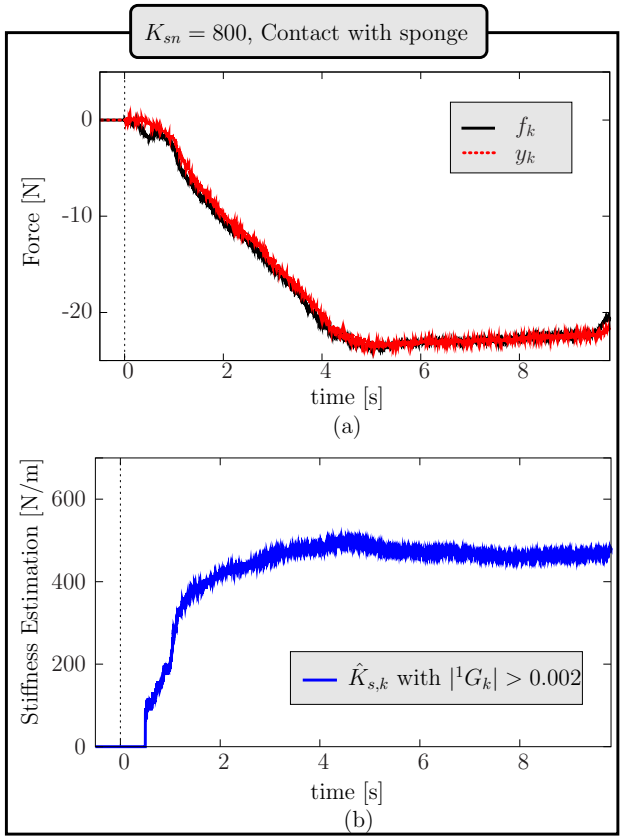


Figure 7.7: COBA - WAM robot interacting with a sponge. f_k and y_k are respectively input and measured forces. $\hat{K}_{s,k}$ is the stiffness estimation. The vertical dotted line represents the starting point of each experiment.

presents force control results. In free-space the measured force is approximately zero, being corrupted by measurement noise and by a small bias. Before entering contact, the teleoperated robot moves up and down along the Z-axis. Once in contact with the sponge, the measured force tracks the input force. In this experiment, contact is broken around 20 [s] and re-established once again around 25 [s]. In Fig. 7.9(b) the stiffness estimation from COBA is presented. Free-space estimation is successfully obtained. Additionally, while in contact, COBA results are consistent and robust to force sensor bias. A consistent stiffness estimation of $\hat{K}_{s,k} \approx 440$ [N/m] is produced during all contact episodes (see Fig. 7.9(b) around 15 [s] and 27 [s]), in line with expected properties of the physical object. This experi-

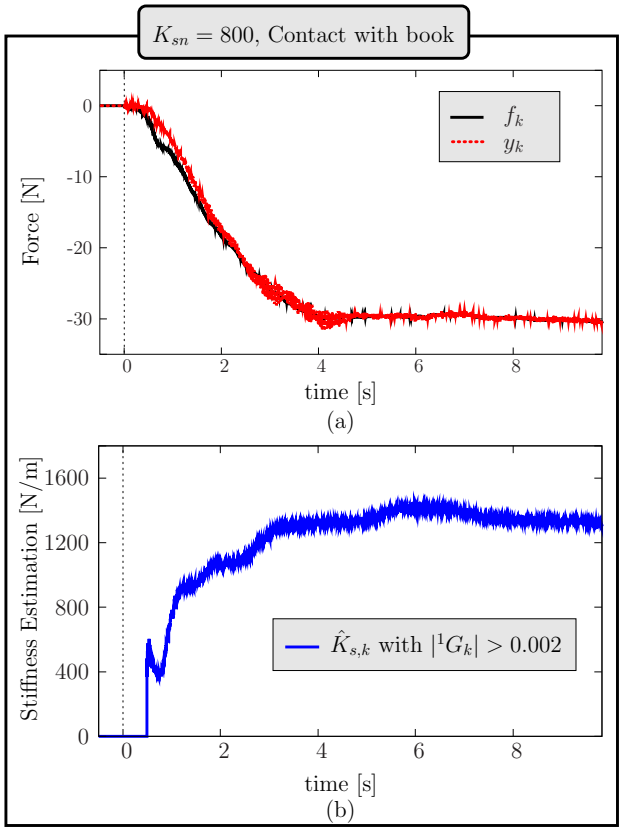


Figure 7.8: COBA - WAM robot interacting with a book. f_k and y_k are respectively input and measured forces. $\hat{K}_{s,k}$ is the stiffness estimation. The vertical dotted line represents the starting point of each experiment.

ment demonstrates that COBA can be used to obtain reliable and consistent stiffness estimations even for environments with complex geometry, since the estimations rely only on force data (i.e., no explicit geometric information is needed).

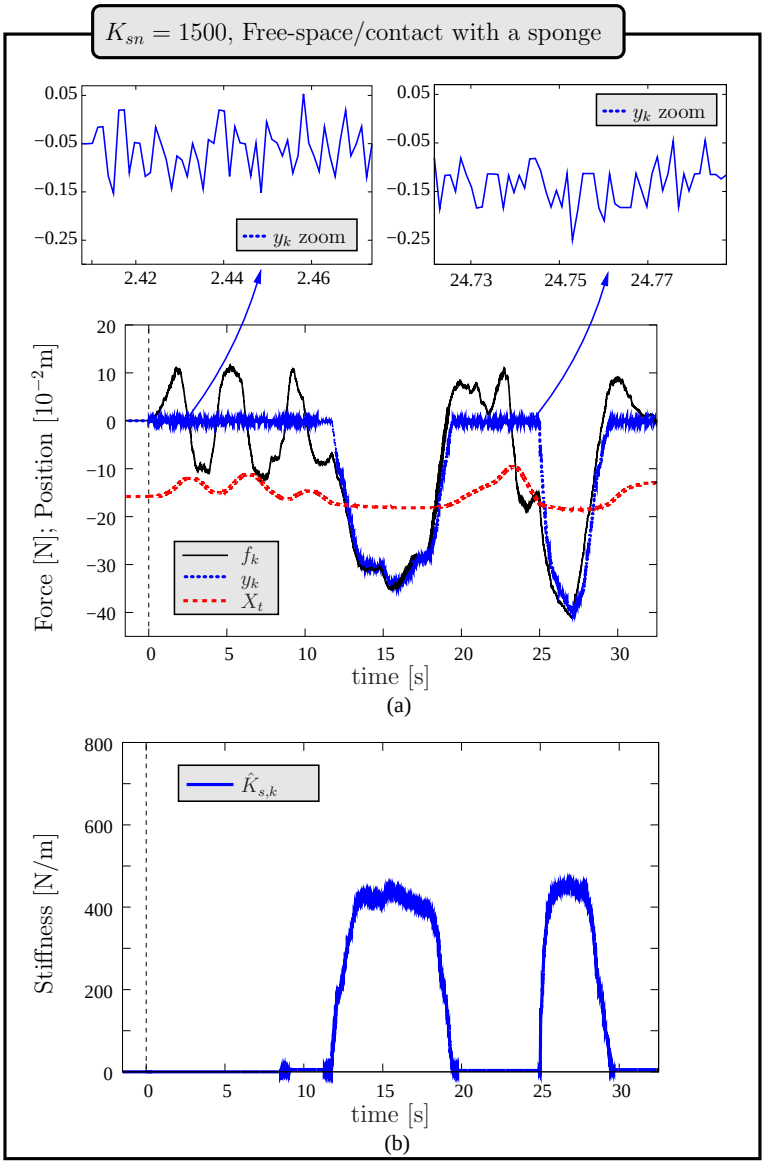


Figure 7.9: COBA - free-space and contact transitions between a WAM robot and a sponge. (a) Reference and measured forces. (b) Stiffness estimation based on COBA. The vertical slashed line denotes the starting point of the experiment.

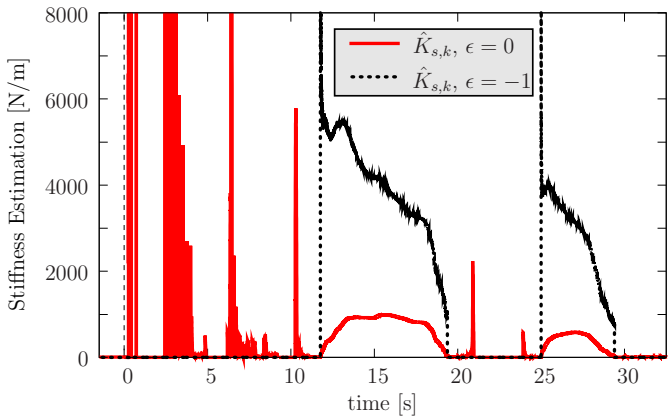


Figure 7.10: Position-dependent RLS: Free-space and contact with a sponge. Stiffness estimation based on RLS technique and two different force thresholds.

7.3.4 Comparison With Position-Based Approach

For comparison purposes, a position-algorithm stiffness estimator, based on RLS techniques [66] and the contact model (1.2), was implemented and applied to the data in Fig. 7.9(a). The following provisions were adopted to enable proper algorithm operation:

- Contact transitions and the rest position $\hat{X}_0$ are identified with recourse to a force threshold parameter ϵ.
- Past estimation information is reseted in contact transition to improve responsiveness.

Fig. 7.10 presents the estimation results for two different force thresholds ϵ. When $\epsilon = 0$ [N], measurement noise triggers too many false positive contacts, making the estimation very noisy in free-space (see the first 10 [s] of Fig. 7.10). Higher contact thresholds can tackle these issues, but high estimation peaks at contact appear due to higher forces required to trigger contact detection, generating also poor X_0 estimates. Non-uniform detection of contact transitions in free-space with $\epsilon = 0$ [N] is caused by a small bias in the force sensor, which is stronger after the first contact (see zooms of Fig. 7.9(a)). While in contact, substantially different stiffness estimations are produced for different values of ϵ (see Fig. 7.10 around 15 [s]). Moreover, no stiffness estimation consistency exists in successive surface contacts (see Fig. 7.10 around 15 [s] and 27 [s]), due to different values of $\hat{X}_0$ for each

contact transition (even for the same ϵ). Force sensor bias changes after contact entail inconsistent $\hat{X}_0$ estimates, having consequences in stiffness estimation. For example, in the experiment with $\epsilon = 0$ [N], a stiffness estimation close to $\hat{K}_{s,k} \approx 1000$ [N/m] is obtained at the first contact (around 15 [s]), while in the second contact an estimation of $\hat{K}_{s,k} \approx 500$ [N/m] is produced (around 27 [s]). These results contrast with those of COBA, presented in Fig. 7.9(b). By avoiding reliance on positional data, COBA produces consistent estimations in both contact episodes, without estimation peaks near free-space/contact transition or spurious estimations in free-space due to sensor noise.

7.4 ANNE

This section describes experiments related to training and evaluation of the ANNE estimator. Two sets of experiments have been conducted. In the first set, the robot interacts with virtual objects with known stiffnesses. This setup is used for training and assessing ANNE in a controlled and known environment. In the second set, ANNE is trained and evaluated using data from interactions of the robot with real objects with unknown stiffnesses. In this case, the object stiffness is not available. To overcome this problem, the stiffness estimations provided by the COBA Algorithm (Section 4.3) are used whenever the stiffness value is required for training or assessment purposes. The experiments described in sections 7.4.1 and 7.4.2 follow the same general process:

(i) The general configuration parameters of the ANNE estimator are determined: number of composing ANNs, their respective nominal stiffness, and the fusion module order. The individual ANNs are trained and tested accordingly.

(ii) The individual ANNs are composed into the overall ANNE estimator, whose performance is then evaluated.

Table 7.3 presents the setup configuration parameters and the ANN design parameters.

7.4.1 WAM Robot Interaction with Virtual Objects

The experiments considered in this section involve robot interactions with virtual objects which are modelled by the spring K_{plane} (see Fig. 7.1). These experiments include both training and assessment of the ANNE estimator.

Table 7.3: ANNE experimental results - design parameters.

AOB Parameters

Description	Parameter	Value
Nominal stiffness	K_{sn}	{400, 500, 800, 900, 1200, 1500, 1750, 2000}

Common Training and Testing Parameters

Description	Value
Experiment duration	variable (6 [s] to 15 [s])
Samples/experiment	4000 random samples
Training set	75% of all samples
Validation set	25% of all samples
Max. training epochs	2000
Convergence criteria	5 epochs w/o improv.
Nr. of neurons on input layer	18
Nr. of neurons on hidden layer	4
Regressor order	1
Regression set dimension	2

Training Parameters for Virtual Objects

Description	Parameter	Value
Training Stiffnesses	K_T	{400,800,1200}
Virtual Plane Stiffness	K_{plane}	{100,150,...,1600}

Training Parameters for Real Objects

Description	Parameter	Value
Training Stiffnesses	K_T	{1500,2000}
Training Objects	-	Pillow, sponge, book

ANN Training and Testing

Training data is collected from telemanipulation interactions within a range of environment stiffnesses (low to medium) typical of robotic-assisted minimally invasive surgery tasks (see Table 7.3). Three individual ANNs were trained: ANN400, ANN800 and ANN1200, so that the space of low to medium stiffness is uniformly occupied. Teleoperated interaction experiments were conducted for collecting the data necessary for training, and then all ANNs were trained (see Table 7.3 for specific details and parameters). To test the performance of ANN400, ANN800 and ANN1200, a novel set of experiments is conducted. These provided new data sets, not used in the training process, so that the generalization capabilities of the networks can be assessed. Fig. 7.11 shows the average estimation results provided by each ANN with K_{plane} ranging from 300 [N/m] to 1300 [N/m], in experi-

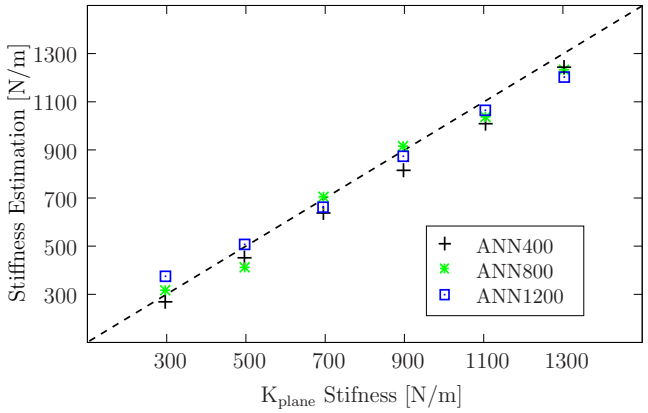


Figure 7.11: Average values of stiffness estimations computed by three ANNs: ANN400, ANN800 and ANN1200. The results for eighteen different robot interaction experiments are shown. The WAM robot interacts with virtual objects with stiffnesses $K_{\text{plane}} \in \{300, 500, 700, 900, 1100, 1300\}$ [N/m]. $K_{sn} \in \{400, 800, 1200\}$ [N/m], according to the ANN being used. The diagonal dotted line represents ideal stiffness identification.

ments where $K_{sn} = K_T$. It can be seen that all networks estimate close to the actual object stiffness. Fig. 7.12 illustrates the instantaneous stiffness estimation provided by each ANN for three experiments sharing the same $K_{plane} = 1000$ [N/m] and $K_{sn} = K_T$. In accordance with the results of Fig. 7.11, it can be observed that all networks provide estimations close to the correct value. Fig. 7.13 shows the performance of ANN400, ANN800 and ANN1200 for three different experiments with $K_{sn} \in \{400, 800, 1200\}$ [N/m] and $K_{\text{plane}} = 1000$ [N/m]. If training and nominal stiffnesses match (i.e., $K_{sn} = K_T$) , the best estimations are obtained, entailing small errors. Otherwise, deterioration occurs.

Fusion Module Design

Fusion module design involves the definition of the regression order m and the size of the regression set n. With this purpose, preliminary results are analysed for evaluating the performance of ANNE in our system for different regression orders and different regression sets. Table 7.4 shows the average estimated polynomial regression coefficients $\hat{A}_k$ (see (5.11)) for different virtual object interactions.

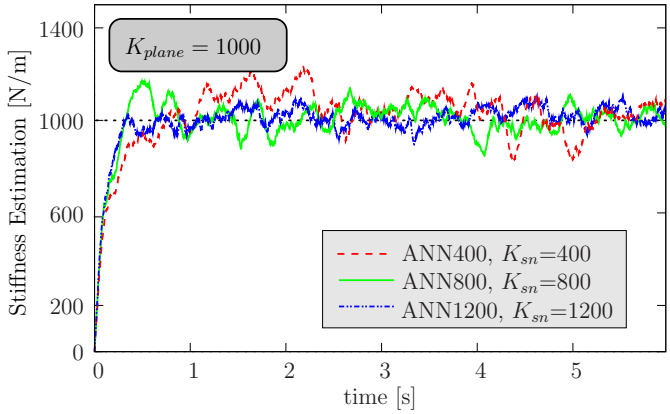


Figure 7.12: ANN400, ANN800 and ANN1200 results for robot interactions with a virtual object. $K_{plane} = 1000$ [N/m]. Each ANN K_T is tested with $K_{sn} = K_T$.

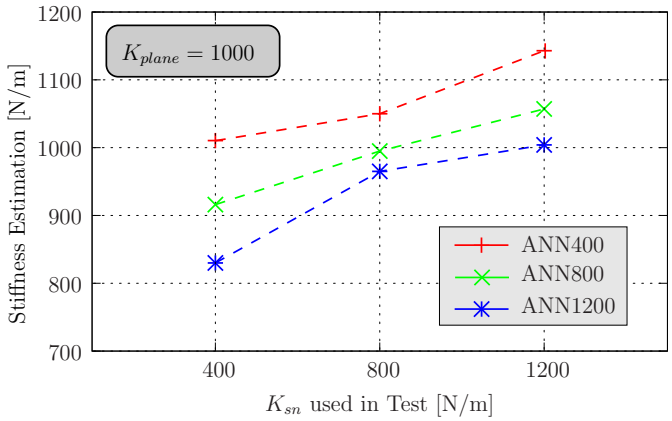


Figure 7.13: ANN400, ANN800 and ANN1200 average estimation results for three different experiments. $K_{sn} \in \{400, 800, 1200\}$ [N/m] and $K_{\text{plane}} = 1000$ [N/m].

Fig. 7.14 shows ANNE results for $K_{sn} = 500$ [N/m] and K_{plane} ranging from 300 to 1300 [N/m] for first and second order regressors and different regression set dimensions. It can be seen that all regressors generate estimations close to the correct value, with the regressor with $m = 1$ and $n = 2$ providing the best results. This suggests adoption of this configuration for the fusion module.

Table 7.4: ANNE - fusion module parameter $\hat{A}_k$.

Virtual Object Stiffness K_{plane}	**Regression Coefficient** a_1	**Regression Coefficient** a_0
300	-0.18	424
500	-0.2425	609
700	-0.58	978
900	-0.8225	1293
1100	-0.64	1395
1300	-1.01	1730

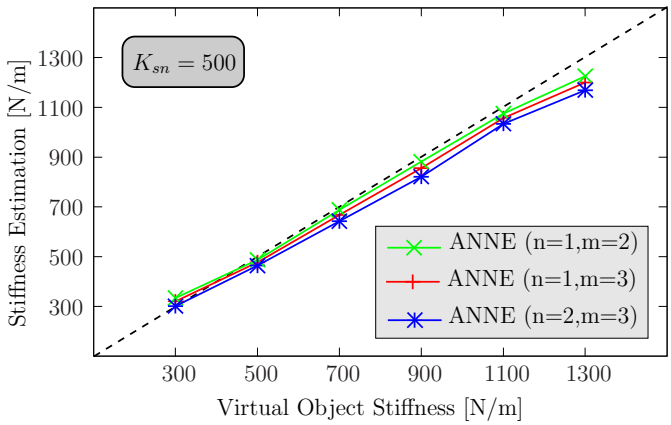


Figure 7.14: ANNE results (average values) for three different configurations of the fusion module. $K_{sn} = 500$ [N/m] and $K_{plane} \in \{300, 500, 700, 900, 1100, 1300\}$ [N/m]. Results are shown from a first order regressor (n=1) with regression set dimensions $m \in \{2, 3\}$ and a second order regressor with $n = 2$ and m=3.

ANNE Testing

Results shown in Fig. 7.13 suggest that the stiffness estimation can be improved by data fusion techniques. Therefore, the individual ANNs are combined to produce the overall ANNE estimator. Fig. 7.15 shows ANNE results with $K_{sn} \in \{500, 900\}$ [N/m] and $K_{plane} \in \{300, 500, 700, 900, 1100, 1300\}$ [N/m]. The data fusion improves the quality of the final estimation in the majority of cases.

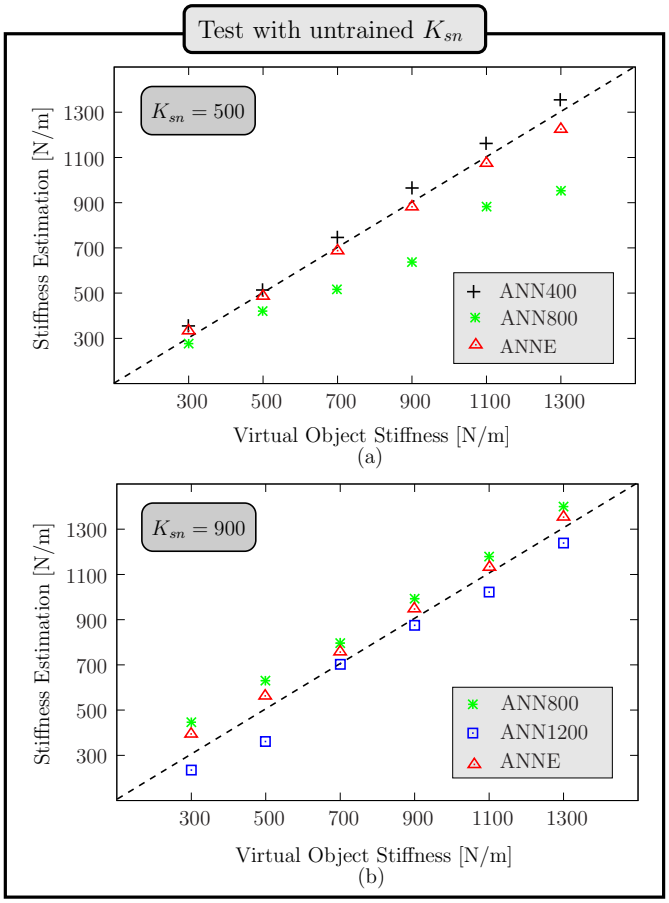


Figure 7.15: ANNE - Stiffness estimation results with objects of stiffnesses in the range 300-1300 [N/m] (average values). (a) Results for $K_{sn} = 500$ [N/m], with the fusion module merging the outputs of networks ANN400 and ANN800. (b) Results for $K_{sn} = 900$ [N/m], with the fusion module merging the outputs of networks ANN800 and ANN1200. The diagonal dotted line represents ideal stiffness identification.

7.4.2 WAM Robot Interaction with Real Objects

The experiments with virtual objects discussed in the previous section are particularly useful since they are performed in a controlled environment, allowing the results to be validated against expected outcome. In this section, the analysis is extended to real objects.

ANN Training and Testing

Since the exact stiffness of real objects is unknown or poorly known, the COBA stiffness estimation algorithm provides the target stiffness data. In all experiments, the robot end-effector begins operation in free-space rather than in contact with the real object, reflecting realistic operating conditions. A pillow and a book have been used for ANN testing purposes. Figs. 7.16 and 7.17 shows results with $K_{sn} = 1500$ [N/m] and $K_{sn} = 2000$ [N/m] for pillow and book interaction, respectively. It can be inferred that good

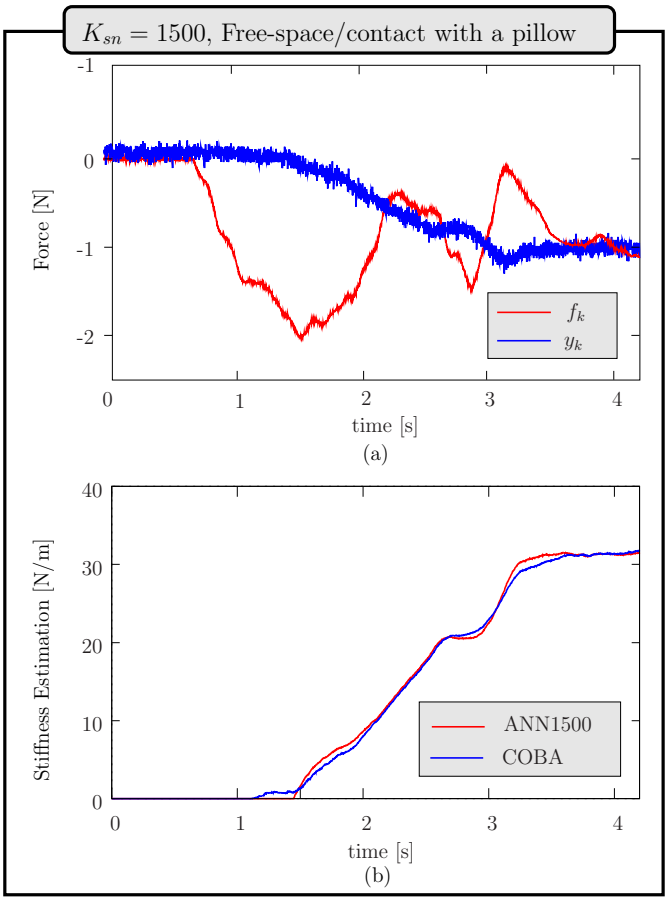


Figure 7.16: WAM robot interaction with a pillow: ANN1500 results. Initially, the robot end-effector is in free-space. f_k and y_k are respectively the input and measured forces. $\hat{K}_{s,k}$ is the stiffness estimation. (a) Force tracking response for pillow interaction. (b) Pillow stiffness estimation obtained with ANN1500.

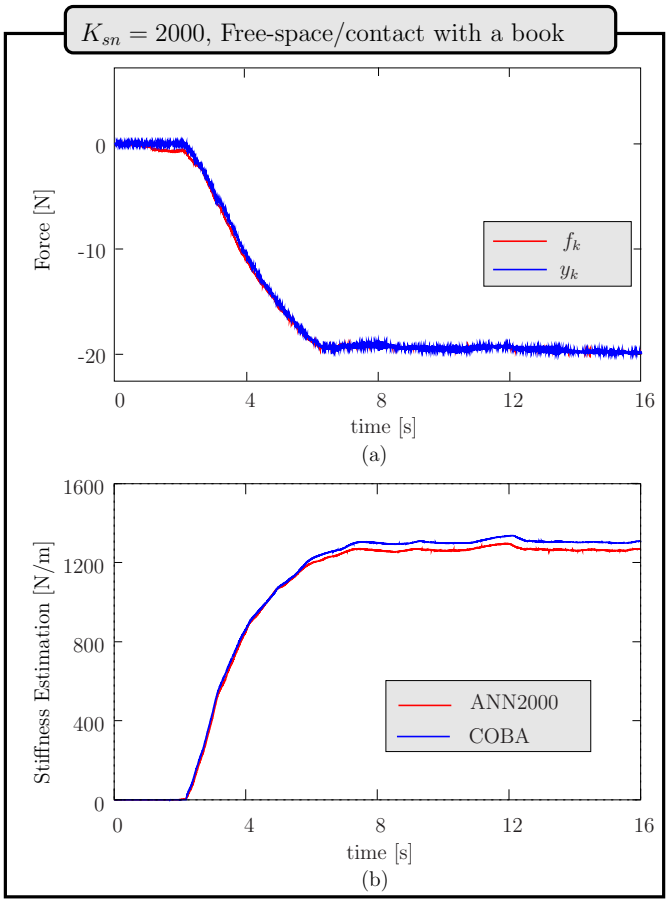


Figure 7.17: WAM robot interaction with a book: ANN2000 results. Initially, the robot end-effector is in free-space. f_k and y_k are respectively the input and measured forces. $\hat{K}_{s,k}$ is the stiffness estimation. (a) Force tracking response for book interaction. (b) Book stiffness estimation obtained with ANN2000.

performance of ANNs is achieved in spite of force control degradation due to stiffness mismatch. The ANNs estimation follows COBA the target estimations provided by COBA. Free-space/contact transition are also handled correctly.

ANNE Testing

While each ANN is tuned for a specific value of K_{sn}, the ANNE estimator is responsible for providing estimations for arbitrary K_{sn}. This is tested

by setting $K_{sn} = 1750$ [N/m] for sponge interaction and configuring the fusion module as described in Section 7.4.1. Results can be observed in Fig. 7.18. Control results can be observed in Fig. 7.18(a), while the estimation is found in Fig. 7.18(b). Even though the ANNE estimation is obtained by fusing the data from ANN1500 and ANN2000, the results are extremely close to those provided by COBA, indicating that the architecture and training methodology are adequate when considering interactions with real objects. Free-space/contact transitions are successfully handled.

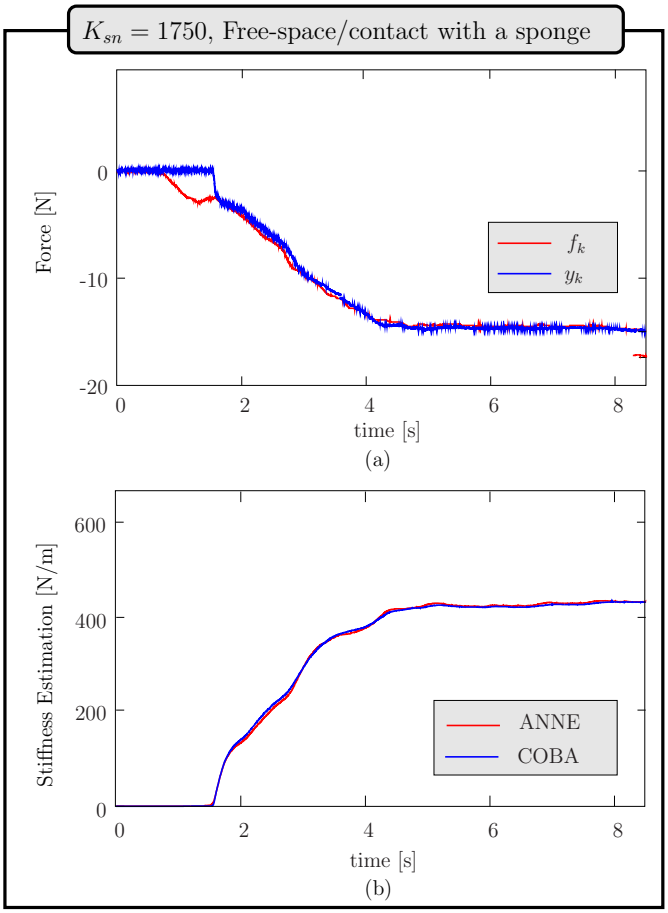


Figure 7.18: ANNE - WAM robot interaction with a sponge. $K_{sn} = 1750$ [N/m]. Initially, the robot end-effector is in free-space. f_k and y_k are respectively the reference and measured forces. $\hat{K}_{s,k}$ is the stiffness estimation obtained by the data fusion of ANN1500 and ANN2000.

7.5 Discussion

The experimental results in this section demonstrate that the proposed techniques (COBA and ANNE) can be successfully used in practical conditions to estimate the environmental stiffness. ASBA has not been included in these experiments as it is less suitable to online operation than either one of the above methods. Interaction experiments involved both virtual and real objects. The main advantage of the former is that the contact parameters can be determined programmatically by the user, allowing results to be constrated against the expected outcome. The algorithms have been tested successfully in conditions reflecting practical scenarios involving free-space/contact transitions. Virtual plane results were also instrumental in determining the general pattern of additional disturbance in our system, allowing proper compensation to be generated. The estimation results of COBA matched expected outcome in virtual plane tests and were consistent with the force tracking results of tasks involving interaction with real objects. The capability to provide consistent estimations across multiple free-space/contact transitions has also been demonstrated, sidestepping problems such as uncertainties in detecting the rest contact position or force sensor bias. The poor results of a position-based estimator when subjected to the same practical conditions have also been demonstrated. Experimental testing demonstrated that the ANNE estimator can be successfully trained to produce the expected results in virtual plane experiments and reproducing the estimation behavior of COBA in interactions with real objects, including in free-space/contact transitions. Both approaches were found to be successful in addressing the main concern of this thesis: estimating the system stiffness without relying on object position data. The relative merits of each approach can be summarized as follows:

- ANNE is less reliant on input signals that are specific to the control approach being used. COBA is more tightly associated with the architecture described in Chapter 3.

- The training process of ANNE requires a data that is correctly matched to a target stiffness. If the environment is unknown, then results from another estimation approach are required to bootstrap the process. In that sense, the estimation performance of ANNE is contingent on performance of the original source. COBA does not require any training prior to online operation and its estimation behavior is supported by theoretical analysis of the properties of the control system.

- A stiffness range must be decided when training ANNE. Estimation results can be expected to decline in quality outside of this range. No explicit, discrete range of operation is assumed in COBA, although assumptions such as the rigidity of the manipulator may be more realistic at the lower end of the stiffness range.

Chapter 8

Conclusions

Contents

8.1 Introduction

Stiffness estimation is a relevant research problem, with significant practical impact and challenging issues to address. One important difficulty is related to uncertainty in the relative position of the manipulator with respect to the environment. This is a significant problem when the manipulator must interact with unstructured, unknown and dynamic environments. The stiffness estimation techniques proposed in this thesis overcome this issue by not requiring object position data. These algorithms can be used to online estimate the system stiffness in tasks where it would be challenging to do so, if such information is required. A critical property of these algorithms is that they are applicable to tasks where stiffness estimation can be beneficial, but it is not normally employed due to practical difficulties. Examples of these tasks include teleoperated robotic-assisted minimally invasive surgery, where stiffness estimation provides improved force tracking and telepresence. Other applications of stiffness estimation include quality control of manufactured objects, identification of irregularities in biological tissues and environment mapping for simulation and task planning purposes. The stiffness estimation algorithms described in this work are embedded in

an explicit force control architecture, using computed torque in the task space. Stiffness estimations are obtained for each uncoupled Cartesian dimension. Force tracking performance has been shown to depend on the mismatch between the real stiffness parameter and the nominal value used in control design. Adaptation of the nominal value during online operation has been found to significantly improve force tracking performance.

8.2 Contributions

Three new stiffness estimation algorithms are the key contributions of this thesis. They have been developed using two alternative strategies: model-based and sensor-based approaches. In the former case, the estimation algorithm is based on properties that can be analytically derived and demonstrated from a model describing the system behavior. The correctness of the estimation algorithms is ensured, provided the models can be safely assumed to properly represent the system under consideration. In the latter case, sensorial data has been used to provide adequate estimations, avoiding explicit modelling. Simulated interactions between the robot and the environment have been used to assess the behavior of the algorithms in nearly ideal conditions, with perfect linearization of rigid manipulator dynamics and perfect modeling of the environment by a linear spring. Simulation studies have also been used to explore the impact of mismatching contact parameters in the dynamic response, and to confirm the force tracking improvement obtained by performing online stiffness estimation and adaptation. Experimental results have been obtained at Institute of Systems and Robotics at the University of Coimbra, with a robotic platform based on a WAM manipulator including a JR3 force sensor and teleoperated through a Phantom haptic device. Explicit force control has been achieved by computed torque control in the task space and active observers. Interactions with both virtual and real objects have been conducted. In the former case, the environment parameters are specified by the user and are therefore well-known, allowing the estimation results to be compared with expected results. In the latter case, objects of distinct (but unknown) stiffnesses have been used.

1) ASBA

The ASBA algorithm is a model-based algorithm, based on a comparison between the actual AOB active state with analytically expected behavior, for different mismatch scenarios. The algorithm pre-computes step responses of the AOB active state, for different stiffnesses and mismatch scenarios. The

actual value of the active state is compared, during online operation, to the predicted response for several hypothetical mismatch scenarios. Stiffness has been estimated by finding the closest match, in terms of least-squared-error. Theoretical analysis and simulation results have demonstrated that stiffness is correctly estimated, for a broad range of stiffness values and mismatches. Simulation tests have been conducted for stiffnesses ranging from very soft (100 [N/m]) to very stiff (6000 [N/m]), including severe under and overestimated scenarios. Exact estimation has been obtained in as few as 4 samples from contact, although a trade-off between noise rejection and responsiveness is necessary, depending on the specific parameter choices. In underestimated systems, faster dynamics of the extra state allow the estimation to be obtained quicker, as it contains rich frequency components. Conversely, the behavior of the estimator in overestimated cases is comparatively slower, as a larger window of observation must be adopted. In any case, estimations close to the correct stiffness have been obtained. Parameterization of ASBA also allows a trade-off between memory and runtime overhead and estimation quality. Online suitability can be achieved by reducing the quantity and dimension of candidate curves analysed by the algorithm. On the other hand, if online overhead is not a concern, a higher memory footprint and execution time may be allowed for achieving better estimations. ASBA successfully achieves the goal of not requiring object position. Subsequent work was aimed at improving suitability for online operation, adaptive behavior, reducing implementation complexity and handling unmodeled disturbances.

2) COBA

The COBA algorithm improves online capabilities by reducing memory and time overhead. Stiffness is estimated by processing the force error of two force observers tuned with different nominal parameters. COBA has been subjected to theoretical analysis, simulation, experimental testing and validation. COBA is very efficient, as the runtime and memory overhead is only due to two additional observers. The estimation has been obtained by analysis of the force error of these observers. The implementation of COBA is straightforward and does not require any simplification of the relevant theoretical results. This has the benefit of ensuring consistent theoretical results. Responsiveness is also improved by not relying on LSE analysis. Low overhead and high responsiveness contribute significantly to the online suitability of COBA. Appropriate measures for compensating additional disturbances, caused by factors other than stiffness mismatch, have also been described and used. Results have demonstrated that COBA is able to provide accurate

stiffness, even in the presence of degraded force tracking in severe mismatch scenarios. Simulation tests in MATLAB Simulink allowed operation under ideal conditions to be assessed and the theoretical results to be confirmed. Experimental results, using both virtual and real objects, have provided the required experimental validation under practical conditions. Simulation tests have explored different scenarios with homogenous and heterogenous environments, that is, constant *versus* dynamic stiffness. Free-space/contact transitions have also been considered. COBA has been shown to converge to the correct stiffness value in just one time sample after a non-null force is applied to the environment, in both under and overestimated scenarios. Similar results have been obtained for both homogenous and heterogenous environments. The negative impact of online adaptation of the control laws in stiffness estimation capabilities is only residual. Similar performance has also been observed in simulations of free-space/contact transitions, without any stationary error or estimation spikes near the transition. Experimental results with virtual objects have been collected for the purpose of identifying the profile of the dominant disturbance in our system. This allowed proper compensation action to be determined. Good estimation behavior has been observed, with estimations converging to the expected value even with contact forces of very low magnitude. An acceptance criteria is described that allows poorly conditioned results to be disregarded. Interactions with real objects of unknown stiffness, including free-space/contact transition, have also been performed. In these, COBA has provided estimations matching the expected physical properties of the objects and actual force tracking performance. Free-space/contact transitions have been handled seamlessly, with consistent estimations being provided in different contact episodes with the same object, even though the force sensor bias changed between successive contact episodes. COBA results have been compared with those of a LSE-based technique that uses contact position data. The latter technique is negatively affected by contact position uncertainty, manifesting as estimation transients in free-space operation, large estimation spikes near contact and inconsistent estimations in successive contacts with the same object. COBA does not exhibit these problems, offering consistent estimations and good transient behavior in free-space and near the contact transition. Results of COBA reveal an interesting non-linear behavior of the stiffness, where the estimation gradually increases with applied force until it saturates at a final level. From this point onward, the estimation remains approximately constant even in the presence of increasing contact forces. In scenarios of contact with soft objects, the final estimation value is approximately obtained for inputs of magnitude between -2 [N] and -3 [N]. This

is notable as the SNR is very low in that force range. In stiffer scenarios, over 80% of the final estimation value is obtained for contact forces bellow -10 [N], approximately.

3) ANNE

Unlike the other techniques presented in this work, the ANNE estimator does not rely on a specific model of environment or control system. It is a sensor-based technique that relies on a neural network model that is trained, using sensorial data, to produce the desired estimations. A set of candidate input variables, based on force data, has been pruned by conducting a correlation analysis. This has allowed a minimal set of inputs with rich information content to be determined, to the benefit of ANN training and performance. Several neural networks have been trained, each using data obtained using a specific nominal stiffness value. In this way, each ANN specializes in estimating the stiffness in the neighborhood of its training stiffness. Results from multiple independent neural nets, trained with data from different stiffness mismatch scenarios, have been fused to generate a final estimation. ANNE has been trained and tested using both simulation and experimental data. Simulation results have demonstrated good performance of ANNE. It has been able to successfully generalize training results to novel input data, that has not been used earlier in the training process. A minimal stationary error of 2% has been detected, demonstrating good accuracy. Similarly, experimental results have also demonstrated good estimation performance of ANNE. Results from interactions with virtual objects have demonstrated that each individual neural network that composes ANNE offers good estimation performance by itself. However, their performance deteriorates slightly as the difference between the training stiffness and nominal stiffness increases. Analysis of the results has revealed that fusing the output of each individual neural net with polynomial interpolation improves the results of the individual nets, particularly when the nominal stiffness does not closely match one of the training stiffnesses. Experiments with data from interactions with real objects have confirmed these results. ANNE has closely replicated COBA estimations, even in interactions with objects that were not considered previously in the training process.

Comparative Analysis

Comparative analysis of the functional and performance profiles of the three algorithms suggests preferable target applications. ASBA offers a parameterizable trade-off between online overhead and estimation quality, making it a flexible technique well suited for a wide range of scenarios. However, lack of disturbance rejection may compromise estimation performance if system operation is significantly affected by disturbances other than stiffness mismatch. COBA has minimal online overhead and includes disturbance compensation, making it a prime candidate for online estimation tasks. Both COBA and ASBA are based on the availability of a system model and a specific control architecture. ANNE does not depend on such model, but only on sensorial data. It is therefore less dependent on the specificities of the control architecture, easing its adoption in other platforms. It requires a preliminary training process, including sensory data including stiffness estimations, that are used to seed the process. These may be obtained by simulation, the use of data from experiments in well known environments or the results of other stiffness estimators. Regardless of the source, ANNE is then able to generalize to other scenarios.

8.3 Future Directions

Several future research directions can be identified:

Estimation of Higher-Order Terms

In this work, the focus is on developing estimation techniques suitable for tasks with low dynamics. Stiffness is the dominant parameter in these tasks, so the effect of higher order terms can be considered negligible. In tasks with higher dynamics, this is not the case and higher order modeling may be necessary. Developing estimation techniques, independent of object position, for high order terms is a possible direction of future research.

Nonlinear Estimation

Non-linear contact models have been proposed in the literature, reported to be more accurate than linear impedance models, in particular in interactions with softer environments. Object-position independent techniques for estimating the parameters of such models can be of value, broadening the range of tasks were estimations based on high-fidelity contact models can be obtained.

Addressing Alternative Control Approaches

Some aspects of the model-based estimation techniques presented in this work, such as the use of the active state of the AOB, are based on specific details of the described control architecture. Other control approaches can be considered so that object-position independent techniques may be obtained. Ideally, a common framework should be developed, with minimal or no dependencies on the control approach, that may be individually tailored to each specific control system.

Appendix A

Theoretical Results

Contents

A.1 Introduction

This appendix collects all the theoretical proofs of theorems, corollaries and lemmas that support ASBA and COBA algorithms. Proofs of Theorem 1 and Lemma 1 of the ASBA Algorithm can be found in Section A.2. Proofs of theorems 2, 3 and 4, lemmas 2 and 3 as well as corollaries 1 and 2 of the COBA Algorithm are demonstrated in Section A.3. In all these proofs the initial conditions are assumed to be known.

A.2 Theoretical Results for ASBA

Lemma 1. *Assuming null initial conditions, the active state $\hat{p}_k$ of the active observer can be given by*

$$\hat{p}_k = KC \sum_{i=1}^{k} \sum_{j=0}^{i-1} Z_{j,i} \Gamma r_{i-1-j}, \tag{A.1}$$

where r, K, C and Γ are the reference force, the Kalman gains, the output and the command matrices, respectively. Z is represented by

$$Z_{j,k} = (\Phi_{n,k-1} + \Delta\Phi_{k-1} - \Gamma L)^j - (\Phi_{n,k-1} - \Gamma L)^j, \tag{A.2}$$

where L is the state feedback matrix, Φ_n is the nominal state transition matrix and $\Delta\Phi$ is the mismatch matrix given by the difference between the real and the nominal state transition matrices.

Proof of Lemma 1. From (3.28), $\hat{p}_k$ is represented by

$$\hat{p}_k = \hat{p}_{k-1} + K_{k,\hat{p}_k}(y_k - \hat{y}_k), \tag{A.3}$$

where $K_{k,\hat{p}_k}$ is the Kalman gain of the $\hat{p}_k$ term and y_k and $\hat{y}_k$ are the real and estimated system outputs (forces), respectively. Knowing from (3.16) and (3.29) that $y_k = Cx_k$ and $\hat{y}_k = C\hat{x}_k^-$, (A.3) becomes

$$\hat{p}_k = \hat{p}_{k-1} + K_{k,\hat{p}_k} C(x_k - \hat{x}_k^-), \tag{A.4}$$

where $\hat{x}_k^-$ is the *a-priori* estimation of x_k. Replacing x_k and $\hat{x}_k^-$ in (A.4) by the corresponding expressions (A.5) and (A.6),

$$x_k = (\Phi_{r,k-1} - \Gamma L)^k \, x_0 + \sum_{j=0}^{k-1} (\Phi_{r,k-1} - \Gamma L)^j \, \Gamma r_{k-1-j} \tag{A.5}$$

and

$$\hat{x}_k^- = (\Phi_{n,k-1} - \Gamma L)^k \, \hat{x}_0 + \sum_{j=0}^{k-1} (\Phi_{n,k-1} - \Gamma L)^j \, \Gamma r_{k-1-j}, \tag{A.6}$$

then, considering zero initial conditions ($x_0 = \hat{x}_0 = 0$), (A.4) becomes:

$$\hat{p}_k = \hat{p}_{k-1} + K_{k,\hat{p}_k} C \sum_{j=0}^{k-1} \left[(\Phi_{r,k-1} - \Gamma L)^j - (\Phi_{n,k-1} - \Gamma L)^j \right] \Gamma r_{k-1-j}. \tag{A.7}$$

A stiffness mismatch occurs when the nominal contact stiffness considered in the control design (K_{sn}) is different from the real system stiffness (K_s). This stiffness mismatch ($\Delta K_s = K_s - K_{sn}$) is reflected on a mismatch matrix $\Delta\Phi_k$ which is represented as $\Delta\Phi_k = \Phi_{r,k} - \Phi_{n,k}$. So, (A.7) becomes:

$$\hat{p}_k = \hat{p}_{k-1} + K_{k,\hat{p}_k} C \sum_{j=0}^{k-1} \left[(\Phi_{n,k-1} + \Delta\Phi_{k-1} - \Gamma L)^j - (\Phi_{n,k-1} - \Gamma L)^j \right] \Gamma r_{k-1-j}. \tag{A.8}$$

Analyzing (A.8), it is clear that $\hat{p}_k$ is dependant on the value of $\Delta\Phi_{k-1}$. If $\Delta\Phi_{k-1} = 0$ then $\hat{p}_k$ remains constant. Defining $Z_{j,k}$ as:

$$Z_{j,k} = (\Phi_{n,k-1} + \Delta\Phi_{k-1} - \Gamma L)^j - (\Phi_{n,k-1} - \Gamma L)^j, \tag{A.9}$$

and replacing (A.9) in (A.8):

$$\Delta\hat{p}_k = \hat{p}_k - \hat{p}_{k-1} = K_{k,\hat{p}_k} C \sum_{j=0}^{k-1} Z_{j,k} \Gamma r_{k-1-j}. \tag{A.10}$$

Since

$$\hat{p}_k = \hat{p}_0 + \sum_{j=1}^{k} \Delta\hat{p}_j, \tag{A.11}$$

then, for zero initial conditions, (A.8) is equivalent to:

$$\hat{p}_k = \sum_{i=1}^{k} K_{i,\hat{p}_k} C \sum_{j=0}^{i-1} Z_{j,i} \Gamma r_{i-1-j}. \tag{A.12}$$

For a constant Kalman gain ($K = K_{i,\hat{p}_k}$) (A.12) becomes:

$$\hat{p}_k = KC \sum_{i=1}^{k} \sum_{j=0}^{i-1} Z_{j,i} \Gamma r_{i-1-j}. \tag{A.13}$$

□

Theorem 1. *If $I^{K_{s,n},K_s}$ is the active state response for constant real and nominal stiffnesses (K_s and $K_{s,n}$ respectively) when the system is excited with a step input, then $\hat{p}_k$ can be computed, for any arbitrary reference input, through the convolution:*

$$\hat{p}_k = (I^{K_{s,n},K_s} * \Delta r)_k \tag{A.14}$$

where $\Delta r_k = r_k - r_{k-1}$ is the reference input difference.

Proof of Theorem 1. (A.14) will be proved to be equivalent to (A.1). Applying a unitary step input ($r_k = 1$, for $k > 0$) in equation (A.12), $I_k^{K_{s,n},K_s}$ can be computed by:

$$I_k^{K_{s,n},K_s} = \sum_{i=1}^{k} K_{i,\hat{p}_k} C \sum_{j=0}^{i-1} Z_{j,i}\Gamma. \tag{A.15}$$

Then, starting from (A.14),

$$\hat{p}_k = \left(I^{K_{sn},K_s} * \Delta r\right)_k = \sum_{n=0}^{k} I_n^{K_{sn},K_s} \Delta r_{k-n} \tag{A.16}$$

and substituting (A.15) in (A.16),

$$\hat{p}_k = \left(I^{K_{sn},K_s} * \Delta r\right)_k = \sum_{n=0}^{k} \left[\left(\sum_{i=1}^{n} K_{i,\hat{p}_k} C \sum_{j=0}^{i-1} Z_{j,i}\Gamma \right) (r_{k-n} - r_{k-n-1}) \right]. \tag{A.17}$$

Under constant K_{sn}, $K_{i,\hat{p}_k}$ will quickly converge to a stationary value after the initial transient phase. Therefore, it is reasonable to assume $K = K_{i,\hat{p}_k}$. Then, applying (A.17) and (A.1), (A.16) holds if:

$$\sum_{n=0}^{k} \left[\left(\sum_{i=1}^{n} KC \sum_{j=0}^{i-1} Z_{j,i}\Gamma \right) (r_{k-n} - r_{k-n-1}) \right] = KC \sum_{i=1}^{k} \sum_{j=0}^{i-1} Z_{j,i}\Gamma r_{i-1-j} \tag{A.18}$$

Referring to the 1st member of (A.18) as B^k:

$$B^k = \sum_{n=0}^{k} \left[\left(\sum_{i=1}^{n} KC \sum_{j=0}^{i-1} Z_{j,i}\Gamma \right) (r_{k-n} - r_{k-n-1}) \right] \tag{A.19}$$

and developing (A.19):

$$B^k = KC \sum_{n=0}^{k} \sum_{i=1}^{n} \left(\sum_{j=0}^{i-1} Z_{j,i}\Gamma r_{k-n} - \sum_{j=0}^{i-1} Z_{j,i}\Gamma r_{k-n-1} \right) \tag{A.20}$$

$$B^k = KC \sum_{n=0}^{k} \left(\sum_{i=1}^{n} \sum_{j=0}^{i-1} Z_{j,i}\Gamma r_{k-n} - \sum_{i=1}^{n} \sum_{j=0}^{i-1} Z_{j,i}\Gamma r_{k-n-1} \right) \tag{A.21}$$

Defining

$$A^i = \sum_{j=0}^{i-1} Z_{j,i}\Gamma, \tag{A.22}$$

the equivalent expression for (A.21) is obtained:

$$B^k = KC \sum_{n=0}^{k} \left(\sum_{i=1}^{n} A^i r_{k-n} - \sum_{i=1}^{n} A^i r_{k-n-1} \right). \tag{A.23}$$

Developing, with the individual terms

$$B^k = KC(\underbrace{A^1 r_{k-1} - A^1 r_{k-2}}_{n=1} + \underbrace{\sum_{i=1}^{2} A^i r_{k-2} - \sum_{i=1}^{2} A^i r_{k-3}}_{n=2} + \cdots + \underbrace{\sum_{i=1}^{k} A^i r_0 - \sum_{i=1}^{k} A^i r_{-1}}_{n=k}) \tag{A.24}$$

and grouping the terms with the same reference, (A.24) is equivalent to

$$B^k = KC(A^1 r_{k-1} + A^2 r_{k-2} + A^3 r_{k-3} + \cdots + A^k r_0). \tag{A.25}$$

Equation (A.25) can be represent as

$$B^k = KC \sum_{i=1}^{k} A^i r_{k-i}. \tag{A.26}$$

Replacing (A.22) in (A.26),

$$B^k = KC \sum_{i=1}^{k} \sum_{j=0}^{i-1} Z_{j,i}\Gamma r_{k-i}. \tag{A.27}$$

Replacing the first member of (A.18) by (A.27), then (A.18) can be represented as

$$KC \sum_{i=1}^{k} \sum_{j=0}^{i-1} Z_{j,i}\Gamma r_{k-i} = KC \sum_{i=1}^{k} \sum_{j=0}^{i-1} Z_{j,i}\Gamma r_{i-1-j}. \tag{A.28}$$

To prove that the equality of (A.28) is true, let us represent both sides of (A.28) in the form:

$$\sum_{i=0}^{\infty} a_i r_i = a_0 r_0 + a_1 r_1 + \cdots, \tag{A.29}$$

finding, after straightforward analysis, that the first term of (A.28) can be developed as

$$\sum_{i=1}^{k}\sum_{j=0}^{i-1} Z_{j,k} r_{k-i} = \sum_{i=0}^{\infty}\sum_{j=0}^{k-i-1} Z_{j,k} r_i. \tag{A.30}$$

Applying similar analysis to the second term of (A.28),

$$\sum_{i=1}^{k}\sum_{j=0}^{i-1} Z_{j,k} r_{i-1-j} = \sum_{i=0}^{\infty}\sum_{j=0}^{k-i-1} Z_{j,k} r_i. \tag{A.31}$$

$\therefore$ (A.30) and (A.31) prove (A.28) and therefore Theorem 1 is proved. □

A.3 Theoretical Results for COBA

Lemma 2, Corollary 1 and Theorem 2 are related to the analysis of COBA operation under ideal conditions. Lemma 3, Corollary 2 and Theorem 3 are the corresponding results for non-ideal operation. Theorem 4 presents conditions for FOB tuning.

A.3.1 Ideal Conditions

Lemma 2. *Under ideal operating conditions, there exists a value G_k, independent of ΔK_{sc} and equal for both FOBs, that verifies:*

$$\Delta x_{c,k} = -\Delta K_{sc}\, G_k, \tag{A.32}$$

where $\Delta x_{c,k}$ is the FOB state estimation error and ΔK_{sc} is the FOB stiffness mismatch.

Proof of Lemma 2. This theorem will be proved by mathematical induction.

Basis Step: (A.32) is proven to hold, for $k = 1$, with G_1 independent of ΔK_{sc} and equal for both FOBs. From (4.2), $x_{c,1}$ is given by

$$x_{c,1} = \Phi_c x_{c,0} + \Gamma u_{c,0} + K_c(y_1 - y_{c,1}^{-}). \tag{A.33}$$

Replacing Φ_c, $x_{c,0}$, y_1 and $y_{c,1}^{-}$ according to (4.8), (4.7), (3.16) and (4.4), (A.33) becomes

$$\begin{aligned} x_{c,1} \;=\; & (\Phi - \Delta\Phi_c)(x_0 - \Delta x_{c,0}) + \Gamma u_{c,0} \\ & + K_c C\,[\Phi x_0 - (\Phi - \Delta\Phi_c)(x_0 - \Delta x_{c,0})]. \end{aligned} \tag{A.34}$$

For known initial conditions ($\Delta x_{c,0} = 0$) and knowing, from (3.16), that $x_1 = \Phi x_0 + \Gamma u_0$, (A.34) becomes,

$$x_{c,1} = x_1 + (K_c C - I)\Delta\Phi_c \, x_0. \tag{A.35}$$

Applying (4.9) and (4.7), (A.35) is equivalent to,

$$\Delta x_{c,1} = -\Delta K_{sc}(K_c C - I)T x_0. \tag{A.36}$$

Making

$$G_1 = (K_c C - I)T x_0, \tag{A.37}$$

(A.36) can be written as

$$\Delta x_{c,1} = -\Delta K_{sc} G_1. \tag{A.38}$$

Since (A.37) is independent of ΔK_{sc} and equal for both FOBs then the basis step is proved.

Inductive Step: Assuming the inductive hypothesis for some $k > 0$, it is proven that it also holds for $k + 1$. This is equivalent to prove that

$$\Delta x_{c,k} = -\Delta K_{sc} \; G_k \Longrightarrow \Delta x_{c,k+1} = -\Delta K_{sc} \; G_{k+1}, \tag{A.39}$$

with G_{k+1} independent of ΔK_{sc} and equal for both COBs. Replacing in (4.2) the terms y_k, $y_{c,k}^-$, $x_{c,k}$, Φ_c by (3.16), (4.4), (4.7) and (4.8), respectively, $x_{c,k+1}$ becomes

$$\begin{aligned} x_{c,k+1} \;\; = \;\; & (\Phi - \Delta\Phi_c)(x_k - \Delta x_{c,k}) + \Gamma u_{c,k} + K_c C \Phi x_k \\ & - K_c C(\Phi - \Delta\Phi_c)(x_k - \Delta x_{c,k}). \end{aligned} \tag{A.40}$$

On the other hand, since

$$u_{c,k} = u_k, \tag{A.41}$$

then ${}^3\Delta x_{c,k} = 0$ and (4.7) becomes,

$$\Delta x_{c,k} = \left[\begin{array}{ccc} {}^1\Delta x_{c,k} & {}^2\Delta x_{c,k} & 0 \end{array} \right]^T. \tag{A.42}$$

Therefore, from (4.9) and (A.42),

$$\Delta\Phi_c \, \Delta x_{c,k} = 0. \tag{A.43}$$

From (3.16) and (A.43), (A.40) becomes

$$x_{c,k+1} = x_{k+1} + (K_c C - I)\Delta\Phi_c x_k + (K_c C - I)\Phi\Delta x_{c,k}. \tag{A.44}$$

Replacing $\Delta x_{c,k}$ by the induction hypothesis and applying (4.9), (A.44) becomes

$$x_{c,k+1} = x_{k+1} + \Delta K_{sc}(K_c C - I)(-\Phi G_k + T x_k). \tag{A.45}$$

Making

$$G_{k+1} = (K_c C - I)(-\Phi G_k + T x_k), \tag{A.46}$$

and applying (4.7), (A.45) can be rewritten as

$$\Delta x_{c,k+1} = -\Delta K_{sc} G_{k+1}. \tag{A.47}$$

From (3.17), (A.41) and (4.9), (A.46) can be written as

$$G_{k+1} = -(K_c C - I)\Phi G_k + \Gamma_u u_{c,k-1}, \tag{A.48}$$

where Γ_u is given by

$$\Gamma_u = (K_c C - I)Z, \tag{A.49}$$

being Z the third column of T. Since (A.48) is independent of ΔK_{sc} and equal for both FOBs, the inductive step is proved. □

Corollary 1. *Under ideal operating conditions,*

$$^1\hat{G}_k = {}^1G_k. \tag{A.50}$$

Proof of Corollary 1. From (4.21), the force estimation error of each FOB is given by

$$\Delta^1 x_{c_1,k} = -{}^1G_k \Delta K_{sc_1} \tag{A.51}$$

and

$$\Delta^1 x_{c_2,k} = -{}^1G_k \Delta K_{sc_2}. \tag{A.52}$$

Replacing (A.51) and (A.52) in (4.12) proves (A.50). □

Theorem 2. *Under ideal operating conditions, COBA estimates the correct system stiffness if* $^1G_k \neq 0$,

$$\hat{K}_{s,k} = K_s. \tag{A.53}$$

Proof of Theorem 2. Replacing (4.21), (4.6) and applying Corollary 1, (4.14) becomes

$$\hat{K}_{s,k} = K_s. \tag{A.54}$$

□

A.3.2 Non-ideal Conditions

Lemma 3. *Under non-ideal operating conditions ($\Delta u_{c,k} \neq 0$), there exists a value G_k and W_k, independent of ΔK_{sc} and equal for both COBs, that verifies:*

$$\Delta x_{c,k} = -\Delta K_{sc} G_k - W_k, \tag{A.55}$$

where $\Delta x_{c,k}$ is the FOB state estimation error and ΔK_{sc} is the FOB stiffness mismatch.

Proof of Lemma 3. This theorem will be proved by mathematical induction.

Basis Step: (A.55) is proven to hold, for $k = 1$, with G_1 and W_1 independent of ΔK_{sc} and equal for both COBs. From (4.2) and knowing that $u_{c,k} = \tilde{u}_k - \Delta u_{c,k}$, $x_{c,1}$ is given by

$$x_{c,1} = \Phi_c x_{c,0} + \Gamma(\tilde{u}_0 - \Delta u_{c,0}) + K_c(y_1 - y_{c,1}^-). \tag{A.56}$$

Replacing $x_{c,0}$, Φ_c, y_1 and $y_{c,1}^-$ according to (4.7), (4.8), (3.16) and (4.4),

$$\begin{aligned} x_{c,1} &= (\Phi - \Delta\Phi_c)(x_0 - \Delta x_{c,0}) + \Gamma(\tilde{u}_0 - \Delta u_{c,0}) \\ &\quad + K_c C\,(\Phi \Delta x_{c,0} + \Delta\Phi_c x_0 - \Delta\Phi_c \Delta x_{c,0}). \end{aligned} \tag{A.57}$$

Assuming known initial conditions, (A.57) becomes

$$x_{c,1} = x_1 - \Delta\Phi_c x_0 - \Gamma\Delta u_{c,0} + K_c C\,\Delta\Phi_c x_0. \tag{A.58}$$

Applying (4.9) and rearranging, (A.58) is equivalent to

$$x_{c,1} = x_1 + \Delta K_{sc}(K_c C - I)T x_0 - \Gamma\Delta u_{c,0}. \tag{A.59}$$

Making

$$G_1 = (K_c C - I)T x_0 \tag{A.60}$$

and

$$W_1 = -\Gamma\Delta u_{c,0}, \tag{A.61}$$

(A.59) can be presented as

$$x_{c,1} = x_1 + \Delta K_{sc} G_1 + W_1. \tag{A.62}$$

Applying (4.7), (A.62) becomes

$$\Delta x_{c,1} = -\Delta K_{sc} G_1 - W_1. \tag{A.63}$$

Since (A.60) and (A.61) are independent of ΔK_{sc} and are equal for both COBs, the basis step is proved.

Inductive Step: Assuming the inductive hypothesis for some $k > 0$, it is shown to also hold for $k + 1$. Considering an input error (4.23), (4.2) becomes

$$x_{c,k+1} = \Phi_c x_{c,k} + \Gamma(\tilde{u}_k - \Delta u_{c,k}) + K_c(y_{k+1} - y^-_{c,k+1}). \tag{A.64}$$

Replacing (3.16), (4.4), (4.7) and (4.8) in (A.64),

$$\begin{aligned} x_{c,k+1} &= x_{k+1} + \Delta K_{sc}(K_c C - I)T x_k - \Delta K_{sc}(K_c C - I)T\Delta x_{c,k} \\ &\quad +(K_c C - I)\Phi\Delta x_{c,k} - \Gamma\Delta u_{c,k}. \end{aligned} \tag{A.65}$$

Using the the induction hypothesis, (A.65) can be written as

$$\begin{aligned} x_{c,k+1} &= x_{k+1} + \Delta K_{sc}(K_c C - I)T x_k - \Delta K_{sc}(K_c C - I)T\Delta x_{c,k} \\ &\quad +(K_c C - I)\Phi(-\Delta K_{sc} G_k - W_k) + \Gamma\Delta u_{c,k}. \end{aligned} \tag{A.66}$$

Applying (4.7), (A.66) can be represented as

$$\Delta x_{c,k+1} = -\Delta K_{sc} G_{k+1} - W_{k+1}, \tag{A.67}$$

with

$$G_{k+1} = (K_c C - I)(T x_k - \Phi G_k - T\Delta x_{c,k}) \tag{A.68}$$

and

$$W_{k+1} = -(K_c C - I)\Phi W_k + \Gamma\Delta u_{c,k}. \tag{A.69}$$

From (3.17), (4.3), (4.7) and (4.9), (A.68) can be written as

$$G_{k+1} = -(K_c C - I)\Phi G_k + \Gamma_u u_{c,k}, \tag{A.70}$$

where Γ_u is defined by (A.49). From (A.69) and (A.70), W_{k+1} and G_{k+1} are independent of ΔK_{sc} and equal for both COBs, therefore the theorem is proved. □

Corollary 2. *Under non-ideal operating conditions,*

$$ {}^1\hat{G}_k = {}^1G_k. \tag{A.71} $$

Proof of Corollary 2. From (4.24), the force estimation error of each FOB is given by

$$ \Delta^1 x_{c_1,k} = -\Delta K_{sc_1}{}^1G_k - {}^1W_k \tag{A.72} $$

and

$$ \Delta^1 x_{c_2,k} = -\Delta K_{sc_2}{}^1G_k - {}^1W_k. \tag{A.73} $$

Replacing (A.72) and (A.73) in (4.12) proves (A.71). □

Theorem 3. *Under non-ideal operating conditions, the estimation computed by COBA is given by*

$$ \hat{K}_{s,k} = K_s + \Delta\hat{K}_{s,k}, \tag{A.74} $$

with

$$ \Delta\hat{K}_{s,k} = -\frac{{}^1W_k}{{}^1G_k}, \tag{A.75} $$

if ${}^1G_k \neq 0$. $\Delta\hat{K}_{s,k}$ *is the stiffness estimation error.*

Proof of Theorem 3. Replacing (4.24), (4.6) and applying Corollary 2, (4.14) becomes

$$ \hat{K}_{s,k} = K_s + \frac{{}^1W_k}{{}^1G_k}. \tag{A.76} $$

Defining $\Delta\hat{K}_{s,k} = K_s - \hat{K}_{s,k}$,

$$ \Delta\hat{K}_{s,k} = -\frac{{}^1W_k}{{}^1G_k}. \tag{A.77} $$

□

Theorem 4. *W_k is bounded if $\Delta u_{c,k-1}$ is bounded and the following conditions hold:*

$$|\lambda_1| < 1 \text{ and } |\lambda_2| < 1, \tag{A.78}$$

where

$$\lambda_1 = \alpha/2 - \sqrt{(\alpha^2/4 + \beta)} \tag{A.79}$$

and

$$\lambda_2 = \alpha/2 + \sqrt{(\alpha^2/4 + \beta)} \tag{A.80}$$

with

$$\alpha = {}^1K_c, \tag{A.81}$$

$$\beta = -2\,{}^1K_c{}^2K_c + {}^1K_c\,c + 2\,{}^2K_c a - c, \tag{A.82}$$

$$a = (1 - e^{-K_D T_s})/K_D \tag{A.83}$$

and

$$c = e^{-K_D T_s}. \tag{A.84}$$

Proof of Theorem 4. If $\Delta u_{c,k}$ is bounded, analysis of (A.69) reveals that W_k is also bounded if the eigenvalues of the transition matrix $-(K_c C - I)\Phi$ are within the unit circle. According to (4.5), ${}^3K_c$ is zero. Therefore, computing the eigenvalues of $-(K_c C - I)\Phi$, λ_0, λ_1 and λ_2,

$$\lambda_0 = 0, \tag{A.85}$$

$$\lambda_1 = \alpha/2 - \sqrt{\alpha^2/4 + \beta} \tag{A.86}$$

and

$$\lambda_2 = \alpha/2 + \sqrt{\alpha^2/4 + \beta}, \tag{A.87}$$

with

$$\alpha = {}^1K_c, \tag{A.88}$$

$$\beta = -2\,{}^1K_c{}^2K_c + {}^1K_c\,c + 2\,{}^2K_c a - c, \tag{A.89}$$

$$a = (1 - e^{-K_D T_s})/K_D \tag{A.90}$$

and

$$c = e^{-K_D T_s}. \tag{A.91}$$

If

$$|\lambda_1| < 1 \text{ and } |\lambda_2| < 1, \tag{A.92}$$

then (A.69) is bounded for a bounded input $\Delta u_{c,k}$ (in practical situations $\Delta u_{c,k}$ is bounded since system disturbances are bounded). □

Bibliography

[1] S. Abrate. *Impact on Composite Structures*. Cambridge University Press, Cambridge, 1998.

[2] C. An, C. Atkeson, and J. Hollerbach. *Model-Based Control Of A Robot Manipulator*. MIT Press, 1988.

[3] C. An and J. Hollerbach. The role of dynamic models in cartesian force control of manipulators. *International Journal of Robotics Research*, 8(4):51–72, 1989.

[4] K. Astrom and B. Wittenmark. *Computer-Controlled Systems: Theory And Design*. Prentice Hall, 1997.

[5] N. Bajçinca, R. Cortesão, and M. Hauschild. Robust control for steer-by-wire vehicles. *Autonomous Robots*, 19(2):193–214, 2005.

[6] P. Barkan. *Impact Design - Mechanical Design and Systems Handbook*. McGraw-Hill, 1974.

[7] J. Berrut and L. Trefethen. Barycentric lagrange interpolation. *SIAM Review*, 3(46):501–517, 2004.

[8] S. Bozic. *Digital and Kalman Filtering : An Introduction to Discrete Time Filtering and Optimum Linear Estimation*. Edward Arnold Publishers Ltd, 1979.

[9] R. Brach. *Mechanical Impact Dynamics: Rigid Body Collisions*. John Wiley & Sons, 1991.

[10] H. Bruyninckx and J. de Schutter. Specification of force controlled actions in the "task frame formalism" – a synthesis. *IEEE Transactions on Robotics and Automation*, 12:581–589, 1996.

[11] S. Chiaverini and L. Sciavicco. The parallel approach to force/position control of robotic manipulators. *IEEE Transactions on Robotics and Automation*, 9:361–373, 1993.

[12] S. Chiaverini, B. Siciliano, and L. Villani. Force/position regulation of compliant robot manipulators. *IEEE Transactions on Automatic Control*, 39:647–652, 1994.

[13] R. Christensen. *Theory of Viscoelasticity.* Academic Press, New York, 1982.

[14] P. Coelho and U. Nunes. Path-following control of mobile robots in presence of uncertainties. *IEEE Transactions on Robotics*, 21(2):252–261, 2005.

[15] R. Colbaugh, H. Seraji, and K. Glass. Direct adaptive impedance control of robot manipulators. *Journal of Robotic Systems*, 10:217–248, 1993.

[16] R. Cortesão. On kalman active observers. *Journal of Intelligent and Robotic Systems*, 48(2):131–155, 2007.

[17] R. Cortesão. *Kalman Techniques for Intelligent Control Systems: Theory and Robotic Experiments.* LAP LAMBERT Academic Publishing, 2010.

[18] R. Cortesao, R. Koeppe, U. Nunes, and G. Hirzinger. Explicit force control for manipulators with active observers. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)*, volume 2, pages 1075–1080, 2000.

[19] R. Cortesao, R. Koeppe, U. Nunes, and G. Hirzinger. Compliant motion control with stochastic active observers. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001)*, volume 4, pages 1876–881, 2001.

[20] R. Cortesão, J. Park, and O. Khatib. Real-time adaptive control for haptic telemanipulation with active observers. *IEEE Transactions on Robotics*, 3(5):987–999, 2006.

[21] F. Coutinho and R. Corteão. Environment stiffness estimation with multiple observers. In *Proc. of the Annual Conference of the IEEE Industrial Electronics Societ (IECON 2009)*, pages 153–1542, 2009.

[22] F. Coutinho and R. Corteão. Contact damping estimation for robotic tasks with candidate observers. In *Proc. of the Portuguese Conference on Automatic Control (CONTROLO 2010)*, 2010.

[23] F. Coutinho and R. Cortesão. Online stiffness estimation for compliant robotic manipulation. In *Proc. of the Portuguese Conference on Automatic Control (CONTROLO 2008)*, 2008.

[24] F. Coutinho and R. Cortesão. System stiffness estimation with the candidate observers algorithm. In *Proc. of the IEEE Mediterranean Conference on Control and Automation (MED 2010)*, pages 796–801, 2010.

[25] F. Coutinho and R. Cortesão. Adaptive stiffness estimation for compliant robotic manipulation using stochastic disturbance models. *International Journal of Systems Science*, 42(8):1241–1252, 2011.

[26] F. Coutinho and R. Cortesão. Comparison of position and force-based techniques for environment stiffness estimation in robotic tasks. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012)*, pages 4933–4938, 2012.

[27] F. Coutinho and R. Cortesão. Force-based stiffness estimation for robotic tasks. In *Proc. of the IEEE Conference on Decision and Control (CDC 2012)*, pages 2164–2170, 2012.

[28] J. Craig. *Adaptive Control of Mechanical Manipulators*. Phd thesis, UMI Dissertation Information Service, Ann Arbor, 1986.

[29] J. Craig. *Introduction To Robotics: Mechanics And Control*. Prentice Hall, 2005.

[30] G. Dahlquist. *Numerical Methods*. Prentice Hall, Englewood Cliffs, N.J, 1974.

[31] T. de Fazio, D. Seltzer, and D. Whitney. The instrumented remote center of compliance. *Industrial Robot: An International Journal*, 11(4):238–242, 1984.

[32] J. de Schutter and H. Van Brussel. Compliant robot motion i. a formalism for specifying compliant motion tasks. *International Journal of Robotics Research*, 7(4):3–17, 1988.

[33] C. de Wit, H. Olsson, K. Astrom, and P. Lischinsky. A new model for control of systems with friction. *IEEE Transactions on Automatic Control*, 40(3):419–425, 1995.

[34] C. de Wit, B. Siciliano, and G. Bastin. *Theory of Robot Control*. Springer, 1996.

[35] N. Diolaiti, C. Melchiorri, and S. Stramigioli. Contact impedance estimation for robotic systems. *IEEE Transactions on Robotics*, 21(5):925–935, 2005.

[36] S. Dubowsky and F. Freudenstein. Dynamic analysis of mechanical systems with clearances - part 1: formation of dynamical model. *Journal of Engineering for Industry*, 93:305–309, 1971.

[37] P. Dupont, C. Schulteis, P. Millman, and R. Howe. Automatic identification of environment haptic properties. *Presence: Teleoperators and Virtual Environments*, 8(4):394–411, 1999.

[38] S. Eppinger and W. Seerin. Introduction to dynamic models for robot force control. *IEEE Control Systems Magazine*, 7(2):48–52, 1987.

[39] D. Erickson, M. Weber, and I. Sharf. Contact stiffness and damping estimation for robotic systems. *International Journal of Robotics Research*, 22(1):41–57, 2003.

[40] K. Fu, R. Gonzales, and C. Lee. *Robotics: Control, Sensing, Vision and Intelligence*. McGraw-Hill, London, 1987.

[41] G. Gilardi and I. Sharf. Literature survey of contact dynamics modelling. *Mechanism and Machine Theory*, 37(10):1213–1239, 2002.

[42] W. Goldsmith. *Impact: The Theory and Physical Behavior of Colliding Solids*. Edward Arnold Publishers Ltd, London, 1960.

[43] A. Haddadi and K. Hashtrudi-Zaad. Real-time identification of hunt-crossley dynamic models of contact environments. *IEEE Transactions on Robotics*, 28(3):555–566, 2012.

[44] D. Haessig and B. Friedland. On the modeling and simulation of friction. *Journal of Dynamic Systems, Measurement, and Control*, 113(3):354–362, 1991.

[45] C. Hansen. *Rank-Deficient and Discrete III - Posed Problems: Numerical Aspects of Linear Inversion*. SIAM, 1998.

[46] H. Hertz. *Miscellaneous Papers*. Macmillan, London, 1896.

[47] N. Hogan. Impedance control: an approach to manipulation: parts i–iii. *ASME Journal of Dynamic Systems, Measurement, and Control*, 107:1–24, 1985.

[48] N. Hogan. On the stability of manipulators performing contact tasks. *IEEE International Journal of Robotics and Automation*, 4:677–686, 1988.

[49] J. Hopfield. Neural networks and physical systems with emergent computational abilities. *Proceedings of the National Academy of Sciences of the United States of America*, 79:2554–2558, 1982.

[50] W. Howard and V. Kumar. A minimum principle for the dynamic analysis of systems with frictional contacts. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA 1993)*, volume 3, pages 437–442, 1993.

[51] K. Hunt and F. Crossley. Coefficient of restitution interpreted as damping in vibroimpac. *Journal of Applied Mechanics*, 42:440–445, 1975.

[52] A. Isidori. *Nonlinear Control Systems*. Springer, 1995.

[53] W. Kalil. *Nonlinear Systems*. Macmillan, 1992.

[54] D. Karnopp. Computer simulation of stick-slip friction in mechanical dynamic systems. *Journal of Dynamic Systems, Measurement and Control*, 107:100–103, 1985.

[55] H. Kazerooni. Contact instability of the direct drive robot when constrained by a rigid environment. *IEEE Transactions on Automatic Control*, 35:710–714, 1990.

[56] H. Kazerooni, T. Sheridan, and P. Houpt. Robust compliant motion for manipulators, part i: the fundamental concepts of compliant motion. *IEEE International Journal of Robotics and Automation*, 2:83–92, 1986.

[57] R. Kelly, R. Carelli, M. Amestegui, and R. Ortega. Adaptive impedance control of robot manipulators. *International Journal of Robotics and Automation*, 4(3):134–141, 1989.

[58] O. Khatib. A unified approach for motion and force control of robots manipulators: the operational space formulation. *IEEE Journal on Robotics and Automation*, 3(1):45–53, 1987.

[59] R. Kikuuwe and T. Yoshikawa. Robot perception of environment impedance. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA 2002)*, volume 2, pages 1661–1666, 2002.

[60] B. Kim, S. Oh, I. Suh, and B. Yi. A compliance control strategy for robots manipulators under unknown environment. *KSME International Journal*, 14(10):1881–1888, 2000.

[61] S. Kim. *Contact Dynamics and Force Control of Flexible Multi-Body Systems*. PhD thesis, Department of Mechanical Engineering, McGill University, Montreal, 1999.

[62] P. Kraus and V. Kumar. Compliant contact models for rigid body collisions. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA 1987)*, volume 2, pages 1382–1387, 1987.

[63] H. Lankarani and P. Nikravesh. A contact force model with hysteresis damping for impact analysis of multi-body systems. *Journal of Mechanical Design*, 112:369–376, 1990.

[64] T. Lee and A. Wang. On the dynamics of intermittent-motion mechanism, part i: Dynamic model and response. *Journal of Mechanism, Transmissions and Automation in Design*, 105:534–540, 1983.

[65] C. Lim and W. Stronge. Oblique elastic–plastic impact between rough cylinders in plane strain. *International Journal of Engineering Science*, 1(37):97–122, 1999.

[66] L. Ljung. *System Identification, Theory for the User*. Prentice Hall, 2007.

[67] L. Love and W. Book. Environment estimation for enhanced impedance control. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA 1995)*, volume 2, pages 1854–1859, 1995.

[68] L. Love and W. Book. Force reflecting teleoperation with adaptive impedance control. *IEEE Transactions on Systems, Man, and Cybernetics*, 34(1), 2004.

[69] R. Lozano and B. Brogliato. Adaptive hybrid force position control for redundant manipulators. *IEEE Transactions on Automatic Control*, 37:1501–1505, 1992.

[70] J. Luh, M. Walker, and R. Paul. Resolved–acceleration control of mechanical manipulator. *IEEE Transations on Automatic Control*, 25(3):468–474, 1980.

[71] L. Luo and M. Nahon. A compliant contact model including interference geometry for polyhedral objects. *Journal of Computational and Nonlinear Dynamics*, 1:150–159, 2006.

[72] O. Ma. Contact dynamics modeling for the simulation of the space station manipulators handling payloads. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA 1995)*, volume 2, pages 1252–1258, 1995.

[73] O. Ma. Contact dynamics modelling for the simulation of the space station manipulators handling payloads. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA 1995)*, volume 2, pages 1252–1258, 1995.

[74] H. Maier and G. Dandy. Neural network based modelling of environment variables: A systematic approach. *Mathematical and Computer Modelling*, 33:669–682, 2001.

[75] D. Marhefka and D. Orin. A compliant contact model with nonlinear damping for simulation of robotic systems. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 29(6):566–572, 1999.

[76] M. Mason. Compliance and force control for computer controlled manipulators. *IEEE Transactions on Systems, Man and Cybernetics*, 11:418–432, 1981.

[77] N. McClamroch and D. Wang. Feedback stabilization and tracking of constrained robots. *IEEE Transactions on Automatic Control*, 33:419–426, 1988.

[78] K. Mirza, M. Hanes, and D. Orin. Dynamic simulation of enveloping power grasps. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA 1993)*, pages 430–435, 1993.

[79] S. Misra and A. Okamura. Environment parameter estimation during bilateral telemanipulation. In *Proc. of the IEEE Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (HAPTICS 2006)*, pages 301–307, 2006.

[80] J. Molina, A. Gonzalez, J. Coronado, J. Moran, and P. Gorce. A neural tactile force architecture applied to aeal-time stiffness estimation for a large scale of robotic grasping systems. *Journal of Intelligent and Robotic Systems*, 49, 2007.

[81] F. Nagata, T. Mizobuchi, T. Hase, Z. Haga, K. Watanabe, and M. Habib. Cad/cam-based force controller using a neural network-based effective stiffness estimator. *Artificial Life Robotics*, 15:101–105, 2010.

[82] J. Park. *Control Strategies for Robots in Contact.* Phd, Stanford University, 2006.

[83] J. Park and O. Khatib. A haptic teleoperation approach based on contact force control. *The International Journal of Robotics Research*, 25(5-6):575–591, 2006.

[84] J. Park and O. Khatib. *Robust Haptic Teleoperation of a Mobile Manipulation Platform*, volume 21, chapter Experimental Robotics, pages 543–554. STAR Springer Tracts in Advanced Robotics, 2006.

[85] J. Park and O. Khatib. Robot multiple contact control. *Robotica*, 26(5):667–677, 2008.

[86] M. Raibert and J. Craig. Hybrid position/force control of manipulators. *ASME Journal of Dynamic Systems, Measurement and Control*, 103:126–133, 1981.

[87] C. Richard. *On the Identification and Haptic Display of Friction.* PhD thesis, Stanford University, Department of Mechanical Engineering, 2000.

[88] F. Rosenblatt. The perceptron: a probablistic model for information storage and organization in the brain. *Psychological Review*, 65:386–408, 1958.

[89] E. Routh. *Dynamics of a System of Rigid Bodies.* Macmillan, London, 1904.

[90] J. Salisbury. Active stiffness control of a manipulator in cartesian coordinates. In *Proc. of the IEEE Conference on Decision and Control (CDC 1980)*, volume 19, pages 95–100, 1980.

[91] L. Santos and R. Cortesão. Joint space torque control with task space posture reference for robotic-assisted tele-echography. In *Proc. of the IEEE International Symposium on Robot and Human Interactive Communication (Ro-Man 2012)*, pages 126–131, 2012.

[92] R. Schilling. *Fundametals of Robotics: Analysis and Control*. Prentice Hall, 1989.

[93] L. Sciavicco and B. Siciliano. *Modeling and Control of Robot Manipulators*. McGraw-Hill, London, 1996.

[94] H. Seraji and R. Colbaugh. Force tracking in impedance control. *International Journal of Robotics Research*, 16:77–117, 1997.

[95] B. Siciliano and O. Khatib. *Handbook Of Robotics*. Springer, 2008.

[96] B. Siciliano and L. Villani. A passivity-based approach to force regulation and motion control of robot manipulators. *Automatica*, 32(3):443–447, 1996.

[97] B. Siciliano and L. Villani. *Robot Force Control*. Kluwer Academic Publishers, 1999.

[98] J. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, 1991.

[99] M. Spong and M. Vidyasagar. *Robot Dynamics And Control*. John Wiley & Sons, New York, 1989.

[100] S. Stramigioli. *Modeling and IPC Control of Interactive Mechanical Systems – A Coordinate Free Approach*. Springer, 2001.

[101] W. Stronge. Unraveling paradoxical theories for rigid body collisions. *Journal of Applied Mechanics*, 58:1049–1055, 1991.

[102] W. Stronge. *Impact Mechanics*. Cambridge University Press, Cambridge, 2000.

[103] R. Sutton and A. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[104] C. Tsaprounis and N. Aspragathos. Sliding mode with adaptive estimation force control of robot manipulators interacting with an unknown passive environment. *Robotica*, 17:447–458, 1999.

[105] A. Tyagi. *MATLAB and Simulink for Engineers*. Oxford University Press, 2012.

[106] D. Verscheure, I. Sharf, H. Bruyninckx, J. Swevers, and J. De Schutter. Identification of contact dynamics parameters for stiff robotic payloads. *IEEE Transactions on Robotics*, 25(2):240–252, 2009.

[107] D. Verscheure, I. Sharf, H. Herman Bruyninckx, J. Swevers, and J. De Schutter. Identification of contact parameters from stiff multi-point contact robotic operations. *The International Journal of Robotics Research*, 29(4):367–385, 2010.

[108] L. Villani, C. de Wit, and B. Brogliato. An exponentially stable adaptive control for force and position tracking of robot manipulators. *IEEE Transactions on Automatic Control*, 44:798–802, 1999.

[109] R. Volpe and P. Khosla. A theoretical and experimental investigation of impact control for manipulators. *International Journal of Robotics Research*, 12:351–365, 1993.

[110] M. Vukobratovic and V. Potkonjak. *Dynamics of Manipulation Robots: Theory and Application*. Springer-Verlag, Berlin, 1982.

[111] M. Vukobratovic and V. Potkonjak. Dynamics of contact tasks in robotics. *Mechanism and Machine Theory*, 34:923–942, 1999.

[112] X. Wang, X. Liu, D. Wang, B. Chebbi, and M. Meng. Design of bilateral teleoperators for soft environments with adaptive environmental impedance estimation. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA 2005)*, pages 1127–1132, 2005.

[113] Z. Wang, A. Peer, and M. Buss. Fast online impedance estimation for robot control. In *Proc. of the IEEE International Conference on Mechatronics (ICM 2009)*, pages 1–6, 2009.

[114] M. Weber, O. Ma, and I. Sharf. Identification of contact dynamics model parameters from constrained robotic operations. In *Proc. of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE 2012)*, pages 1–10, 1995.

[115] J. Wen and S. Murphy. Stability analysis of position and force control for robot arms. *IEEE Transactions on Automatic Control*, 36:365–371, 1991.

[116] L. Whitcomb, S. Arimoto, T. Naniwa, and F. Ozaki. Adaptive model-based hybrid control of geometrically constrained robots. *IEEE Transactions on Robotics and Automation*, 13:105–116, 1997.

[117] D. Whitney. Force feedback control of manipulator fine motions. *ASME Journal of Dynamic Systems, Measurement and Control*, 99:91–97, 1977.

[118] E. Whittaker. *A Treatise on the Analytical Dynamics of Particles and Rigid Bodies*. Cambridge University Press, Cambridge, 1904.

[119] T. Yamamoto, M. Bernhardt, A. Peer, M. Buss, and A. Okamura. Techniques for environment parameter estimation during telemanipulation. In *Proc. of the IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob 2012)*, pages 217–223, 2008.

[120] B. Yao, S. Chan, and D. Wang. Unified formulation of variable structure control schemes for robot manipulators. *IEEE Transactions on Automatic Control*, 39:371–376, 1992.

[121] T. Yoshikawa. Dynamic hybrid position/force control of robot manipulators – description of hand constraints and calculation of joint driving force. *IEEE Journal of Robotics and Automation*, 3:386–392, 1987.

[122] W. Zarrad, P. Poignet, and R. Cortesão. Haptic feedback control in medical robotics trough stiffness estimation with extended kalman filter. In *Proc. of the IEEE International Conference on Advanced Robotics (ICAR 2007)*, pages 81–87, 2007.