UNIVERSITÀ DELLA CALABRIA

Dipartimento di Ingegneria Meccanica, Energetica e Gestionale

Dottorato di Ricerca in Ingegneria Civile e Industriale
Ciclo XXXV

Development of advanced strategies for modeling, design and virtual sensing of multimaterial metallic-composite robots

Settore Scientifico Disciplinare ING/IND 13

Coordinatore: Ch.Mo Prof. Enrico Conte

Firma: _____

Supervisore/Tutor: Ch.Mo Prof. Giuseppe Carbone

Firma: _____

Co-supervisore: Dr. Tomma[redacted]

Firma: _____

Dottorando: Dott. Cristian Enrico Capalbo

Firma: [redacted]

La borsa di dottorato è stata cofinanziata con risorse del Programma Operativo Regionale Calabria FSE/FESR 2014 – 2020 (CCI 2014IT16M2OP006)

Acknowledgments

"And once the storm is over you won't remember how you made it through, how you managed to survive. You won't even be sure, in fact, whether the storm is really over. But one thing is certain. When you come out of the storm you won't be the same person who walked in. That's what this storm's all about."

- Haruki Murakami

The knowledge one acquires in scientific research has the beautiful and frightening effect of magnifying doubts and uncertainties. Now that this storm called PhD is almost over, the changes it brought to my life are the only certainty I have. It has been an amazing journey during which I grew up immensely, both from a personal and professional point of view. I survived the storm and arrived to the safe haven of these mature research results only thanks to the amazing people who shared the path with me.

First of all I want to thank my supervisor Prof. Carbone, for giving me this opportunity and for always providing precious insight on the intricate academic world. My gratitude goes also to my co-supervisor Tommaso. Each of our conversations teaches me what integrity really is, and how to be utterly honest and direct. The way you put a great technical knowledge at the service of an endless source of ideas is an example for me.

My very first day of PhD was marked by a request that I struggled with at first: "From now on, I am only Domenico for you". The reason for that request was soon clear: you treat me and each of your collaborators as equals, while putting all the effort you can in making sure that our careers take the best direction. Thank you, Domenico, for always caring about my work and for making all of this possible.

Regardless of my beloved books and papers, the most valuable source of knowledge has been a much more interesting one. Daniel, thank you! You are the single person I learned the most from. I will forever treasure our long technical discussions and the moments of simple conversation. You have been my guide through the research and industry worlds.

My gratitude goes to all the beautiful people of the DMS Lab at Unical: Rocco, Walter, Francesco, Michele, Anna. Sharing knowledge (and food) with you has helped me expanding my point of view. A big contribution (most but not only in terms of caffeine and glucose) came also from the other side of the corridor, from the technology group: David, Romina, Francesco G., Giuseppe, Francesco B., Michela, Carmine and Giusy. Thank you all for the endless coffee breaks and the nice talks.

Each of my days at Siemens has been made amazing by a great group who made me feel like I belonged there from the very start. Thanks to Onur, Ali, Roberta, Pavel, Matijs, Daeyun and Silvia.

This research path gave me the opportunity to find a place that I consider home: Leuven. This beautiful little town gifted me so many special moments, endless opportunities to face different cultures and stimulating ideas. Every corner of that city carries a memory now. Most importantly, what makes it really special are the beautiful people I crossed ways with, be it for a night or for months.

Leuven gave me the chance to meet incredible people who started as colleagues but soon became friends first. Thank you Davide, for sharing most of this path with me and for being a careful and amazing friend. I know I will always be able to rely on you for a deep conversation and a great plate of pasta. In my very first week at Siemens, a peculiar dude with a colorful tie popped in the office to introduce, and I already there understood what an incredible person he was. Thank you Filippo for always bringing your energy and creativity wherever you are, for all the parties and most important for making me discover Bibimbap.

While I am sure I will forget someone, I still want to name those people who shared so many amazing memories with me in Leuven: Maria, Yesim, Lorenzo, Lorenzo, Lorenzo (come on guys...), Andrea, André, Huma, Lorenzo (yes I know...), and all the Waterview community.

I'm lucky enough to have friends who are still with me regardless of the distance and me never showing up even when I was physically close. It was never easy to turn down an invitation from you, but I treasure each and every of those times we are together. Thanks to Giuseppe, Enrico, Simone, Marco, Francesco, Alessia, Mariafrancesca, Alessandro and Domenico. Thanks to Angelica and Damiano, my favorite cretini. And thanks to Ingrid for always being supportive despite the huge distance.

Finally, I would not have become who I am today without my family. I always found in my aunt Anna a reference point, an anchor. Knowing that you believe in me and that you were always there for me has been the biggest support. Thanks to Luigi for the precious advice and for always be there in the important moments.

The most important of all, the person who gave everything for me, is my mother. I am well aware of the sacrifices you made for me to achieve results such as this PhD and the many more that will come. I want to apologize for the many moments in which I let tension and stress overcome the feelings I have for you. And nonetheless, you were always there. All I am is for you.

Ringraziamenti

"Quando la tempesta sarà finita, probabilmente non saprai neanche tu come hai fatto ad attraversarla e a uscirne vivo. Anzi, non sarai neanche sicuro se sia finita per davvero. Ma su un punto non c'è dubbio. Ed è che tu, uscito da quel vento, non sarai lo stesso che vi è entrato. Questo é il vero significato della tempesta."

- Haruki Murakami

La conoscenza che si acquisisce con la ricerca scientifica ha l'effetto meraviglioso e spaventoso di ingigantire dubbi ed incertezze. Ora che questa tempesta chiamata PhD è quasi finita, i cambiamenti che ha portato nella mia vita sono le uniche certezze che ho. È stato un viaggio incredibile, nel quale sono cresciuto immensamente sia dal punto di vista professionale che da quello personale. Sono sopravvissuto alla tempesta per arrivare al porto sicuro di questa tesi solo grazie alle persone meravigliose che hanno condiviso il percorso con me.

Prima di tutto voglio ringraziare il mio supervisore, Prof. Carbone, per avermi dato questa opportunità e per i sempre preziosi suggerimenti sull'intricato mondo accademico. La mia gratitudine va anche al mio co-supervisore, Tommaso. Ogni nostra conversazione mi insegnata il vero significato della parola integrità, e come essere sempre onesti e diretti. Il modo in cui metti una enorme conoscenza tecnica a disposizione di una fonte inesauribile di idee è un esempio per me.

Il mio primo giorno di Dottorato è stato segnato da una richiesta con cui inizialmente ho avuto molta difficoltà: "Da oggi, sono solo Domenico per te". La ragione è diventata subito chiara: tu tratti me e tutti i tuoi collaboratori da pari mentre fai ogni sforzo possibile per assicurarti che le nostre carriere prendano la direzione migliore. Grazie, Domenico, per esserti sempre interessato al mio lavoro e per aver reso tutto ciò possibile. Nonostante i miei amati libri ed articoli, la fonte di conoscenza piú preziosa è stata un'altra, molto piú interessante. Daniel, grazie! Sei la persona da cui ho imparato di più. Ho fatto tesoro di ogni nostra discussione tecnica e dei momenti di semplice conversazione. Sei stato la mia guida nel mondo della ricerca e dell'industria.

La mia gratitudine va a tutte le bellissime persone del DMS Lab all'Unical: Rocco, Walter, Francesco, Michele ed Anna. Condividere le conoscenza (e tanto cibo) con voi mi ha aiutato ad espandere i miei punti di vista. Un grande contributo (non solo in termini di caffeina e glucosio) è arrivato anche dall'altro lato del corridoio, dal gruppo dei Tecnologi: David, Romina, Francesco G., Giuseppe, Francesco B., Michela, Carmine e Giusy. Grazie per le infinite pause caffè e le chiacchiere.

Ognuno dei miei giorni in Siemens è stato reso speciale da un gruppo di colleghi che mi hanno fatto sentire parte integrante fin dal primo giorno. Grazie a Onur, Ali, Roberta, Pavel, Matijs, Daeyun e Silvia.

Questo percorso mi ha permesso di scoprire un posto che ora per me è casa: Leuven. Questa piccola ed affascinante cittá mi ha regalato momenti speciale ed infinite occasioni di confrontarmi

con culture diverse ed idee stimolanti. Ogni angolo della cittá porta con se un ricordo. Ancora piú importante, ció che la rende davvero speciale sono le belle persone che hanno incrociato il proprio percorso con il mio, per una notte o per mesi. Leuven mi ha dato l'opportunitá di conoscere persone incredibili che hanno iniziato come colleghi ma sono presto diventati amici prima di tutto. Grazie Davide, per aver condiviso la maggior parte di questo percorso con me e per essere un amico straordinario ed attento. So che potró sempre contare su di te per una conversazione profonda ed un buon piatto di pasta. Nella mia prima settimana in Siemens, un tipo decisamente particolare con una cravatta colorata è comparso nel mio ufficio per presentarsi, ed ho subito capito quanto fosse speciale. Grazie, Filippo, per l'energia e la creativitá che porti dovunque vai, per tutte le feste e per avermi fatto scoprire il Bibimbap.

Sono sicuro che dimenticheró qualcuno, ma voglio comunque ringraziare le persone che hanno reso speciali i miei ricordi di Leuven: Maria, Yesim, Lorenzo, Lorenzo, Lorenzo (poca fantasia...), Andrea, André, Huma, Lorenzo (lo so...), e tutta la comunitá di Waterview.

Sono fortunato ad avere amici che sono ancora vicino a me nonostante la distanza e la mia abitudine a sparire anche quando sono vicino. Rifiutare una vostra proposta non è mai facile, ma faccio tesoro di ogni momento in cui siamo insieme. Grazie a Giuseppe, Enrico, Simone, Marco, Francesco, Alessia, Mariafrancesca, Alessandro e Domenico. Grazie ad Angelico e Damiano, i miei cretini preferiti. E grazie ad Ingrid per avermi sempre supportato nonostante la distanza.

In fine, non sarei diventato la persona che sono oggi senza la mia famiglia. Ho sempre visto in mia zia Anna un punto di riferimento, un'ancora. Sapere che credi in me a che sei sempre lí per me è stato il piú grande supporto che potessi avere. Grazie a Luigi per i sempre preziosi consigli e per essere sempre presente nei momenti importanti.

Piú importante di qualsiasi altra cosa, la persona che ha dato tutto per me, è mia madre. Sono consapevole dei sacrifici perché io raggiungessi questo risultato e tutti quelli futuri. Voglio chiederti scusa per tutti i momenti in cui ho lasciato che la tensione e lo stress vincessero i sentimenti che ho per te. E nonostante tutto, tu ci sei sempre stata. Tutto ció che sono è per te.

Abstract

Robotic manipulators with lightweight design are becoming increasingly popular given their superior mobility, ease of setup and reduced dangers in case of collisions with humans. In contrast with what traditionally done for bulky manipulators, design of lightweight manipulators requires to take compliance of the components into account. Compliance should in fact be minimized in order to ensure accuracy, while keeping energy usage as low as possible to ensure efficient actuation. The requirements for energy efficiency and stiffness are often contrasting and difficult to address with traditional materials and design methodologies. Great performance improvements can be achieved employing advanced materials such as composites, thanks to their low specific weight and high stiffness. Lightweight manipulators made of composite materials would allow for better dynamic performances, improved safety and reduction of actuation power. The use of composites in robotic applications has been particularly limited both in research and industry, mainly due to the high cost and complex design choices required to avoid damage during machining. Hybrid multimaterial designs have been proposed in literature to solve these issues, with few components in composite materials and the most complex ones in traditional materials.

The design of multimaterial mechanical systems results in a complex challenge. Given a set of requirements, their dependency on design parameters is often highly nonlinear and complicated to extract. Furthermore, the solutions require high quality while satisfying numerous and often contrasting requirements. Simulation based on numerical models can predict the performances and in turn enable automated design methods based on optimization algorithms. The numerical models employed in the design phase can then be updated based on experimental data to closely match any physical instance of the systems, constituting a Digital Twin (DT). The availability of a DT enables in turn applications in the operation phases such as Virtual Sensing (VS), which allows to indirectly estimate quantities that can result difficult, expensive or even impossible to directly measure. Models employed for optimization and VS need to satisfy two often contrasting requirements: high accuracy and computational efficiency. Common industrial-level numerical models require large sizes in order to guarantee accuracy, resulting in inefficient simulation and model updating procedures.

This work aims at the development of accurate and efficient numerical modeling strategies for multimaterial mechanical systems that enable design optimization and VS applications. The systems on which the methodologies are applied are multimaterial robotic manipulators in which composite materials are employed to achieve high performance and energy savings.

The work initially focuses on modeling strategies based on the Finite Element (FE) and Flexible MultiBody (FMB) methodologies. The use of component-level parametric Model Order Reduction (pMOR) techniques allows to define component-level and system-level models with high accuracy and efficiency both in terms of performance evaluation and design update. The proposed methodology is validated showing good results, with particular focus on the model of a 5-DOF robotic manipulator. A multimaterial version of the robotic manipulator is then designed through a MultiObjective Optimization (MOO) technique. The use of the developed efficient system-level models proves fundamental in this phase to grant efficiency and accuracy. The results demonstrate the gains in terms of maneuver accuracy and energy consumption resulting from the use of composite materials. The final part of the thesis employs the models for VS applications based on the Kalman Filter (KF) framework for the joint estimation of states, inputs and material parameters. The methodology is developed both for component-level and system-level applications, allowing to track the evolution of the system in time through a limited set of output-only measurements. A wide set of numerical and experimental validations shows good results.

Sommario

I manipolatori robotici dal design *leightweight* stanno diventando sempre più popolari, grazie alla loro superiore mobilità, alla facilità di installazione e alla riduzione dei danni in caso di collisione con umani. A differenza di quanto avviene per i manipolatori tradizionalmente rigidi ed ingombranti, la progettazione di manipolatori leggeri richiede di tenere conto della deformabilità dei componenti. La cedevolezza deve infatti essere ridotta al minimo per garantire un'adeguata accuratezza, mantenendo il consumo di energia il più basso possibile per assicurare una attuazione efficiente. I requisiti di efficienza energetica e rigidezza sono spesso contrastanti e difficili da soddisfare con materiali e metodologie di progettazione tradizionali. È possibile ottenere grandi miglioramenti delle prestazioni utilizzando materiali innovativi come i compositi, grazie al loro basso peso specifico e all'elevata rigidezza. Manipolatori realizzati in materiali compositi consentirebbero migliori prestazioni dinamiche, maggiore sicurezza e riduzione della potenza di attuazione. L'uso dei compositi in applicazioni robotiche è stato particolarmente limitato sia nella ricerca che nell'industria, soprattutto a causa del costo elevato e delle complesse scelte progettuali necessarie per evitare danni durante la lavorazione. Questi problemi sono stati superati in letteratura tramite sistemi ibridi multimateriale, con pochi componenti in materiali compositi e quelli più complessi in materiali tradizionali.

La progettazione di sistemi meccanici multimateriale rappresenta una sfida complessa. Dato un insieme di requisiti, la loro dipendenza dai parametri di progettazione è spesso altamente non lineare e complicata da estrarre. Inoltre, le soluzioni richiedono un'elevata qualità, pur soddisfacendo requisiti numerosi e spesso contrastanti. Simulazioni basate su modelli numerici possono prevedere le prestazioni e, a sua volta, abilitare metodi di progettazione automatizzati basati su algoritmi di ottimizzazione. I modelli numerici utilizzati nella fase di progettazione possono poi essere aggiornati sulla base di dati sperimentali per corrispondere fedelmente a qualsiasi istanza fisica dei sistemi, costituendo un Digital Twin (DT). La disponibilità di un DT consente a sua volta applicazioni nelle fasi operative come il Virtual Sensing (VS), che permette di stimare indirettamente quantità che possono risultare difficili, costose o addirittura impossibili da misurare direttamente. I modelli utilizzati per l'ottimizzazione e il VS devono soddisfare due requisiti spesso contrastanti: alta precisione ed efficienza computazionale. I modelli numerici comunemente usati a livello industriale richiedono grandi dimensioni per garantire l'accuratezza, con conseguente inefficienza delle procedure di simulazione e di aggiornamento del modello.

Questo lavoro mira allo sviluppo di strategie di modellazione numerica accurate ed efficienti per sistemi meccanici multimateriali, che consentano l'ottimizzazione del design e le applicazioni di VS. I sistemi su cui vengono applicate le metodologie sono manipolatori robotici multimateriale in cui sono usati materiali compositi per ottenere elevate prestazioni e risparmio energetico.

Il lavoro si concentra inizialmente sulle strategie di modellazione basate sulle metodologie Finite Element (FE) e Flexible MultiBody (FMB). L'uso di tecniche di parametric Model Order Reduction (pMOR) a livello del singolo componente consente poi di definire modelli, sia dei componenti che del sistema, accurati ed efficienti sia in termini di valutazione delle prestazioni che di aggiornamento del design. La metodologia proposta viene validata con buoni risultati usando il modello di un manipolatore robotico a 5 gradi di libertà. Una versione multimateriale del manipolatore robotico viene quindi progettata attraverso una tecnica di ottimizzazione multiobiettivo (MOO). L'uso dei modelli efficienti sviluppati si rivela fondamentale in questa fase per garantire efficienza e accuratezza. I risultati dimostrano i guadagni in termini di precisione nelle manovre e di consumo energetico derivanti dall'uso di materiali compositi. La parte finale della tesi impiega i modelli per applicazioni di VS basate sul Kalman Filter (KF) per la stima congiunta di stati, input e parametri dei materiali. La metodologia è stata sviluppata sia per applicazioni a livello di componente che di sistema, consentendo di seguire l'evoluzione del sistema nel tempo basandosi su un set limitato di misure. Un'ampia serie di validazioni numeriche e sperimentali mostrano buoni risultati.

Contents

List of Figures

List of Tables

List of Symbols

Notation

$\dot{\square}$	First time derivative
$\ddot{\square}$	Second time derivative
$\tilde{\square}$	Skew-symmetric matrix
$\|\cdot\|$	L^2 norm
$\mathbf{I}$	Identity matrix
$\mathbf{0}$	Zeros matrix
$\mathbf{1}$	Ones matrix
$\overline{\square}$	Maximum value of a time signal
$\nabla\square$	Gradient operator

Generic

Δt	Integration time step
H	FRF
ω	Frequency
e	Error
$\mathbf{X}$	Generic matrix
$\mathfrak{I}$	Inertia tensor

Materials modeling

$\boldsymbol{\sigma}$	Stress tensor
$\boldsymbol{\epsilon}$	Strain tensor
$\mathbf{E}$	Constitutive matrix
E	Young's modulus
ν	Poisson's ratio
λ, μ	Lame's parameters
α, β	Rayleigh damping parameters
Θ	Fibers angle for laminate
h	Layer thickness
$\overline{\mathbf{E}}$	Lamina constitutive matrix in relative frame of reference
$\mathbf{T}$	Rotation matrix used for composites constitutive relation
n_l	Number of layers

Finite Elements - full order model

$\square_e$	Relative to element
$\mathbf{q}_f$	Nodal DOFs
$\mathbf{M}_f$	FE full mass matrix
$\mathbf{C}_f$	FE full mass matrix
$\mathbf{K}_f$	FE full mass matrix
$\mathbf{S}_u$	FE loads shape matrix
$\mathbf{N}_e$	Element shape functions
$\mathbf{B}_e$	Element strain-displacement matrix
m	Mass of the mesh
n_{nodes}	Number of nodes in the mesh
n_{DOF}	Number of DOFs of the mesh
$\mathbf{S}_e$	Element connectivity matrix
n_{DOF}	Number of nodes in the mesh
$\mathbf{\Psi}$	Reduction basis
$\mathbf{q}_{f_r}$	Reduced DOFs
n_{red}	Number of reduced DOFs
$\mathbf{M}_{f_r}$	FE reduced mass matrix
$\mathbf{C}_{f_r}$	FE reduced mass matrix
$\mathbf{K}_{f_r}$	FE reduced mass matrix
$\mathbf{S}_{u_r}$	FE reduced loads shape matrix
$\mathbf{p}$	Parameters set
$\mathbf{\Omega}$	Parameters space
$\mathbf{y}$	Measurements set
$\mathbf{h}$	Measurements equation
$\boldsymbol{\vartheta}$	Measurement direction
$\mathbf{N}_s$	Strain measurement basis matrix
$\mathbf{N}_d$	Displacement measurement shape matrix

Multibody

$\mathbf{q}$	Coordinates
n_{sys}	Number of DOFs of the system
$\mathbf{q}_R$	Position rigid coordinates
$\mathbf{q}_\Theta$	Rotation rigid coordinates
$\mathbf{v}$	Velocities
$\mathbf{M}$	System mass matrix
$\mathbf{C}$	System damping matrix
$\mathbf{K}$	System stiffness matrix
$\mathbf{\Phi}$	Constraints
$\mathbf{\Phi}_q$	Constrains Jacobian
n_c	Number of constraints
$\boldsymbol{\lambda}$	Lagrange multipliers
$\boldsymbol{\mu}$	Extra Lagrange multipliers for GGL formulation
$\mathbf{f}_e$	Generalized external forces
$\mathbf{f}_v$	Quadratic velocity term
$\hat{\mathbf{M}}$	Mass invariants
d	Trajectory deviation with respect to ideal one

Optimization

n_{ojb}	Number of objective functions
n_{con}	Number of constraints
n_p	Number of parameters
f^{obj}	Objective function
f^{con}	Constraint function
ϱ	Paerson correlation coefficient

Kalman Filter

$\mathbf{x}$	System states
$\mathbf{A}_c$	Continuous state-update matrix
$\mathbf{B}_c$	Continuous input matrix
$\mathbf{H}$	Output matrix
$\mathbf{D}$	Feedthrough matrix
$\mathbf{A}$	Discrete state transition matrix
$\mathbf{B}$	Discrete input matrix
$\mathbf{w}$	Additive noise
$E[\square]$	Expected value operator
$\mathbf{Q}$	States covariance matrix
$\mathbf{R}$	Measurements covariance matrix
$\square^+$	A-priori estimate
$\square^-$	A-posteriori estimate
$\mathbf{P}$	Error covariance matrix
$\boldsymbol{\mathcal{K}}$	Kalman gain
$\mathbf{f}_c$	Continuous non-linear time-update equation
$\mathbf{h}$	Measurements equation
$\mathbf{f}_c$	Discrete non-linear time-update equation
$\mathbf{g}_d$	Discrete implicit time-update equation
$\mathbf{J}$	States Jacobian
$\mathbf{J}_m$	Measurements Jacobian
$\square^*$	Relative to augmented model
$\mathbf{u}^{un}$	Unknown inputs
$\mathbf{u}^{kn}$	Known inputs
$\mathbf{B}_{un}$	Unknown input matrix
$\mathbf{B}_{kn}$	Known input matrix
a	Expected amplitude value of an unknown estimated quantity
δ_p	Scaling factor to set covariance of unknown parameters
$\boldsymbol{\mathcal{N}}$	Constraints equation nullspace

List of Acronyms

AAE	Averaged Absolute Error
AKF	Augmented Kalman Filter
BDF	Backward Differentiation Formula
CF	Constraint Function
CFRP	Carbon Fiber Reinforced Polymer
CLT	Classical Lamination Theory
CM	Condition Monitoring
CMS	Component Modes Synthesis
DAE	Differential Algebraic Equation
DKF	Dual Kalman Filter
DOF	Degree Of Freedom
DV	Design Variable
EKF	Extended Kalman Filter
EOM	Equation Of Motion
ESLT	Equivalent Single Layer Theories
FE	Finite Elements
FFR	Floating Frame of Reference
FMB	Flexible MultiBody
FOM	Full Order Model
FRF	Frequency Response Function
GGL	Gear-Gupta-Leimkuhler
KF	Kalman Filter
LHS	Latin Hypercube Sampling
LSV	Left Singular Vector
MAC	Model Assurance Criterion
MB	MultiBody
MOO	Multi-Objective Optimization
MOR	Model Order Reduction
MRA	Maximum Response Amplitude
ND	Non-Dominated
ODE	Ordinary Differential Equation
OF	Objective Function
OSP	Optimal Sensors Placement
OSS	Optimal Sensors Selection

PBH	Popov-Belevitch-Hautus
PF	Pareto Front
pFMB	parametric Flexible MultiBody
pMOR	parametric Model Order Reduction
pROM	parametric Reduced Order Model
RB	Rigid Body
ROM	Reduced Order Model
SPKF	Sigma-Point Kalman Filter
SS	State Space
SHM	Structural Health Monitoring
SV	Singular Value
SVD	Singular Value Decomposition
UD	UniDirectional
VS	Virtual Sensing

Introduction

Chapter 1

Introduction

This introductory chapter starts by discussing context and motivations of the thesis in Section 1.1. Based on these, the research objectives are defined in Section 1.2 and the methodologies selected are listed in Section 1.3. In Section 1.4 the main contributions of the work are then summarized. The outline of the thesis structure is finally presented in Section 1.5.

1.1 Context and motivation

Arguably the most interesting and innovative industrial engineering applications in recent years can be framed in the context of the fourth industrial revolution (*Industry 4.0*) and *Smart Manufacturing* [99, 154]. The technologies that characterize applications in these fields are defined as the Key Enabling Technologies (KETs). Despite the different definitions of KETs presented in the literature, most agree on *Advanced Automation* (or *Advanced Manufacturing*) being one of the main drivers for Industry 4.0. Advanced automation relies on automated manufacturing systems with high level of interaction with the environment and adaptability to different tasks. Robotic manipulators, which allow to execute a wide range of operations with a minimal re-adaptation effort, are among the most ubiquitous examples of such technologies [81, 87].

The wide spread of robotic applications in the industry has been sided by the development of lightweight manipulators. Lightweight designs are becoming increasingly popular for manipulators given their superior mobility and ease of setup; furthermore, the reduced inertias make collisions with humans less dangerous for collaborative applications [70, 47]. In terms of mechanical properties, one main challenge arises: while traditional industrial manipulators exhibit bulky designs for which structural deformations are negligible, novel lightweight designs require to take compliance into account [127, 124, 77]. Ideally, such compliance should be minimized in order to ensure accuracy. Also energy usage should be kept into account in the design phase to ensure efficient actuation [61, 150], especially e.g. in cases when a large number of instances of the system are deployed in one factory. The requirements for energy efficiency and stiffness are often contrasting and difficult to address with traditional materials and design methodologies.

In many engineering fields, great improvements in terms of performance and energy usage have been achieved employing *Advanced Materials*, such as composites, an additional technology considered fundamental for the Industry 4.0. This is made possible by composite's mechanical properties such as low specific weight and high resistance [34, 21].

There would be several important advantages in a lightweight manipulator made of composite materials compared to traditional ones. First of all, lighter links allow for better dynamic performances, therefore with a reduction of single operations process time that implies also a reduction

in the overall production time of a product. Reduced weight also implies reduced inertia, and this implies an improved safety since it would make a collision with humans or other objects less dangerous. If, instead, the performances are kept at the same level, a lightweight manipulator needs less actuation power; this would make the actuation cheaper and in turn the entire robotic system more sustainable and environmental-friendly.

To avoid fiber damage caused by machining, composite materials have been mainly used for bulk parts such as cars, planes or other vehicles' frames. For these applications even a small weight reduction can cause great reductions of fuel usage, also considering the strict regulations on CO2 emissions. The use of composites in robotic applications has been particularly limited both in research and industry. The main reasons that limited this evolution are the expensiveness of composite materials, high risk of damaging them with machining procedures [153, 92] and the limited understanding of fatigue behaviors [69]. In the last years research focused on hybrid multimaterial systems [34], with few components in composite materials and the most complex ones, subject to stress concentration and wear or difficult to manufacture, in traditional materials (metallic alloys or polymers). This idea can be applied to robotic manipulators, designing the links that have the larger impact on deformation and energy usage employing composites while using traditional materials for the remaining components [158]. Such a hybrid design would allow to take advantage of the strengths of both material families while reducing costs and damaging risks.

Design complexity is another reason limiting the use of composites in robotics [133]. The design of multimaterial mechanical systems results in a complex challenge. Given a set of requirements, their dependency on design parameters is often highly nonlinear and complicated to extract. Furthermore, the solutions require high quality while satisfying numerous and often contrasting requirements. The use of traditional analytical, empirical or trial-and-error methods becomes impractical or even unfeasible. *Simulation* (another KET) employing physics-based numerical models can predict the performances and in turn enable automated design methods based on optimization algorithms [148, 16, 65]. This can reduce the required design time while allowing for improved performances.

The numerical models employed in the design phase can be updated based on experimental data to closely match any physical instance of the systems, constituting what is defined as a Digital Twin (DT) [146, 139]. The availability of a DT enables in turn a set of applications in the subsequent operation phases such as e.g. Virtual Sensing (VS). VS techniques consist in indirectly estimating quantities that can result difficult, expensive or even impossible to directly measure [7]. This commonly relies on merging information from measurements of quantities easier and cheaper to extract with the knowledge of the system granted by the accurate numerical model. Monitoring the evolution of specific parameters of the systems can allow to evaluate its condition and health, applications known as Condition Monitoring (CM) and Structural Health Monitoring (SHM). Monitoring the health of a system during its entire lifetime is of particular interest when composite materials are employed, given the higher damage risk and difficulty in predictions.

Models employed for optimization and estimation need to satisfy two often contrasting requirements: high **accuracy** and computational **efficiency**. The accuracy requirement consists in ensuring that the behavior is matched both on a component level and on a system level, keeping into account how the properties of single components impact performances of the whole system. Efficiency is required first of all on simulation, allowing for a wide range of possibilities explored in the design phase and VS applications readily providing information during operations (ideally in real-time). Also model updating efficiency is desired, given that each modification in the design parameters coming either from the optimization procedure or the estimation requires the model to be modified accordingly. The two requirements are often contrasting. Common

industrial-level numerical models generally require large computational resources in order to guarantee accuracy [136, 106], resulting in inefficient simulation and model updating procedures. The use of approximate models of small size would instead result in sub-optimal design choices and inaccurate estimates.

1.2 Goals

Given the context and motivations defined in the previous section, the broader objective of this research work can be defined as: *the development of accurate and efficient numerical modeling strategies for multimaterial mechanical systems that enable optimization and Virtual Sensing applications. The systems considered are robotic manipulators in which composite materials are employed to achieve high performance and energy savings.*

More in detail, the goals of the work are:

- Development of accurate and efficient modeling methodologies for advanced multi-material mechanisms. Both performance evaluation and design update efficiency should be considered. The models must furthermore keep into account the coupling between the design of components and the behavior of the overall system;
- Development of methods for improvement of the dynamic performances of robotic manipulators by employing advanced materials with an optimal design. These optimization techniques based on the developed accurate models must be employed in the design phase and guarantee the satisfaction of multiple objectives and constraints;
- Development of Virtual Sensing techniques for component-level and system-level identification and monitoring. These must be able to reliably estimate the full states, inputs and parameters based on a reduced set of measurements easy to obtain;
- Numerical and experimental validation of the proposed methodologies on complex industrial-sized application cases.

1.3 Methodologies

Given the context and objectives defined in the previous sections, the goals have been tackled by employing a set of fundamental numerical methodologies. This section gives a brief overview, while motivations, state of the art and detailed explanations for each will be given in the subsequent chapters. The core methodologies are listed above:

- linear structural *Finite Element* (FE) models are used to predict the dynamic behavior of components;
- component-level *Model Order Reduction* (MOR) methodologies are employed to reduce the size of FE models;
- *parametric Model Order Reduction* (pMOR) allows to retain the parametric dependency in the reduced models;
- *Flexible MultiBody* (FMB) models are used for system-level simulation in order to account for the effect of component deformations on the whole system, therefore enabling optimization;

- *MultiObjective Optimization* (MOO) methodologies are employed in the design optimization phase to identify the set of designs that jointly satisfy sets of contrasting requirements;
- optimal state estimation within the *Kalman Filter* (KF) framework is applied for Virtual Sensing by estimation of states, inputs and parameters based on merging information from measurements with predictions from component-level and system-level models.

1.4 Contributions

The research efforts based on previous considerations resulted in the following innovative contributions to the state of the art:

- the development of a component-level pMOR methodology for structural dynamics applications that allows to retain an explicit dependency of the reduced model on material properties, with particular focus on composite materials;
- the development of a FMB formulation that, through the use of component-level pMOR, exhibits a system-level explicit dependency on material properties of the flexible bodies;
- the definition of a workflow for the strongly coupled structural MOO of flexible components of a MultiBody system;
- the development of a component-level VS technique for the joints estimation of state, inputs and material properties;
- the extension of a system-level estimation technique for system-level joint estimation of states, inputs and material parameters of the flexible components.

A part of the contributions is presented in publications on international journals [32], while the remaining are described in works either under review for publications or in the writing stage.

1.5 Outline

The rest of the work is structured in three main parts. Each part is composed of a first chapter containing the theoretical background of the methodologies employed, followed by chapters presenting the main contributions.

Part I presents the component-level and system-level numerical modeling methodologies developed, together with descriptions of the considered case studies and related numerical and experimental validations. **Part II** discusses the workflow for a model-based design optimization and its application to a multi-material robotic manipulator. **Part III** presents component-level and system-level Virtual Sensing methodologies and their validation on the considered use cases. Finally, conclusions and future possible research efforts are discussed in **Conclusions**.

The parts are structured as follows:

Part I

Chapter 2 provides the theoretical background of the modeling methodologies employed. The constitutive relationships of different kinds of materials are presented. The component-level Finite Elements modeling methodology and typical Model Order Reduction techniques are then summarized. System-level modeling through Flexible MultiBody is finally introduced.

Chapter 3 presents a parametric Model Order Reduction method for linear structural dynamics applications. The methodology is numerically validated on two case studies, namely a scaled wind-turbine blade and a link of a robotic manipulator made of composite material. Their experimental model updating is presented as well.

Chapter 4 describes the Flexible MultiBody formulation with explicit dependency on parameters of the flexible components. The methodology is numerically validated on an open-chain and a closed-chain kinematic mechanisms, namely a 4-bar linkage and a 5 DOF robotic manipulator. The experimental model updating procedure for the manipulator is also discussed.

Part II

Chapter 5 provides an overview on the common design optimization approaches with a focus on Multi-objective optimization problems. Applications for design of robotic manipulators and multi-material structures are discussed.

Chapter 6 discusses the optimization workflow for a strongly-coupled approach employing the parametric Flexible MultiBody models proposed. The particular case of use of composite links for the robotic manipulator is presented, with results discussed for two different scenarios.

Part III

Chapter 7 provides the theoretical background of the estimation methodologies employed. The classical Kalman Filter is introduced, together with its extensions to non-linear systems and for estimation of unknown inputs and parameters. The observability properties are discussed, together with practical issues for implementation and setup of the algorithms.

Chapter 8 presents a component-level technique for the estimation of state, inputs and parameters of Finite Element models. The choice of sensors in order to guarantee observability is analyzed. Furthermore, the setup of the filter is discussed. The methodology is validated both numerically and experimentally on a scaled wind turbine blade case.

Chapter 9 presents a system-level technique for the estimation of states, inputs and parameters of Flexible MultiBody models. The sensor choice is discussed. A numerical validation on

the four-bar linkage is presented. The methodology is furthermore validated both numerically and experimentally on the robotic manipulator case.

Conclusions

Chapter 10 summarizes the conclusions for the main contributions of the work and discusses possible future research efforts.

A graphical representation of the structure of the thesis is presented in the following scheme:

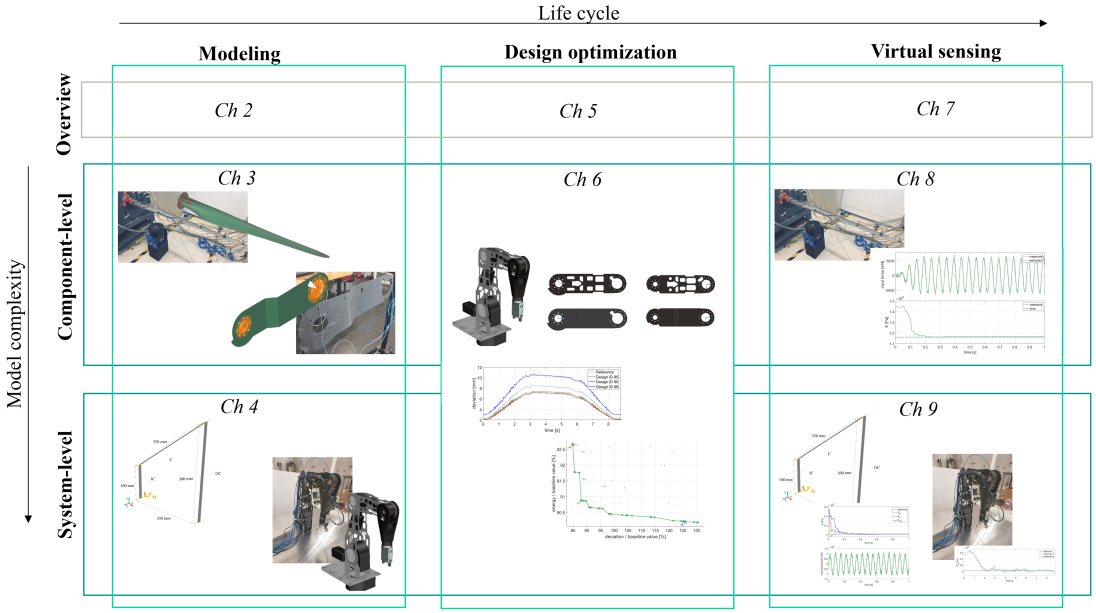


Figure 1.1: Graphical outline of the thesis structure.

Part I

Numerical modeling methodologies

Chapter 2

Mechanical systems modeling: Overview

2.1 Introduction

This chapter presents an overview on the numerical modeling techniques on which the methodologies developed in this work are based. Constitutive relationships of several linear elastic materials are presented in Section 2.2 and used within the Finite Element (FE) methodology for component-level modeling in Section 2.3. The analysis then focuses on system-level models through Flexible Multibody methodology in Section 2.4. Time integration schemes are finally discussed in Section 2.5.

2.2 Linear elastic material models

The first step in modeling a flexible component is the description of the load-deformation relationship. In this work, bodies are assumed to be linear-elastic with homogeneous material properties, with equivalent homogeneous models used for composite materials. Under such assumptions, the relation between the strain tensor $\boldsymbol{\epsilon} \in \mathbb{R}^6$ in a point in the volume of the body and the stress tensor $\boldsymbol{\sigma} \in \mathbb{R}^6$ is described by the constitutive relation:

$$\boldsymbol{\sigma} = \mathbf{E}\boldsymbol{\epsilon} \tag{2.1}$$

where $\mathbf{E} \in \mathbb{R}^{6\times6}$ is the constitutive matrix which depends only on the material model that has been employed. The material models used in this thesis will be briefly shown in the rest of this section, with particular focus on the parameters needed to fully define their elastic behavior.

2.2.1 Generic anisotropic material

The most generic linear-elastic constitutive relation is that of a completely anisotropic materials, in which each element of $\mathbf{E}$ is independent. The only constraint comes from the symmetry property of $\mathbf{E}$. This results in 21 independent parameters:

$$\mathbf{E} = \begin{bmatrix} E_{11} & E_{12} & E_{13} & E_{14} & E_{15} & E_{16} \\ & E_{22} & E_{23} & E_{24} & E_{25} & E_{26} \\ & & E_{33} & E_{34} & E_{35} & E_{36} \\ & & & E_{44} & E_{45} & E_{46} \\ & sym & & & E_{55} & E_{56} \\ & & & & & E_{66} \end{bmatrix} \tag{2.2}$$

Any further model that assumes a particular behavior for the material can be seen as a particular case of this first one with a reduced number of independent parameters.

2.2.2 Isotropic materials

For isotropic 3D materials, the behavior is dictated by 2 independent parameters: the Young's Modulus E and the Poisson's Ratio ν, resulting in:

$$\mathbf{E} = \begin{bmatrix} \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} & \frac{E\nu}{(1+\nu)(1-2\nu)} & \frac{E\nu}{(1+\nu)(1-2\nu)} & 0 & 0 & 0 \\ \frac{E\nu}{(1+\nu)(1-2\nu)} & \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} & \frac{E\nu}{(1+\nu)(1-2\nu)} & 0 & 0 & 0 \\ \frac{E\nu}{(1+\nu)(1-2\nu)} & \frac{E\nu}{(1+\nu)(1-2\nu)} & \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{E}{2(1+\nu)} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{E}{2(1+\nu)} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{E}{2(1+\nu)} \end{bmatrix} \tag{2.3}$$

The constitutive relation can be rendered in a compact form by employing the so-called Lame parameters:

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \tag{2.4}$$

$$\mu = \frac{E}{2(1+\nu)} \tag{2.5}$$

so that

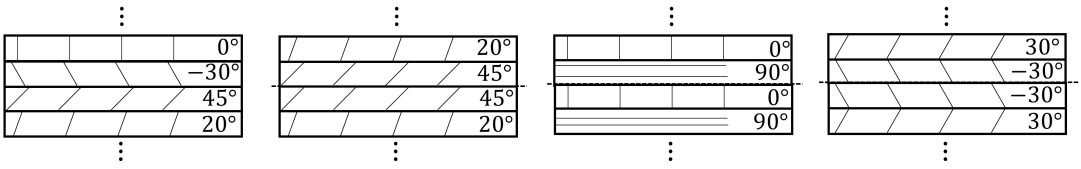
$$\mathbf{E} = \begin{bmatrix} \lambda+2\mu & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda+2\mu & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & \lambda+2\mu & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{bmatrix} \tag{2.6}$$

2.2.3 Composite laminate materials

The family of composite materials is another interesting class to analyze, given its properties highlighted in Chapter 1. These are made of two or more distinct materials, called the *reinforcement* and the *matrix*, which are combined together to create a material with enhanced properties. The reinforcement material provides strength and stiffness, while the matrix material holds the reinforcement in place and protects it from damage [21, 64].

This work focuses on Long Fiber Reinforced composites, a type of composite material combining fibers and a matrix (commonly polymeric) to create a lightweight, high-strength material. Such materials are made of plies (also called laminae) stacked to create a laminate. Different laminates with a wide range of properties can be achieved based on the architecture of the single plies and how they are assembled [21].

In unidirectional (UD) plies, the fibers are aligned along one direction. The laminates are then characterized by the number of layers and the stacking order, with each layer at a specific orientation angle relative to the global reference. Different stacking sequences (Figure 2.1) guarantee different behavior in terms of isotropy and resistance to loads in specific directions.



(a) Generic laminate (b) Symmetric laminate (c) Cross-ply laminate (d) Angle-ply laminate

Figure 2.1: Examples of common stacking sequences for laminates made of UD plies.

Equivalent Single Layer Theories (ESLT) are among the most common modeling approaches for laminates [6]. These methods are based on the assumption that the laminate behaves as if it were a single layer with the same mechanical properties as the original multi-layered structure. In this context, Classical Lamination Theory (CLT) is the most common approach [75, 64].

For the considered approach, the mechanical behavior of such materials is characterized first on the ply level and then assembled to represent the full laminate. UD layers present an orthotropic behavior with three planes of elastic symmetry. For the k-th layer, a local coordinate system $\begin{bmatrix}\overline{x} & \overline{y} & \overline{z}\end{bmatrix}$ is defined so that the the $\overline{x}$ axis is aligned with the fibers and the $\overline{z}$ axis is normal to the plane of the ply (Figure 2.2). The $\overline{\mathbf{E}}_k$ matrix relates stresses and strains of the k-th layer in the local frame:

$$\overline{\mathbf{E}}_k = \begin{bmatrix} \overline{c}_{11} & \overline{c}_{12} & \overline{c}_{13} & 0 & 0 & 0 \\ \overline{c}_{12} & \overline{c}_{22} & \overline{c}_{23} & 0 & 0 & 0 \\ \overline{c}_{13} & \overline{c}_{23} & \overline{c}_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & \overline{c}_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & \overline{c}_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & \overline{c}_{66} \end{bmatrix}_k \tag{2.7}$$

The $\overline{c}_{ij}$ elastic coefficients can be calculated based on material parameters of the matrix (E_m, ν_m), of the fibers (E_f, ν_f) and the volumetric percentage of fibers (V_f) [21].

The global coordinate system of the laminate $\begin{bmatrix}x & y & z\end{bmatrix}$ does not coincide with the layer local ones, given that each of them can have a different fibers orientation, so the global and local frames have the x axis rotated by an angle Θ_k (Figure 2.2). The elastic matrix can be rotated using the tensor relationship $\mathbf{E}_k(\Theta_k) = \mathbf{T}(\Theta_k)\overline{\mathbf{E}}_k\mathbf{T}_k^T(\Theta_k)$ where $\mathbf{T}(\Theta_k)$ is a rotation matrix based on the transformation between the two coordinate systems [75].

ESLT allows to calculate an equivalent elasticity matrix $\mathbf{E}^c$ relating forces and strain field on the laminate. This can be defined as [112]:

$$\mathbf{E}^c = \begin{bmatrix} \mathbf{E}^c_m & \mathbf{E}^c_{mb} & \mathbf{0} \\ \mathbf{E}^{c^T}_{mb} & \mathbf{E}^c_b & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{E}^c_s \end{bmatrix} \tag{2.8}$$

where the subscript m refers to membrane loads and deformations, b to bending and s to shear

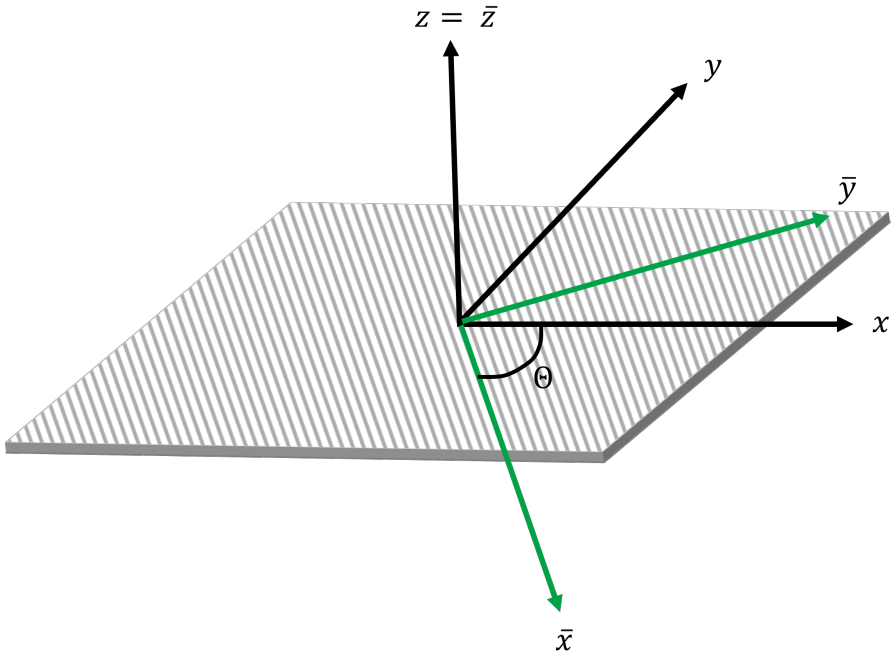


Figure 2.2: Local and global frames of reference for a layer.

ones. For a laminate with n_l layers, the four components can be calculated as:

$$\begin{aligned}
\mathbf{E}_m^c &= \sum_{k=1}^{n_l} \left(\mathbf{E}_k \mathbf{S}_m \left(z_{k+1} - z_k\right)\right) \\
\mathbf{E}_b^c &= \frac{1}{2} \sum_{k=1}^{n_l} \left(\mathbf{E}_k \mathbf{S}_b \left(z_{k+1}^2 - z_k^2\right)\right) \\
\mathbf{E}_{mb}^c &= \frac{1}{3} \sum_{k=1}^{n_l} \left(\mathbf{E}_k \mathbf{S}_{mb} \left(z_{k+1}^3 - z_k^3\right)\right) \\
\mathbf{E}_s^c &= \sum_{k=1}^{n_l} \left(\mathbf{E}_k \mathbf{S}_s \left(z_{k+1} - z_k\right)\right)
\end{aligned} \tag{2.9}$$

where $\mathbf{S}_m$, $\mathbf{S}_b$, $\mathbf{S}_{mb}$ and $\mathbf{S}_s$ are shape matrices used to select a subset of required components of $\mathbf{E}_k$. z_k and z_{k-1} represent the coordinate along the thickness of the bottom and top faces of the layer, respectively.

The assembled laminate matrices depend on independent material properties, thickness h and fiber angle of each layer, resulting in a large set of parameters for a single component. Without loss of generality, this work focuses on a particular configuration that is the symmetric angle-ply (Figure 2.1d), for which the layers are oriented alternately in the directions Θ and $-\Theta$; if the number of layers is even, the symmetry is maintained by repeating the same orientation for the two central layers. Furthermore, layers are assumed to have the same material properties and thickness. This choice allows to have only one parameter for the entire stack, reducing the dimension of the parametric problems faced in the following chapters.

2.3 Component-level modeling

The elastic material behavior described in the previous section relates loads and deformations in a continuous 3D space. This relationship extends to components by considering their geometry, the distribution of materials, and the points of application of loads. This section introduces the basic concept of the FE method [39, 18], used to operate a spacial discretization of the volume of a component. FE models generated for industrial application typically exhibit a large size that requires intensive computational resources. A method for reducing the size of such models is also presented.

2.3.1 Linear structural Finite Element method

The FE method consists in discretizing a physical domain, that for structural applications is the body of a component, into sub-domains called *elements*. The state of each element is evaluated only at a finite set of points within its domain, typically placed on the boundaries, called *nodes*. The coordinates describing the state of the element with $n_{nodes,e}$ nodes are the nodal displacements, or nodal Degrees Of Freedom (DOFs), $\mathbf{q}_{f,e} \in \mathbb{R}^{n_{DOF,e}}$. $n_{DOF,e} = 6n_{nodes,e}$ is the number of nodes of the element for the most generic element kinds for which all 6 spatial DOFs are considered.

2.3.1.1 Single elements

For an element with homogeneous material, the displacements $\mathbf{d}_e \in \mathbb{R}^3$ in its volume are assumed to be a linear combination of the nodal displacements, where the weights are the shape functions $\mathbf{N}_e \in \mathbb{R}^{3\times n_{DOF,e}}$, so that:

$$\mathbf{d}_e = \mathbf{N}_e \mathbf{q}_{f,e} \tag{2.10}$$

The strain tensor $\boldsymbol{\epsilon}_e \in \mathbb{R}^6$ in a point of the element is related to the nodal displacements as:

$$\boldsymbol{\epsilon}_e = \mathbf{B}_e \mathbf{q}_{f,e} \tag{2.11}$$

where $\mathbf{B}_e \in \mathbb{R}^{6\times n_{DOF,e}}$ is the strain-displacement matrix calculated from a spatial differentiation of the shape functions. In turn, the stress tensor $\boldsymbol{\sigma}_e \in \mathbb{R}^6$ is related to the strains through the constitutive matrix $\mathbf{E}_e \in \mathbb{R}^{6\times 6}$ as:

$$\boldsymbol{\sigma}_e = \mathbf{E}_e \boldsymbol{\epsilon}_e \tag{2.12}$$

where $\mathbf{E}_e$ depends on the material model used.

The dynamic evolution in time of the element's states $\mathbf{q}_{f,e}$ and their derivatives $\dot{\mathbf{q}}_{f,e}$ and $\ddot{\mathbf{q}}_{f,e}$ under a set of external loads $\mathbf{u} \in \mathbb{R}^{n_u}$ is expressed by the Equations Of Motion (EOMs):

$$\mathbf{M}_{f,e}\ddot{\mathbf{q}}_{f,e}(t) + \mathbf{C}_{f,e}\dot{\mathbf{q}}_{f,e}(t) + \mathbf{K}_{f,e}\mathbf{q}_{f,e}e(t) = \mathbf{S}_{u,e}\mathbf{u}(t) \tag{2.13}$$

where $\mathbf{M}_{f,e} \in \mathbb{R}^{n_{DOF,e}\times n_{DOF,e}}$, $\mathbf{D}_{f,e} \in \mathbb{R}^{n_{DOF,e}\times n_{DOF,e}}$ and $\mathbf{K}_{f,e} \in \mathbb{R}^{n_{DOF,e}\times n_{DOF,e}}$ are respectively the mass, damping and stiffness matrices and $\mathbf{S}_{u,e} \in \mathbb{R}^{n_{DOF,e}\times n_u}$ is the shape matrix that distributes the loads on the DOFs. The stiffness matrix is calculated by integration over the volume of the element:

$$\mathbf{K}_{f,e} = \int_V \mathbf{B}_e^T \mathbf{E}_e \mathbf{B}_e dV \tag{2.14}$$

The consistent mass matrix is:

$$\mathbf{M}_{f,e} = \int_V \rho \mathbf{N}_e^T \mathbf{N}_e dV \tag{2.15}$$

where ρ is the material density. A common technique employed for computational reasons is placing the masses on the nodes, resulting in a diagonal "lumped" mass matrix [39].

Finally, the consistent damping matrix is:

$$\mathbf{C}_{f,e} = \int_V c\mathbf{N}_e^T \mathbf{N}_e dV \tag{2.16}$$

where c is the material damping. The modeling of damping at the element level would need an understanding of the physical laws governing the phenomenon much deeper than what currently known in literature. For this reason, it is common to define the damping on the component level with simplified models.

Lumped elements

A particular kind of elements allows to model lumped properties. A lumped stiffness element acts as a spring between two DOFs of the mesh. For a lumped stiffness element with a certain stiffness k, the element stiffness matrix is:

$$\mathbf{K}_{f,e}^l = k\mathbf{I}_2 \tag{2.17}$$

The same applies to a lumped damping element with damping coefficient c:

$$\mathbf{C}_{f,e}^l = c\mathbf{I}_2 \tag{2.18}$$

A lumped mass element applies an additional mass and inertia on one node of the structure. For a lumped mass element of mass m and inertia tensor $\mathfrak{I} \in \mathbb{R}^{3\times3}$ on the i-th node:

$$\mathbf{M}_{f,e}^l = \begin{bmatrix} m\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathfrak{I} \end{bmatrix} \tag{2.19}$$

2.3.1.2 Mesh assembly

The DOFs of the entire structure $\mathbf{q}_f \in \mathbb{R}^{n_{DOF}}$ are represented by the concatenation of the ones from every element in it, so that:

$$\mathbf{q}_f = \sum_{i=1}^{n_e} \mathbf{S}_e^i \mathbf{q}_{f,e}^i \tag{2.20}$$

where n_e is the number of elements in the mesh and $\mathbf{S}_e \in \mathbb{R}^{n_{DOF}\times n_{DOF,e}}$ is a connectivity matrix containing only ones and zeroes that associates the local DOFs of the element with the correspondent global ones.

The system matrices are also assembled using the same connectivity matrix $\mathbf{S}_e$:

$$\mathbf{K}_f = \sum_{i=1}^{n_e} \mathbf{S}_e^i \mathbf{K}_{f,e}^i \mathbf{S}_e^{i^T} \tag{2.21}$$

$$\mathbf{M}_f = \sum i = 1^{n_e} \mathbf{S}_e^i \mathbf{M}_{f,e}^i \mathbf{S}_e^{i^T} \tag{2.22}$$

This work employs a typical and well-known modeling approach for damping in structural applications: the proportional Rayleigh damping model [96]. This consists in defining the damping matrix as a linear combination of mass and stiffness matrices as:

$$\mathbf{C}_f = \alpha\mathbf{M}_f + \beta\mathbf{K}_f \tag{2.23}$$

where α and β are the Rayleigh damping parameters.

The EOMs for the structure are then:

$$\mathbf{M}_f\ddot{\mathbf{q}}_f(t) + \mathbf{C}_f\dot{\mathbf{q}}_f + \mathbf{K}_f\mathbf{q}_f = \mathbf{S}_u\mathbf{u} \tag{2.24}$$

2.3.2 Model Order Reduction

For industrial applications, given the complexity of the modeled systems and the level of accuracy needed, FE models can reach sizes of tens of thousands or even millions of DOFs. The direct use of these Full Order Models (FOMs) can result in an unfeasible computational load. Their dimensions are usually reduced by employing Model Order Reduction (MOR) techniques [125, 22].

This work in particular focuses on the class of methods called projection-based MOR [22, 116]. The number of DOFs is reduced by employing a set of n_{red} shape vectors $\mathbf{\Psi} \in \mathbb{R}^{n_{DOF} \times n_{red}}$ that span a subspace over which the evolution of the FOM DOFs is approximated, so that:

$$\mathbf{q}_f \approx \mathbf{\Psi} \mathbf{q}_{f_r} \tag{2.25}$$

where $\mathbf{q}_{f_r} \in \mathbb{R}^{n_{red}}$ is the reduced set of DOFs.

Embedding this in the FOM's EOMs leads to:

$$\mathbf{M}_f \mathbf{\Psi} \ddot{\mathbf{q}}_{f_r} + \mathbf{C}_f \mathbf{\Psi} \dot{\mathbf{q}}_{f_r} + \mathbf{K}_f \mathbf{\Psi} \mathbf{q}_{f_r} = \mathbf{S}_u \mathbf{u} + \boldsymbol{\delta} \tag{2.26}$$

where $\boldsymbol{\delta}$ represents the residual due to the approximation done in the reduction of coordinates. A second reduction space orthogonal to the residual is then used to reduce the number of equations in the reduced EOMs. In the general case of an oblique Petrov-Galerkin projection, the two reduction spaces can differ [116]. This work considers only the particular case of an orthogonal projection in which only one set of bases is used for both spaces, resulting in the reduces EOM:

$$\mathbf{M}_{f_r} \ddot{\mathbf{q}}_{f_r} + \mathbf{C}_{f_r} \dot{\mathbf{q}}_{f_r} + \mathbf{K}_{f_r} \mathbf{q}_{f_r} = \mathbf{S}_{u,r} \mathbf{u} \tag{2.27}$$

where $\mathbf{M}_{f_r} = \mathbf{\Psi}^T \mathbf{M}_f \mathbf{\Psi} \in \mathbb{R}^{n_{red} \times n_{red}}$, $\mathbf{C}_{f_r} = \mathbf{\Psi}^T \mathbf{C}_f \mathbf{\Psi} \in \mathbb{R}^{n_{red} \times n_{red}}$, $\mathbf{K}_{f_r} = \mathbf{\Psi}^T \mathbf{K}_f \mathbf{\Psi} \in \mathbb{R}^{n_{red} \times n_{red}}$ and $\mathbf{S}^r = \mathbf{\Psi}^T \mathbf{S} \in \mathbb{R}^{n_{red} \times n_u}$. This effectively reduces both the number of equations and DOFs, resulting in a so-called Reduced Order Model (ROM) with a decreased associated computational load. Several projection-based MOR methods exist in literature, mainly differing in the choice of the reduction basis. This work focuses on the Component Modes Synthesis (CMS) approach [42]. The DOFs or the system are divided between internal and boundary DOFs, with the latter being the ones subject to boundary conditions or external loads. Different CMS techniques use different sets of vector [41]. The one employed in this work uses:

- $\mathbf{\Psi}_{nm}$, a truncated set of normal modes of the system.
- $\mathbf{\Psi}_{sm}$, a set of static modes that represent the static deformation due to external loads. They are calculated by applying one-by-one a unitary displacement to the boundary DOFs while keeping all the other constrained.

The entire reduction basis is obtain by concatenation of the mentioned sets:

$$\mathbf{\Psi} = \begin{bmatrix} \mathbf{\Psi}_{nm} \mathbf{\Psi}_{sm} \end{bmatrix} \tag{2.28}$$

One of the main challenges in the creation of a projection-based ROM lies in choosing the size of the reduction basis. A large set of normal modes allows to have a better accuracy at larger frequency ranges, at the expense of a larger model and less computational gain. Commonly, a limited set of normal modes in the frequency range of interest is used. If the model is matched with a physical component for which experimental data is available, modes that are validated experimentally are privileged. Adding static modes relative to a large number of input DOFs enables the use of the ROM in a wider range of applications, while also making the model heavier. For both classes of modes, a solution needs to be chosen that guarantees the right trade-off between accuracy and efficiency.

2.3.3 Measurement equations

The solution of EOMs for a FE model does not directly disclose the values of some physical quantities of interests that are commonly measured through sensors on physical systems. These quantities can be computed from the states of the system, resulting in what are commonly referred to as *virtual measurements*. The generic set of virtual measurements $\mathbf{y} \in \mathbb{R}^{n_m}$ is:

$$\mathbf{y} = \mathbf{h}(\mathbf{q}_{f_r}, \dot{\mathbf{q}}_{f_r}, \mathbf{u}) \tag{2.29}$$

where $\mathbf{h}$ is commonly defined as the measurement equation.

The rest of this section will present the formulation of the virtual measurements considered in this work.

Strain Measurements

One of the most common measurement in engineering structural applications is the strain of a specific point on the surface of a component. The full strain status of a point is defined by its strain tensor $\boldsymbol{\epsilon}$. The strain tensor in a point can be obtained through Equation 2.11, employing the shape matrix $\mathbf{S}_e$ relative to the element containing the point to select only the element-related DOFs:

$$\boldsymbol{\epsilon} = \mathbf{B}_e \mathbf{S}_e^T \mathbf{q}_f = \mathbf{B}_e \mathbf{S}_e^T \boldsymbol{\Psi} \mathbf{q}_{f_r} \tag{2.30}$$

The strain tensor can then be projected along the unit vector $\boldsymbol{\vartheta} = \begin{bmatrix} \vartheta_1 & \vartheta_2 & \vartheta_3 \end{bmatrix}^T$ by [117]:

$$\epsilon_{\boldsymbol{\vartheta}} = \overline{\boldsymbol{\vartheta}}^T \boldsymbol{\epsilon} = \begin{bmatrix} \vartheta_1^2 & \vartheta_2^2 & \vartheta_3^2 & \vartheta_1\vartheta_2 & \vartheta_2\vartheta_3 & \vartheta_3\vartheta_1 \end{bmatrix} \boldsymbol{\epsilon} \tag{2.31}$$

The strain measurement equations finally results in:

$$h_s(\mathbf{q}_{f_r}) = (\overline{\boldsymbol{\vartheta}}^T \mathbf{B}_e \mathbf{S}_e^T \boldsymbol{\Psi}) \mathbf{q}_{f_r} = \mathbf{N}_s \mathbf{q}_{f_r} \tag{2.32}$$

where $\mathbf{N}_s \in \mathbb{R}^{1 \times n_{red}}$ is a strain basis matrix introduced to simplify the notation.

Displacement Measurements

Another quantity of interest can be the displacement of a point with respect to the undeformed configurations, still along the direction $\boldsymbol{\vartheta}$.

Equivalently to what seen for strains, also the displacement of a point on the FE mesh can be obtained using Equation 2.10 as:

$$\mathbf{d} = \mathbf{N}_e \mathbf{S}_e^T \mathbf{q}_f = \mathbf{N}_e \mathbf{S}_e^T \boldsymbol{\Psi} \mathbf{q}_{f_r} \tag{2.33}$$

Projection over the sensors direction yields:

$$h_d(\mathbf{q}_{f_r}) = (\boldsymbol{\vartheta}^T \mathbf{N}_e \mathbf{S}_e^T \boldsymbol{\Psi}) \mathbf{q}_{f_r} = \mathbf{N}_d \mathbf{q}_{f_r} \tag{2.34}$$

Acceleration Measurements

Derivation of the former equation allows to evaluate the acceleration of a point in a direction $\boldsymbol{\vartheta}$:

$$h_a(\ddot{\mathbf{q}}_{f_r}) = (\boldsymbol{\vartheta}^T \mathbf{N}_e \mathbf{S}_e^T \boldsymbol{\Psi}) \ddot{\mathbf{q}}_{f_r} = \mathbf{N}_d \ddot{\mathbf{q}}_{f_r} \tag{2.35}$$

that can be expressed in terms of the system states by employing the EOMs in Equation 2.27:

$$h_a(\mathbf{q}_{f_r}, \dot{\mathbf{q}}_{f_r}, \mathbf{u}) = \mathbf{N}_d \mathbf{M}_{f_r}^{-1} \left(-\mathbf{C}_{f_r} \dot{\mathbf{q}}_{f_r} - \mathbf{K}_{f_r} \mathbf{q}_{f_r} + \mathbf{S}_{u,r} \mathbf{u} \right) \tag{2.36}$$

2.3.4 State-space representation of the EOMs

Most numerical methods for the integration in time of dynamical systems are formulated on first-order ODEs. The second-order EOMs for FE models can be transformed to a first-order form via a State-Space (SS) representation [45]. The states of the system $\mathbf{x} \in \mathbb{R}^{2n_{red}}$ are defined as the concatenation of coordinates and their first derivatives as:

$$\mathbf{x} = \begin{bmatrix} \mathbf{q}_{f_r} \\ \dot{\mathbf{q}}_{f_r} \end{bmatrix} \tag{2.37}$$

The state-space equations can then be defined as:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}_c \mathbf{x}(t) + \mathbf{B}_c \mathbf{u}(t) \\ \mathbf{y} = \mathbf{H}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \end{cases} \tag{2.38}$$

in which $\mathbf{A}_c \in \mathbb{R}^{2n_{red} \times 2n_{red}}$ and $\mathbf{B}_c \in \mathbb{R}^{2n_{red} \times 2n_{red}}$ are respectively defined as the state and input matrices. These can be computed as:

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}_{f_r}^{-1}\mathbf{K}_{f_r} & -\mathbf{M}_{f_r}^{-1}\mathbf{C}_{f_r} \end{bmatrix} \tag{2.39}$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbf{M}_{f_r}^{-1}\mathbf{S}_{u,r} \end{bmatrix} \tag{2.40}$$

The output and feedthrough matrices $\mathbf{H} \in \mathbb{R}^{n_m \times 2n_{red}}$ and $\mathbf{D} \in \mathbb{R}^{n_m \times n_u}$ depend on the quantity measured. From the derivations of Subsection 2.3.3:

$$\mathbf{H}_s = \begin{bmatrix} \mathbf{N}_s & \mathbf{0} \end{bmatrix} \qquad \mathbf{D}_s = \begin{bmatrix} \mathbf{0} \end{bmatrix} \tag{2.41}$$

$$\mathbf{H}_d = \begin{bmatrix} \mathbf{N}_d & \mathbf{0} \end{bmatrix} \qquad \mathbf{D}_d = \begin{bmatrix} \mathbf{0} \end{bmatrix} \tag{2.42}$$

$$\mathbf{H}_a = \mathbf{N}_d \mathbf{M}_{f_r}^{-1} \begin{bmatrix} -\mathbf{K}_{f_r} & -\mathbf{C}_{f_r} \end{bmatrix} \qquad \mathbf{D}_a = \mathbf{N}_d \mathbf{M}_{f_r}^{-1} \begin{bmatrix} \mathbf{S}_{u,r} \end{bmatrix} \tag{2.43}$$

2.4 System-level modeling

Linear structural dynamics FE models deal with single bodies undergoing small deformations and no rigid body motion. This section will extend the discussion to models that can represent large nonlinear rigid-body motions of a mechanical system under the MultiBody (MB) framework [73, 58]. While a subset of the bodies in such systems can be modeled under the rigid-body assumption, it is often of paramount importance to keep into account how the deformation of certain components impacts the system behavior. Small deformations of the bodies are represented in this work using the Floating Frame of Reference (FFR) approach [126], resulting in a FMB model.

2.4.1 The Floating Frame of Reference approach

The idea behind the FFR modeling approach is that of decoupling the large translations and rotations of a flexible body and its (small) deformations. A body-attached frame of reference is defined whose configuration in space represents the rigid motion of the body. The deformation is then described with respect to such frame, typically using a FE formulation. This in turn enables the use of common component-level linear MOR techniques to reduce the flexible coordinates [151] , such as the one described in Section 2.3.

The configuration of the i-th flexible body is described using a set of coordinates:

$$\mathbf{q}^i = \begin{bmatrix} \mathbf{q}^i_R \\ \mathbf{q}^i_\Theta \\ \mathbf{q}^i_{fr} \end{bmatrix} \tag{2.44}$$

where $\mathbf{q}^i_R \in \mathbb{R}^3$ are the coordinates of the center of the body-attached frame in global coordinates, and $\mathbf{q}^i_\Theta \in \mathbb{R}^4$ are rotational coordinates representing the orientation of its axes. The choice of rotational coordinates in 3D space is not uniquely defined, with different parameter sets available in literature (e.g. Euler parameters, Euler angles, Rodriguez parameters, direction cosines) [126, 73]. In this work, the Euler parameters are employed as angular coordinates. These allow to avoid singular configuration, at the expense of a redundant additional DOF for each body and a consequential constraint to be added. $\mathbf{q}^i_{f_r} \in \mathbb{R}^{n^i_{red}}$ are the reduced flexible coordinates representing the deformation with respect to the body-attached frame. The considerations made on the need for dimensionality reduction applies also when FE models are used to represent components of a FMB model. Also in this case, the CMS approach is employed in order to obtain a reduction basis. Among the many possible approaches [41], the one employed in this work uses normal modes $\mathbf{\Psi}_{nm}$ calculated relative to the unconstrained component. This would results in also Rigid Body (RB) modes present in the basis, that are removed during the pre-processing steps given that the RB behavior of the component is already represented by the body-attached frame.

If a body is modeled under the rigid-body assumptions, only the rigid coordinates $\mathbf{q}^i_R$ and $\mathbf{q}^i_\Theta$ are employed to represent its configuration.

2.4.2 Flexible Multibody formulation

The system's coordinates $\mathbf{q} \in \mathbb{R}^{n_{sys}}$ are given by the concatenations of the ones relative to the n_b single bodies:

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}^1 \\ \vdots \\ \mathbf{q}^{n_b} \end{bmatrix} \tag{2.45}$$

The relative and absolute motion of the bodies is subject to constraints described by the set of algebraic equations $\mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \in \mathbb{R}^{n_c}$. In this work, constraints that depend only on position-level coordinates, also called *holonomic stationary constraints*, are considered.

The constrained EOMs generate a set of index-3 Differential-Algebraic Equations (DAEs) governing the behavior of the system:

$$\begin{cases} \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} + \mathbf{\Phi}^T_q(\mathbf{q})\boldsymbol{\lambda} = \mathbf{f}_e(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{f}_v(\mathbf{q}, \dot{\mathbf{q}}) \\ \mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \end{cases} \tag{2.46}$$

where:

- $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n_{sys} \times n_{sys}}$ is the configuration-dependent mass matrix
- $\mathbf{K} \in \mathbb{R}^{n_{sys} \times n_{sys}}$ is the stiffness matrix
- $\mathbf{C} \in \mathbb{R}^{n_{sys} \times n_{sys}}$ is the damping matrix
- $\mathbf{\Phi}_q(\mathbf{q}) = \frac{\partial \mathbf{\Phi}(\mathbf{q})}{\partial \mathbf{q}} \in \mathbb{R}^{n_c \times n_{sys}}$ is the constraints equation's Jacobian
- $\boldsymbol{\lambda} \in \mathbb{R}^{n_c}$ are the Lagrange multipliers relative to the constraints

- $\mathbf{f}_e(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n_{sys}}$ are the generalized external forces
- $\mathbf{f}_v(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n_{sys}}$ are the quadratic velocity forces

The quadratic velocity term for the i-th body is:

$$\mathbf{f}_v^i(\mathbf{q}^i, \dot{\mathbf{q}}^i) = \left(\frac{\partial}{\partial \mathbf{q}^i} \left(\dot{\mathbf{q}}^{i^T} \mathbf{M}^i(\mathbf{q}^i) \dot{\mathbf{q}}^i \right) \right)^T - \dot{\mathbf{M}}^i(\mathbf{q}^i, \dot{\mathbf{q}}^i) \dot{\mathbf{q}}^i \tag{2.47}$$

The system mass matrix $\mathbf{M}$ consists of the block-diagonal concatenation of the individual components mass matrices:

$$\mathbf{M}(\mathbf{q}) = \begin{bmatrix} \mathbf{M}^1(\mathbf{q}^1) & & & & \\ & \ddots & & & \mathbf{0} \\ & & \mathbf{M}^i(\mathbf{q}^i) & & \\ sym & & & \ddots & \\ & & & & \mathbf{M}^{n_b}(\mathbf{q}^{n_b}) \end{bmatrix} \tag{2.48}$$

The mass matrix of the i-th body, if flexible, can be defined as a block matrix:

$$\mathbf{M}^i(\mathbf{q}^i) = \begin{bmatrix} \mathbf{M}_R^i & \mathbf{M}_{R\Theta}^i(\mathbf{q}^i) & \mathbf{M}_{Rf}^i(\mathbf{q}^i) \\ & \mathbf{M}_{\Theta}^i(\mathbf{q}^i) & \mathbf{M}_{\Theta f}^i(\mathbf{q}^i) \\ sym & & \mathbf{M}_{f_r}^i \end{bmatrix} \tag{2.49}$$

In particular, two of the terms are constant:

$$\mathbf{M}_R^i = \mathbf{I}^{3\times 3} m^i \tag{2.50}$$

and $\mathbf{M}_{f_r}^i$ is the reduced mass matrix of the FE model for the body. The other 4 components are instead configuration-dependent. The configuration-dependent mass matrix is one of the main disadvantages of the FFR formulation. Different approaches have been proposed that allow for a constant mass matrix at the cost of different kinds of complications arising. A set of invariant quantities, the so-called mass invariants, is commonly pre-computed to improve the computational efficiency of updating the mass matrix every time the state of the system is updated. The generic expression for the j-th mass invariant is:

$$\hat{\mathbf{M}}_j = \sum_{i=1}^{n_{nodes}} m_i \mathbf{h}_j(\mathbf{r}_0^i, \mathbf{\Psi}) \tag{2.51}$$

where $\mathbf{h}$ can be a scalar, matrix or tensor operator according to which mass-invariant is considered. The formulas for all of the mass invariants and their use to calculate $\mathbf{M}$ are provided in Appendix A.

If the body is modeled as rigid, instead, the terms of the mass matrix relative to flexibility disappear. Term relative to translation remains the same as defined for flexible bodies, while the rotation components are defined based on the body's inertia tensor [73].

It is assumed in this work, as common practice in FMB modeling, that discrete elastic components in the system (e.g. springs, bushings) are modeled as force elements and are thus included in the force terms of the EOMs. The only components that appear in the stiffness matrices are the FE damping and stiffness matrices relative to internal elastic forces of the flexible bodies. Thanks to this, also damping and stiffness matrices are block-diagonal matrices containing the ones relative to single bodies:

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}^1 & & & & \\ & \ddots & & & \mathbf{0} \\ & & \mathbf{K}^i & & \\ & & & \ddots & \\ sym & & & & \mathbf{K}^{n_b} \end{bmatrix} \tag{2.52}$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}^1 & & & & \\ & \ddots & & & \mathbf{0} \\ & & \mathbf{C}^i & & \\ & & & \ddots & \\ sym & & & & \mathbf{C}^{n_b} \end{bmatrix} \tag{2.53}$$

For the i-th body:

$$\mathbf{K}^i = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ & \mathbf{0} & \mathbf{0} \\ sym & & \mathbf{K}^i_{f_r} \end{bmatrix} \tag{2.54}$$

$$\mathbf{C}^i = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ & \mathbf{0} & \mathbf{0} \\ sym & & \mathbf{C}^i_{f_r} \end{bmatrix} \tag{2.55}$$

where $\mathbf{K}^i_{f_r}$ is the reduced stiffness matrix of the FE model as obtained with the reduction method in Subsection 2.3.2, and $\mathbf{C}^i_{f_r}$ the correspondent damping matrix. It can be appreciated how system's stiffness and damping matrices, in the FFR approach, result in constant terms, i.e. not dependent on configuration.

2.4.3 Measurement equations

Virtual measurements are defined on FMB models equivalently to what seen in the previous section for FE models. This section will briefly present measurement equations for the types of sensors considered in this work. The generic measurement equation for FMB applications is defined as:

$$\mathbf{y} = \mathbf{h}(\mathbf{q}, \mathbf{v}) \tag{2.56}$$

Strain Measurements

The strain measurement on a point of the i-th flexible component only depends on the flexible coordinates $\mathbf{q}^i_{f_r}$ and is defined equivalently as in Equation 2.32:

$$h_s(\mathbf{q}^i_{f_r}) = \mathbf{N}_s \mathbf{q}^i_{f_r} \tag{2.57}$$

Position Measurements

The position measurement in a MB system is defined with respect to the global frame of reference. Let us consider a point on the i-th body with relative coordinates $\mathbf{r}_0$ in the body-attached frame. This coordinate can be arbitrary in the case of a rigid body, or the position of a node in the undeformed configuration for a flexible component.

For a flexible body, the global position can be computed, based on the assumptions of the FFR formulation [126] and employing Equation 2.34, as:

$$h_d(\mathbf{q}) = \boldsymbol{\vartheta}^T \mathbf{q}_R^i + \mathbf{A}_\Theta^i(\boldsymbol{\vartheta}^T \mathbf{r}_0 + \mathbf{N}_d \mathbf{q}_{f_r}^i) \tag{2.58}$$

where $\mathbf{A}_\Theta^i(\mathbf{q}_\Theta^i)$ is a rotation matrix representing the orientation of the body-attached frame with respect to the global one (see Appendix A). When a rigid body is considered, the measurement equation above is modified by removing the term relative to flexible coordinates.

Acceleration Measurements

Derivation of the former equation allows to evaluate the acceleration of a point in a direction $\boldsymbol{\vartheta}$:

$$h_a(\ddot{\mathbf{q}}) = \boldsymbol{\vartheta}^T \ddot{\mathbf{q}}_R^i + (\ddot{\mathbf{A}}_\Theta^i)(\boldsymbol{\vartheta}^T \mathbf{r}_0 + \mathbf{N}_d \mathbf{q}_{f_r}^i) + \mathbf{A}_\Theta^i \mathbf{N}_d \ddot{\mathbf{q}}_{f_r}^i \tag{2.59}$$

2.4.4 Stabilized index-2 formulation of the EOMs

The second-order set of DAEs of Equation 2.46 can be recast as an first-order form by introducing the velocity variable $\mathbf{v} = \dot{\mathbf{q}}$:

$$\begin{cases} \dot{\mathbf{q}} - \mathbf{v} = \boldsymbol{\Phi}_q^T(\mathbf{q})\boldsymbol{\mu} \\ \mathbf{M}(\mathbf{q})\dot{\mathbf{v}} + \mathbf{C}\mathbf{v} + \mathbf{K}\mathbf{q} + \boldsymbol{\Phi}_q^T(\mathbf{q})\boldsymbol{\lambda} = \mathbf{f}_e(\mathbf{q}, \mathbf{v}) + \mathbf{f}_v(\mathbf{q}, \mathbf{v}) \\ \boldsymbol{\Phi}_q^T(\mathbf{q})\mathbf{v} = \mathbf{0} \\ \boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0} \end{cases} \tag{2.60}$$

where $\boldsymbol{\mu} \in \mathbb{R}^{n_c}$ is an additional set of Lagrange multipliers that approach zero for the exact solution of the equations. This stabilized index-2 form of the EOMs is commonly referred to as Gear-Gupta-Leimkuhler (GGL) formulation [62]. The constraint equation is included also in its velocity-level form, allowing to avoid drift-off errors and enabling the solution with common integration schemes.

2.5 Time integration

Practical numerical implementations for the solution of ODEs require a time-discretization. The transition from continuous-time to discrete-time is done through one of the many integration schemes available in literature [71, 72]. A main distinction can be made between explicit and implicit schemes. Explicit schemes provide a direct way of computing the states at time k given information on the previous step $k-1$. These are characterized by computational efficiency and ease of implementation, at the cost of a small time step required in order to ensure stability. On the other end, implicit integration schemes are more complicated to implement and do not allow for closed-form solutions, requiring instead the use of an iterative procedure (e.g. Newton-Raphson method). Implicit schemes offer the advantage of guaranteeing stability even for larger integration steps compared to explicit ones.

Without loss of generality, this work employs an implicit first-order Backward Differentiation Formula (BDF) for the solution of EOMs both in the FE and in the FMB domains. Given the generic state $\mathbf{x}$, the order-1 BDF assumes that the states evolution between steps $k-1$ and k follows the equation:

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \Delta t \dot{\mathbf{x}}_k \tag{2.61}$$

The time step $\Delta t = t_k - t_{k-1}$ is assumed to not vary in the solution process.

Chapter 3

Parametric Model Order Reduction for structural dynamics applications

3.1 Introduction

The numerical models used in many practical applications (e.g. optimization, Virtual Sensing) depend on parameters that need to be updated frequently. The efficient parameters update is then of paramount importance in order to guarantee computational feasibility. Such a requirement is often not guaranteed by standard projection-based ROMs. Simulation and update efficiency are then two contrasting requirements. The explicit parametric dependency can be retained by applying a parametric Model Order Reduction (pMOR) technique. The use of a parametric Reduced Order Models (pROMs) can be beneficial in many applications in which the parameters of a FE require frequent update [152, 10].

This chapter presents a pMOR approach that exploits models with an affine parameter dependency and uses a constant global reduction basis [20]. More specifically, linear structural FE models show an affine dependency on material parameters. In this case, a global reduction basis, i.e. not depending on parameters, can be defined and used to reduce the so-called affine components in an off-line phase. The evaluation of the parametric model becomes then straightforward in the on-line phase. Only one global basis and a small set of matrices - the affine components - thus have to be stored, leading to significant gains with respect to storage and a positive impact on the computational performances. Component-level pROMs enable applications in which frequent updates are needed and where the explicit dependency is leveraged to compute the sensitivity of the component's behavior on parameters.

The methodology is validated both in time and frequency domain by comparison with standard non-parametric ROMs. Two different models are considered, namely a Titanium scaled wind turbine blade and the link of a robotic manipulator in fiber-reinforced laminate composites. This allows to evaluate the performance of the methodology on both isotropic materials, one of the most used classes of materials in engineering applications, and composites, that given their anisotropic behavior and complex formulation offer a tougher challenge for the parametric dependency representation.

In Section 3.2 the problems arising by applying standard projection-based MOR techniques to parametric problems are analyzed. In Section 3.3 the affine dependency of the FE matrices on parameters is demonstrated for different kinds of materials. Section 3.4 describes the pMOR scheme and the way the global reduction basis is selected. Finally, in Section 3.5 the pMOR is

validated for two models with different materials. The results of experimental model updating procedures for both cases are also discussed.

The content of this chapter is partially presented in a publication from the author [32].

3.2 The challenge of projection-based MOR for parametric models

Let us consider the case in which the FE model depends on a set of parameters $\mathbf{p}(t) \in \mathbb{R}^{n_p}$. This work restricts the analysis to parameters relative to the material properties or lumped properties. In this case, the EOMs become:

$$\mathbf{M}_f(\mathbf{p})\ddot{\mathbf{q}}_f + \mathbf{C}_f(\mathbf{p})\dot{\mathbf{q}}_f + \mathbf{K}_f(\mathbf{p})\mathbf{q}_f = \mathbf{S}_u\mathbf{u} \tag{3.1}$$

The EOMs are generally derived by assuming constant mass for the system, while if any mass-related parameter is time-variant an additional related term appears in the equations [136]. The assumption made in this work is that density varies slowly in time, a realistic behavior for system identification or monitoring applications, so that the additional term is assumed to be negligible. Furthermore, the considered parameters do not have an influence on the topology of the system, that is assumed to be constant, so that also the input shape matrix $\mathbf{S}_u$ in constant.

The modal reduction method shown in the previous chapter presents an intrinsic inefficiency when applied to such parametric systems. The modal properties of the FE model are in fact parameter-dependent, so in turn also the reduction basis exhibits such dependency ($\mathbf{\Psi} = \mathbf{\Psi}(\mathbf{p})$). Applying the projection results in:

$$\begin{aligned}\mathbf{\Psi}^T(\mathbf{p})\mathbf{M}_f(\mathbf{p})\mathbf{\Psi}(\mathbf{p})\ddot{\mathbf{q}}_{f_r}(t) + \mathbf{\Psi}^T(\mathbf{p})\mathbf{C}_f(\mathbf{p})\mathbf{\Psi}(\mathbf{p})\dot{\mathbf{q}}_{f_r} + \\ + \mathbf{\Psi}^T(\mathbf{p})\mathbf{K}_f(\mathbf{p})\mathbf{\Psi}(\mathbf{p})\mathbf{q}_{f_r} = \mathbf{\Psi}^T(\mathbf{p})\mathbf{S}_{u,r}\mathbf{u}\end{aligned} \tag{3.2}$$

Once the projection step is done, the parametric dependency does not appear explicitly in the matrices of the ROM. Even the parametric dependency of the reduction basis is complex to define explicitly. The update of the model for a given set of parameters requires then to re-create the FOM, to compute the reduction basis and project the equations. This procedure would need to be done for each new parameter value, resulting in infeasible computational times for applications such as optimization or estimation where a large number of updates is required.

Several challenges arise from the need to achieve ROMs that can be easily and efficiently updated for parameter variations. The explicit parametric dependency can be retained by applying a pMOR technique. A comprehensive overview on projection-based pMOR techniques can be found in the work by Benner et al. [20]. As a first challenge, the dependency on parameters of $\mathbf{\Psi}$ needs to be properly treated by either making it explicit, approximating or removing it. Typical ways to approach the issue are by constructing a global reduction space that represents the entire parameter space, or by sampling and interpolating local bases. Interpolation approaches have the main disadvantage of requiring the sampling and storing of ROMs for a large pool of parameter combinations in the offline phase, together with pre-processing steps and added computational efforts in the online phase. Furthermore, intermediate steps are required in order to ensure that the reduced coordinates are consistent during the interpolation of the local reduction spaces. Global approaches, on the other end, can be more efficient but it is often complex to identify set of bases representative of the entire parameter space. A second challenge consists in the efficient computation of the FOM's matrices to be reduced. Particular properties of the system matrices can be exploited in order to make the reduction step efficient.

This chapter will present a novel pMOR methodology that focuses on material parameters and employs a constant reduction basis. The efficiency of the proposed approach relies on the affine structure of the dependency of system matrices on the parameter of interest, that will be demonstrated in the next section.

3.3 Affine dependencies for different material models and parameters

The system matrices should exhibit an affine dependency on the considered parameters to guarantee the efficiency of the global basis pMOR approach. The affine representation for a generic matrix $\mathbf{X}(\mathbf{p})$ can be stated as:

$$\mathbf{X}(\mathbf{p}) = \mathbf{X}^0 + \sum_i \mathbf{X}^i f^i(\mathbf{p}) \tag{3.3}$$

where $\mathbf{X}^0$ is the constant term, $\mathbf{X}^i$ are the affine components and $f^i(\mathbf{p})$ are the affine functions.

It is possible to demonstrate that, if an affine relationship is present in the constitutive relation of the model's material, this is preserved also in the element's and mesh's stiffness matrix. Let us assume that $\mathbf{E}$ has the form:

$$\mathbf{E}(\mathbf{p}) = \mathbf{E}^0 + \sum_i \mathbf{E}^i f^i(\mathbf{p}) \tag{3.4}$$

If the material properties are assumed constant over the element volume, it follows from Equation 2.14 and Equation 3.4 that:

$$\begin{gathered} \mathbf{K}_{f,e}(\mathbf{p}) = \int_V \mathbf{B}^T \left(\mathbf{E}^0 + \sum_i \mathbf{E}^i f^i(\mathbf{p}) \right) \mathbf{B} dV = \\ \left(\int_V \mathbf{B}^T \mathbf{E}^0 \mathbf{B} dV \right) + \sum_i \left(\int_V \mathbf{B}^T \mathbf{E}^i \mathbf{B} dV \right) f^i(\mathbf{p}) = \\ \mathbf{K}^0_{f,e} + \mathbf{K}^i_{f,e} f^i(\mathbf{p}) \end{gathered} \tag{3.5}$$

Using Equation 2.21 then:

$$\begin{gathered} \mathbf{K}_f(\mathbf{p}) = \sum^{n_e} \mathbf{S}_e \left(\mathbf{K}^0_{f,e} + \mathbf{K}^i_{f,e} f^i(\mathbf{p}) \right) \mathbf{S}_e^T = \\ \left(\sum^{n_e} \mathbf{S}_e \mathbf{K}^0_{f,e} \mathbf{S}_e^T \right) + \left(\sum^{n_e} \mathbf{S}_e \mathbf{K}^i_{f,e} \mathbf{S}_e^T \right) f^i(\mathbf{p}) = \\ \mathbf{K}^0_f + \mathbf{K}^i_f f^i(\mathbf{p}) \end{gathered} \tag{3.6}$$

Following the same procedure, it can be shown that the affine relationship is retained from the element to the mesh level also for the mass matrix, so that its affine form is:

$$\mathbf{M}_f(\mathbf{p}) = \mathbf{M}^0_f + \mathbf{M}^i_f f^i(\mathbf{p}) \tag{3.7}$$

The proportional damping model used in this work has the advantage of retaining the parametric dependencies of mass and stiffness matrices also for the damping matrix:

$$\mathbf{C}_f(\mathbf{p}) = \alpha(\mathbf{M}^0_f + \mathbf{M}^i_f f^i(\mathbf{p})) + \beta(\mathbf{K}^0_f + \mathbf{K}^i_f f^i(\mathbf{p}) \tag{3.8}$$

In the following sections, the affine representation of the matrices of the system is explicitly described for different material models and for lumped components.

3.3.1 Generic anisotropic materials

For the generic anisotropic material, every entry of the upper triangle and the main diagonal of $\mathbf{E}$ is an independent parameter (Equation 2.2), so:

$$\mathbf{E} = \sum_{i=1}^{6} \sum_{j=i}^{6} \mathbf{1}_{ij} E_{ij} \tag{3.9}$$

where $\mathbf{1}_{ij}$ is a zeros matrix with ones in position (i, j) and (j, i). The resultant stiffness matrix is:

$$\mathbf{K}_f^{an}(\mathbf{p}) = \sum_{i=1}^{6} \sum_{j=i}^{6} \mathbf{K}^{E_{ij}} E_{ij} \tag{3.10}$$

The mass matrix of a single element, assuming constant material parameters over the volume, can be represented with an affine dependency on the material density ρ as:

$$\mathbf{M}_{f,e}^{an} = \mathbf{M}_e^{\rho} \rho \tag{3.11}$$

Following the same considerations made for the stiffness matrix, the affine representation of the parameterized mass matrix is:

$$\mathbf{M}_f^{an}(\mathbf{p}) = \mathbf{M}^{\rho} \rho \tag{3.12}$$

3.3.2 Isotropic materials

The constitutive matrix of an isotropic material (Equation 2.3) has an affine relationship with the Young's modulus E and Poisson coefficient ν of the material, that can be simplified employing the Lame parameters:

$$\mathbf{E} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \frac{E\nu}{(1+\nu)(1-2\nu)} + \begin{bmatrix} 2\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \frac{E}{2(1+\nu)} = \mathbf{E}^{\lambda}\lambda + \mathbf{E}^{\mu}\mu \tag{3.13}$$

It is clear that the two Lamé parameters themselves are the needed affine functions if the system is parameterized for both E and ν. Also if only ν is considered, the affine functions are still the two Lamé parameters. The affine representation of the parametric stiffness matrix is then:

$$\mathbf{K}_f^{is}(\mathbf{p}) = \mathbf{K}^{\lambda}\lambda + \mathbf{K}^{\mu}\mu \tag{3.14}$$

In an equivalent way, E is itself the affine function needed if ν is not considered in the parameterization, so that:

$$\mathbf{K}_f^{is}(\mathbf{p}) = \mathbf{K}^{E} E \tag{3.15}$$

The mass matrix results equivalent to the anisotropic case, given that the mass in both cases is modeled in the same way:

$$\mathbf{M}_f^{is}(\mathbf{p}) = \mathbf{M}^{\rho}\rho \tag{3.16}$$

3.3.3 Composite laminate materials

For laminate materials with an angle-ply configuration, under the assumption that every layer has same materials and thickness, the affine functions for two separate parameters, the angle of the fibers or the thickness of the layers, has been demonstrated in [27] and will be represented here for reference. It should be noted that no affine representation is currently available in literature to parametrize for both at the same time.

Angle ply - Angle

When the angle Θ is considered, only the stiffness matrix is dependent on it through 5 affine functions:

$$\mathbf{K}_f^c(\mathbf{p}) = \sum_{i=1}^{5} \mathbf{K}^{\Theta,i} f^{\Theta,i}(\Theta) \tag{3.17}$$

where:

$$\begin{aligned} f^{\Theta,1} &= cos^4(\Theta) \\ f^{\Theta,2} &= sin^4(\Theta) \\ f^{\Theta,3} &= cos^2(\Theta)sin^2(\Theta) \\ f^{\Theta,4} &= cos(\Theta)sin^3(\Theta) \\ f^{\Theta,5} &= cos^3(\Theta)sin(\Theta) \end{aligned} \tag{3.18}$$

Angle ply - Thickness

For the layer thickness h, three affine functions relative to the stiffness matrix are identifiable, so that:

$$\mathbf{K}_f^c(\mathbf{p}) = \sum_{i=1}^{3} \mathbf{K}^{h,i} f^{h,i}(h) \tag{3.19}$$

where

$$f^{h,1} = h \tag{3.20}$$

$$f^{h,2} = h^2 \tag{3.21}$$

$$f^{h,3} = h^3 \tag{3.22}$$

while for the mass matrix only the parameter h itself is sufficient.

3.3.4 Lumped components

Identifying the affine functions relative to lumped elements is straightforward, since they correspond to the stiffness, damping or mass coefficients associated with the elements themselves. This results in the following affine representations:

$$\mathbf{K}_f^l(\mathbf{p}) = \mathbf{K}^k k \tag{3.23}$$

$$\mathbf{C}_f^l(\mathbf{p}) = \mathbf{C}^c c \tag{3.24}$$

$$\mathbf{M}_f^l(\mathbf{p}) = \mathbf{M}^m m \tag{3.25}$$

3.3.5 Considerations on the affine representation

Once defined the affine representations for the material classes of interest, few additional considerations are required and will be presented in this section.

Affine Matrices assembly

For multi-material components, different sections of a single FE model are described with independent material properties. Once the affine representations for the single zones are defined, these can be assembled into the whole matrices. These still exhibit an affine relationship on parameters, given that the sum of affine representations is still affine. An extra constant term appears to account for parts of the model that are not parameterized:

$$\mathbf{K}_f(\mathbf{p}) = \mathbf{K}_f^0 + \sum \mathbf{K}_f^{an}(\mathbf{p}) + \sum \mathbf{K}_f^{is}(\mathbf{p}) + \sum \mathbf{K}_f^{comp}(\mathbf{p}) + \sum \mathbf{K}_f^{l}(\mathbf{p}) \tag{3.26}$$

$$\mathbf{M}(\mathbf{p}) = \mathbf{M}_f^0 + \sum \mathbf{M}_f^{an}(\mathbf{p}) + \sum \mathbf{M}_f^{is}(\mathbf{p}) + \sum \mathbf{M}_f^{comp}(\mathbf{p}) + \sum \mathbf{M}_f^{l}(\mathbf{p}) \tag{3.27}$$

$$\mathbf{C}(\mathbf{p}) = \mathbf{C}_f^0 + \alpha\mathbf{M}_f(\mathbf{p}) + \beta\mathbf{K}_f(\mathbf{p}) + \sum \mathbf{C}_f^{l}(\mathbf{p}) \tag{3.28}$$

Affine Components Extraction

FOMs are often obtained through commercial FE tools that do not allow the user to have direct access to the underlying formulation. In this case, the affine components cannot be directly extracted. They can still be computed by knowing the expected structure of the affine parametric dependency and having a sufficient number of FOMs for different parameter values.

Starting from a set of matrices calculated at n_s different parameter values, the so-called sampling points, a least squares problem is solved to calculate the affine components. For a generic matrix, given the samples $[\mathbf{p}_1 ... \mathbf{p}_{n_s}]$ and the corresponding matrices $[\mathbf{X}_1 ... \mathbf{X}_{n_s}]$, the affine components are calculated by solving for each element $X_{(a,b)}$ of the matrices:

$$\min_{X^0_{(a,b)},\dots,X^i_{(a,b)},\dots} \sum_{h=1}^{n_s} \left\| X_{h(a,b)} - X^0_{(a,b)} - \sum_i X^i_{(a,b)} f^i(\mathbf{p}_h) \right\| \tag{3.29}$$

where i represents the affine component index, as in Equation 3.3. This procedure requires the sampling of the FOM a number of times at least equal to the number of components to be identified. An n_s lower than the number of unknowns in the least squares problem would in fact result in an underdetermined problem. The full matrices can be stored and used for the identification during the sampling phase of the pROM creation (described below) at a minimum added computational cost.

Existence and compatibility

The main limit of relying on the affine representation in order to achieve an efficient model reduction is that, for certain parameters, such dependency might not exist or be complicated to retrieve theoretically.

Even if for a single parameter it is possible to retrieve an affine dependency, considering multiple parameters for the same model might render the affine decomposition infeasible. One such example is parametrizing for both fiber angle and thickness of the composite materials considered in this work. To the author's knowledge, no parametrization for this case is available in literature.

If a direct access to the FE formulation is not available and the least square procedure needs to be applied, an extra limitation arises. The affine functions cannot in fact be linearly dependent, otherwise the minimization problem would be ill-defined and have infinite solutions. Such is the case when a model is parameterized for the angle of a composite with a non-parametric portion represented in the affine dependency by a constant term. The affine functions relative to the fiber angle are in fact linearly dependent on the constant term ($f^{\Theta,1} + f^{\Theta,2} + 2f^{\Theta,3} = 1$), resulting in infinite solutions for the least-squares problem.

3.4 Parametric Model Order Reduction scheme

It has been discussed how the modal reduction method applied to a parametric problem results in a parameters-dependent reduction basis. The pMOR scheme proposed in this work employs a constant reduction basis instead, not depend on time or on the value of the parameters of interest:

$$\mathbf{q}_f \approx \mathbf{\Psi}\mathbf{q}_{f_r} \tag{3.30}$$

The Galerkin projection of the EOMs onto the reduced space yields the reduced parametric EOMs:

$$\mathbf{M}_{f_r}(\mathbf{p})\ddot{\mathbf{q}}_{f_r} + \mathbf{C}_{f_r}(\mathbf{p})\dot{\mathbf{q}}_{f_r} + \mathbf{K}_{f_r}(\mathbf{p})\mathbf{q}_{f_r} = \mathbf{S}_{u,r}\mathbf{u} \tag{3.31}$$

where $\mathbf{M}_{f_r}(\mathbf{p}) = \mathbf{\Psi}^T\mathbf{M}_f(\mathbf{p})\mathbf{\Psi}$, $\mathbf{C}_{f_r}(\mathbf{p}) = \mathbf{\Psi}^T\mathbf{C}_f(\mathbf{p})\mathbf{\Psi}$, $\mathbf{K}_{f_r}(\mathbf{p}) = \mathbf{\Psi}^T\mathbf{K}_f(\mathbf{p})\mathbf{\Psi}$ and $\mathbf{S}_{u,r} = \mathbf{\Psi}^T\mathbf{S}_u$.

Using Equation 3.3, a generic matrix with an affine dependency can be reduced as:

$$\begin{aligned} \mathbf{X}_r(\mathbf{p}) &= \mathbf{\Psi}^T\mathbf{X}(\mathbf{p})\mathbf{\Psi} = \\ &= \mathbf{\Psi}^T\mathbf{X}^0(\mathbf{p})\mathbf{\Psi} + \sum_i \mathbf{\Psi}^T\mathbf{X}^i\mathbf{\Psi}f^i(\mathbf{p}) = \\ &= \mathbf{X}_r^0 + \sum_i \mathbf{X}_r^i f^i(\mathbf{p}) \end{aligned} \tag{3.32}$$

The efficient reduced representation applies to every parametric matrix of the FE model, given that they all exhibit an affine dependency as previously shown.

3.4.1 Global reduction basis selection

The reduction space has to be properly chosen as to be representative of the full parameter space while keeping the size as small as possible in order to ensure computational efficiency [20]. In this work the global reduction basis $\mathbf{\Psi}$ is constructed by concatenating local bases $\mathbf{\Psi}_i$, corresponding to different parameter samples $\mathbf{p}_i$, and applying a Singular Value Decomposition (SVD) [116].

The reduction basis is representative of the system for a set of configurations that correspond to specific parameters values. This poses the challenge of identifying a global reduction space that properly represents the system's behavior over the range of parameter values of interest. This is particularly true for a modal basis, which contains a subset of the eigenmodes of the system, since the eigenmode shapes can vary greatly for different values of the parameters. The final reduction basis thus has to accurately represent possible deformations across the parameter range in its entirety while having a reasonably small dimension. The steps required for the basis selection are shown in Figure 3.1, and are further described below.

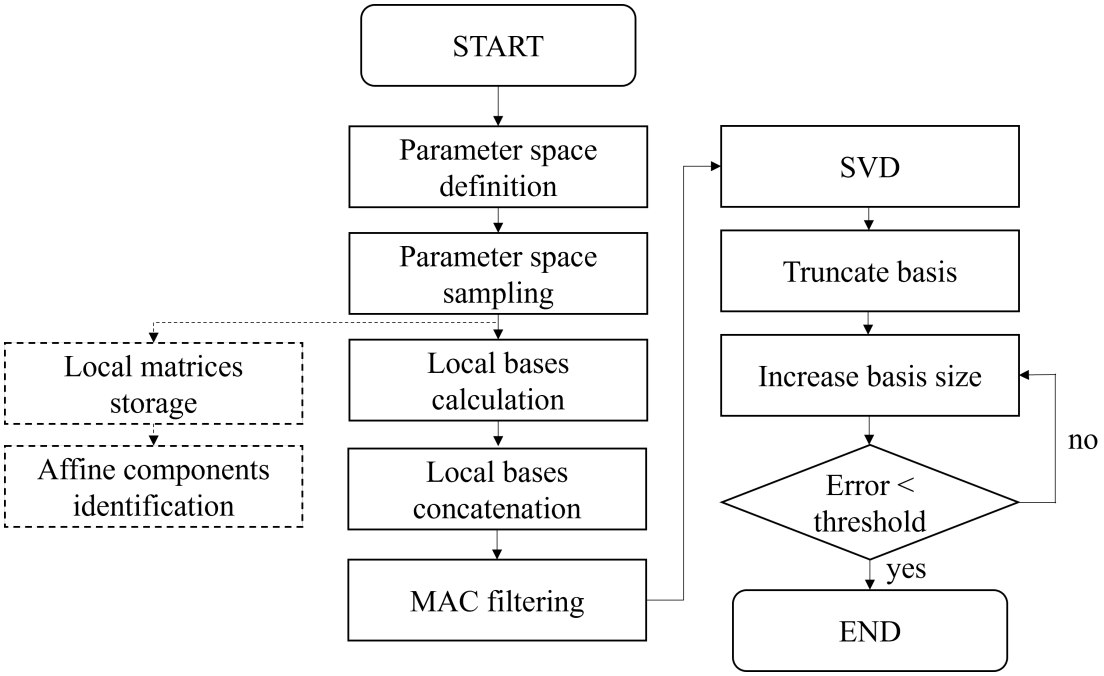


Figure 3.1: Basis selection workflow

Parameter Space Definition and Sampling

The parameter space $\mathbf{\Omega}$ is defined by setting upper and lower limits for each parameter based on the application of interest: $\mathbf{\Omega} = \{p_i \in \mathbb{R}, p_i^{min} < p_i < p_i^{max} \forall p_i\}$.

Once defined, the parameter space has to be sampled in order to get a set of configurations that is representative of the entire space. Several sampling methods have been proposed in literature for pMOR. Benner et al. [20] state that, for a small or moderate number of parameters ($n_p < 10$), a Latin Hypercube Sampling (LHS) method [104] is effective. Several methods have been tested for the application at hand (e.g. uniform sampling, LHS) and LHS proved to be the most effective. Therefore this method is adopted in this work. In case of a large parameter space, for which the LHS would require a high number of samples, other techniques (such as e.g. greedy search [93] or local sensitivity analysis [26]) could be adopted.

Local Bases and Matrices Generation

For each parameter sample $\mathbf{p}_i$ the full system, consisting of the matrices $\mathbf{M}_{f,i}$ and $\mathbf{K}_{f,i}$, is assembled. These FOM matrices are used to extract the corresponding local basis. This basis consists of a set of eigenmodes $\mathbf{\Psi}_{nm,i}$ concatenated with a set of static modes $\mathbf{\Psi}_{sm,i}$ [45, 22], as explained in Subsection 2.3.2, so that $\mathbf{\Psi}_i = \mathbf{\Psi}(\mathbf{p}_i) = [\mathbf{\Psi}_{nm,i}\mathbf{\Psi}_{sm,i}]$. The FOM matrices are stored for the affine component identification as explained in Section 3.3.5.

Global Reduction Basis Construction

The global basis is constructed by concatenating the local bases:

$$\mathbf{\Psi}_{global} = [\mathbf{\Psi}_1 ... \mathbf{\Psi}_{n_s}] \tag{3.33}$$

In general, this matrix of local bases can contain redundant and linearly dependent columns. In order to remove these linear dependencies, an SVD is performed:

$$\boldsymbol{\Psi}_{global} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T \tag{3.34}$$

The Left Singular Vectors (LSVs) $\mathbf{U}$ represent an orthonormal set of vectors that, spanning the range of $\boldsymbol{\Psi}_{global}$ [116], is suitable to be used as a reduction basis. In order to only keep the vectors that contain most of the information and therefore obtain a global basis of small enough dimension, a subset of the LSVs is used as the global truncated basis $\boldsymbol{\Psi}$. This truncated LSVs basis exhibits the property of being the optimal low-rank approximation for the full basis [116]. The choice of how many LSVs to retain is done through an iterative procedure that subsequently increases the number of retained basis vectors and compares the Frequency Response Functions (FRFs) of the pROM over a set of n_{in} input DOFs and n_{out} output DOFs and in a specific frequency range until the relative error becomes lower than a user-defined threshold.

Let $H_{ij}(\omega, \boldsymbol{\Psi}, \mathbf{p})$ be the FRF for a unitary force on the input DOF i and displacement of the output DOF j, using the pROM created using the basis $\boldsymbol{\Psi}$ evaluated for the parameter values $\mathbf{p}$. Furthermore, the frequency range is discretized as $[\omega_1, \ ... \ , \omega_{n_\omega}]$. At each iteration step k, the relative error between the FRFs calculated using the current reduction basis $\boldsymbol{\Psi}_k$ and the one relative to the previous step $\boldsymbol{\Psi}_{k-1}$ is calculated as:

$$e_k(\mathbf{p}) = \sum_{i=1}^{n_{in}} \sum_{j=1}^{n_{out}} \left(\frac{1}{n_\omega} \sum_{l=1}^{n_\omega} \frac{\|(\|H_{ij}(\omega_l, \boldsymbol{\Psi}_{k-1}, \mathbf{p})\| - \|H_{ij}(\omega_l, \boldsymbol{\Psi}_k, \mathbf{p})\|)\|}{\|H_{ij}(\omega_l, \boldsymbol{\Psi}_{k-1}, \mathbf{p})\|} \right) \tag{3.35}$$

This metric allows to achieve a good accuracy of the pROM in the frequency range of interest while limiting the number of evaluations needed for the FRFs to only the set of input/output DOFs. Furthermore, the metric also allows to compare both the shape of the responses and the amplitude. The error can be evaluated on a single parameter configuration of interest or averaged over a set of configurations. In this work, the error is calculated for the parameter values at the center of the sampled range.

3.4.1.1 Correspondent strain basis reconstruction

Of particular interest is the calculation of strain measurements starting from the reduced coordinates relative to the global reduction basis using Equation 2.32. If $\mathbf{B}_e$ is known, the strain basis matrix $\mathbf{N}_s$ can be directly calculated as $\mathbf{N}_s = \overline{\boldsymbol{\vartheta}}^T \mathbf{B}_e \mathbf{S}_e^T \boldsymbol{\Psi}$. In the case in which the reduction basis is extracted from a commercial software, the strain-displacement matrix is not known. The local strain basis $\boldsymbol{\epsilon}_i$ relative to one sample Ψ_i can be extracted from the commercial FE software and employed to calculate the local strain basis matrix $\mathbf{N}_{s,i} = \overline{\boldsymbol{\vartheta}}^T \boldsymbol{\epsilon}$. These can then be concatenated in $\mathbf{N}_{s,global}$, that needs then to be made consistent with the SVD truncation operation done to extract the basis $\boldsymbol{\Psi}$.

From Equation 3.34 and if only the first n left singular vectors are retained, it follows that:

$$\boldsymbol{\Psi}_{global,n} \mathbf{V}_n \boldsymbol{\Sigma}_n^{-1} = \mathbf{U}_n \approx \boldsymbol{\Psi} \tag{3.36}$$

The strain basis can then be approximated as:

$$\mathbf{N}_s = (\overline{\boldsymbol{\vartheta}}^T \mathbf{B}_e \mathbf{S}_e^T) \boldsymbol{\Psi} \approx (\overline{\boldsymbol{\vartheta}}^T \mathbf{B}_e \mathbf{S}_e^T) \mathbf{U}_n = (\overline{\boldsymbol{\vartheta}}^T \mathbf{B}_e \mathbf{S}_e^T) \boldsymbol{\Psi}_{global,n} \mathbf{V}_n \boldsymbol{\Sigma}_n^{-1} \tag{3.37}$$

and finally:

$$\mathbf{N}_s = \mathbf{N}_{s,global,n} \mathbf{V}_n \boldsymbol{\Sigma}_n^{-1} \tag{3.38}$$

This final equation allows to transform the concatenated strain basis to make it consistent with the truncated displacement basis obtained through the SVD.

3.4.2 Filtering of concatenated basis using MAC metric

While the simple concatenation and truncation through SVD of the basis is sufficient to achieve a representative and small reduction space, few considerations are still required.

The local bases that are concatenated, while relative to different parameter values, can contain redundant information. Given the iterative approach used in this work, it is possible to extract a number of singular vectors higher than the number of independent vectors in the concatenated basis. In such a case, the SVD procedure returns vectors orthogonal to the previous ones but with no real information, that can be seen as spatial noise. The presence of such vectors in the basis can cause numerical instability. Any excitation of these vectors would in fact result in an unpredictable and unrealistic deformation of the structure which would not respect boundary conditions and continuity properties of the original model. To tackle this issue, this work proposes a filtering of the concatenated basis based on the Modal Assurance Criterion (MAC) metric [114, 11]. The MAC between the basis vectors $\mathbf{\Psi}^i$ and $\mathbf{\Psi}^j$ can be calculated as:

$$MAC_{ij} = \frac{|\mathbf{\Psi}^{iT}\mathbf{\Psi}^j|^2}{\left(\mathbf{\Psi}^{iT}\mathbf{\Psi}^j\right)\left(\mathbf{\Psi}^{jT}\mathbf{\Psi}^i\right)} \tag{3.39}$$

The **MAC** matrix relative to a set of vectors is defined as a matrix with the MAC_{ij} in position (i, j). This metric is used in modal analysis to determine the similarity of two modal shapes. A value close to 1 indicates that the two shapes are identical (they can still be scaled by a constant factor).

After the concatenation, the MAC filtering is done by computing the MAC matrix for the whole set of vectors and iteratively removing one from each couple that has a MAC higher than a certain threshold, chosen as 0.99 in this work. This procedure allows to obtain a set of basis vectors that still contain the same information as the starting one but with no redundancy.

A simple example is presented here to prove how a linear dependency can arise in the basis vectors. Let us consider a FE model with one material property for the whole mesh and only one type of elements. In this case, the matrices are $\mathbf{M}_f = \rho\mathbf{M}_f^\rho$ and $\mathbf{K}_f = E\mathbf{K}_f^E$. For a specific set of parameters (E_i, ρ_i), the basis is extracted through the eigenvalue problem as:

$$\left(\mathbf{K}_{f,i} - \mathbf{M}_{f,i}\mathbf{\Omega}_i\right)\mathbf{\Psi}_i = \mathbf{0} \tag{3.40}$$

that, using the affine representations for the matrices yields:

$$\left(\mathbf{K}_f^E - \frac{\rho_i}{E_i}\mathbf{M}_f^E\mathbf{\Omega}_i\right)\mathbf{\Psi}_i = \mathbf{0} \tag{3.41}$$

If now a second set of parameters (E_j, ρ_j) is considered, the equation becomes:

$$\left(\mathbf{K}_f^E - \frac{\rho_j}{E_j}\mathbf{M}_f^E\mathbf{\Omega}_j\right)\mathbf{\Psi}_j = \mathbf{0} \tag{3.42}$$

A solution to this problem is obtained by imposing $\mathbf{\Omega}_j = \frac{\rho_i}{E_i}\frac{E_j}{\rho_j}\mathbf{\Omega}_i$ and $\mathbf{\Psi}_j = \mathbf{\Psi}_i$. Given the uniqueness of the eigenspace of a set of matrices, this proves how the eigenvectors of the system are unaltered by a change in parameters, where only the eigenvalues are scaled consequently. This in turn demonstrates how, in this particular case, the concatenated basis cannot theoretically contain more information than one of the local bases. The behavior over the entire parameter space can therefore be fully represented by one sample.

The FE model of a link of the robotic manipulator, that will be introduced in the following section, has been used to generate a pROM with the discussed parametric dependency in order to

showcase the behavior of its eigenvectors. The local bases contain 22 modes, and 5 samples have been acquired. The final size of the global basis, if no MAC filtering is applied, is 28. Figure 3.3 shows a graphical representation of the basis functions in the global basis. In particular, the ones up to the 22nd vector exhibit a regular shape, while the last ones are pure spatial noise as expected.

Figure 3.2 shows the MAC matrix of the concatenated basis before and after application of the MAC filtering. It can be seen that all the local basis sets are the same, as theoretically expected. Only one local basis is then retained regardless of the number of samples.

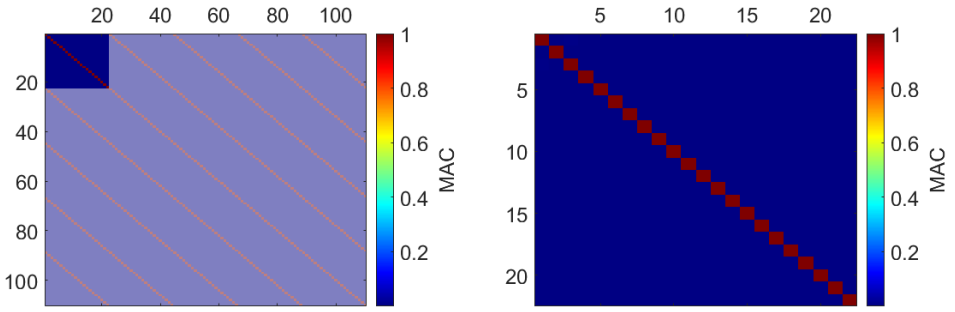


Figure 3.2: MAC matrix of the concatenated basis (left) and of the MAC-filtered one (right). On the left plot, the dim rows and columns are correspondent to vectors removed by the filtering procedure.

The analytical explanation given for the extremely simple example above cannot be generalized for more complicated models, therefor the need for a numerical filtering of the starting basis.

The proposed approach for the iterative basis size selection presents the advantage of not requiring any additional computational effort after the initial sampling, and the easy choice of the threshold for the MAC value given its upper limit of 1 (0.99 is employed in this work). Furthermore, filtering the concatenated basis before the SVD allows to reduce its computational cost. Other approaches for the basis size truncation are possible and should be investigated in future research efforts. One of these consists in using the error with respect to a FOM in the convergence analysis, resulting in a better guarantee of accuracy but at the cost of the expensive evaluation of the FOM frequency domain solution. This issue could be mitigated by evaluating the error with respect to a ROM, reducing the extra computational effort but also the accuracy. Finally, a common approach in SVD applications is truncating any LSV with a relative SV lower than a certain threshold. Such approach might prove ideal but the choice of the threshold value is more complicated. This could be done for example by setting it not directly on the SVs but on the percentage of energy retained in the truncated basis [116] or by identifying a so-called elbow in the curve of decay of the SVs [27]. The downside of this approach is the higher computational effort of the SVD on the full concatenated basis.

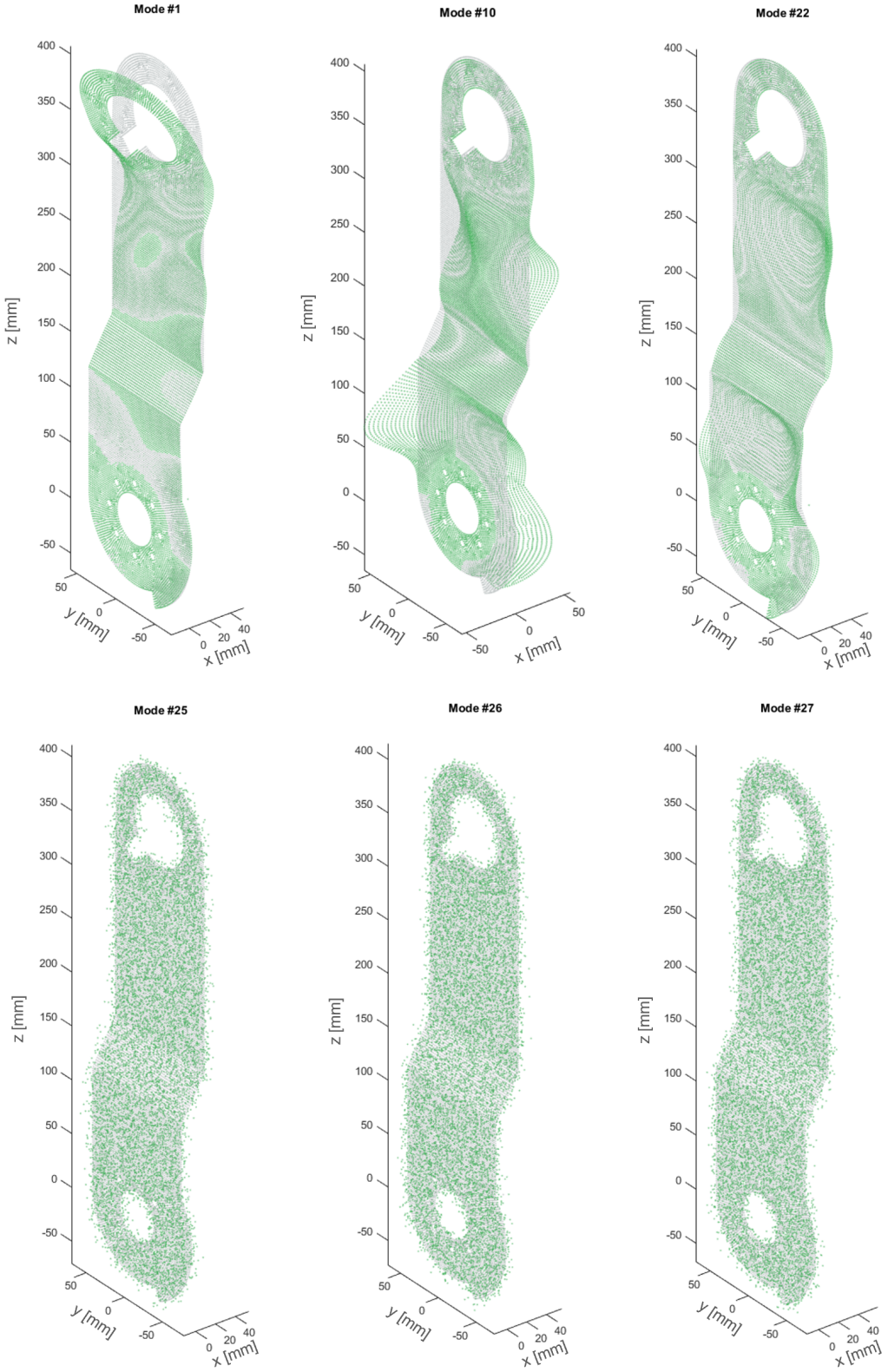


Figure 3.3: Graphical representation of the components of the global reduction basis of a pROM with spatial noise components (bottom) caused by the linear dependency of the original set of concatenated local bases.

3.5 Case study n. 1: isotropic wind turbine blade

A numerical validation of the proposed pMOR scheme against a standard ROM is presented in this section for a component with isotropic material. The performance of the parametric model is verified for simulations both in frequency and in time domain on several different combinations of the parameter values not initially sampled.

3.5.1 Description of the benchmark model

The component considered for the validation with isotropic material is a scaled wind turbine blade (Figure 3.4) present at the testing facilities of Siemens Industry Software NV in Leuven (Belgium). The blade consists of Titanium Gr23 and has been produced using additive manufacturing followed by heat treatments. A wind turbine blade is a particularly interesting case given its complex geometry that allows to obtain a non-trivial dynamic response with relatively simple experimental setup and boundary conditions.

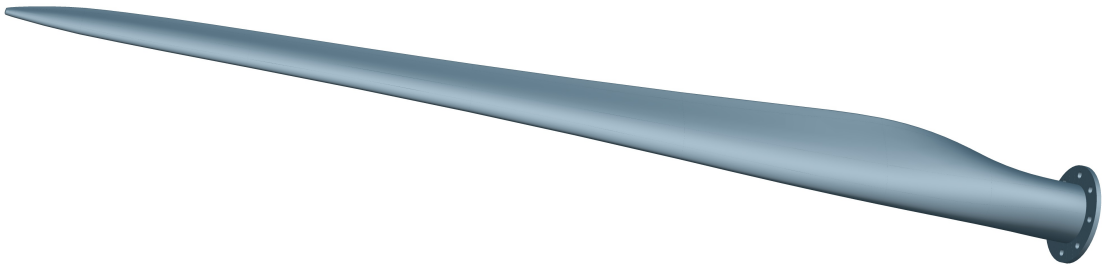
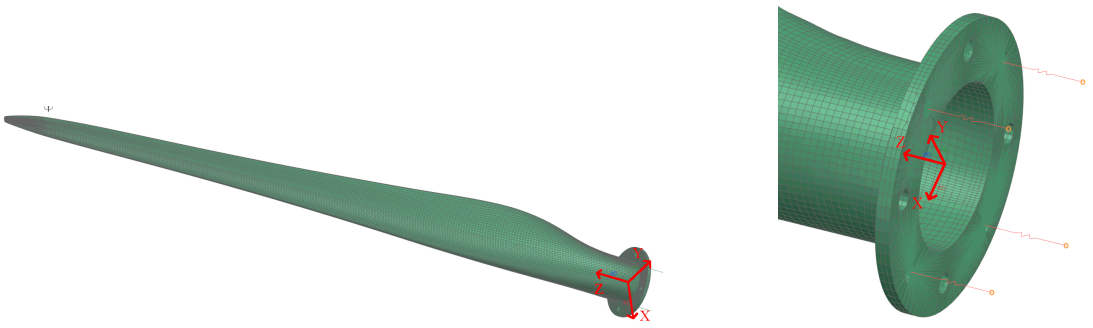


Figure 3.4: CAD model of the scaled wind turbine blade

The linear FE model of the blade used in this work (Figure 3.5a) mainly consists of linear hexahedral elements and a minority of linear pentahedral elements, for a total of 65264 nodes and 391584 DOFs. Rigid elements connect the entire face of the flange to four spring elements that model the compliance of the bolts, as shown in Figure 3.5b.

The pROM has been created considering the Young's modulus and the density as parameters. This choice aims at giving a smaller set of representative parameters that can be easily represented in results. Two different versions of the pROM have been created; in a first one the parameters are sampled in a range between -40% and 40% of the reference values, while in the second



(a) (b)

Figure 3.5: Finite element model of the blade

one the range has been reduced to be between -10% and 10%. This is aimed at analyzing the extrapolation performance of the models outside the range for which they are initially generated.

A commercial FE solver [1] has been used to extract the full model information needed for the pROM generation. The least-squares procedure identification described in Subsection 3.4.1 has been applied for the affine component identification. The settings used for the creation of both pROMs can be seen in Table 3.1. The number of eigenvectors of the structure with an associated eigenfrequency in the range of interest is higher for configurations with low stiffness and high density. The size of the local basis has therefore been chosen to contain the entire set of modes even for these extreme cases. Figure 3.6 shows the convergence of the relative error in the basis size selection procedure, in which the basis starts with a dimension of 24 and is increased by 12 basis vectors at each iteration. The relative error decreases monotonically to reach a value beyond the threshold in 3 iterations. The final dimension of the pROM is 60, with a reduction factor of more than 6000 with respect to the full model.

Table 3.1: Blade's pROM information

size of Ψ_i	24 (15 eigenmodes + 9 static modes)
number of samples	5
relative error threshold	10^{-3}
frequency range	0-500Hz
number of nodes	65264
DOFs of the full model	391584
size of Ψ	60

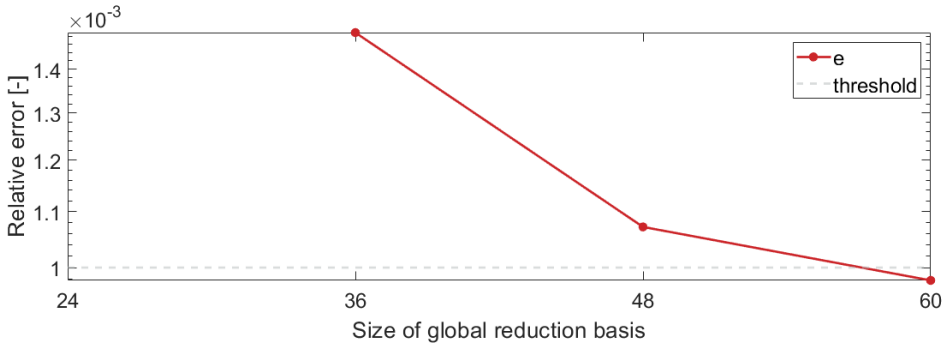


Figure 3.6: Relative error for the global basis selection iterative procedure of the blade's pROM. The basis size starts at 24 and the first relative error is calculated at the second step where the size increased to 36. The same convergence behavior is exhibited by both pROM models of the blade.

3.5.2 Experimental model updating

The numerical model of the blade has been updated in order to match the physical test setup (Figure 3.7) in a work by Vettori et al. [30].

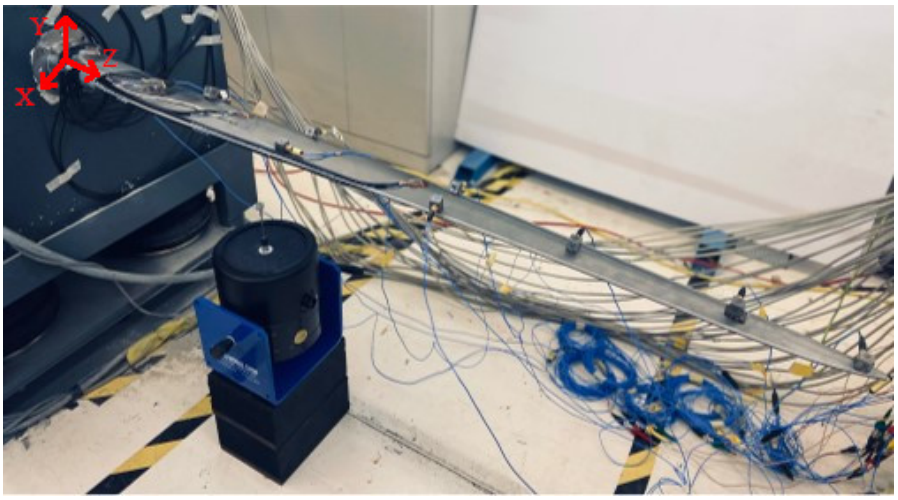


Figure 3.7: Experimental setup for the scaled wind turbine blade

The main parameters of the FE model (Figure 3.5a) representing the physical blade, namely Young's modulus, Poisson's ratio, density and stiffness of the bolts, have been identified by a modal testing procedure and model updating [30]. Impact tests measurements have been used to identify the modal properties of the blade. Modal frequencies and shapes have been extracted from input-output measurements using the Polymax algorithm implemented inside Simcenter Testlab [3]. The updating of the numerical model has been done through a Simcenter Nastran Solution 200 [2] that applies an optimization procedure to minimize the error between numerical and experimental frequencies and to maximize the MAC for the modes of interest. The damping parameters for the proportional Rayleigh model have been identified using an optimization procedure aimed at finding the values that minimize the error between the experimental and numerical FRFs for one input point and two representative acceleration outputs. The identification has been carried out in the 0-50Hz range. The final identified parameters can be seen in Table 3.2, while the matching of numerical and experimental modes is shown in Table 3.3.

Table 3.2: Blade material parameters

$E\ [GPa]$	$\rho\ [Kg/m^3]$	ν	α	β	$K\ [N/mm]$
116.04	4873	0.34	0.02	0.0005	33167.2

Table 3.3: Experimentally identified model parameters

Mode #	**Ref. freq. [Hz]**	**Model freq. [Hz]**	**Error**	**MAC**
1	25.020	24.758	-1.047%	0.985
2	55.004	54.512	-0.894%	0.977
3	90.426	93.405	3.294%	0.949
4	193.201	187.902	-2.743%	0.909
5	236.416	242.358	2.513%	0.881
6	343.176	349.907	1.932%	0.916

3.5.3 Application cases for validation

In the time-domain validation procedure, a broadband continuous random load (Figure 3.9) is applied on the bottom face and two measurements are extracted: the strain on the center of the top face and the acceleration on the vertical direction close to the tip (Figure 3.8). For frequency domain results, the input is a unit force applied and the output is the displacement of its application point.

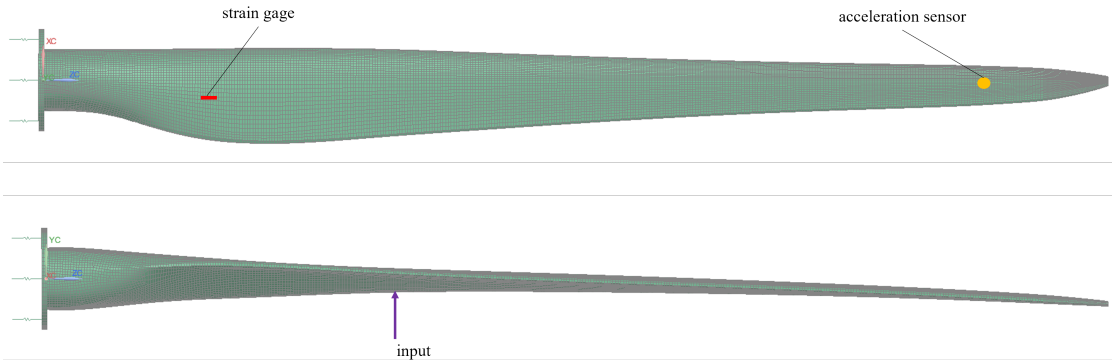


Figure 3.8: Input point and sensors on the blade used for validation.

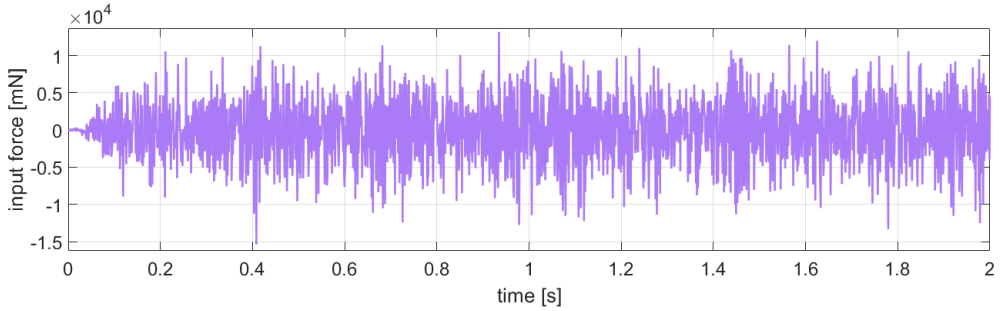


Figure 3.9: Continuous random input

3.5.4 Frequency domain analysis

In order to check that the pROM properly interpolates the behavior of the model in the frequency range and parameter space considered, the FRFs can be calculated and compared for a set of input/output DOFs of interest with the ones of non-parametric ROMs created for a specific set of parameters. The results in Figure 3.10 show the FRFs of standard, non-parametric ROMs and the pROM for parameter values not used in the pROM generation, namely the nominal values as well as two extremes of the parameter range, for input and output on the force application point at the bottom face of the blade. A good correspondence can be observed.

In Figure 3.11 the average error (calculated as $\frac{|(|FRF_{pROM}|-|FRF_{ROM}|)|}{|FRF_{ROM}|}$) on the FRFs between ROM and pROM is plotted over a parameter range larger than the one used for generating the pROM, corresponding to 11×11 samples, for the pROM with the larger parameter space. This shows as, in the original range, the interpolation by the pROM is accurate and provides good results even far from sampling points. Outside of the range the error is still low, but it shows a

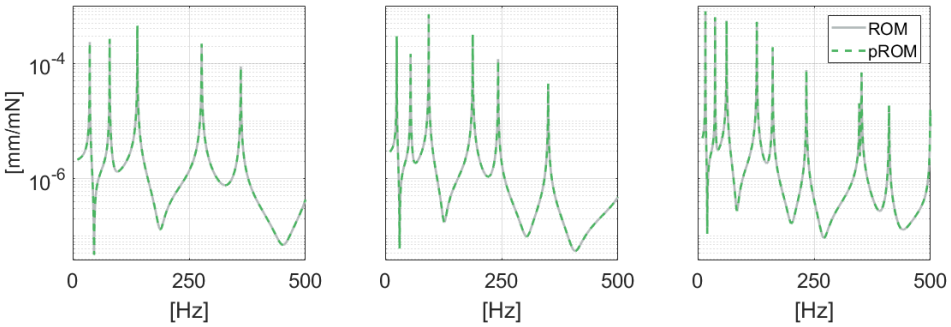


Figure 3.10: FRFs comparison on 3 parameter sets (High E and low rho on the left, reference in the middle, low E and high rho on the right).

tendency to grow in the low stiffness zone where the frequency range includes a higher number of eigenfrequencies associated with eigenmodes of the structure (Figure 3.12).

Figure 3.13 shows the average FRFs error relative to the pROM generated considering a smaller parameter space. It can be seen how the performance in this case is not influenced by the sampling. Such behavior can be explained with the simple parametric dependency of the considered model; the eigenmodes do not in fact change much over the parameter space, and the eigenvalues have a smooth and regular variation.

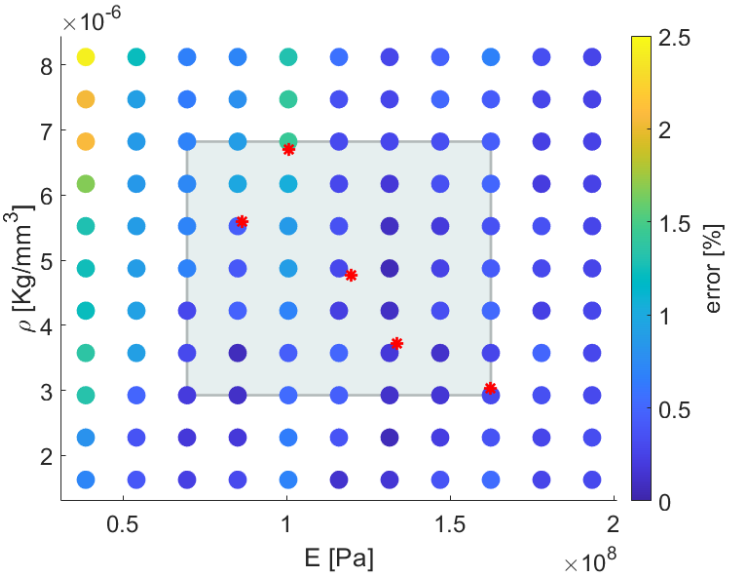


Figure 3.11: Relative error between ROM and pROM FRFs over the parameter space sampled on a 11x11 grid. The gray box and the red stars are respectively the parameter space and the sampled points used for the pROM generation.

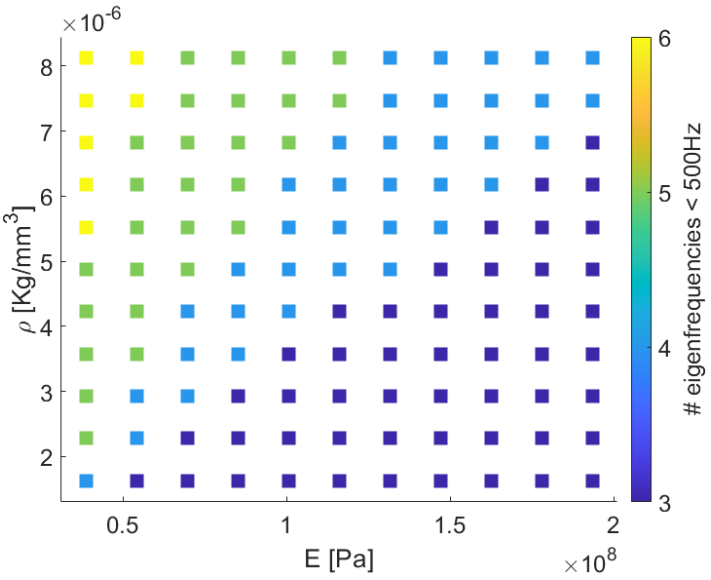


Figure 3.12: Number of eigenfrequencies in the considered frequency range for the blade model.

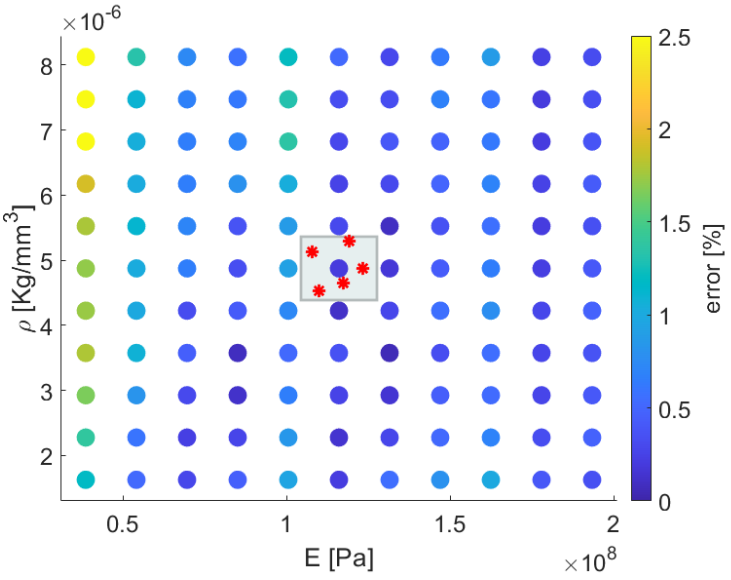


Figure 3.13: Relative error between ROM and pROM FRFs over the parameter space for the model generated by sampling over a smaller range of parameters.

3.5.5 Time domain analysis

Since instabilities can arise when employing a ROM for time domain simulations if not handled properly [13, 145], the comparison of FRFs might not be sufficient to properly asses the quality of the proposed pMOR technique. Therefore, a comparison is also carried out in the time domain. The results are shown in Figure 3.14, where a continuous random signal with a frequency content in the range of 0-500Hz is applied to the pROM. The acceleration is calculated at the same

location as the input force, while the strain is calculated at a location on the top face in the direction of the axis of the blade. A first-order BDF time integration scheme was used for the forward simulation presented in this section with an integration step $\Delta t = 0.1ms$. A good correspondence between the ROM and pROM can again be observed.

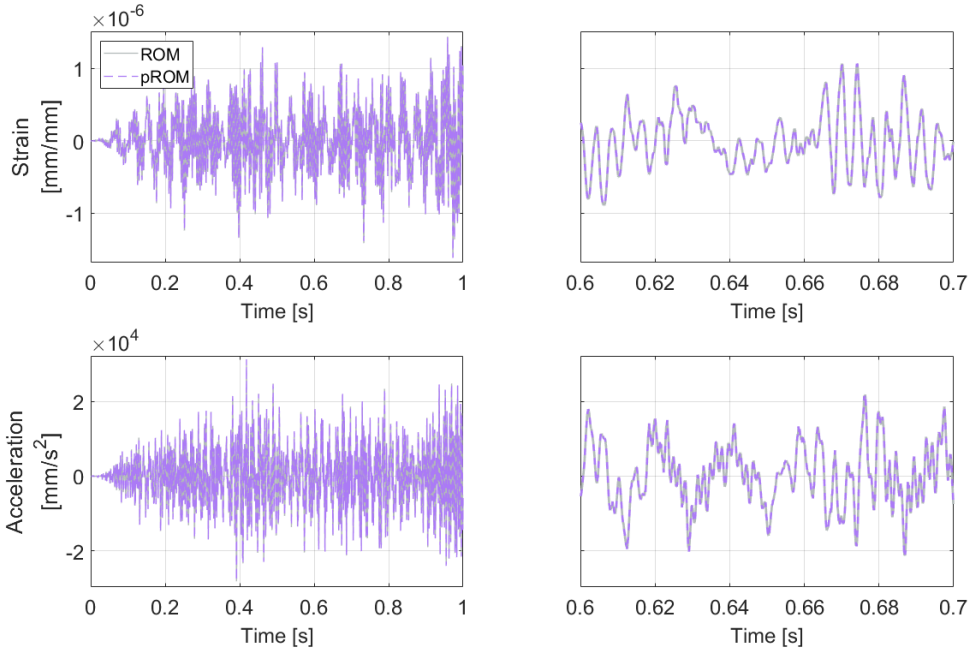


Figure 3.14: Comparison of time-domain simulations for a continuous random input. The right figures show a zoom of the 0.6s to 0.7s range.

3.6 Case study n. 2: composite robotic manipulator link

The system considered for the validation using composite materials is one of the links of a robotic manipulator that will be described in detail in Chapter 4. The link consists of a thin (3 mm) folded sheet of steel, whose geometry has been simplified with respect to the original to adapt it to composites manufacturing. More details on the geometry simplification will be given in Chapter 6.

3.6.1 Description of the benchmark model

The linear FE model of the link (Figure 3.15) consists of 4-nodes and 3-nodes shell elements, that are the most suitable choice for such a model with small thickness compared to the other two dimensions. The material considered is a Carbon Fiber Reinforced Polymer (CFRP). The laminate consists in 8 layers and the reference direction of the fibers is the global z axis. Rigid elements connect both faces in contact with the motors of the robot to two central nodes.

The pROM has been created considering only the fibers angle as a parameter. Also in this case, a model considering the full [0° 90°] range has been generated together with one sampling only the range [0° 45°]. Details on the settings and final results for both models can be seen in Table 3.4. The size required for this pMOR, with respect to the blade's one, is much larger. This

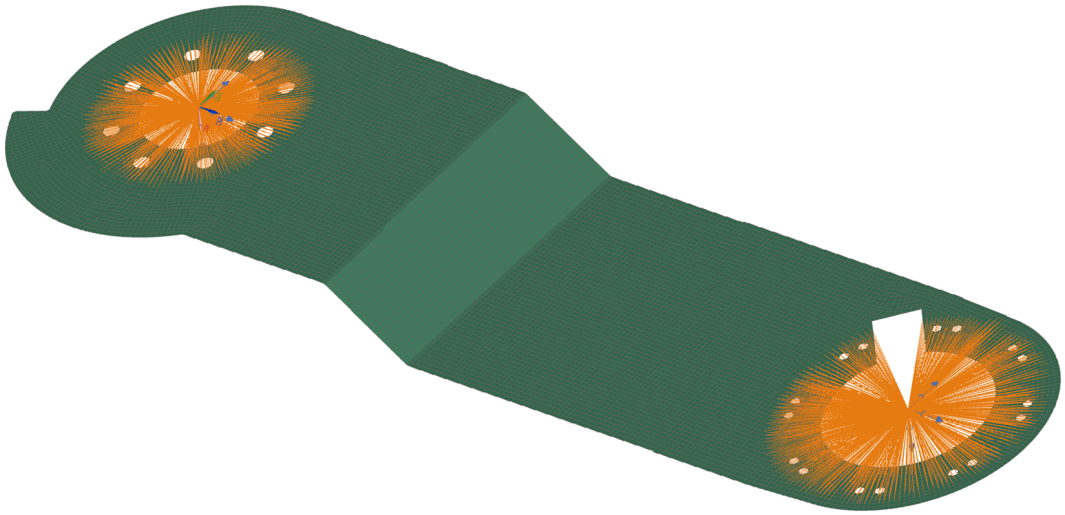


Figure 3.15: Finite element model of the Robolink's second link with simplified geometry. The orange lines are RBEs that connect a master node to the faces in contact with the motors in the manipulator.

can be explained with the larger number of terms in the affine dependency of the matrices. Also the width of the sampling space has an influence in this case, with the smaller one requiring a model 30% smaller.

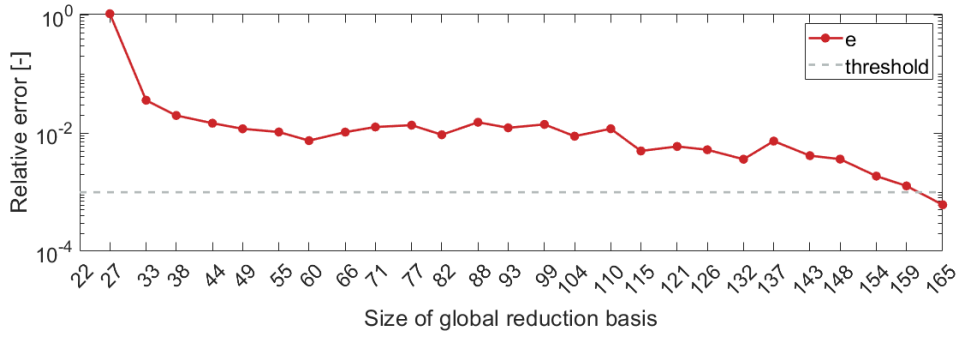
Table 3.4: Simplified Link 2's pROM information

size of Ψ_i	22 (16 eigenmodes + 6 static modes)	
number of samples	10	
relative error threshold	10^{-3}	
frequency range	0-500Hz	
number of nodes	12608	
DOFs of the full model	75648	
	Full range	**Half range**
size of Ψ	165	115

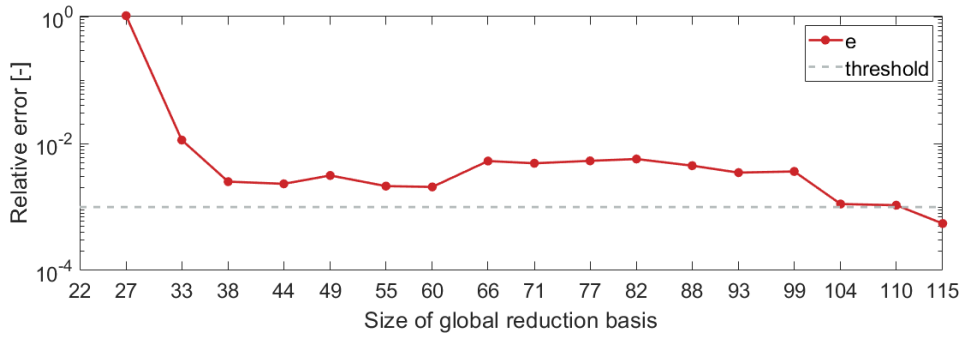
Figure 3.16a shows the error convergence behavior of the full-range pROM. For small sizes of the basis the pROM is not able to correctly reproduce the frequency responses of the ROM, and the convergence requires a large number of iterations. It can also be appreciated how a narrower parameter space (Figure 3.16b) requires a smaller basis size to be properly represented. In both cases, the complicated parametric dependency needed to approximate makes the convergence less smooth and not-monotonical, in contrast to what seen previously for the isotropic case.

3.6.2 Experimental model updating

The properties of the composite material employed in this work have been experimentally validated. A physical prototype of the component presented in this section has been manufactured in the laboratories of the University of Calabria. The single pre-impregnated plies have been



(a)



(b)

Figure 3.16: Relative error for the global basis selection iterative procedure of the composite link's pROM created on [0° 90°] (a) and [0° 45°] (b) parameter ranges. The basis size starts at 22 and the first relative error is calculated at the second step where the size increased to 27.

manually placed to achieve a final design with 8 layers in angle-ply configuration with $\Theta = 20°$. In an ideal condition, multiple components with different angle values should be available for testing in order to fully validate the parametric dependency of the component properties. Given practical budget and effort limitations, it has been chosen to test only one representative design. This already enables the identification of material properties, assuming the composite parameters (fiber angles, number of layers, thickness) as known. The angle value has been chosen based on the best designs obtained in the optimizations described in Chapter 6.

A preliminary weighting of the component has allowed to compute the material density, given that the geometry is known.

The link has been suspended elastically (Figure 3.17) in order to replicate free-free conditions. A set of 5 uniaxial accelerometers with axis on the out-of-plane direction has been mounted on the link (Figure 3.18). The choice is aimed at observing the first 5 modal shapes. The same impact test experimental procedure as in Subsection 3.5.2 has been followed. The results of the comparison of experimental and numerical modal properties are summarized in Table 3.5, showing a good matching between experimental and numerical modal data both in terms of frequency and MAC. It can be noted in particular how most of the mode couples exhibit a MAC value close to ideal, with only the third one exhibiting a lower value. Also the frequencies show a good matching, with errors always lower than 5%. The errors present even after the updating procedure can be attributed to manufacturing errors and uncertainty deriving from the thermal treatment executed in the curing phase of the component.

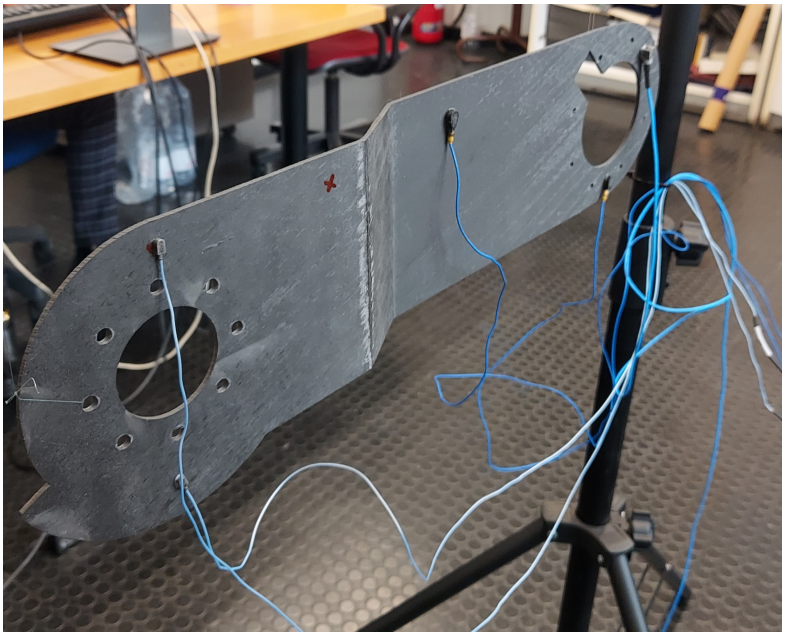


Figure 3.17: Experimental setup for the composite L2r.

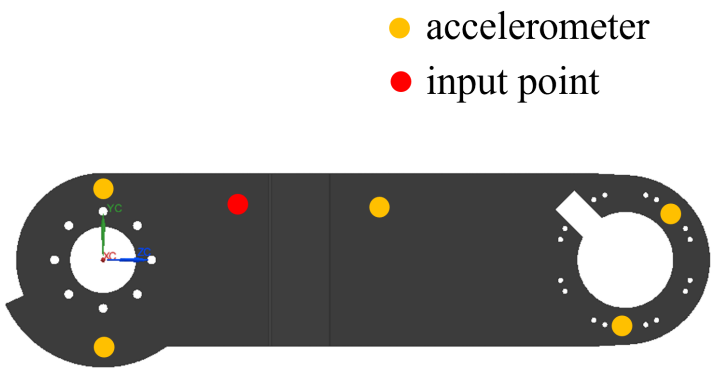


Figure 3.18: Position of input and output points for modal analysis on composite L2r.

Table 3.5: Modal matching for composite L2r

Mode #	**Ref. freq. [Hz]**	**Model freq. [Hz]**	**Error**	**MAC**
1	87.50	91.50	4.57%	0.968
2	149.00	148.50	0.34%	0.984
3	206.50	213.50	3.39%	0.801
4	276.50	283.50	2.53%	0.981
5	373.50	381	2.01%	0.947

The modal properties are based on real eigenvalues and eigenvectors, thus not keeping into account damping. In order to tune the damping parameters, the FRFs have been considered. The two Rayleigh parameters have been tuned in order to match the curves. Results show good match especially close to the resonances (Figure 3.19) and more generally on the shape of the curve. In particular, the most prominent peak exhibits an almost ideal matching. The mismatches can be mainly attributed to the approximate boundary conditions and the difficulty in ensuring a load direction normal to the face of the component in the testing phase.

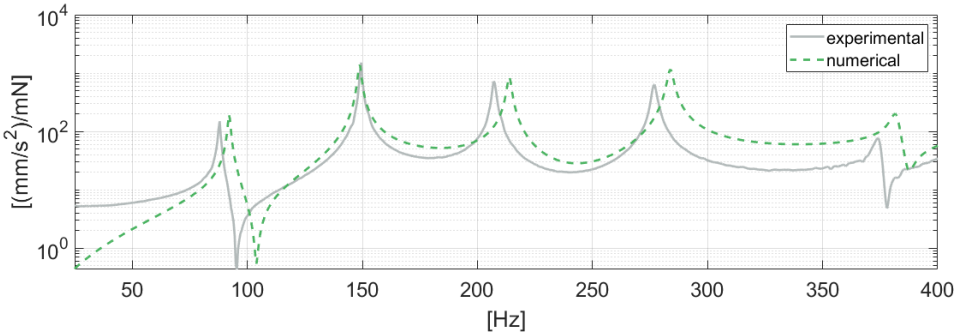


Figure 3.19: Comparison of experimental and numerical FRFs in free-free conditions for composite L2r.

The results presented in this section are based on nominal material properties. A model updating procedure on composite properties results in a more complicated problem given the large number of parameters to update. Given this issue and the already good matching demonstrated by the nominal properties, is has been chosen to not execute any further material model updating. It will be of interest for future research efforts to further improve the material model by a model updating.

Finally, the properties for the composite component are summarized in Table 3.6.

Table 3.6: Composite link model material reference parameters

E_m $[GPa]$	ν_m	ρ_m $[Kg/m^3]$	E_f $[GPa]$	ν_f	ρ_f $[Kg/m^3]$
3.00	0.33	1200	240	0.34	1800

V_f	Θ [°]	n_l	h $[mm]$	α	β
0.6	20	8	0.3	5	0.000003

3.6.3 Application cases for validation

For this application in which the component is isolated, one of the two sides is fixed and a load in the x direction is applied to the other (Figure 3.20). In the validation procedure in time domain, the same input as the previous case (Figure 3.9) is applied in the out-of-plane direction, and two measurements are extracted: the strain on the center and the acceleration on the input point (Figure 3.20). For frequency domain, force and displacement on the same point are used respectively as input and output to calculate the FRFs.

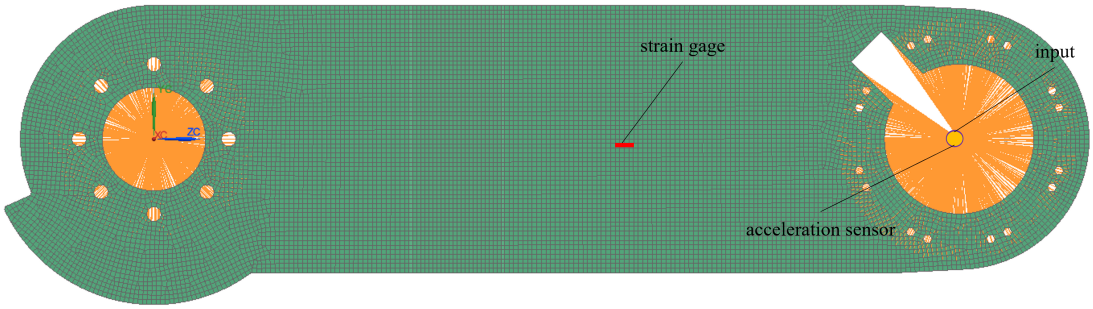


Figure 3.20: Input point and sensors on the composite link used for validation.

3.6.4 Frequency domain analysis

For the model with full sampling over the parameters space, the comparison of FRFs on 3 parameter values (0°, 45° and 90°) is shown in Figure 3.21. The curves show a good correspondence even for the extremes of the parameter range.

In Figure 3.22 the average error (calculated as $\frac{|(|FRF_{pROM}|-|FRF_{ROM}|)|}{|FRF_{ROM}|}$) over the parameter range is shown. The metric shows satisfying values over the entire range, with an increase around 40° where the modal behavior is more sensitive to parameters variations.

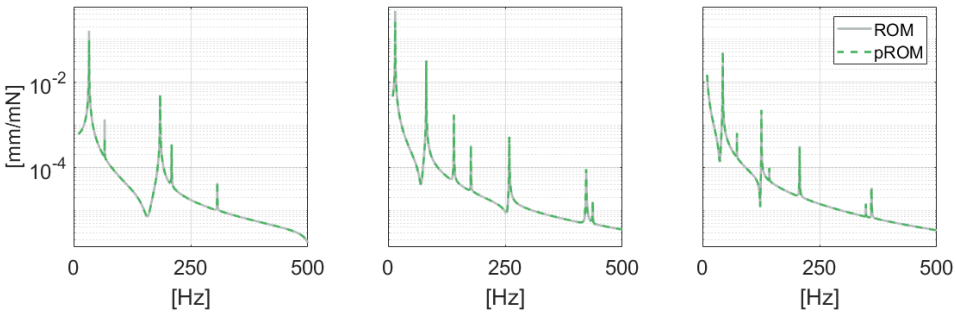


Figure 3.21: FRFs comparison on 3 angle values (0 on the left, 45 in the middle, 90 on the right).

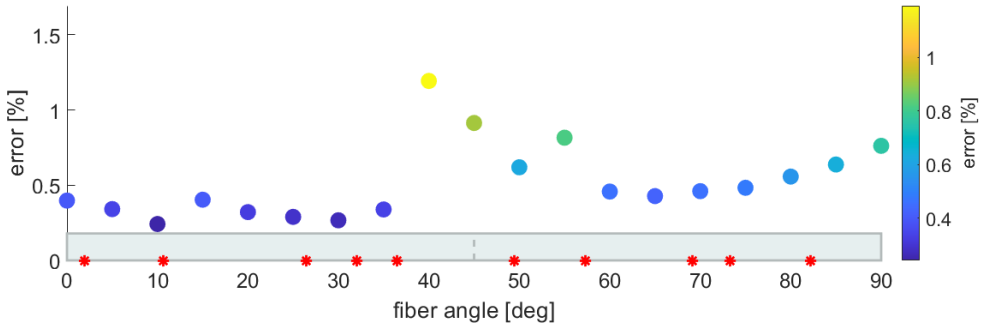


Figure 3.22: Relative error between ROM and pROM FRFs over the parameter space. The grey box and the red stars are, respectively, the parameter space and the sampled points used for the pROM generation.

The extrapolation behavior of the pROM for this composite application case differs from what seen in the previous one. In Figure 3.23 and Figure 3.24 it can be seen how the pROM generated on the range [0° 45°] is not able to capture the dynamic behavior of the system out of the sampled range. This can again be imputed to the much more complicated dependency of the modal properties on the parameter considered (Figure 3.25), that causes the modal shapes to be largely different in the unsampled area.

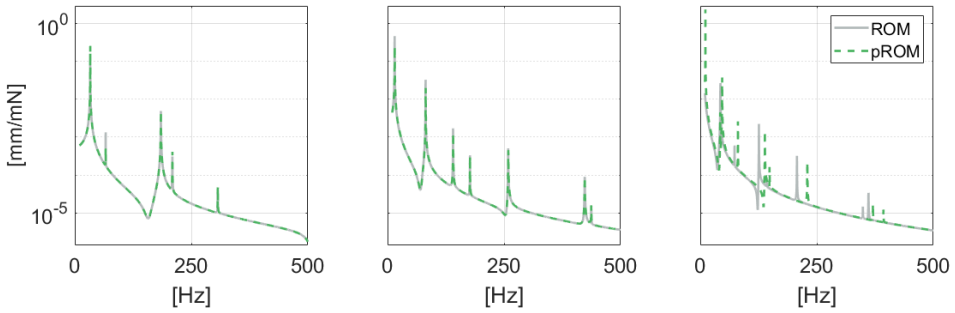


Figure 3.23: FRFs comparison on 3 angle values (0 on the left, 45 in the middle, 90 on the right) for a small sampling space.

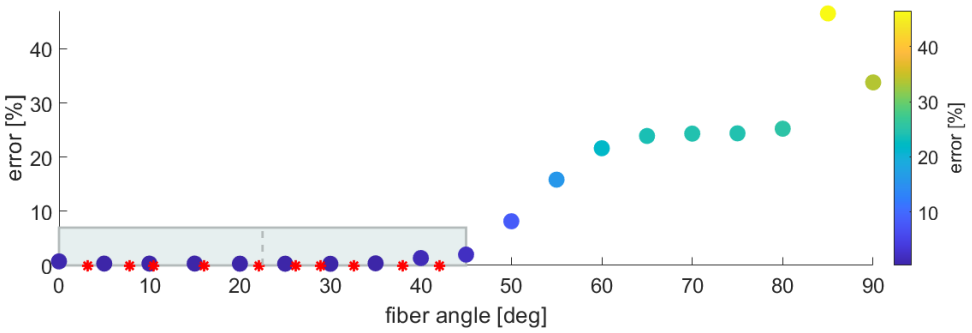


Figure 3.24: Relative error between ROM and pROM FRFs over the parameter space, for a small sampling space. The grey box and the red stars are, respectively, the parameter space and the sampled points used for the pROM generation.

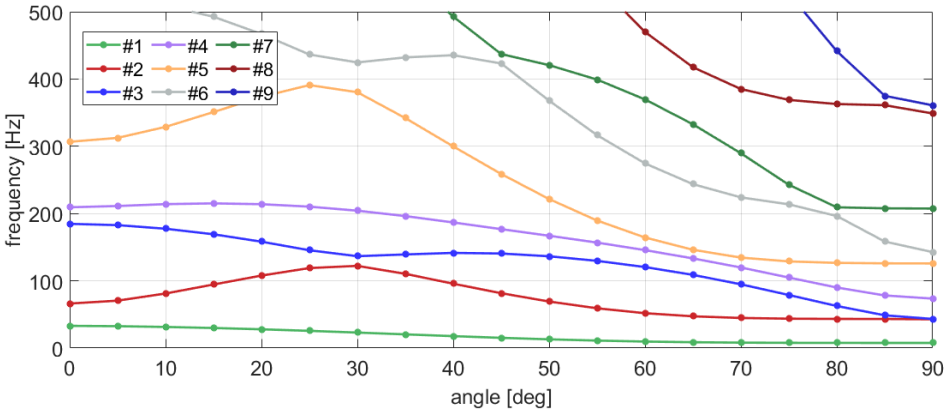


Figure 3.25: Eigenfrequencies of the composite link model in the frequency range of interest.

The results of this section show how the sampling range should be carefully selected keeping into account the expected use of the model. By sampling a narrow range, a pROM of smaller size can be achieved at the cost of not being able to represent the system for other values of the parameter.

3.6.5 Time domain analysis

The validation in time domain is done by applying the same continuous random input to the composite link. Same integration settings as the previous case are also employed. A strain measurement on the center of the top face in the z direction is extracted. The results in Figure 3.26 show a good matching for both strain and acceleration responses.

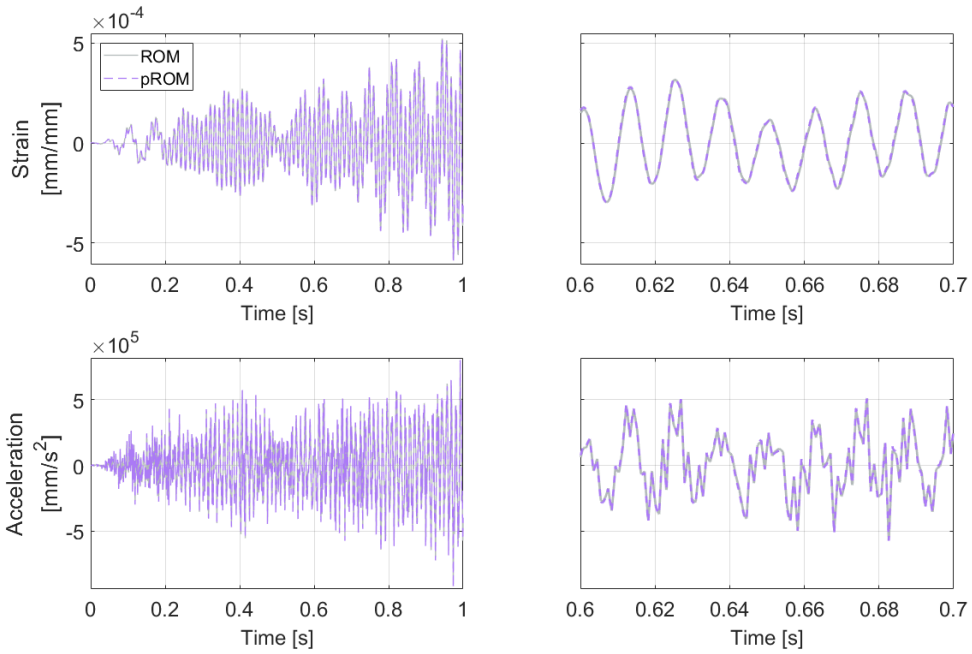


Figure 3.26: Comparison of time-domain simulations on the composite component for a continuous random input. The right figures show a zoom of the 0.6s to 0.7s range.

3.7 Conclusions and possible future work

This chapter presented a projection-based pMOR scheme that exploits the affine dependency on the parameters of the full model and uses a constant global reduction basis to achieve an efficient reduced model that maintains an explicit dependency on the parameters. The methodology was validated both in time and frequency domain for two different components with different materials, showing good results. The experimental model updating and validation of both models employed has been presented.

Future research efforts could be aimed at identifying affine representation for different kinds of materials or shape parameters, or at investigating methods to identify approximate affine dependencies. It will be of interest to also analyze more efficient sampling methods and algorithms for the selection of the size of the global reduction basis.

The component-level modeling technique will be embedded in a system-level model in the next chapter, and the parametric models obtained will enable applications such as optimization and sensing that will be presented in the rest of the work.

Chapter 4

Parametrization of Flexible Multibody models

4.1 Introduction

As discussed for component-level applications, also the capability to quickly evaluate FMB models under parameter changes is of paramount importance. Unfortunately, satisfying both evaluation and updating efficiency requirements while maintaining a high accuracy is one of the main challenges of most mechanical modeling applications.

As mentioned in Chapter 2, a main advantage of the FFR approach is that, by representing the deformations in a body-fixed floating frame, it allows to employ well-established component-level MOR techniques for linear FE models [22]. As is the case for structural applications described in Chapter 3, also for FMB the reduction of the flexible component models renders the parametric dependency implicit. The ROM of a flexible body and its respective mass-invariants have to be re-created every time such parameters are updated, resulting in a large computational effort. Model updating of a flexible multibody model by means of updating the material parameters is therefore typically computationally inefficient.

The parametric dependency of FMB models on the parameters of flexible bodies is treated in literature mainly by interpolation of local ROMs either at system level [100, 132] or at component level [19, 56, 55]. The main issues of interpolation approaches (large storage requirements and intermediate steps to enforce consistency) have been discussed in Chapter 3. Hou et al. [78] propose a component-level method that allows to retrieve a global reduction basis that has no parametric dependency. This approach focuses on large non-linear deformations and employs the Proper Orthogonal Decomposition (POD) applied to time-domain snapshots of FOMs, resulting in an expensive training procedure involving simulations of the entire system, i.e. not on an individual body level.

Considering the case of linear deformation and focusing on the material properties of flexible bodies, this chapter proposes a FMB formulation that employs the component-level pROMs presented in Chapter 3 to represent the flexible bodies, resulting in a parametric Flexible Multibody (pFMB). The use of pFMB models enables efficient model updating of flexible multibody models that maintain low size and high accuracy. The availability of parametric models enables, among the many possible applications, their use in design optimization procedures and estimation of parameters in operation conditions.

The proposed modeling approach is validated by comparison with standard non-parametric

models covering the desired parameter range. Both open-loop and closed-loop mechanisms are considered by analyzing case studies representative of the two classes, namely a four-bar linkage and a serial robotic manipulator. The two applications present different modeling challenges and it is therefore of interest to evaluate the performance of the methodology on both. Furthermore, without loss of generality, two representative families of material models are used for the flexible components: isotropic materials and fiber-reinforced laminates.

In Section 4.2 the adaptation of the pMOR technique to make pROMs usable in pFMB models is presented. The parametric version of the system EOMs and of the invariants terms used to compute the mass matrix is described in Section 4.3. A validation of the methodology by comparison of simulation results with the ones of non-parametric models is presented for a closed-chain mechanism in Section 4.4. Finally, is Section 4.5 the model of a robotic manipulator is first described together with its experimental updating procedure, and it is then used for validation of the methodology on an open-chain mechanism.

4.2 Adaptation of component-level pROMs for use in pFMB models

The pROM generation procedure presented in the previous chapter requires few considerations and adjustments in order to make the models compatible for use in a pFMB model as well.

Section 3.4 presented the pROM methodology applied to structural cases, in which the models are fully constraint as to not exhibit any rigid-body (RB) modes between the ones extracted through a modal analysis. For the component reduction basis employed in this work, the normal modes present in the reduction base are relative to the unconstrained condition [126] and the RB modes are filtered out in a pre-processing phase (Subsection 2.4.1). This step has then to be introduced also in the global basis selection workflow of the pMOR technique, in order to ensure that the final reduction basis does not contain any RB mode. The approach proposed in this work consists in removing the RB modes from each of the local bases $\mathbf{\Psi}_i$ calculated in the sampling phase. This results in a smaller concatenated basis and faster SVD computational times. This also guarantees that, during the iterative procedure to increase the size of the global basis, no RB behavior will be included in the final global basis.

Calculation of the components of the EOMs for a pFMB system require the knowledge of the nodal masses of the component's FE model. The affine dependency used to represent the mass matrix of the pROMs can be recast on the nodal mass of the j-th node m_j and in the mass of the whole component m as:

$$m_j(\mathbf{p}) = m_j^0 + \sum_k m_j^k f^k(\mathbf{p}) \tag{4.1}$$

$$m(\mathbf{p}) = m^0 + \sum_k m^k f^k(\mathbf{p}) \tag{4.2}$$

In the sampling stage of the workflow, together with the local matrices and bases also the local nodal masses need to be stored. These are then used to identify the affine representation using the least-squares approach presented in Section 3.4:

$$\min_{m_j^0,\dots,m_j^i,\dots} \sum_{h=1}^{n_s} \left\| m_{h,j} - m_j^0 - \sum_k m_j^k f^k(\mathbf{p}_h) \right\| \tag{4.3}$$

The updated workflow with additional steps can be seen in Figure 4.1.

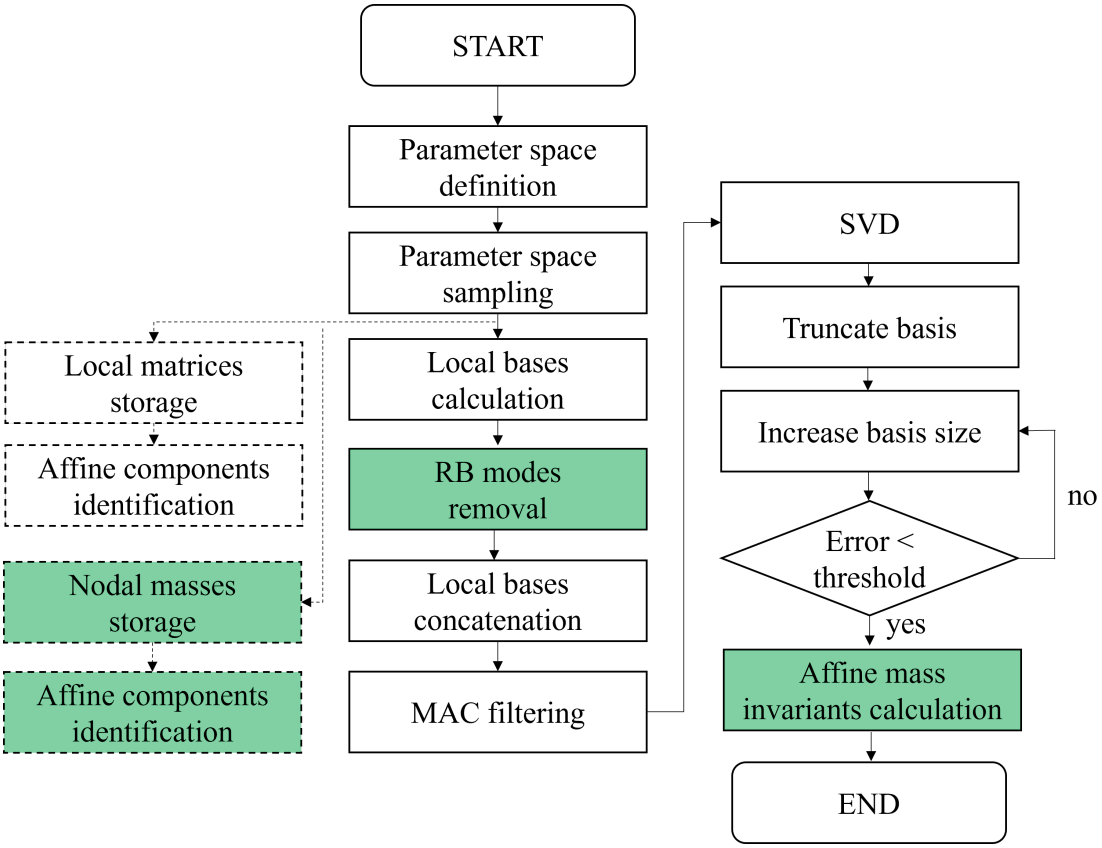


Figure 4.1: Basis selection workflow for pROMs used in a pFMB model. The steps added with respect to the original workflow (Figure 3.1) are highlighted in green.

4.3 System-level parametric Equations of Motion

Using pROMs results in an explicit parametric dependency of the mass, damping and stiffness matrices of the single bodies. These matrices appear in the calculation of several terms of the EOMs of the whole pFMB system, so that the dependency is carried on. The parametric EOMs are:

$$\begin{cases} \mathbf{M}(\mathbf{q},\mathbf{p})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{p})\dot{\mathbf{q}} + \mathbf{K}(\mathbf{p})\mathbf{q} + \boldsymbol{\Phi}_q^T(\mathbf{q})\boldsymbol{\lambda} = \mathbf{f}_e(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{f}_v(\mathbf{q},\dot{\mathbf{q}},\mathbf{p}) \\ \boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0} \end{cases} \tag{4.4}$$

The EOMs are derived considering parameters constant in time while, if they vary dynamically during the simulation, extra terms dependent on the time derivative of parameters arise. The assumption made in this work is that parameters are constant or vary slowly in time, allowing to neglect the extra terms. Such a behavior is realistic in most applications for the material parameters of interest.

One of the main features of the proposed pMOR methodology, the use of a constant reduction basis, results in a great advantage for pFMB application as well. The constraint equations and the external generalized forces, while depending on the reduced flexible coordinates and therefore on the reduction basis as well, do not depend on the parameters. This is of particular importance in applications such as estimation or gradient-based optimization, where the derivatives of the EOM

terms with respect to parameters are needed and such a dependency would greatly complicate the calculations.

The explicit parametric dependency of the stiffness and damping components matrices is retained also in the system ones, given that those results from a simple block-diagonal concatenation. This allows for an efficient update procedure, in which the element-level affine components are combined and then the matrices are concatenated:

$$\mathbf{K}^i(\mathbf{p}) = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ & \mathbf{0} & \mathbf{0} \\ sym & & \mathbf{K}^i_{f_r}(\mathbf{p}) \end{bmatrix} \tag{4.5}$$

$$\mathbf{C}^i(\mathbf{p}) = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ & \mathbf{0} & \mathbf{0} \\ sym & & \mathbf{C}^i_{f_r}(\mathbf{p}) \end{bmatrix} \tag{4.6}$$

For the mass matrix:

$$\mathbf{M}^i(\mathbf{q}^i, \mathbf{p}) = \begin{bmatrix} \mathbf{M}^i_R(\mathbf{p}) & \mathbf{M}^i_{R\Theta}(\mathbf{q}^i, \mathbf{p}) & \mathbf{M}^i_{Rf}(\mathbf{q}^i, \mathbf{p}) \\ & \mathbf{M}^i_{\Theta}(\mathbf{q}^i, \mathbf{p}) & \mathbf{M}^i_{\Theta f}(\mathbf{q}^i, \mathbf{p}) \\ sym & & \mathbf{M}^i_{f_r}(\mathbf{p}) \end{bmatrix} \tag{4.7}$$

where the term $\mathbf{M}^i_R(\mathbf{p}) = m^i(\mathbf{p})\mathbf{I}^{3\times 3}$ is explicitly dependent on parameters through the mass dependency in Equation 4.2, and $\mathbf{M}^i_{f_r}(\mathbf{p})$ is the reduced mass matrix of the FE model for the body. The parameter dependency of the mass-invariant terms can be made explicit in order to enable an efficient update. Equation 4.1 can be inserted in Equation 2.51 in order to obtain an explicit parametric dependency:

$$\begin{gathered} \hat{\mathbf{M}}^i(\mathbf{p}) = \left(\sum_{j=1}^{n_{nodes}} m^{i,0}_j \mathbf{h}(\mathbf{r}^i_0, \mathbf{\Psi}) \right) + \sum_k \left(\sum_{j=1}^{n_{nodes}} m^{i,k}_j \mathbf{h}(\mathbf{r}^i_0, \mathbf{\Psi}) \right) f^{i,k}(\mathbf{p}) = \\ \hat{\mathbf{M}}^{i,0} + \sum_k \hat{\mathbf{M}}^{i,k} f^{i,k}(\mathbf{p}) \end{gathered} \tag{4.8}$$

The affine components of each mass invariant can be pre-computed in the offline phase, at the end of the pROM generation algorithm (Figure 4.1), and stored as already done for the affine data relative to the system matrices. These will then need only to be combined in the online phase for any new parameters combination.

Finally, the quadratic velocity forces term depends on parameters through the mass matrix:

$$\mathbf{f}^i_v(\mathbf{q}^i, \dot{\mathbf{q}}^i, \mathbf{p}) = \left(\frac{\partial}{\partial \mathbf{q}^i} \left({\dot{\mathbf{q}}^i}^T \mathbf{M}^i(\mathbf{q}^i, \mathbf{p}) \dot{\mathbf{q}}^i \right) \right)^T - \dot{\mathbf{M}}^i(\mathbf{q}^i, \dot{\mathbf{q}}^i, \mathbf{p}) \dot{\mathbf{q}}^i \tag{4.9}$$

It should be noted how the exact representation of mass invariants is important in particular for highly dynamic applications in which the inertial terms have a noticeable impact on the overall behavior. In quasi-static or slow applications, instead, even an approximate representation of such terms would be acceptable.

4.4 Case study n. 1: a four-bar linkage

The proposed methodology is validated by means of comparing time domain simulation results of pFMB models with non-parametric, i.e. regular FMB models, for different material models and parameter values. This section presents the results of such validation procedure on a closed-chain kinematic chain model, known from literature to be a non-trivial simulation challenge [103, 38].

4.4.1 Description of the benchmark model

A four-bar linkage mechanism (Figure 4.2) consisting of one rigid body and three flexible bodies is considered. More specifically, the ground is considered to be the rigid body, whereas the links are modeled as flexible. These are indicated with the following codes in the rest of this work:

- **IC**: Input Crank
- **C**: Coupler
- **OC**: Output Crank

The links have varying lengths (shown in Figure 4.2) and a constant cross-section of $10mm \times 2mm$. A revolute joint is defined between each set of consecutive links, for a total of four revolute joints (Figure 4.3). It should be noted that such a modeling choice would cause redundant constraint equations in the case of a rigid bodies model, while it can be employed with no issues in the considered model given the added DOFs deriving from the flexibility of the bodies. A revolute driver imposes an oscillation of the input crank (as denoted on Figure 4.2).

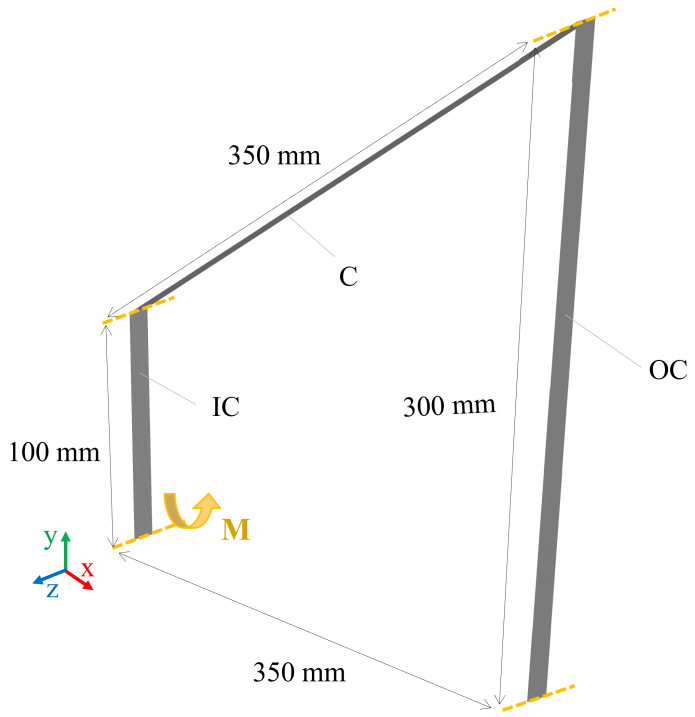


Figure 4.2: Flexible model of the four bars mechanism used for validation of the pFMB methodology. The revolute joints axes are shown in yellow, and in particular the actuated joint (M) is placed between input crank and ground.

A FMB model has been generated first in a commercial tool [1]. This has then been used to extract the needed data to model the same system in an in-house Matlab pFMB framework in which the rest of the modeling and simulations are done (the commercial tool is still used for visualization purpose).

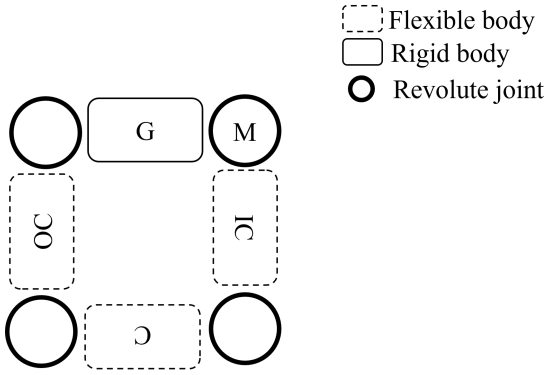


Figure 4.3: Scheme of the four-bar linkage flexible model.

The FE meshes of the flexible bodies consist of 4-node linear shell elements. Each flexible body is modeled using pROMs, of which the properties are summarized in Table 4.1. For all of them, in the global reduction basis size selection, a threshold of 10^{-3} has been used for the FRFs error, in order to achieve comparable accuracy. An isotropic material model is considered (Table 4.2), with the Young's Modulus and density as parameters of interest. The range of parameter value variation is set between 60% and 140% of the reference values. A CFRP composite material (Table 3.6) is considered as well for the coupler link. The composite material is comprised of 8 layers, with the fiber angle as the parameter of interest varying between 0° and 90°. Each flexible body is also modeled using a standard, i.e. non-parametric, ROM for each set of parameter values considered for validation.

All the reference FMB models contain 94 DOFs. The complete pFMB model contains 94 DOFs when considering isotropic material properties, resulting in the same size as the non-parametric equivalent models. When the composite material model is used for the coupler pROM, the pFMB model size increase to 270 DOFs. While being larger than the correspondent non-parametric model, the size is still small especially if considering the sizes of the FOMs of the flexible components used.

Table 4.1: Four-bar linkage ROMs and pROMs properties. In the global reduction basis size selection a threshold for the FRFs error of 10^{-3} has been used.

Component	n_{dof} **FOM**	n_{red} **ROM**	n_{red} **pROM**	**Parameters**
IC	25338	22	22	E,ρ
C (steel)	88338	22	22	E,ρ
C (composite)	88338	22	198	Θ
OC	75738	22	22	E,ρ

Table 4.2: Isotropic four-bar material reference parameters

	E $[GPa]$	ρ $[Kg/m^3]$	ν	α	β	h $[mm]$
IC, C, OC	201.99	7990	0.34	2.5	0.00001	2

4.4.2 Application cases for validation

The driver profile has been defined to excite the system dynamics and is shown in Figure 4.4. An oscillation of the input crank is imposed with a 10Hz sinusoidal angle variation and amplitude

gradually increasing and decreasing in the range between 0° to 20°. Furthermore, gravity is acting along the global y axis.

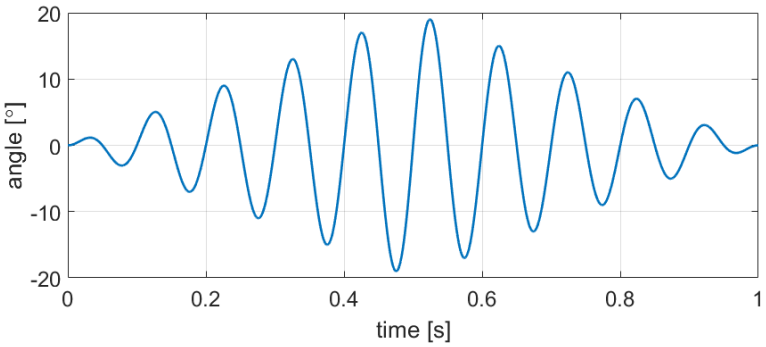


Figure 4.4: Drivers time values of the trajectory considered for validation of the four-bar linkage pFMB model.

A set of measurements is extracted from each simulation to compare the pFMB and FMB models (as shown in Figure 4.5). One virtual strain gage is defined on the top faces of both input and output cranks, and the absolute position and acceleration of the center of gravity of the coupler are also measured. For the case where a composite material variant is used, a strain gage on the top face of the coupler link is defined as well. Comparing the entire set of virtual measurements allows to validate that the system dynamic behavior is properly captured by the pFMB. The deviation, i.e. difference in absolute position, of the center of the coupler link with respect to a simulation of a reference fully rigid model is used for comparison. The absolute position of a material point of the flexible and rigid models is denoted as $\mathbf{x}(t)$ and $\mathbf{x}_R(t)$, respectively, and allows to define the deviation d as:

$$d(t) = \|\mathbf{x}(t) - \mathbf{x}_R(t)\| \tag{4.10}$$

where $\|\square\|$ denotes the L^2 norm.

The virtual measurements extracted from the FMB model y_{FMB} and the pFMB model y_{pFMB} are compared by evaluating an error metric defined as:

$$error(t) = \frac{\|y_{pFMB}(t) - y_{FMB}(t)\|}{\overline{y}_{FMB}} \tag{4.11}$$

where $\overline{y}$ represents the maximum value of the measurement over the entire simulation.

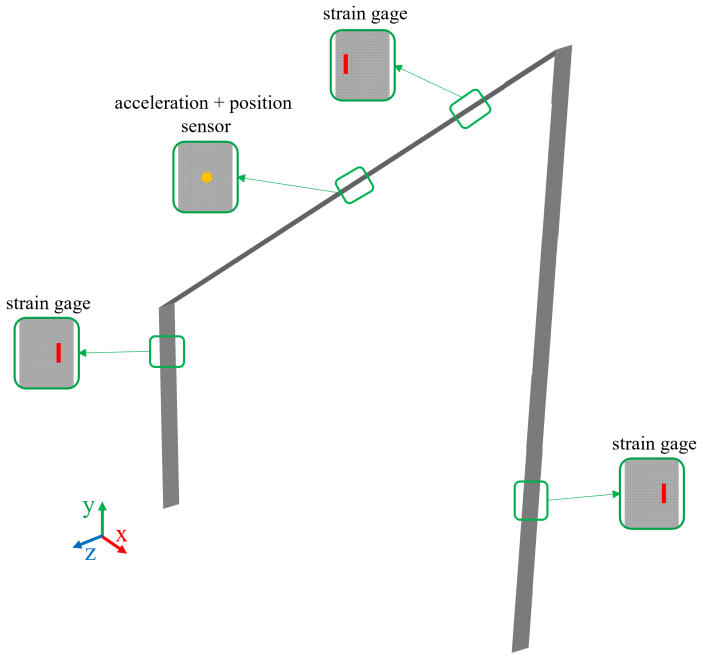


Figure 4.5: Sensors extracted on the four bars flexible model.

4.4.3 Results for the isotropic material model

For the isotropic material model case, simulations are carried out considering three parameter value sets that represent the entire parameter space used to generate the pROMs:

- **Set 1**: E at its minimum value and ρ at its maximum value
- **Set 2**: Reference values for E and ρ
- **Set 3**: E at its maximum value and ρ at its minimum value

The same parameter values are used for each body per simulation run. These parameter values are not part of the sample set used during the training of the pROMs.

The first-order BDF time integration scheme is used for the forward simulation presented in this section with an integration step $\Delta t = 1ms$.

A good correspondence can be observed both for the deviation (Figure 4.6) and acceleration results (Figure 4.7). As expected from what seen in the previous chapter on pROMs approximation errors, the error is smaller for the reference parameter value set considered to generate the pROMs and increases near the boundaries of the parameter space domain. Even in those cases, the error is in an acceptable range, typically being below 1% of the maximum value of the measurements.

A comparison of the strain measurements, as shown in Figure 4.8, displays the same trend seen in the previous results, with an overall satisfying error that is at a minimum for reference parameter values and increases near the parameter range boundaries.

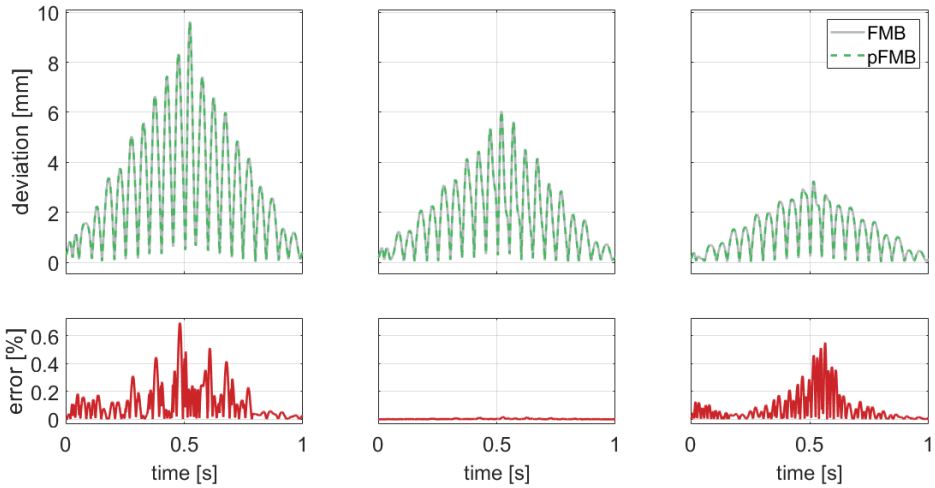


Figure 4.6: Deviation on the center of C on parameter Set 1 (left), Set 2 (center) and Set 3 (right) for the isotropic material case.

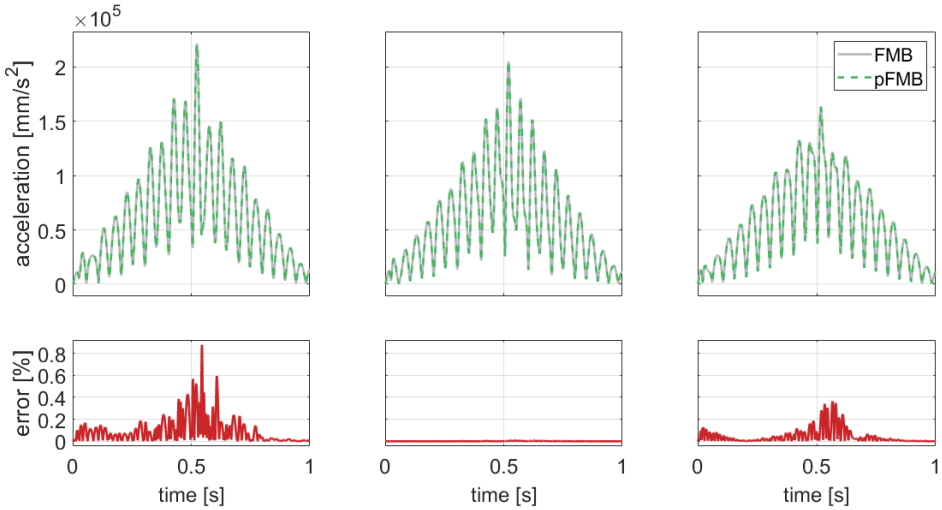


Figure 4.7: Acceleration on the center of C on parameter Set 1 (left), Set 2 (center) and Set 3 (right) for the isotropic material case.

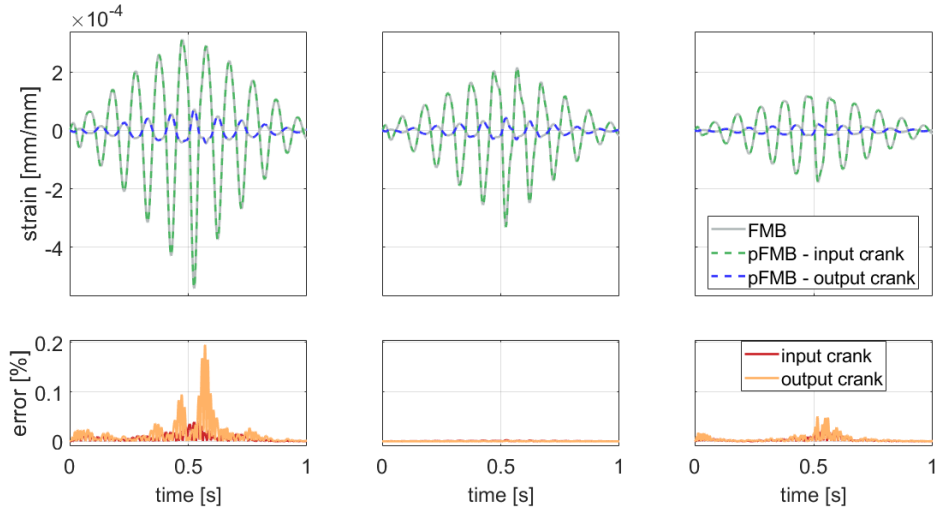


Figure 4.8: Strain of IC and OC on parameter Set 1 (left), Set 2 (center) and Set 3 (right) for the isotropic material case

4.4.4 Results for the model with a composite coupler

A second validation case is performed using a composite material model for the coupler body. The isotropic material model parameter values of both crank links are kept equal to their reference values and three fiber angle parameter values, not used for pROM training, (0°, 45°, and 90°) are considered instead. The same integrator settings as for the isotropic material case are used.

Deviation and acceleration results for coupler link are shown in Figure 4.9 and Figure 4.10 respectively, again showcasing satisfying matching with errors below 1% of the maximum measurement values. In this case, the lowest error is achieved for the 0° angle parameter value and increases for higher angle parameter values. The same behavior can also be observed for the strain measurements extracted from the coupler link (Figure 4.11). Such a behavior is consistent with the approximation error of composite pROMs seen in Section 3.6.

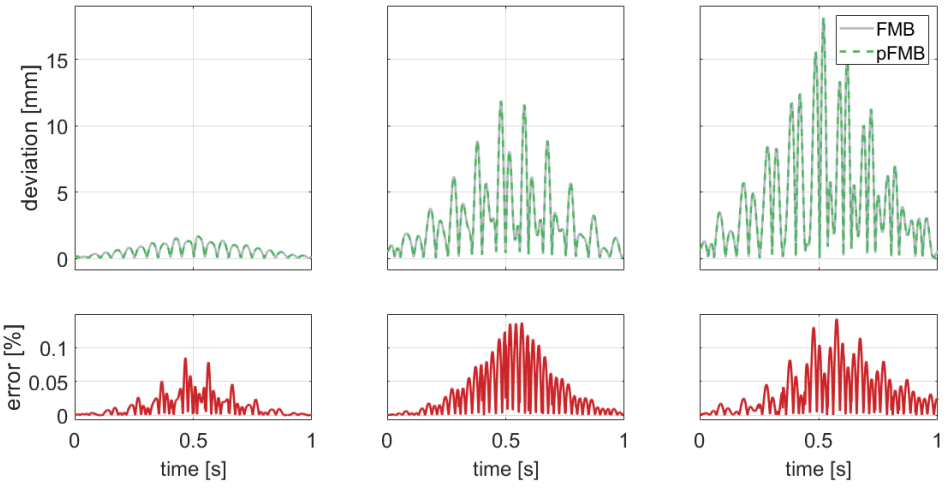


Figure 4.9: Deviation of C's center for the composite material case with fiber angle at 0° (left), 45° (center) and 90° (right).

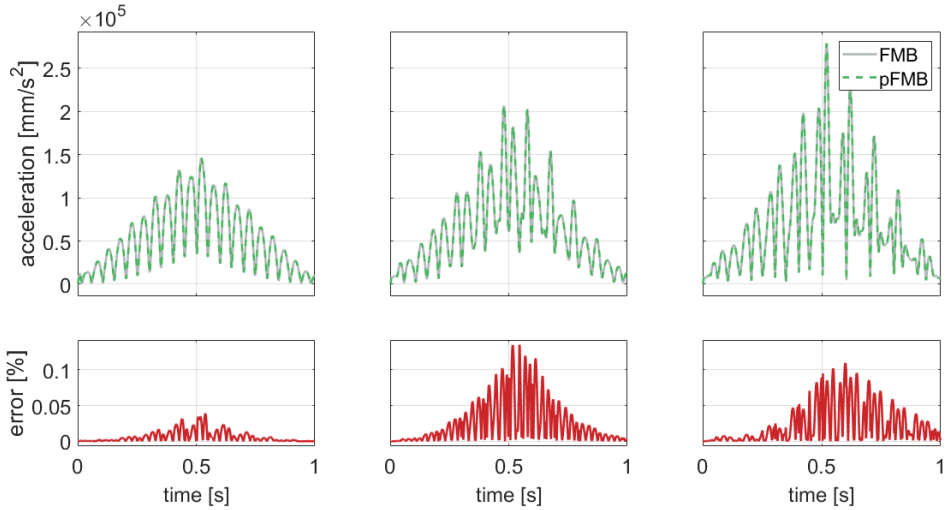


Figure 4.10: Acceleration of C's center for the composite material case with fiber angle at 0° (left), 45° (center) and 90° (right)

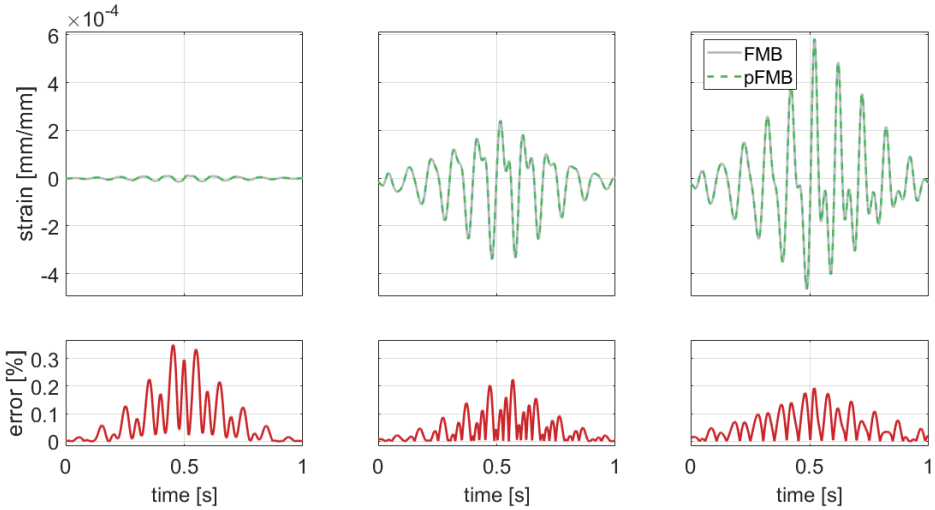


Figure 4.11: Strain on C for the composite material case with fiber angle at 0° (left), 45° (center) and 90° (right)

4.5 Case study n. 2: 5 DOF robotic manipulator

A second validation application is proposed in order to consider a more complex model of a real system, with a higher number of bodies and constraints and interactions between rigid and flexible bodies. Furthermore, by also considering an open-loop kinematic chain, the proposed pFMB methodology is validated on both types of kinematic chains. In particular, robotic manipulators are an interesting family of open-chain mechanism with a wide range of applications and complex dynamic behavior [49].

4.5.1 Benchmark model: Robolink manipulator

The Robolink manipulator manufactured by IGUS [4] is considered for validation. The considered model has 5 rigid DOFs. The first 3 links consist of two folded sheets of steel, connected directly to the joints and constituting the links. The joints in turn consist of the assembly of a stepper motor and a transmission gearbox. The robot is equipped with a gripper. One physical model of the manipulator is available at the testing facilities of Siemens Industry Software NV in Leuven (Belgium), mounted on a test-rig that allows for easy movement and a large working surface.

The Robolink is a small-scale lightweight manipulator with a modular structure. This has the advantage of allowing for an easy disassembly and replacement of single components and easy mounting of sensors. Furthermore, the simple design of the links makes it an ideal case for the replacement with composite parts. Even if simple, the 5-DOF configuration allows for a good mobility in the workspace, thus enabling complex maneuvers.

A complete CAD model of the manipulator comprising all of the main components has been provided by the manufacturer (Figure 4.12). The geometrical representations for the simplest components, such as the links, is close to reality. Other more complicated components, that are themselves sub-assemblies of many small parts, such as the motors and the gripper, are only represented in the model by a single piece with the same external geometry but uniform density on the inside. This simplification is used throughout the entire modeling procedure, keeping into account that some mismatches can arise due to the different mass and compliance distribution with respect to the original bodies. Furthermore, the models available do not include the cables that connect motors and grippers to the control and power unit. These are not modeled in the rest of this work, but it should be noted that, given the lightweight design of the manipulator, their inertial, damping and elastic effect is not always negligible [97, 82].

Figure 4.12: Starting CAD of the Robolink manipulator as provided by the vendor.

Starting from the CAD, the first modeling step consists in a simplification of the model by removing the components that have a negligible influence on the static and dynamic behavior such as bolts and screws. Furthermore, the robot is stripped of the plastic covers of the links, given that these have no structural function. This results in the simplified model shown in Figure 4.13.

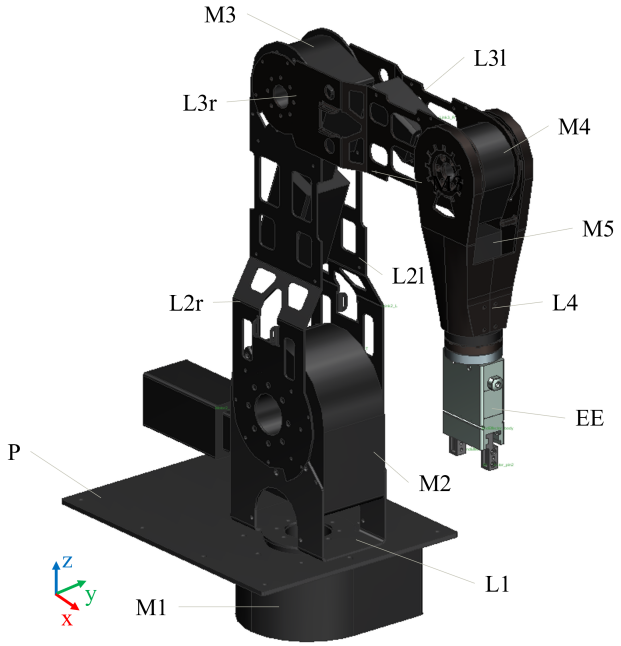


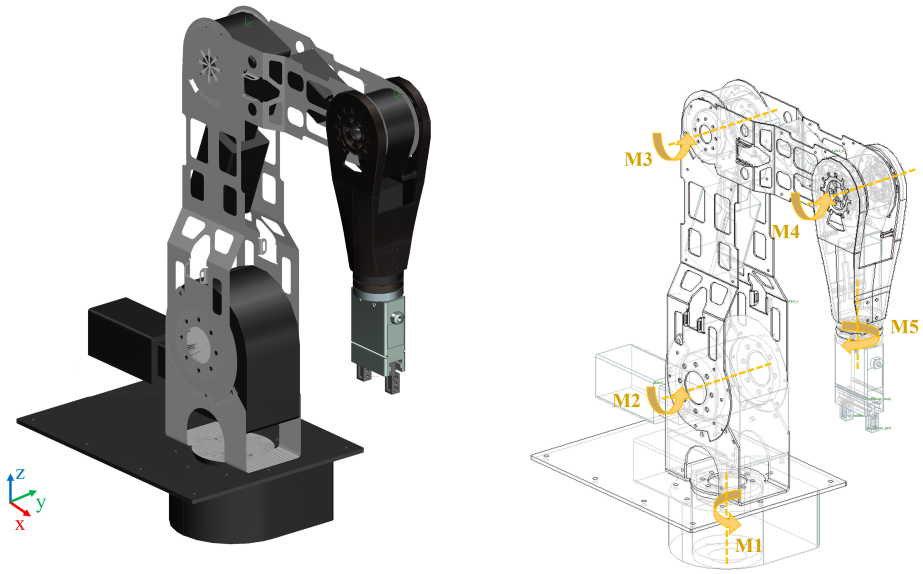
Figure 4.13: Simplified CAD of the Robolink manipulator with labels for the names of each component.

The components of the manipulator are indicated in Figure 4.13 and in the rest of this work with the following codes:

- **G**: ground body
- **P**: plate on which the first motor is fixed
- **Li**: link number i
  - **Lil,Lir**: left and right parts of the link
- **Mi**: motor number i
  - **Mii,Mie**: internal and external part of the motor
- **EE**: end effector

Starting from the simplified CAD, also in this case a FMB model has been generated first in a commercial tool [1], which is illustrated in Figure 4.14. This was used to extract the needed data to then model the same system in the in-house Matlab pFMB framework. Only the first, second and third links are modeled as flexible bodies, given that their compliance is predominant

with respect to the other bodies since these are made of a thin sheet metal. In particular, L2 and L3 have two sides modeled as two separate flexible bodies, each based on the same FE model consisting of linear shell elements.



(a) (b)

Figure 4.14: Flexible multibody model of the Robolink manipulator showing meshes of the flexible bodies (a) and revolute joints axes (b).

Each of the 5 rigid DOFs is modeled by a revolute joint actuated by a driver (Figure 4.14b). In robotic manipulators, compliance and damping of the joints is known to have an important role in the overall response [46] and are therefore modeled as well. A wide range of modeling techniques have been presented in literature. In this work only compliance and damping around the rotational axis of the revolute joints is modeled [134]. The modeling approach for a rigid revolute joint consists of splitting the body of the motor in two concentric components, each of which is rigidly connected to one of the links via a bracket (i.e. fixed) joint (Figure 4.15a). When the joint is modeled as compliant, an extra intermediate fictitious body with zero mass is added. The ideal revolute joint is then defined between the external body and the fictitious one, while the middle and internal bodies are connected by a lumped rotational spring and damper system in parallel to a revolute joint (Figure 4.15b). This approach allows for a simplified modeling of a compliant joint that adds a minimal number of DOFs and constraints to the pFMB model.

Furthermore, the robot is mounted on a test-bench whose flexibility is not negligible. The top part of the bench, made of a 5 *mm* thick aluminum sheet, is also modeled as flexible.

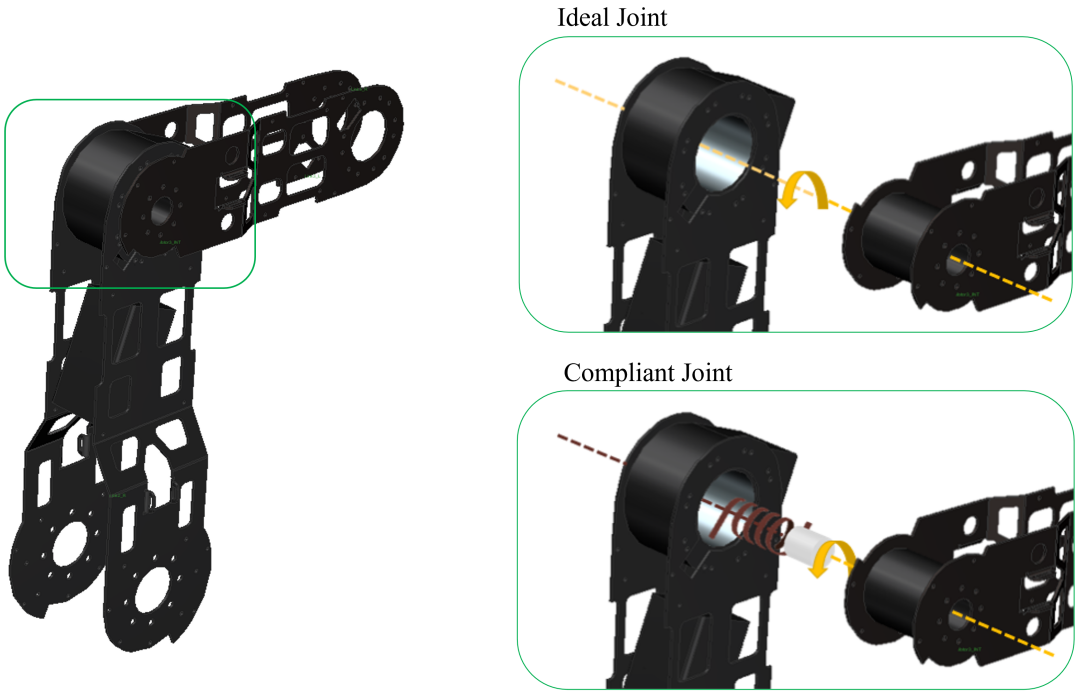
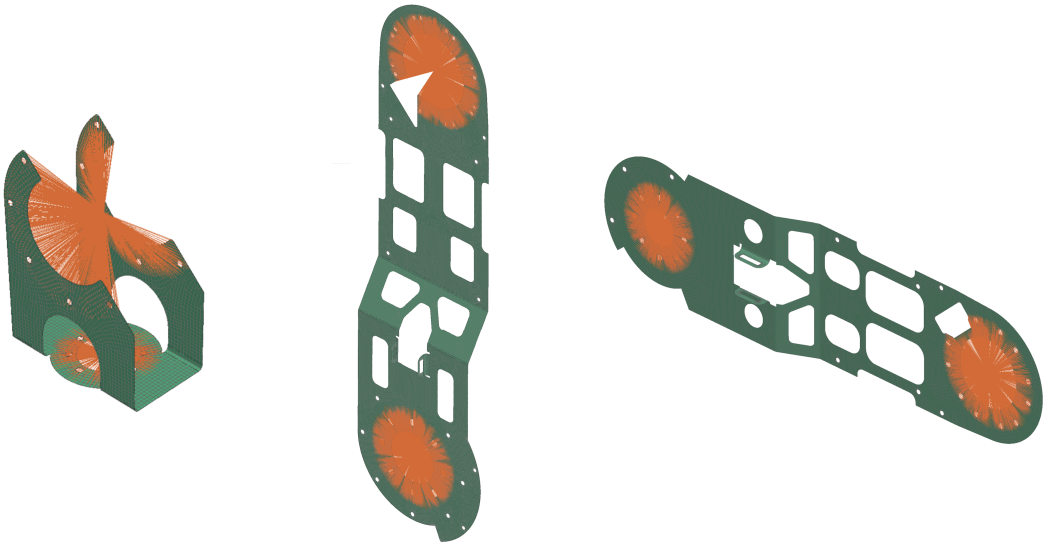


Figure 4.15: Scheme of modeling technique for rigid and compliant revolute joints using as example the sub-assembly of L2, M3 and L3.

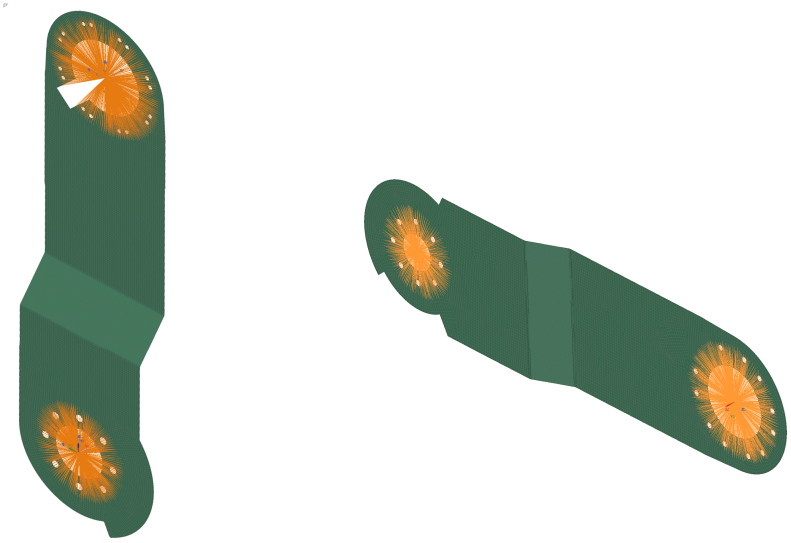
The flexible bodies of the three links are modeled using the proposed pROMs. The meshes of each flexible body, shown in Figure 4.16, consist of 3 and 4-nodes shell elements, and the faces in contact with motors in the assembly are connected to a single master node by Rigid Body Elements (RBEs). The master nodes are then used to define static modes and in turn the joints locations. The properties of the models are summarized in Table 4.3. Also in this case, a threshold of 10^{-3} has been used for the FRFs error in the global reduction basis size selection. All of the links are modeled using an isotropic material model with the Young's Modulus and density as parameters varying between 60% and 140% of the reference value. Furthermore, the second and third links are also modeled using the composite material discussed in Subsection 3.6.2 based on a simplified geometry that would allow easier and safer manufacturing (Figure 4.17). The fibers angle is the parameter of interest, varying between 0° and 90°, and the material used is the one extracted from experiments (Table 3.6) with 8 layers.

For validation purposes, also a set of equivalent regular ROMs with the same parameter values used in the simulations has been created for each flexible body.



(a) (b) (c)

Figure 4.16: Mesh of L1 (a), L2 (b) and L3 (c). The orange lines are RBEs that connect a master node to the faces in contact with the motors in the assembly.



(a) (b)

Figure 4.17: Mesh of L2 (b) and L3 (c) with simplified design to facilitate composite manufacturing. The orange lines areRBEs that connect a master node to the faces in contact with the motors in the assembly.

Table 4.3: Robolink ROMs and pROMs properties

Component	n_{dof} FOM	n_{red} ROM	n_{red} pROM	Parameters
P	252408	22	22	E,ρ
L1	21174	22	22	E,ρ
L2	247038	22	22	E,ρ
L2 composite	75750	22	181	Θ
L3	153876	22	22	E,ρ
L3 composite	55536	22	231	Θ

The complete model results in 14 rigid bodies and 6 flexible bodies, for a total of 272 DOFs when the isotropic models are used and 1020 DOFs if the composite models are used instead. As in the previous application case, also here the isotropic model has the same size as the non-parametric FMB model while the increase in size is limited for the composite pFMB model. An overall scheme of the model showing all the bodies and their relative connection is shown in Figure 4.18.

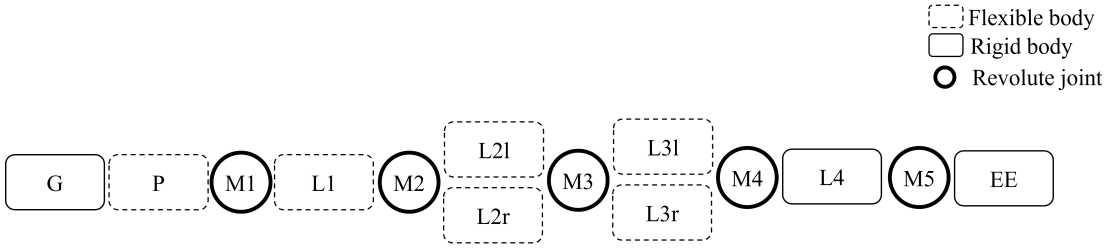


Figure 4.18: Scheme of the Robolink model.

4.5.2 Experimental model updating

For most applications, it is desirable that numerical models closely match the behavior of the physical systems they represent. With this goal, a set of experimental tests has been executed on the physical Robolink manipulator (Figure 4.19). Given its complexity, the manipulator represents a non-trivial modeling challenge. On a component level, each of the bodies is characterized by specific inertia, stiffness and damping properties. On a system level, the connections between the bodies are often far from ideal, introducing non-linear behaviors. Also electronic components such as motor and sensors need to be kept into account given their influence on the overall behavior. Motors and sensors are also connected with cables that run through the whole manipulator and influence its dynamics. Given the many challenges, a trade-off modeling solution is required in order to achieve the needed accuracy without over-complicating the models. The assumptions made in this work are the following:

- The only compliant bodies are the sheet metal links and the base plate, modeled with a linear elastic material;
- The fourth link, made of plastic and with a bulkier design than the first three, is modeled as rigid;
- Motors, gearboxes and gripper are modeled as rigid bodies with uniform density;
- Connections are modeled as ideal;
- Revolute joints have a compliance only around their rotation axis;

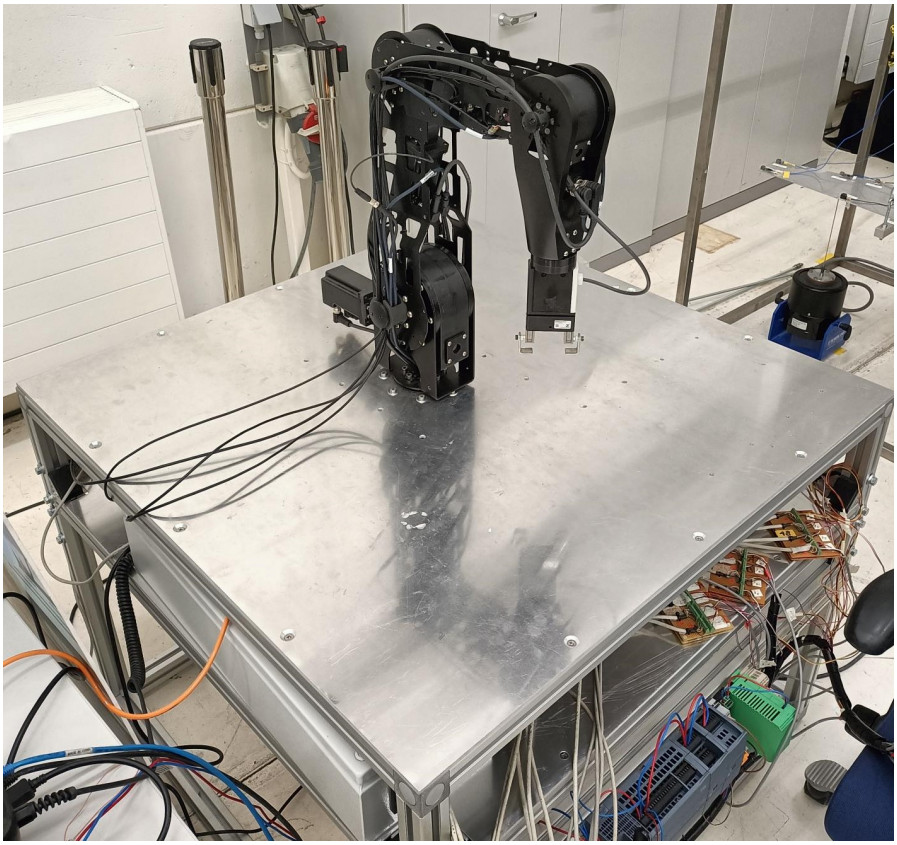


Figure 4.19: Experimental setup for robotic manipulator.

- The behavior of electric motors and gearboxes is not modeled, and the angle for each revolute joint is assigned as a time-dependent constraint in the simulations;
- The cables connecting motors, end effector and sensors to the control electronics are not modeled;

It will be interesting in future efforts to also investigate how removing some of the assumptions and complicating the model benefits the matching and consequent applications of the model.

A first phase of the update has been focused on the separate components, with a second phase considering the entire system. For system-level validation only the isotropic links have been employed. Employing the experimentally validated composite components in the model allows for a good level of confidence also in the composite version of the manipulator.

4.5.2.1 Component-level identification

The goal of the first updating steps is to ensure that the models of all components are as accurate as possible. With this aim, the manipulator has been disassembled to make every component accessible singularly. Given the mounting setup, it has not been possible to disassemble L1, M1 and the support plate, nor to separately test the supporting frame. For M1, the nominal properties provided by the manufacturer have been considered. L1 has instead been modeled using the same material as L2. Testing data for the material used to manufacture the support

plate was available and has been used as a starting point for its modeling, with only its stiffness tuned in the system-level phase as explained later.

In order to characterize the inertia properties, every body has been weighted separately (the results are summarized in Table 4.4). The mass data has been used to then identify the densities applied to the CADs of the rigid bodies to compute their inertial tensors. For the flexible bodies, the computed density is used directly to model material properties of the relative FE models.

Table 4.4: Robolink model bodies properties

Component	**Weight** [Kg]
P	13.05
M1	6.00
L1	0.93
M2	4.35
L2r/L2l	0.58
M3	1.84
L3r/L3l	0.36
M4	0.83
L4 + M5	0.82
EE	0.63

While rigid bodies are completely defined by inertial properties, flexible ones also require identification of stiffness and damping. Impact test measurements have been used to identify the model properties of L2r and L3r, while the correspondent left sides are assumed to have the same properties. The same procedure employed for the composite link in Section 2.3 has been applied also in this case. Both links have been suspended elastically (Figure 4.20) and a set of 5 uniaxial accelerometers with axis on the out-of-plane direction has been mounted on each link (Figure 4.21). The input has been applied by an instrumented hammer. The identified parameters are Young's moduli and Poisson's ratios. The results of the updating procedure, aimed at minimizing the difference between frequencies and maximizing the MACs (Section 2.3), are summarized in Table 4.5, showing a good matching between experimental and numerical modal data both in terms of frequency and MAC.

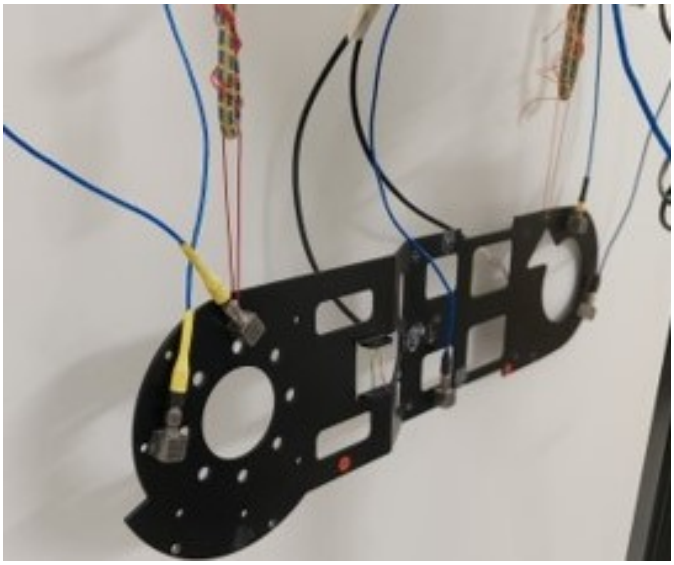


Figure 4.20: Experimental setup for impact testing on L2r.

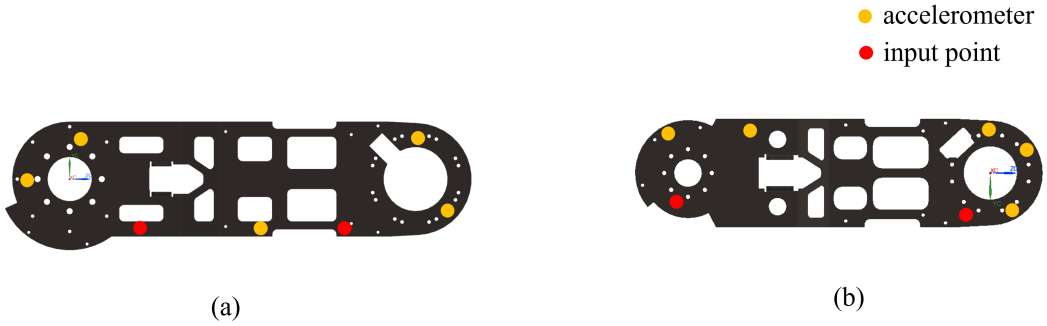
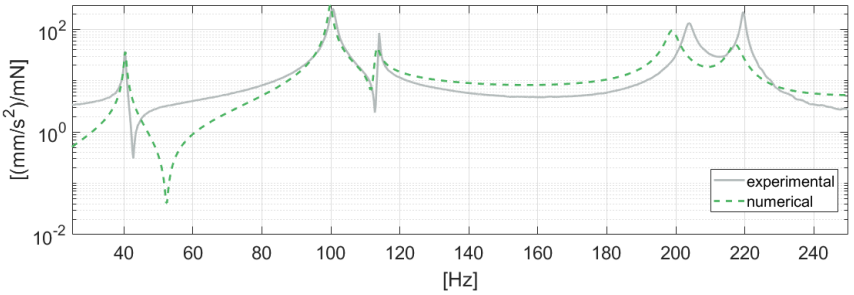


Figure 4.21: Position of input and output points for modal analysis on L2r (a) and L3r (b)

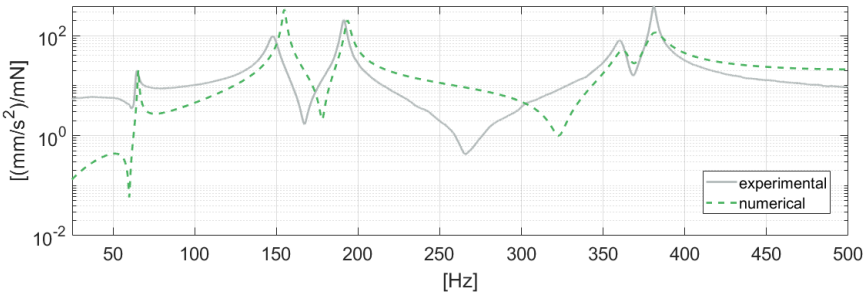
Table 4.5: Modal matching for L2r and L3r

Mode #	**Ref. freq.[Hz]**	**Model freq. [Hz]**	**Error**	**MAC**
		L2r		
1	40.00	40.25	0.83%	0.998
2	100.25	99.50	0.83%	0.998
3	113.75	113.25	0.48%	0.994
4	203.5	198.75	2.45%	0.987
5	219.25	217.00	1.08%	0.976
		L3r		
1	64.50	65.25	1.38%	0.999
2	147.75	154.50	4.9%	0.996
3	191.00	193.25	1.24%	0.995
4	359.75	361.75	0.57%	0.986
5	380.75	381.50	0.2%	0.995

The two Rayleigh parameters of each body have been tuned in order to match the FRFs. Since only 2 parameters are considered, it has been chosen to aim the procedure at matching the peak relative to the first and second natural frequencies for L2, while for L3 the third peak was preferred given its prominence. Results show good match especially close to the resonances (Figure 4.22). The mismatches can be mainly attributed to the approximate boundary conditions and the difficulty in ensuring a load direction normal to the face of the component in the testing phase.



(a)



(b)

Figure 4.22: Comparison of experimental and numerical FRFs in free-free conditions for L2r (a) and L3r (b).

Finally, the identified material parameters are summarized in Table 4.6. It can be noted how the identified parameters for both links are almost identical, given that they are nominally made of the same material. A small difference is noticeable in the Young's modulus, that can reasonably be explained by small geometry discrepancies between models and components.

Table 4.6: Isotropic links material reference parameters

	$E\ [GPa]$	$\rho\ [Kg/m^3]$	ν	α	β	$h\ [mm]$
L1, L2	201.99	7990	0.34	2.5	0.00001	2
L3	194.37	7990	0.34	2.5	0.00001	2

4.5.2.2 Component-level strain gages validation

The estimation application discussed in the next chapters requires position-level measurements to be acquired during the maneuvers of the manipulator. With this aim, one uni-axial strain gage has been mounted both on L2r and L3r (Figure 4.23). In common experimental applications, the system is over-instrumented by applying a larger-than-required number of sensors. Such practice guarantees robustness to errors in the measurements. For the considered application, it has been noted that the system is sensitive to inertial and tension loads applied by cables. It has been then chosen to only employ a minimal number of sensors in order to minimize such disturbances.

The strain gages position has been verified and validated by a static test on the single links. Both links have been fixed on one side, and known static loads have been applied on two different points in three mutually normal directions (Figure 4.24).

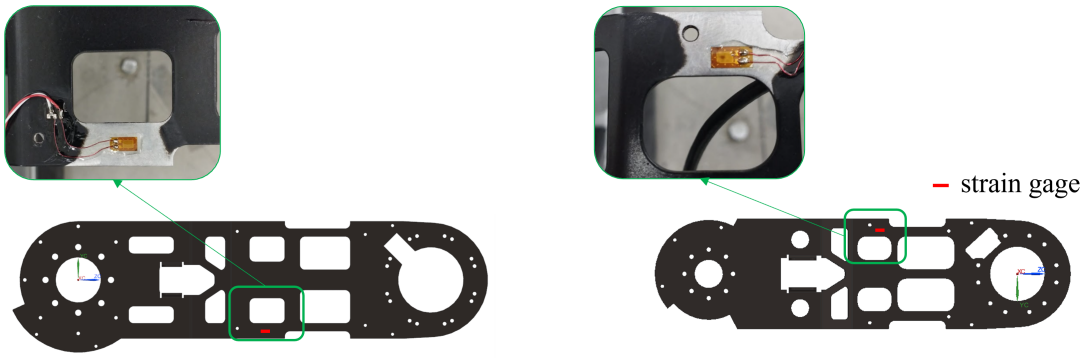


Figure 4.23: Positions of the strain gages on L2r (a) and L3r (b) with a detail of the physical sensors setup.

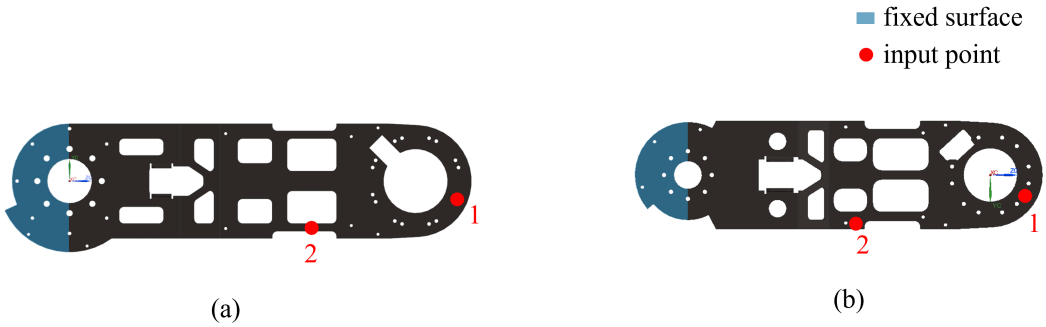


Figure 4.24: Input points and boundary conditions for static tests on L2r (a) and L3r (b)

Figure 4.25 shows the comparison between numerical and experimental measurements for the 6 load cases. For both bodies a good matching can be observed.

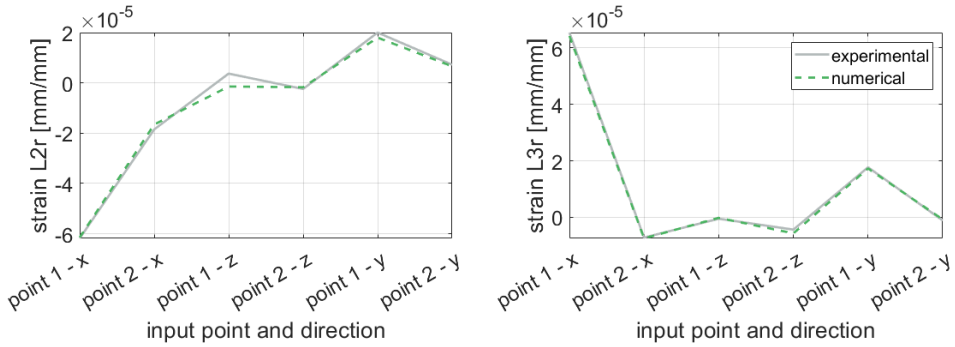


Figure 4.25: Comparison of measured and simulated strains for static loads on L2r and L3r on 2 different input points and 3 load directions.

4.5.2.3 System-level model updating

After the component-level updating phase, the manipulator has been re-assembled in order to evaluate the behavior of the whole system. The aim of the second system-level phase is to identify the remaining components, namely the compliant joints and the mounting frame. Furthermore, analyzing the results of the system-level matching allows to verify if the modeling assumptions hold.

A set of impact tests has been executed on the robot by applying the impact load at the End Effector on vertical and side directions in a configuration with the last link fully extended (Figure 4.26), as to facilitate the procedure. The measurements of the two strain gages previously mentioned have been considered as outputs. The joints stiffness and damping, together with the Young's Modulus of the base plate, have been updated by identification of the values that minimize the distance between numerical and experimental FRFs. The plate material is theoretically known from existing material data. The plate is though only one of the components of a larger test bench whose flexibility would have been impractical to model and has been assumed as rigid. The flexibility of the plate has then been used as a parameter approximatively representing the entire support structure requiring an update in this phase.

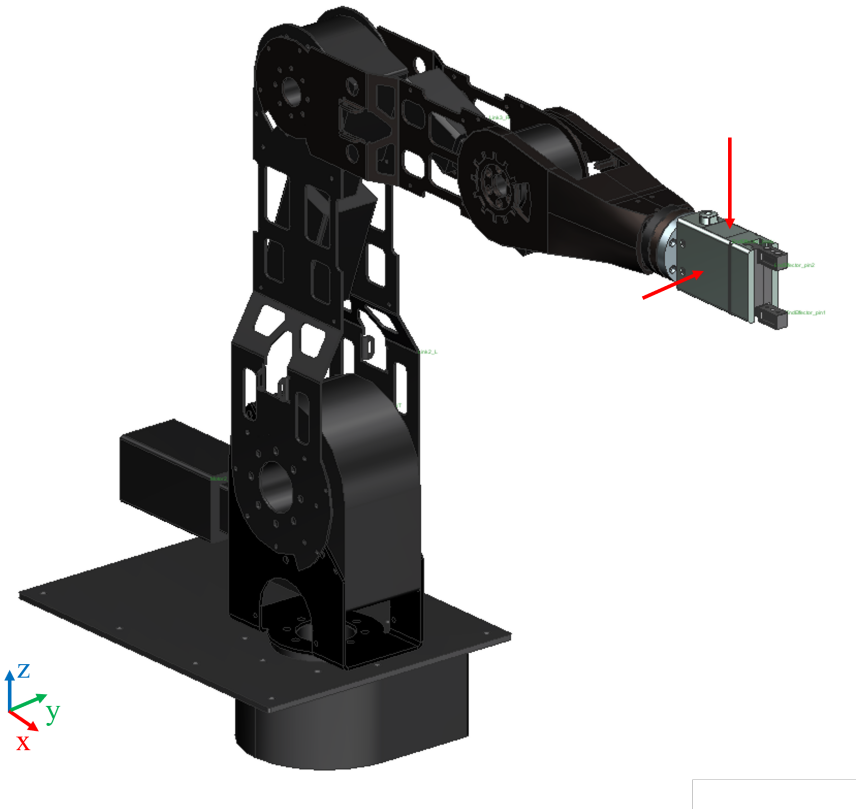
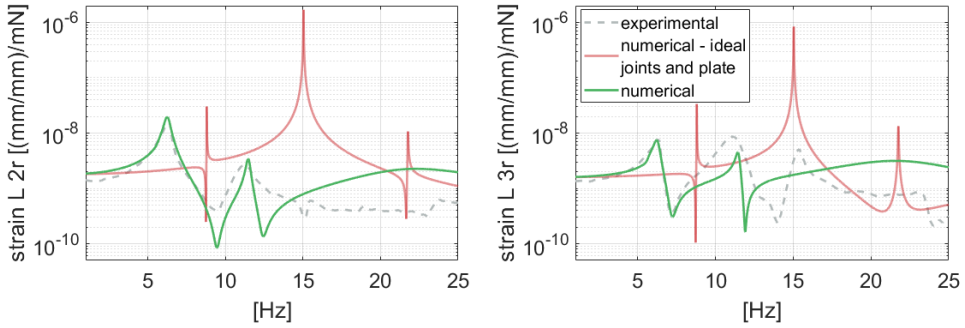


Figure 4.26: Configuration of the manipulator used in the impact tests, with indication of the two input points and directions used.

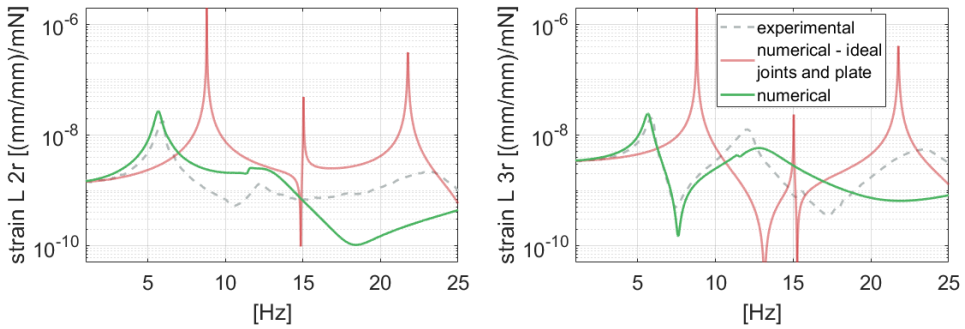
The range between 0Hz and 25Hz, that contains the first 3 experimental resonances of the system, has been considered for the update. This range allows to capture the main dynamic

components while keeping the matching problem simple enough. Tuning of the mentioned parameters allowed to reach an acceptable level of matching between experimental and numerical FRFs at least on the first peaks. The second peaks of the curves are matched in terms of frequency but differ slightly in terms of amplitude. The FRFs of L3r exhibit also a third peak that is not present in the numerical curves. This can be imputed to the assumptions made on the compliance of bodies and joints. The plots in Figure 4.27 also show the same FRFs evaluated on the model with ideal joints and rigid support plate. The wide difference demonstrates the impact of these sources of compliance on the overall behavior of the system and in turn the importance of the system-level model updating.

Finally, the identified joints parameters are summarized in Table 4.7 and the plate material in Table 4.8.



(a)



(b)

Figure 4.27: Comparison of measured and simulated FRFs on the robotic manipulator for 2 strain outputs on L2r (left) and L3r (right) and load directions z (a) and y(b).

Table 4.7: Identified stiffness and damping values for revolute joints.

	k $[Nm/rad]$	c $[Nms/rad]$
M1	2000	20
M2	16000	500
M3	8000	50
M4	8000	50

Table 4.8: Identified material parameters for the base plate

$E\ [GPa]$	$\rho\ [Kg/m^3]$	ν	α	β
28.8	2630	0.33	12.5	0.00005

4.5.2.4 System-level strain gages validation

It is of interest to validate the static response of the strain gages also once the links are assembled on the manipulator. Many unexpected static loads can in fact arise from connections, cables and unbalanced mass distributions.

The treatment of strain gage measurements in such a system requires a few practical considerations. Before each set of acquisitions, strain gages require a stabilization phase in order to remove the thermal drift effect. After this, a calibration procedure is done. The values measured during such calibration are used by the acquisition software as the zero reference configuration. Since it would be impractical to disassemble the links at each acquisition for the calibration, this has been done directly on the assembled system. The reference strains are then relative to the system already deformed by gravity. The measurements acquired are then only relative strains and not the absolute value (i.e. a zero value will be measured in the reference configuration even though gravity is deforming the bodies). In a forward simulation comparison with numerical models, this requires to also reference the virtual measurements using the static solution at the zero configuration.

With the aim of verifying the static behavior of the system, a set of 5 static poses representative of the entire workspace has been chosen (Table 4.9). On each pose a static acquisition of the strains under gravity load has been made. Experimental and simulated static strain measurements are compared in Figure 4.28. It can be appreciated how the measurements on L2r properly match, while the sensor on L3r exhibits an error while maintaining the same order of magnitude and trend of change between different poses. Such behavior can be attributed to one of the aforementioned not modeled behaviors. In particular, during the testing campaign it has been noted how this measurement is sensitive to the position of the cables, an effect complex to mitigate or to model.

Table 4.9: Motor angles for the poses used for static analysis

Pose	$M1$ [°]	$M2$ [°]	$M3$ [°]	$M4$ [°]
1	0	0	0	90
2	0	-45	45	0
3	0	-45	45	90
4	0	-30	70	0
5	0	-30	70	60

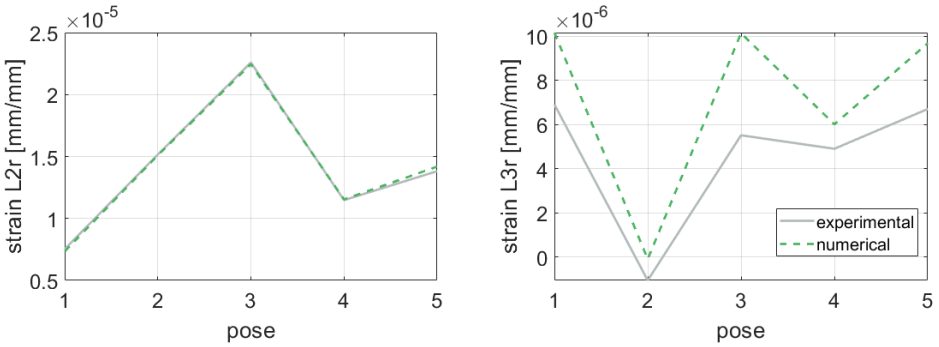


Figure 4.28: Comparison of measured and simulated strains for static tests poses on L2r and L3r on 5 different poses.

4.5.2.5 Driver extraction

The manipulator is controlled using a proprietary software that does not allow to directly input the desired motor trajectory at each time step, but only discrete poses that are then interpolated. Furthermore, the velocity value cannot be directly set. Two desired trajectories have been first generated in the control software. The motor angular values have then been acquired in order to replicate physically realistic trajectories in simulation and in estimation applications.

The first trajectory (referred to as *trajectory #1* in the rest of the work) consists in moving the robotic manipulator from its reference configuration to two poses (Figure 4.29) . The last motor, that rotates the end effector, is kept fixed. The duration of the trajectory is 9 seconds. This trajectory mimics a typical task done by such a manipulator in its lifecycle and explores a wide portion of the workspace while loading all the joints [101, 54]. Here, each link velocity is sufficiently high as to properly excite the global and local system dynamics.

The second trajectory (referred to as *trajectory #2* in the rest of the work) consists in a simpler motion in which the manipulator, starting from the reference configurations, reaches a poses at the extreme of its workspace (Figure 4.30). This pose is then kept fixed for 6 seconds. The aim of this second application is to have a more static set of measurements under the maximum gravity load available, to be used for estimation applications.

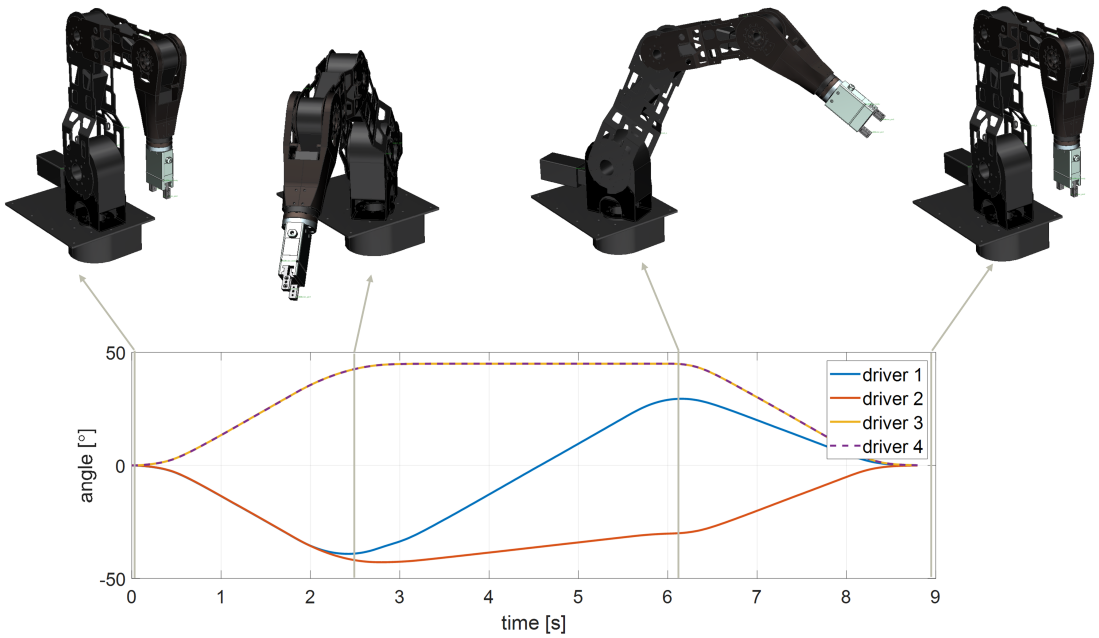


Figure 4.29: Time-history of driver angles and representative poses of the trajectory # 1 considered for validation of the Robolink pFMB model.

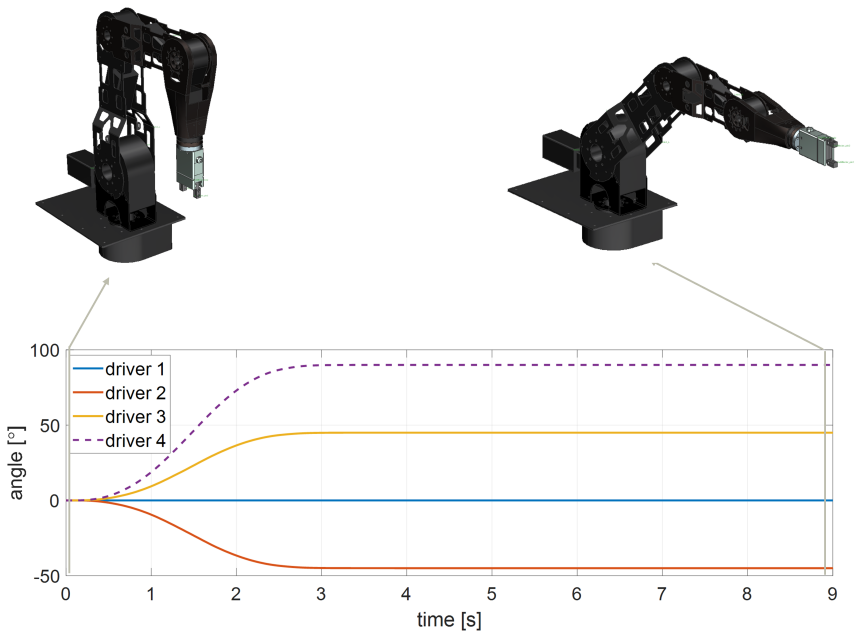


Figure 4.30: Time-history of driver angles and representative poses of the trajectory # 2 considered for validation of the Robolink pFMB model.

4.5.2.6 Final considerations

Few considerations can be made based on the results shown in this section. The achieved accuracy of the model is considered satisfying in order to provide an experimental validation of the methodologies proposed in this work. Nonetheless, many critical points have arisen during the process that would need further consideration and modeling efforts in future work.

On a component level, first of all, a more accurate modeling of the mass distribution could be considered in order to mitigate the static mismatches observed. Also the small difference in material parameters between the two links shows the need to further improve their models. On a system level, the main critical point has been the influence of the cables on the manipulator's behavior. This should be mitigated by properly fixing them and minimizing the forces they apply on the links. Even after these steps it would be interesting to investigate the modeling of such effect. Finally, the base mounting frame should be modeled more in detail.

It will be of interest in future research efforts also to validate the system with composite components.

4.5.3 Application cases for validation

A first validation of the proposed pFMB methodology on the Robolink model is done considering the static behavior of the system under gravity in a set of representative configurations in its workspace. While the main focus of FMB models is the dynamic modeling, the static response makes for an important component of measurements such as local strain or EE's deviation. It is therefore important to ensure that the proposed methodology can accurately represent such behavior.

For the exploration of the workspace in static configurations, the angles of M2 and M3 vary in the range $[0° \ 90°]$ with the other 3 motors fixed in the reference $0°$ angle. This choice is aimed at having a 2-dimensional configuration space that, while being representative, is easy to visualize. It should be noted that the angle of M1, given the kinematic configuration of the robot, does not influence the static response. Furthermore, given their position in the kinematic chain, also M4 and M5 have a less prominent effect than the two motors used for the validation.

For dynamic applications, the trajectory #1 presented in the previous section is simulated. During operation, the system is also subject to gravity along the global z axis.

The virtual measurements extracted for performance assessment are strains on L2r and L3r in the same locations as the physical system, the deviation of a point on the end effector with respect to the reference kinematic trajectory calculated with a rigid MB model (Equation 4.10), and its acceleration (Figure 4.31). Also in this case, the difference between FMB and pFMB models is evaluated by the error metric defined in Equation 4.11.

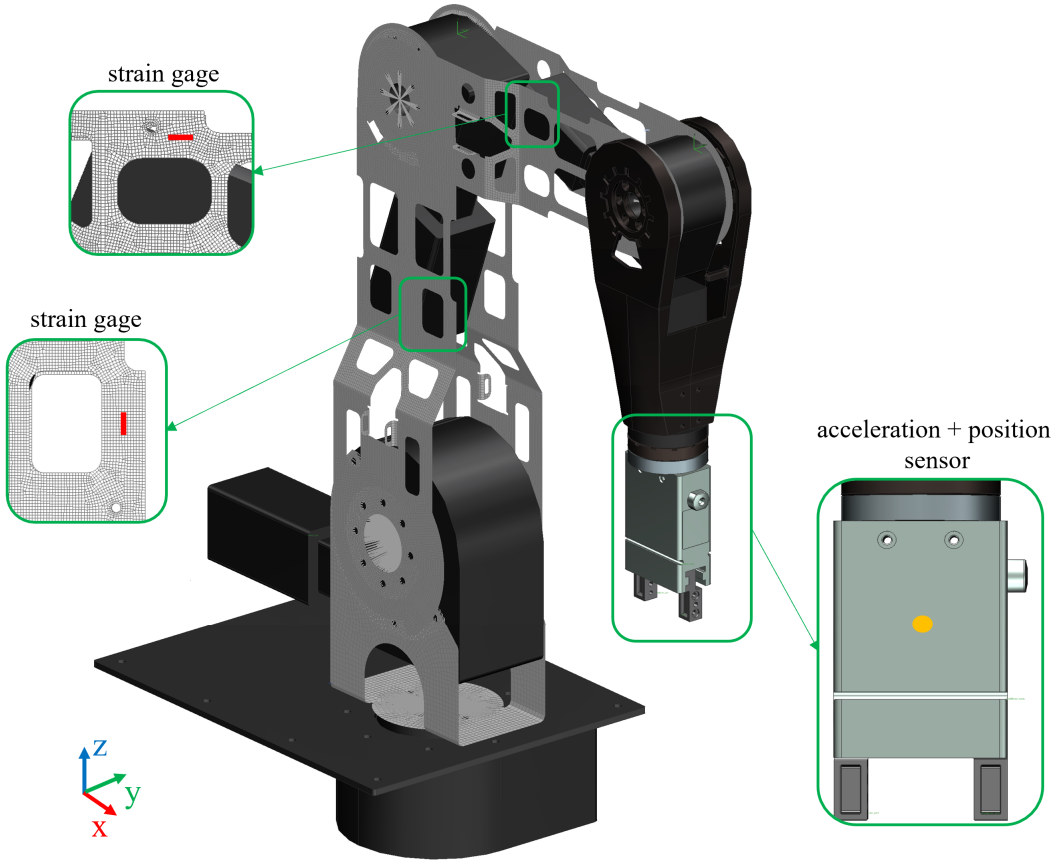


Figure 4.31: Sensors used for validation on the Robolink model

4.5.4 Results for the isotropic material model

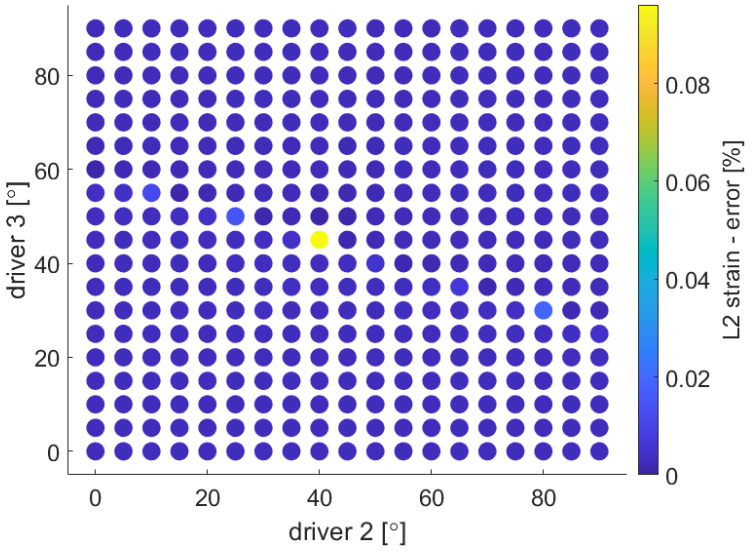
A first set of results is obtained using the pFMB model with isotropic material properties for each link. Three different combinations of parameter values not used in the training of the pROMs, defined as in the previous application case, are considered and the values are used for all the pROMs. A non-parametric FMB model evaluated for the same set of parameter values is used for comparison. The same integrator as in the previous section is used.

4.5.4.1 Static validation

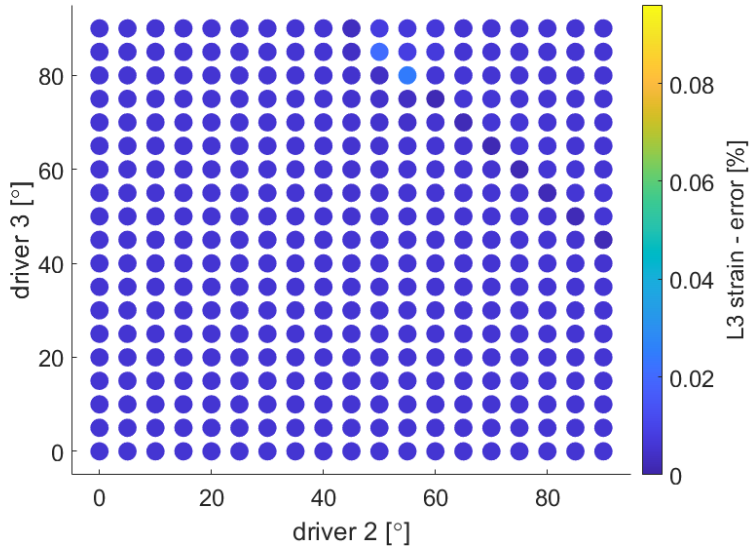
For the static validation, the joints configuration space for M2 and M3 is discretized on a 19x19 grid with 5° resolution in the [0° 90°] range. Both pFMB and FMB model in which the links have the reference material parameter values are used to run a static solution under gravity load. The position of the center of the EE $\mathbf{x}_{EE}$ is extracted and compared with the one of the rigid model $\mathbf{x}_{EE,R}$ by calculating the deviation:

$$d = \|\mathbf{x}_{EE} - \mathbf{x}_{EE,R}\| \tag{4.12}$$

In Figure 4.32 it can be seen as the percentage error of the EE deviation calculated using the two models is almost negligible over the entire configuration space. The same consideration can be made also by analyzing the error on the strain calculated on one point of L2r and one of L3r in the longitudinal direction (Figure 4.33).



(a)



(b)

Figure 4.32: Error between FMB and pFMB model with isotropic material and reference parameter values, for the strain on L2r (a) and L3r (b) over a 19x19 grid of configurations.

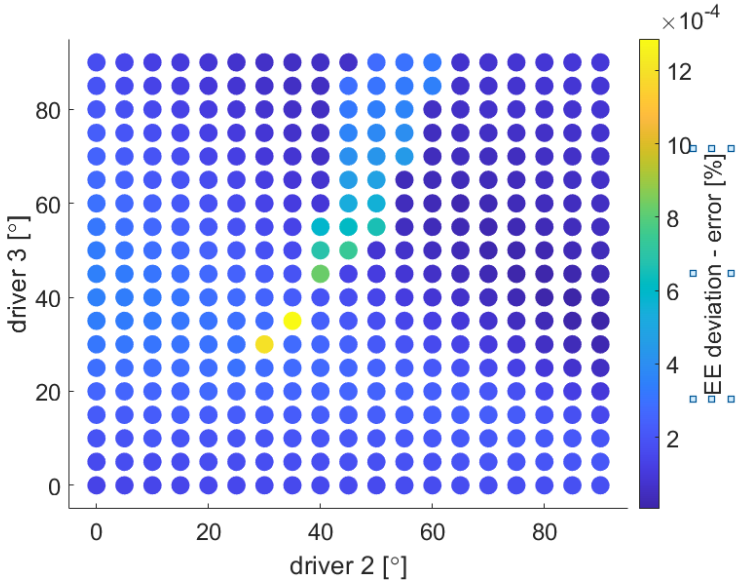


Figure 4.33: Error between FMB and pFMB model with isotropic material and reference parameter values, for the static EE deviation over a 19x19 grid of configurations.

4.5.4.2 Time domain dynamic validation

The validation on a dynamic application is done by using both FMB and pFMB models to perform the reference trajectory described in the previous section.

The EE deviation in this case is calculated as:

$$d(t) = \|\mathbf{x}_{EE}(t) - \mathbf{x}_{EE,R}(t)\| \tag{4.13}$$

The first-order BDF time integration scheme is used for the forward simulation presented in this section with an integration step $\Delta t = 1ms$.

As for the static results, a good matching between pFMB and FMB can be observed both for the deviation measurements and the acceleration of the same point on the EE (Figure 4.34, page 84) and for the strains (Figure 4.36).

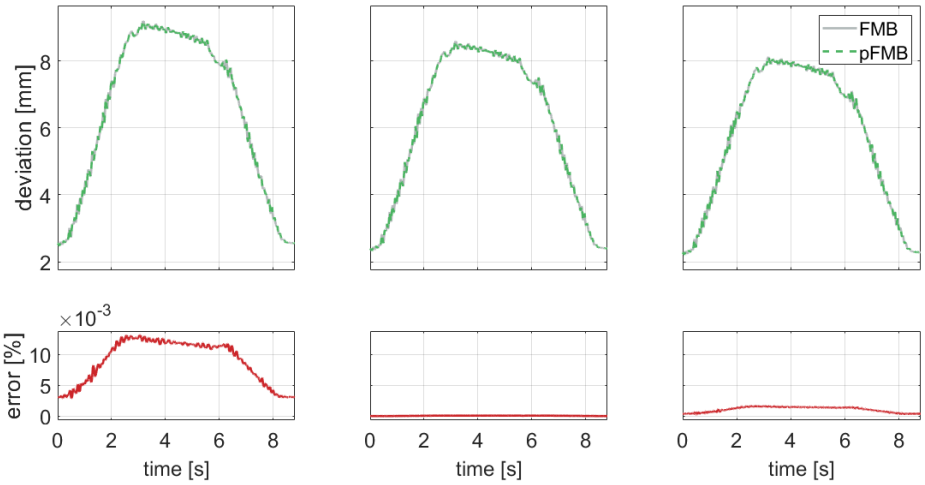


Figure 4.34: EE deviation for the isotropic material with parameter Set 1 (left), Set 2 (center) and Set 3 (right).

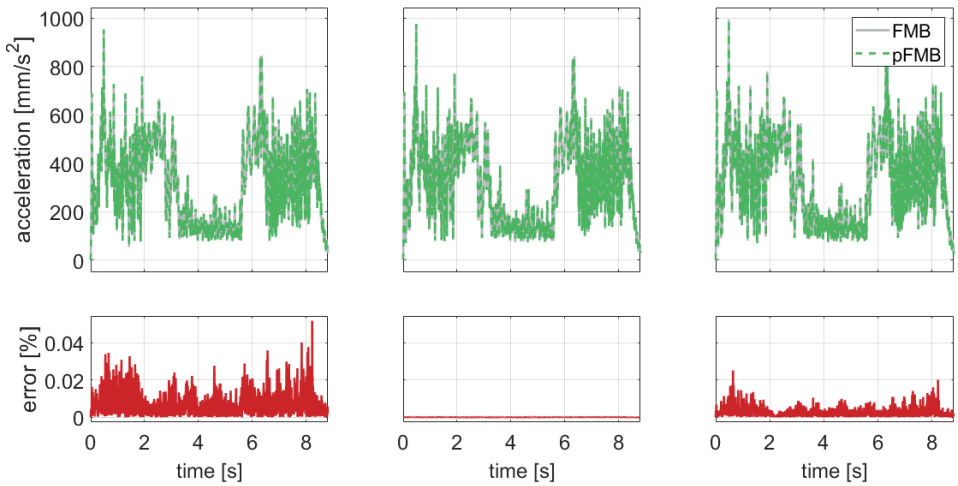


Figure 4.35: EE acceleration for the isotropic material with parameter Set 1 (left), Set 2 (center) and Set 3 (right).

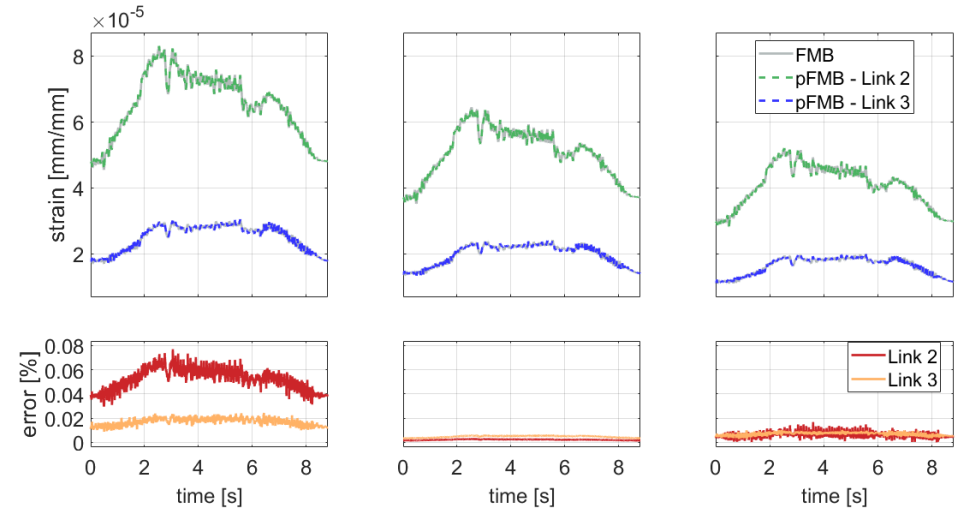


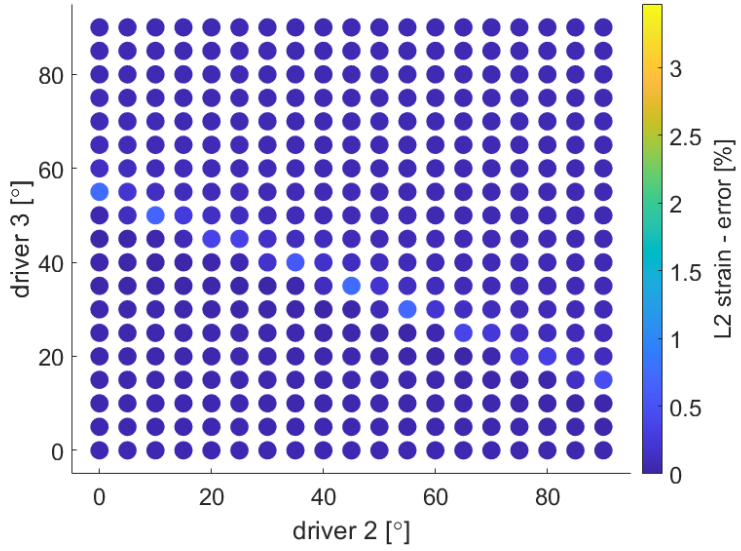
Figure 4.36: Strain of L2r and L3r for the isotropic material with parameter Set 1 (left), Set 2 (center) and Set 3 (right).

4.5.5 Results for the model with composite links

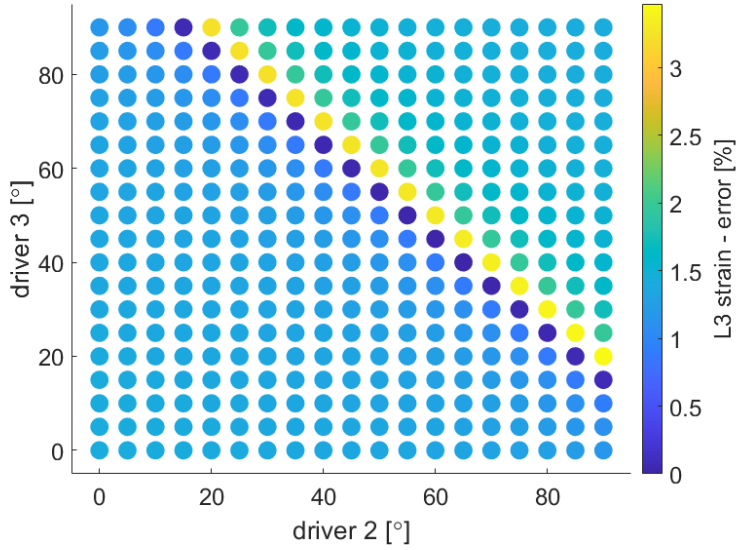
A second set of validation results is generated by replacing the pROMs of both Link 2 and Link 3 with their composite material variants. This allows to evaluate the performance of such models where the parametric dependency is more difficult to capture. All the other links use the same pROMs as in the previous section, evaluated for the same reference parameter values as discussed above.

4.5.5.1 Static validation

For the static comparison, the reference value of 0° for the fibers direction parameter is used. In the static results, it can be seen how the EE deviation matches properly (Figure 4.37). The strain measurement on the composite link (Figure 4.38) shows a satisfying error on average, with a zone in which it tends to increase but remaining in an acceptable range comparable with the error seen in the previous chapter for the component-level models.



(a)



(b)

Figure 4.37: Error between FMB and pFMB model with composite material for the strain on L2r (a) and L3r (b) over a 19x19 grid of configurations.

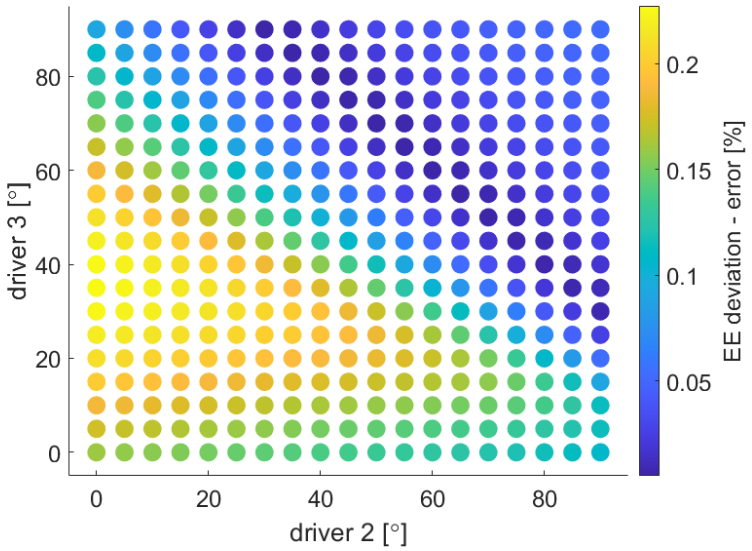


Figure 4.38: Error between FMB and pFMB model with composite material for the static EE deviation over a 19x19 grid of configurations.

4.5.5.2 Time domain dynamic validation

For the dynamic validation, the same trajectory and integration settings as for the isotropic model pROMs are used. Three parameter values at the center and at the extremes of the range of interest, namely 0° 45° and 90°, are considered. Again, a non-parametric FMB model evaluated for the same parameter values is used for comparison. For all of the parameter values, deviation (Figure 4.39), acceleration (Figure 4.40) and strain (Figure 4.41) virtual measurements show a satisfactory matching. It should be noted that the deviation, for this application case, exhibits a small sensitivity to low values of the fiber angles, with similar results for 0° and 45°.

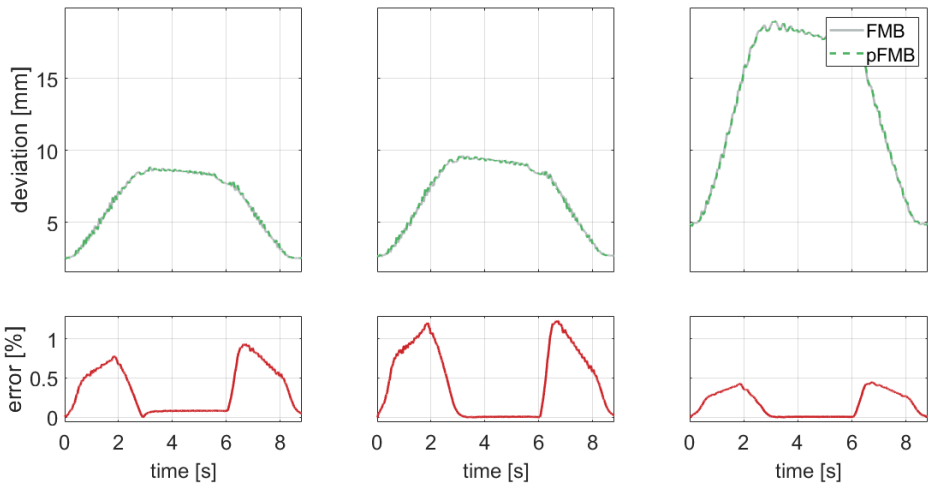


Figure 4.39: EE deviation for the composite material case with fiber angle at 0° (left), 45° (center) and 90° (right).

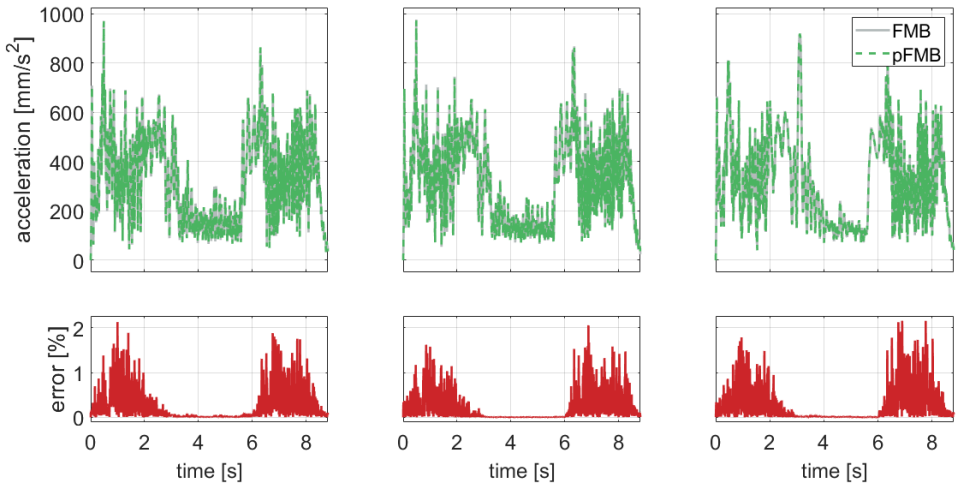


Figure 4.40: EE acceleration for the composite material case with fiber angle at 0° (left), 45° (center) and 90° (right).

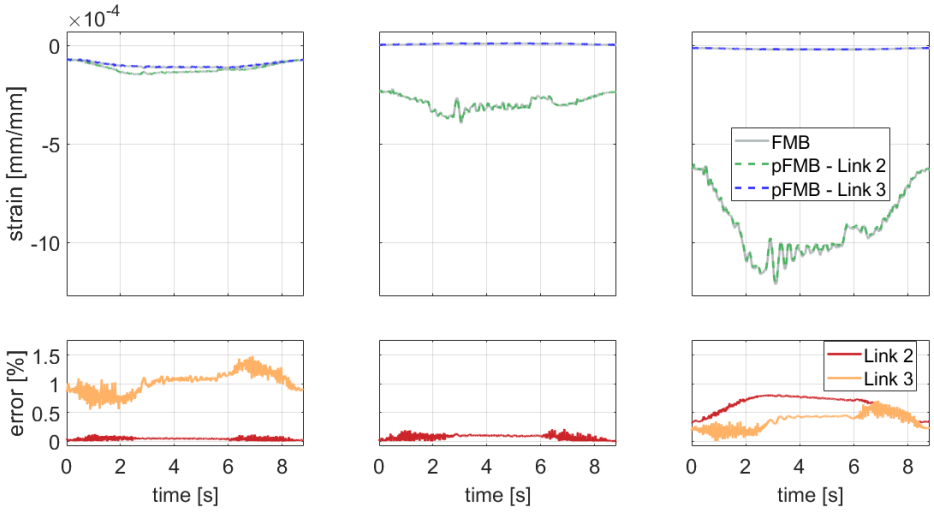


Figure 4.41: Strain of L2r for the composite material case with fiber angle at 0° (left), 45° (center) and 90° (right).

4.6 Conclusions and possible future work

This chapter proposed a novel parametric Flexible Multibody (pFMB) formulation that retains an explicit dependency on material parameters of the flexible bodies by employing a component-level parametric Model Order Reduction (pMOR) technique. The use of pROMs allows to have an explicit dependency on parameters in the stiffness and damping matrices of the system, and on the mass-invariant terms used to compute the mass matrix and the quadratic velocity terms. The remaining terms in the Equations Of Motion (EOMs) do not depend on parameters, given the constant reduction basis employed by the selected pMOR scheme. The resulting pFMB

models allow an efficient updating of parameters while maintaining a size in the same order as the standard non-parametric models, granting also computational efficiency.

The proposed methodology is validated on both open-chain and closed-chain mechanisms, using as benchmarks the flexible multibody models of a four-bar linkage and a robotic manipulator. Two representative classes of materials, i.e. isotropic and composite, are employed to model the flexible components. A good correspondence with non-parametric flexible multibody models over a wide range of parameter values for different kinds of virtual measurements extracted from time-domain simulations can be observed, confirming the accuracy of the proposed methodology. The procedure to update the manipulator model in order to match experimental measurements has been presented. The results show a good matching with few critical point for which explanations and possible future actions have been discussed.

Future efforts could be aimed at evaluating the effect of the time-varying parameters of the EOMs, and at considering how exact representation of the mass invariants is beneficial in high dynamics applications and in which case this can instead be simplified to further improve efficiency.

The methodology offers benefits for applications in which a large number of model updatings and solutions are required. The following chapters will apply it in the fields of design optimization as well as for the estimation and monitoring of the flexible multibody material parameters using Kalman Filtering approaches.

Part II

Optimal design

Chapter 5

Structural optimization for dynamic mechanical systems: Overview

5.1 Introduction

Given the complexity of most of the modern engineering applications, making design choices becomes a challenging task for which the application of traditional analytical, empirical or trial-and-error methods becomes impractical or even unfeasible. Furthermore, the solutions require high quality while satisfying numerous and often contrasting requirements. The use of design optimization methodologies to automatically find the best solutions to such problems has become common in most industrial applications. Most optimization methods employ numerical models to predict the performance of the system under different design choices. The models need to exhibit high accuracy while maintaining a good efficiency in terms of solution and design update.

For multibody mechanical systems, optimization is typically applied to kinematic synthesis, control or structural optimization of the components. This work in particular focuses on the structural optimization of flexible components of FMB systems, where the system topology (number of components, position and type of joints between them, boundary conditions) is considered fixed.

Three main kinds of structural optimization can be identified in literature based on the level of complexity of the problems [144]:

- **Parameter optimization**: the simplest strategy, focused on parameters of components (e.g. sizes of beams and shells structural elements, material or lumped elements properties) [66]. Shape and connectivity are not modified, so that the same FE models can be used while only varying their parameters;
- **Shape optimization**: the boundaries of the components' structure are optimized keeping their topology fixed [24]. This approach requires modifications on the CAD level and in turn also of the FE meshes;
- **Topology optimization**: no a-priori assumption is made on the layout of the component. A spacial design domain is defined, and only the boundary conditions and external loads are known a-priori [52].

The reduction of constraints to the possible designs deriving from optimization of topology or shape generally allows to identify designs with better performance with respect to simpler

approaches. Such gain comes at the cost of a larger number of variables, resulting in more complicated and computationally demanding problems to solve. On the other end, even a simple parameter optimization can allow to achieve substantial performance gains while keeping the problems relatively simple in terms of formulation. Such problems also present interesting computational challenges, given that often the considered design parameters require expensive updates of the FE models even when the topology of the mesh is kept fixed. Given the main applications of interest of this work, only parameter optimization problems will be considered in the following.

In this chapter, an overview on the theoretical background of structural parameter optimization methods is given. In Section 5.2 Multi-Objective Optimization problems and solution strategies are discussed. Section 5.3 discusses different approach to optimize components of FMB models. Finally, the main literature works on the optimization of robotic manipulators and composite materials are presented in Section 5.4

5.2 Multi-objective optimization

The choice of design parameters for a mechanical system commonly requires to take in to account multiple competing goals while satisfying constraints linked to e.g. feasibility, cost and safety. Multi-objective optimization (MOO) techniques are commonly employed in such problem [128, 115, 9, 102]. The goal of a MOO is to find the combination of Design Variables (DVs) that minimizes a defined set of Objective Functions (OFs) while satisfying a set of Constrain Functions (CFs). This work focuses on the application of such optimization methods to design of mechanical systems considering dynamical applications. The performance is evaluated on the behavior of the system in one or more specified tasks, while it is also possible to keep into account task-independent features (e.g. manufacturing cost, weight) as will be explained later.

5.2.1 Problem formulation

The generic mathematical formulation of the dynamic MOO problem is:

$$\begin{aligned} \min_{\mathbf{p}} \quad & f_i^{obj}(\mathbf{p}, \mathbf{q}(t), \dot{\mathbf{q}}(t), \boldsymbol{\lambda}(t), t) && i = 1, ..., n_{obj} \\ & && t \in [t_0, t_f] \\ s.t. \quad & f_j^{con}(\mathbf{p}, \mathbf{q}(t), \dot{\mathbf{q}}(t), \boldsymbol{\lambda}(t), t) \leq \overline{f}_j^{con} && j = 1, ..., n_{con} \\ & \underline{p}_k \leq p_k \leq \overline{p}_k && k = 1, ..., n_p \end{aligned} \tag{5.1}$$

The parameters of the system $\mathbf{p}$ are commonly referred to in optimization literature as independent DVs, being the values that are selected by the optimization algorithm. Each specific combination of the independent DVs is defined in optimization literature and in the rest of this work as a *design*. The states of the system $(\mathbf{q}(t), \dot{\mathbf{q}}(t), \boldsymbol{\lambda}(t))$ are instead dependent DVs, deriving from the solution of the EOMs given the design and the application case. DVs can vary continuously in the parameter space, or only be allowed to take values from a discrete set (e.g. when choosing between alternative solutions already available). Their number should be kept into account since, making the optimization problem tougher, it can greatly influence the performance of search algorithms.

The OF set $\begin{bmatrix} f_1^{obj} & \cdots & f_{n_{obj}}^{obj} \end{bmatrix}$ represents the performance indexes of the design. The models used to evaluate OFs should be efficient in terms of updating for the DVs, ideally having an explicit dependency on them. The update need to be done at each iteration of the search,

easily resulting in a computational bottleneck if not efficiently treated. Furthermore, the explicit dependency simplifies the calculation of sensitivities in case of a gradient-based approach. A set of CFs $\begin{bmatrix} f_1^{con} & \cdots & f_{n_{con}}^{con} \end{bmatrix}$ is defined in order to limit the solution space to feasible designs. The feasible values of the independent DVs are bounded to realistic and physically meaningful values, keeping into account also what solutions are commercially available

5.2.2 Solution process

The OFs commonly refer to contrasting performance indexes, resulting in the impossibility to find a unique set of parameters that jointly minimize all of them or even in objectively comparing two designs. Solutions are most commonly compared based on their Pareto dominance relationship [115, 9]. A design $\mathbf{p}$ is non-dominated if there is no other design that improves it on every OF. The set of non-dominated solutions is commonly referred to as the Pareto Front (PF):

$$PF = \{\mathbf{p} \in \mathbf{\Omega} : \{\mathbf{p}' \in \mathbf{\Omega} : f_i^{obj}(\mathbf{p}) \geq f_i^{obj}(\mathbf{p}') \; \forall i \in [1 : n_{obj}]\} = \emptyset\} \tag{5.2}$$

Given the non-unique nature of the solution to MOO problems, the designer, referred to in literature as Decision Maker (DM), needs then to step into the process by expressing a preference. Solution processes are classified according to when the decision is made [115, 51]:

- **A priori preference methods:** the DM preference is provided before the optimization starts and then used by the algorithm to find a unique solution. The main downsides are that the preference is expressed when the DM might have limited knowledge on the problem, and that several runs are required to compute a PF;
- **Interactive preference methods:** the DM articulates its preference during the optimization process, for example after a fixed number of ND solutions is found or at each iteration. This method has the advantage of using the knowledge acquired by the DM to direct the search, but result impractical given the continuous feedback required by the user;
- **A posteriori preference methods:** the DM does not express any preference until the end of the process. The algorithm directly calculates the entire PF and only then the DM chooses the preferred design. While often requiring a larger number of design evaluations, this approach allows for a wider exploration of the problem thus giving the DM the best possible view on the solutions available.

The a priori approaches often reduce to single-objective problems, resulting in less interesting applications. Interactive approaches, while of interest for real world applications, result in the mix of techniques from both a priori and a posteriori methods. In the rest of this work, only a posteriori methods are considered. These allow, as mentioned, to have a better view on the trade-off between solutions while offering also an interesting challenge.

Several a-posteriori search algorithms for the optimal solutions are available. Most of the common search algorithms consist in evaluating the performance of one or more designs on which the choice for the consequent design to test is based. Such algorithms aim at finding a set of designs that better approximates the entire PF.

A large variety of algorithms is available from literature to solve optimization problems. Different classifications are possible based on e.g. their deterministic nature, the use of derivatives of the problem, or the phenomena on which they are inspired. One main distinction can be done between algorithms that require the evaluation of derivatives of OFs and CFs with respect to DVs, defined as gradient-based, and others that only require the evaluation of the functions themselves, named zeroth-order algorithms.

Gradient-based algorithms employ knowledge of the sensitivity of OFs and CFs of the DPs in order to select at each iterations the parameters update that guarantees the largest increase in performance [57]. The gradients of objective functions $\nabla \mathbf{f}^{obj}$ and constraints $\nabla \mathbf{f}^{con}$ need to be calculated, where:

$$\nabla \mathbf{f} = \frac{\partial \mathbf{f}((\mathbf{p}, \mathbf{q}(t), \dot{\mathbf{q}}(t), \boldsymbol{\lambda}(t), t))}{\partial \mathbf{p}} = \begin{bmatrix} \partial f_1/\partial p_1 & \cdots & \partial f_1/\partial p_{n_p} \\ \vdots & \ddots & \vdots \\ \partial f_{n_f}/\partial p_1 & \cdots & \partial f_{n_f}/\partial p_{n_p} \end{bmatrix} \tag{5.3}$$

The calculation of the gradients is commonly referred to as sensitivity analysis. The main methods to compute sensitivities of OFs and CFs are [148]:

- **Numerical differentiation:** each design parameters is perturbed by a small quantity and the updated model is solved. The responses are employed in a finite differences method to compute their sensitivities.
- **Direct differentiation:** the EOMs are analytically differentiated to obtain state and Lagrange multipliers sensitivity ($\partial \mathbf{q}/\partial \mathbf{p}$, $\partial \dot{\mathbf{q}}/\partial \mathbf{p}$, $\partial \ddot{\mathbf{q}}/\partial \mathbf{p}$, $\partial \boldsymbol{\lambda}/\partial \mathbf{p}$). These are in turn employed to analytically calculate the OFs gradients using the differentiation chain rule.
- **Adjoint variable method:** The EOMs are enforced as constraints in the optimization problem using the Lagrange optimization method. This avoids the need to compute the states and Lagrange multipliers sensitivity, only requiring to calculate the sensitivity of a set of so-called adjoint variables.
- **Automatic differentiation:** The functions are decomposed down to basic operations through repeated application of the chain rule of differentiation. This allows to automatically compute back the sensitivity functions without need of manual implementation.

Numerical differentiation methods , while being straightforward to implement, require the update and forward simulation of the model a number of time equal to the number of design parameters at each iteration, resulting in a large computational effort. Furthermore, the accuracy is not comparable with the one of the objective function. The analytical and automated approaches instead allow for a more efficient and accurate calculation, given that the derivatives of EOMs on parameters have an analytical expression, that in turn requires an explicit dependency. If this is not the case, as in FMB models when e.g. material properties of the flexible bodies are considered, approximations need to be introduced.

Given that such approaches could lead to local optima, in the case of non-convex design dominions it is common to execute multiple searches starting from different points in the design space. The main advantage of gradient-based method is the reduced number of iterations required to reach an optimal solution. The calculation of sensitivities can be computationally and theoretically complex for complicated systems, and furthermore such approaches require smooth and regular problems. Furthermore, in MOO, gradient-based methods often do not allow to reconstruct the entire PF.

Zeroth-order approaches, on the other end, do not require the computation of gradients. The algorithms rely on function evaluations at various points to estimate the direction of the optimum and adjust the search direction accordingly. These algorithms may use heuristics, search patterns, or model-based approaches to navigate the search space and find the optimum. Recent years have seen a large number of literature works proposing different methods, and consequently also many different classifications. Given that most zeroth-order approaches are inspired by natural phenomena [28], one of the most common categorization uses their inspiration

[115] as based on evolution [37], physical phenomena, behaviors related to humans and swarms [36]. The reader is referred to the cited works for a detailed overview of such methods, while only generic considerations on how they operate and their main advantages and disadvantages are given in this and subsequent sections.

Employing such algorithms enables optimization of more complex and irregular problems with a high number of DPs and constraints, given that they do not require an in-depth knowledge of the problem. The objective functions can be irregular and non-convex, without risking to get stuck in local optima as would be the case with gradient-based methods. On the other end, given that the only information of the problem is retrieved by evaluation of the functions at different designs, a larger number of evaluations is required; this in turn makes such approaches computationally more demanding. Furthermore, no strict optimality criteria can be defined resulting in no guarantee that the solutions proposed are the optimal ones.

5.2.3 Treatment of time-dependent functions

Each of the OFs and CFs has an implicit dependency on DVs though the states of the system. These are calculated by solving the EOMs for the considered design in the time window of interest at each iteration of the optimization algorithm. This solution is commonly solved by a discretization of the time interval so that states, and in turn functions, are computed at each time step:

$$\begin{aligned} \mathbf{f}^{obj} &= \begin{bmatrix} f^{obj}(t_0) & \cdots & f^{obj}(t_k) & \cdots & f^{obj}(t_{n_{step}}) \end{bmatrix} \\ \mathbf{f}^{con} &= \begin{bmatrix} f^{con}(t_0) & \cdots & f^{con}(t_k) & \cdots & f^{con}(t_{n_{step}}) \end{bmatrix} \end{aligned} \tag{5.4}$$

where $t_k = k\Delta t$ is the time value at the k-th step. In an ideal scenario the sets of functions of interest would be defined by their value at each time step, so that the best performance is guaranteed during the entire dynamic application case considered. This would results in an extremely large problem often too expensive and impractical to solve [85]. One possible approach is to calculate them as the integrand of a function φ over the time window of interest, so that the generic function f_i (either representing an OF or a CF) becomes in continuous time:

$$f_i = \int_{t_0}^{t_f} \varphi_i(\mathbf{p}, \mathbf{q}(t), \dot{\mathbf{q}}(t), \boldsymbol{\lambda}(t), t) dt \tag{5.5}$$

or in discrete time:

$$f_i = \sum_{k=0}^{n_{step}} \varphi_i(\mathbf{p}, \mathbf{q}(t_k), \dot{\mathbf{q}}(t_k), \boldsymbol{\lambda}(t_k), t_k) \tag{5.6}$$

This solution allows to average the performance index or constraint over the entire time window, reducing each of them to a single scalar value. Information on time-local behavior is lost using this approach, resulting in potential parts of the application with performances worse than desider.

A second approach commonly employed consists in extracting the maximum of the function in the time window:

$$f_i = \max_t(\varphi_i(\mathbf{p}, \mathbf{q}(t), \dot{\mathbf{q}}(t), \boldsymbol{\lambda}(t), t)) \tag{5.7}$$

that in discrete time becomes:

$$f_i = \max_k(\varphi_i(\mathbf{p}, \mathbf{q}(t_k), \dot{\mathbf{q}}(t_k), \boldsymbol{\lambda}(t_k), t_k)) \tag{5.8}$$

Opposite to the previous definition, in this case a bounding of the function is guaranteed, but the overall behavior is not kept into account. Furthermore, the time in which the maximum

occurs can vary between designs, making the function irregular and difficult to treat if gradient-based approached are employed.

It is also possible to consider the static response of the system in one particular configuration under a set of loads. In this case:

$$f_i = \varphi_i(\mathbf{p}, \mathbf{q}, \boldsymbol{\lambda}) \tag{5.9}$$

A third possible definition is that of global, i.e. task-independent, functions. These are not dependent on the particular application case and thus the time dependency is directly eliminated. The function can then be only dependent on parameters, such as the case when some physical property inherent to the system is considered:

$$f_i = \varphi_i(\mathbf{p}) \tag{5.10}$$

Also the modal properties of the considered structures are often employed as a global performance indicator.

5.3 Optimization using Flexible Multibody models

Design optimization has been initially developed for static responses of structures, for which is is a well established industry tool. Recent years have seen focus placed also on considering response to transient loads [85] and applying it to systems made up of multiple components [143, 29].

Traditionally, the structural design of the components of a mechanism has been done on the component level, ignoring or highly approximating its interaction with the entire system by replicating the boundary conditions and static loads generated during operation. In designs for which the flexibility of the components plays a major role (e. g. lightweight and highly dynamic systems), the interaction between components cannot be ignored anymore. Flexible multibody models allow to account for the effect of design variation of one component on the dynamic performance of the entire system, making for ideal models to use in optimization applications [67, 31].

Two main classes of techniques exist for structural optimization of FMB systems: weakly and fully coupled methods [142]. In the weakly coupled approach, the static response of a component under a set of equivalent static loads is optimized [86]. At each iterations, a FMB simulation is performed from which the dynamic loads on the components of interest are extracted. A set of equivalent static loads with the same effect are then computed and used for a static optimization on the components [86]. This cycle is iterated so that the static loads are relative to the most recent components design. Such approach offers the advantage of a much easier objective function to optimize, that results particularly advantageous when gradient-based algorithms are used, for which computation of the sensitivity of the system-level responses is a challenging task [148, 76, 162].

The fully coupled approach employs the time response of the entire MB system in the optimization problem [144]. When such strongly coupled approach is employed, the optimization problem can fully capture the behavior of the system.

If the problem concerns component-based OFs and CFs, i.e. each component is evaluated only based on its own response, strongly and weakly coupled approaches offer equivalent results. When, instead, the functions of interest for the problem depend on the response of multiple components, the weakly coupled approach is generally not suitable if not for few particular cases [142]. The secondary structural static optimization problem solved in the weakly coupled approach loses in fact any information on the interaction between components.

The main disadvantage of the strongly coupled approach is the growth in complication and cost of the problem to solve. When a sensitivity analysis is required by the optimization algorithm

employed, such methods are particularly challenging given that the FMB EOMs are commonly implicitly dependent on the independent DVs of interest (e.g. material parameters, components size). Such effect is mitigated when zeroth-order algorithms are employed, at the costs highlighted in the previous section for such approaches

Given the considerations made in this section, it results of particular interest to investigate how a strongly coupled problem can be solved through zeroth-order methods employing efficient models. Such application will be described in the next chapter.

5.4 Optimization for robotics and composite applications

Optimization has proven an essential design tool in almost all mechanical engineering applications. This works focuses in particular on bringing together robotic manipulators and composite materials, so the main efforts in literature to apply optimization to these fields will be briefly analyzed in this section.

Optimization in the robotics field is widely applied for control or mechanisms synthesis applications. A smaller number of works apply optimization for structural optimization of the components. Of particular interest is the analysis done by Russo [121] on performance metrics for robotic manipulators. The distinction between global and task-based indexes is discussed, together with the most common metrics used in design optimization. Among the main objectives of design optimization for robotic manipulators is energy consumption [61, 150]. [135] points out three main approaches to minimize energy consumption: topology optimization, trajectory optimization and lightweight components. The last solution is of particular interest for the application considered in this work. Often the mass of the single links or of the overall manipulator is used as a performance index (e.g. [157] optimize structure of links and motor models). Such choice does not keep into account weight distribution and how the inertia impacts the performance in particular dynamic applications. Optimization of robotic manipulators employing FMB models has been faced in few literature works (e.g. [68, 95, 113]).

Given their highly complicated dependency on design parameters, the design of composite components is a well established field of application for optimization techniques [133]. Ambrosio et al. [12] consider the optimization of composite components of flexible multibody models. The use of composite materials in robotic manipulators has been analyzed in few seminal papers in the 90s [123, 33]. More recently Yin et al. [159, 158] employ multi-objective optimization for the choice of material parameters of links in composite material, considering mass and first eigenfrequency as OFs. Their approach is based on structural response, not considering true dynamics, and furthermore the optimization uses a weighted sum of objective functions that reduces the problem to a single-objective optimization.

Chapter 6

Design optimization of parametric Flexible Multibody models

6.1 Introduction

Most of common optimization algorithms employing high-fidelity numerical models require the model to be recursively updated at each iteration. The updated model is then used to evaluate the objective functions. An efficient model both in terms of update and solution is consequently of paramount importance in order to reduce the computational time for each step of the search. Such efficiency results in both a speed-up of the whole procedure and possibly more explored designs. Evaluating as many designs as possible is especially important in zeroth-order search algorithms in which there is no optimality guarantee. If gradient-based techniques are used, instead, the explicit dependency of models on Design Variables is desirable in order to enable efficient computation of the sensitivities.

Component-level structural optimization is challenging when standard ROMs are employed. The updating results in fact inefficient, and calculation of the sensitivities of responses to parameters is not straightforward given the implicit dependency of the ROMs on the parameters. Exploiting efficient parametric reduced models such as the pROMs presented in Chapter 3 can greatly benefit and even enable structural optimization procedures. Such challenge is commonly addressed in literature (e.g. [43, 161, 160]).

For the system-level optimization of flexible components exploiting FMB models, in literature it is common to either employ small-size models or re-reduce the models at each iteration. Employing the parametric models presented in Chapter 3 and Chapter 4 can enable the use of accurate models while granting the efficiency of the optimization procedures by removing the need to re-compute the component models. This chapter in particular focuses on the most challenging application, consisting in the optimization of pFMB models using a strongly coupled approach. The considerations given can be applied seamlessly to the optimization of single isolated bodies using pROMs.

In Section 6.2 a generic optimization workflow that employs pFMB models is described. The methodology is then applied to the design optimization of the Robolink manipulator in Section 6.3. Its principal steel links are replaced with composite ones, of which the optimal fiber angles and number of layers combinations are investigated. Two different scenarios are presented, in which first only L2 and then both L2 and L3 are replaced. The use of an experimentally validated baseline and composite material models grants a good level of confidence in the numerical

results to be representative of realistic behaviors.

6.2 Design optimization of mechanisms using pFMB models

This section discusses a generic optimization workflow employing pFMB models, highlighting the main advantages for different classes of search algorithms.

The workflow for the generic optimization algorithm, at each iteration, consists in (Figure 6.1):

- **DVs selection:** the algorithm selects the design to explore at the current iteration based on its decision logic and the information it acquired on the system in previous iterations;
- **Model update:** the FMB model is updated with the selected set of DVs;
- **Time integration:** a forward simulation of the considered dynamic application is done in order to extract the time history of the system's states;
- **Sensitivity analysis:** if a gradient-based optimization algorithm is employed, also the sensitivities of OFs and CFs with respect to DVs are computed;
- **Constraints and performance evaluation:** the functions of interest for the optimization problem are evaluated using dependent and independent DVs relative to the current design;
- **Constraints and convergence check:** if the constraints are not met, the current design is discarded. Otherwise, its performance is employed to evaluate convergence. If not converged, the algorithm proceeds with the following iterations.

After convergence is reached, the entire PS is evaluated based on a search of the non-dominated designs.

The steps in which the model is employed are in particular the forward simulation required to extract the states evolution, and the sensitivity calculation if a gradient-based algorithm is used. At each iteration, the model is updated to match the DPs chosen by the algorithm. The update procedure using a pFMB model is inexpensive and straightforward given the explicit dependency of the model on parameters (Figure 6.2b), while it would require a solution of the FOMs and reduction step for a classical FMB (Figure 6.2a). Such computational advantage comes at the price of a pre-processing step in which the pROMs need to be computed. The impact of this on the overall optimization process is small in cases where the parametrized models are generated once and then several optimization runs are done. In the case for which the pFMB models need to be generated at the start of each new optimization, the impact of the pre-processing phase depends on the number of designs evaluated during the search. Such number is commonly large for zeroth-order methods in comparison to gradient-based ones, so it is in the former that employing pFMBs allows for the larger gain in the update phase.

Another aspect to keep into account is the increase in size of the models when parameterized. This can impact the simulation time, so the balance between the two effects needs to be considered. In a discussion about pROM that can be extended also to pFMB models, Benner et al [20] point out how the use of pROMs trained on the full parameter space might be inefficient in optimization algorithms, given that some portions of the space might never be explored by the search algorithm. This would in turn result in wasted training time and useless increase in the model size. This issue can be mitigated by adaptively re-generating the pROMs for the portions of parameter space explored during the optimization. Such an approach, while of interest and offering advantages in certain applications, is considered out of the scope of this work and not

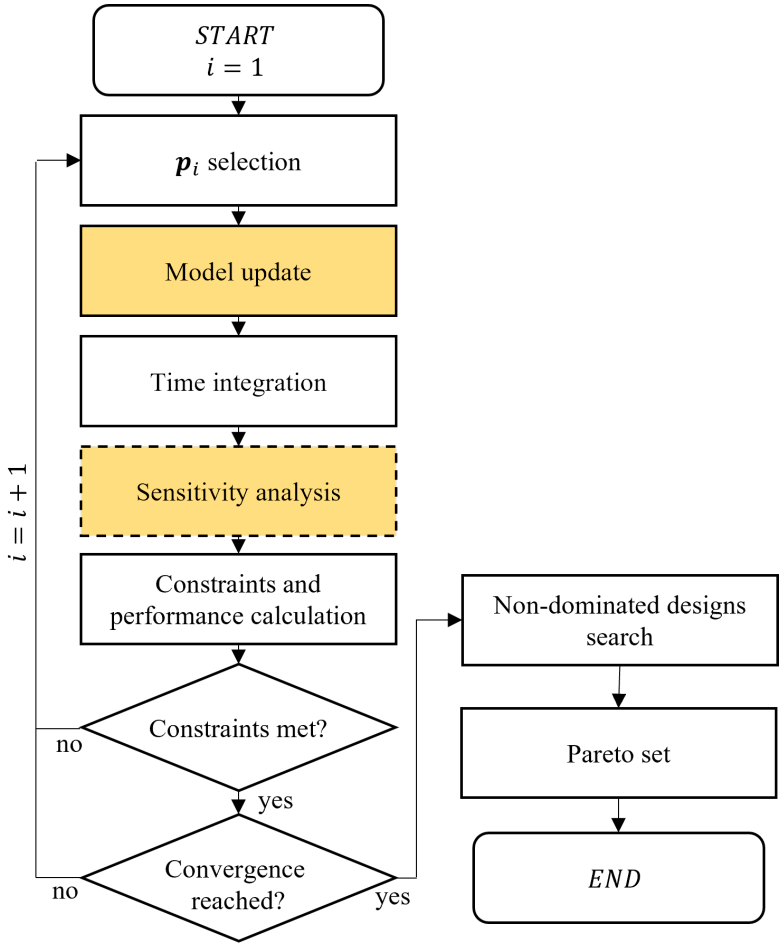


Figure 6.1: Generic optimization workflow The steps in which the use of a pFMB model is beneficial are highlighted.

further explored. The main aim of the work is in fact showcasing the feasibility of using pFMB models in optimization and not achieving the best possible performance. Furthermore, such approach would mainly benefit gradient-based searches where the path in the parameter space is regular and moves to the final designs quickly. This work focuses on the use of zeroth-order algorithms that commonly explore a large portion of the parameter space, where such an approach would be of less benefit.

While offering a smaller advantage in terms of solution time, in the gradient-based approaches the use of a pFMB model can still result in interesting advantages in the sensitivity analysis phase. If analytical methods are employed, in the calculation of the derivative a sub-step often requires to calculate the derivative of system matrices with respect to parameters $\frac{\partial \mathbf{M}}{\partial p_i}$, $\frac{\partial \mathbf{D}}{\partial p_i}$, $\frac{\partial \mathbf{K}}{\partial p_i}$ for each parameter of interest. This computation requires the sensitivity information regarding the FOM matrices, that can be commonly stored at a large memory cost. Furthermore, the challenging problem of computing the sensitivity of the reduction basis need to be tackled, commonly approached by either complicated calculations or simplifications that impact the quality of the calculated gradients. Employing a pFMB model allows to overcome both challenges by making the analytical differentiation immediate and exact, given the form of parametric dependency

in the pROMs employed to model the components. If finite difference approaches are used to compute the sensitivity, the model needs to be updated one or two extra times at each iterations. Benefits of pFMB models in the update phase have been already discussed and apply also for this case.

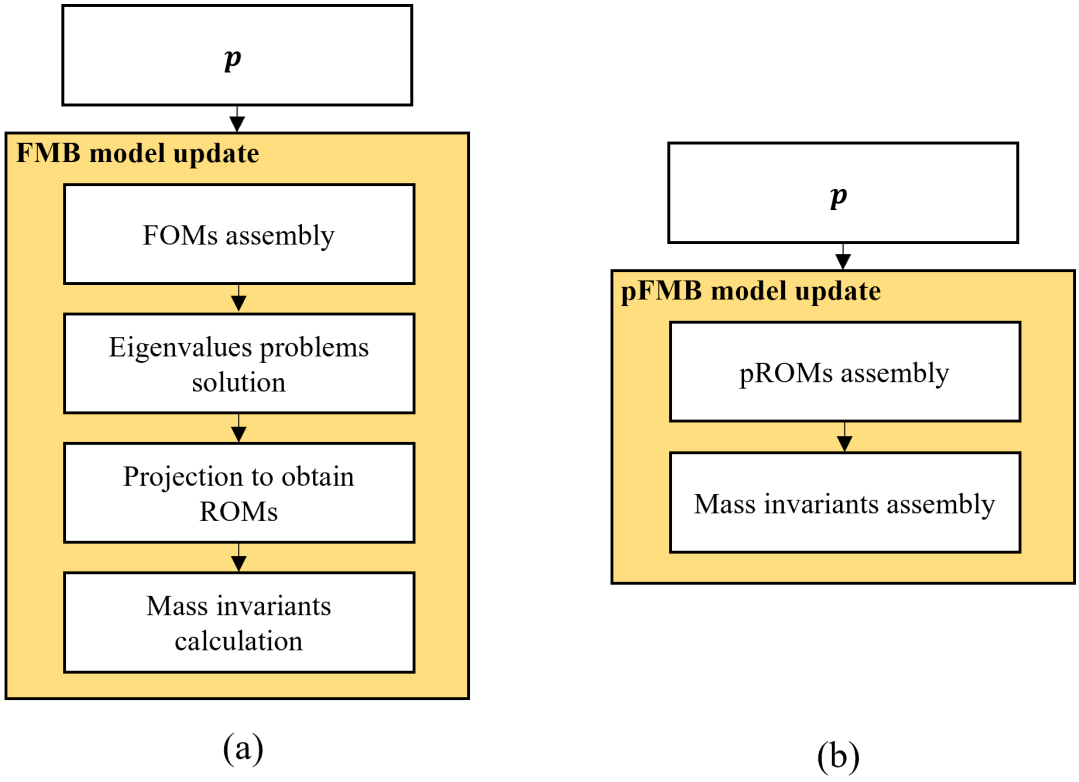


Figure 6.2: Model update step using a non-parametric FMB model (a) and a pFMB model (b).

6.3 Study-case: optimization of composite links of a robotic manipulator

The main application areas for design optimization are the initial design of a system and the improvement of an existing one. The proposed modeling methodology can be seamlessly implemented in both application cases. The focus of this work is on the second case, aiming at optimizing the performance of the Robolink manipulator by the use of composite materials.

This section presents a study-case for the use of pFMB models inside a design optimization procedure. The experimentally updated model presented in Chapter 4 is used as the baseline design. The aim of the design optimization is the improvement of performance, both in terms of accuracy and energy consumption. With this goal in mind, the main links of the manipulator (L2 and L3) are replaced with composite ones. These are in fact the components that undergo the largest deformation during standard maneuvers, and they also present a geometry that facilitates manufacturing. The optimal composite parameters in terms of fiber angle and number of layers are identified. The composite links are modeled employing the experimentally validated material model discussed in Chapter 3. Given the usage of validated models, the numerical results achieved

are expected to closely match their physical prototypes. Nonetheless, it will be of interest for future research efforts to experimentally validate such results with tests on the manipulator with composite links.

An overview of the application with considerations on the design parameters and objective functions will be given. Then the results for two cases will be presented, in which first L2 and then both L2 and L3 are replaced with composite parts.

6.3.1 Composite links

The original models for L2 and L3 in isotropic material have been initially presented and experimentally validated in Chapter 4. The composite versions present a simplified design with respect to the original isotropic components (Figure 6.3). Such simplification is done in order to minimize the post-processing in the manufacturing phase. Composite materials are in fact prone to damages when machined [153, 50, 92]. The original design presents holes with the purpose of reducing the weights of the components, whose removal in the composite versions is balanced by the lower density of the material. The composite versions allow in fact for a mass reduction of the components between 50% and 80% (Table 6.1, Table 6.2). The ending sections are left unmodified given that they connect to the motors, allowing for the same assembly procedure without further modifications to the system.

The ply properties depend on the particular material employed, chosen based on what is available on the market. The design parameters left to the designer to choose are fiber direction and number of layers. The fiber direction dependency is directly available through the use of pROMs, but no affine representation is available for the number of layers. Furthermore, being a parameter that varies discretely in a set of reduced size, the parametrization cost in terms of size of the pROMs would not be justified. The approach employed in this work consists in creating a library of models, differing from each other only for the number of layers. For both links, four different values for n_l (6, 8, 10, 12) are considered. All the pROMs are created in the pre-processing phase of the optimization and switched at each search step according to the parameter selected by the algorithm.

Table 6.1: L2 (sum of both sides) - isotropic vs composite

model	thickness [mm]	mass [Kg]	mass reduction
isotropic	2	1.1702	
composite (6 layers)	1.8	0.2652	77.3372%
composite (8 layers)	2.4	0.3536	69.7829%
composite (10 layers)	3.0	0.4420	62.2287%
composite (12 layers)	3.6	0.5304	54.6744%

Table 6.2: Comparison of isotropic and composite models of L3 (considering the sum of both sides for mass)

model	thickness [mm]	mass [Kg]	mass reduction
isotropic	2	0.7126	
composite (6 layers)	1.8	0.1680	76.4244%
composite (8 layers)	2.4	0.2240	68.5658%
composite(10 layers)	3.0	0.2800	60.7073%
composite(12 layers)	3.6	0.3360	52.8487%

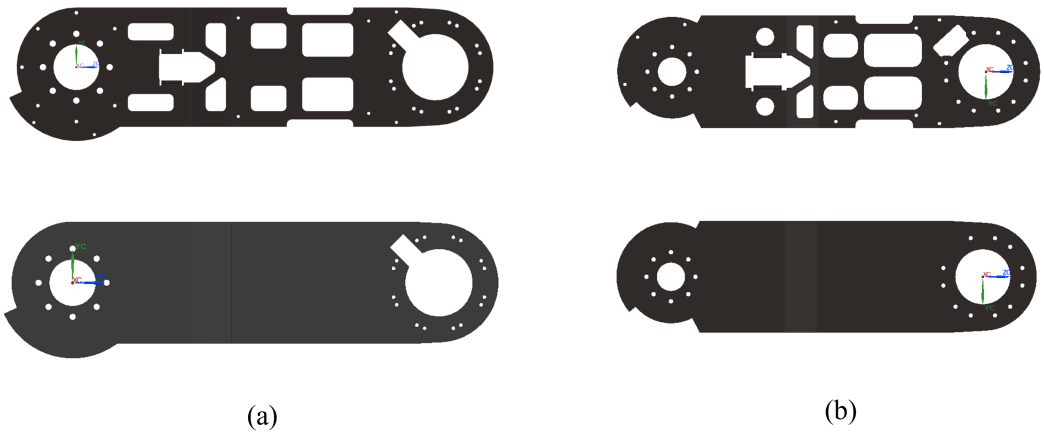


Figure 6.3: Original geometries (top) of links 2 (a) and 3 (b) compared with their simplified versions (bottom) designed to minimize damaging risk in the manufacturing phase.

6.3.2 Task selection

The different categories of performance that can be optimized have been outlined in the previous chapter. In the case of the considered robotic manipulator, interesting objective functions are:

- **Task based:** the performance is evaluated based on the simulation of one or more tasks of interest. Such choice of the objectives allows for maximum performance on the considered tasks but does not guarantee a manipulator that behaves properly in unforeseen scenarios, resulting in lower application flexibility. Examples of these are:
  - **Accuracy:** calculated as the deviation from an idea trajectory or configuration during a time window of interest;
  - **Energy:** the total energy required to actuate the manipulator during the task;
  - **Durability:** keeps into account the accumulated damage during one task repetition and in turn the number of repetitions before failure;
- **Global:** functions that depend on the design of the manipulator and not on a particular task. These allow for more balanced designs but do not keep into account the effect on practical tasks. Examples can be:
  - **Weight:** the weight of the entire mechanism or sub-structures can be minimized since it is proportional to the actuation energy;
  - **Cost:** some parameters can have an impact on the production cost of components, so this should be kept into account;
  - **Compliance:** the average compliance of the robot within its workspace or a subset of it can be considere;
  - **Eigenvalues:** having higher eigenvalues means that the corresponding modes are less likely to be excited during operation.

For the presented application, a task-based optimization has been chosen. The considered task is the maneuver #1 presented in Section 4.5, for which the manipulator is actuated to follow a given path in 9 seconds. The goals are to improve the accuracy, measured as the deviation of

the end-effector actual pose with respect to the kinematic one, while also guaranteeing energy efficiency. The set of objective functions is defined as follows:

$$\begin{aligned} f_1^{obj}(\mathbf{s}(t,\mathbf{p})) = &\quad \frac{1}{(t_f - t_0)} \int_{t_0}^{t_f} d(t,\mathbf{p}) dt \\ f_2^{obj}(\mathbf{s}(t,\mathbf{p})) = &\quad \sum_{i=1}^{5} \int_{t_0}^{t_f} \omega_i(t) T_i(t,\mathbf{p}) dt \end{aligned} \tag{6.1}$$

where d is the deviation as defined in Equation 4.10, ω_i is the angular velocity of the i-th driver and T_i its applied torque. It should be noted that ω does not depend on parameters, being assigned once the trajectory is chosen. The first OF f_1^{obj} represents the average deviation from the reference trajectory of the EE over the entire movement. Minimizing such quantity allows to maintain a good accuracy over the entire trajectory. f_2^{obj} is instead the total mechanical energy spend by the 5 motors to perform the assigned maneuver. In an ideal scenario, the electrical energy absorbed by the motors would be used as the objective function. This would though require modeling of electrical behavior of the motors and in turn bring to a much more complicated multi-physics model that is not in the scope of this work. The total mechanical energy can be considered an approximation of the true energy required by the manipulator, and it is expected to have the same kind of dependency on the chosen design parameters as the electrical energy. This allows to assume that the optimal solutions for the chosen OF would be close to optimal also considering an electro-mechanical model.

The two goals are typically contrasting, making it a perfect and interesting application for multi-objective optimization. A reduction in deformation would in fact require stiffer components, typically achieved by increasing their size and in turn weight. This last modification brings to higher torques and energy required to follow the same trajectory. The use of composites allows for a reduced weight of the components, with a benefit in terms of required energy, and also enables an extra design parameter (i.e. fiber angle) that has an impact on the stiffness without modifying the weight.

Given the higher cost of composite components and the more difficult manufacturing process, it would be ideal to replace as few parts as possible with composite ones. In a first trial, only the second link is replaced with a composite part, since it is the one that contributes the most to the deformation of the system and also the heaviest one. In a second trial, also the third link is replaced. This increases the manufacturing costs and complexity, but allows for a much wider impact on performance. The set of design parameters with their ranges of variation is summarized in Table 6.3.

Table 6.3: Composite parameters

Bodies	**Parameter**	**Values**
L2r,L2l	Θ_2	0° - 90°
L2r,L2l	n_{l2}	6, 8, 10, 12
L3r,L3l	Θ_3	0° - 90°
L3r,L3l	n_{l3}	6, 8, 10, 12

Different kinds of constraints can be applied to the considered case:

- **maximum deviation:** during specific parts of the trajectory, a guaranteed accuracy level might be required;
- **maximum torque:** physical motors have specifics that limit the maximum torque that can be generated, thus the need to also impose such limits to the models;

- **maximum strain:** if the maximum allowed strain of the components is known, an upper limit to the strains generated can be imposed in order to ensure that they do not undergo any damage.

Integrating any of the constraints mentioned above in the proposed workflow would be straightforward. For sake of simplicity and without loss of generality, it was chosen to not use any constraint for the results shown in subsequent sections.

6.3.3 HEEDS optimization software and SHERPA algorithm

The commercial software Simcenter HEEDS is employed for the optimization [5]. The software allows to integrate and connect external software of different kinds to generate the process. The simulation part of the workflow is based on an in-house pFMB framework implemented in Matlab. HEEDS can directly input the design parameters chosen by the search algorithm, run the simulation and extract results.

The proprietary HEEDS algorithm MO-SHERPA is chosen in this work. SHERPA is a hybrid and adaptive search strategy that simultaneously applies elements from different common search strategies adapting to the considered problem [140]. MO-SHERPA is a modified version of SHERPA adapted for MOO and Pareto search. Being a zeroth-order algorithm, SHERPA does not require the calculation of gradients of the responses, resulting in easier implementation of the whole workflow. Furthermore, the only setting to be selected from the user is the number of designs to be explored. This results in easier setup without need to further tuning of the hyper-parameters of the algorithm as is the case for other common algorithms. On the other side, this can result in less flexibility given that the user has almost no space of decision and no insight in how the algorithm behaves. The common issue of any heuristic search, namely the high number of designs to be explored, is present also in this case.

The choice of the MO-SHERPA algorithm in this work has been dictated by the ease of use. Tuning of hyper-parameters or calculation of gradients are interesting applications to be considered in the future, promising to greatly improve the performance of the optimization workflow, but would have been out of the scope of this work.

6.3.4 Optimization workflow

The whole optimization workflow consists in an off-line step in which the models are generated and in the core search iterations.

In the off-line phase, the ROMs and pROMs required are generated to create a library of models. A set of different pROMs for the composite components, each with different numbers of layers and with Θ_2 and Θ_3 as parameters, are created and stored. These compose a library of models use then in the on-line phase.

In the on-line phase, for each design explored by the algorithm a first step consists in assembling the pFMB model with the required pROMs (based on material and number of layers). The model is then updated with the parameters of interest. This procedure requires a minimal computational effort, given the efficiency of the pFMB models in updating the parameters. The updated model is finally used for the simulation whose results are used to evaluate the objective functions. At this point, if any constraint is defined, also the constraint functions are evaluated to check if the proposed design is feasible. If feasible, the results are used by the algorithm to select the successive step in the search.

6.4 Results

This section presents the results of the two optimization procedures. The baseline performance indexes, relative to the experimentally validated model with links in isotropic material and to the task presented as *trajectory #1* in Subsection 4.5.2, are summarized in Table 6.4.

Table 6.4: Baseline results (isotropic design)

Average deviation	6.0926 mm
Energy	38.5796 J

The running times discussed in this section are relative to HEEDS and Matlab run on a machine with an Intel Core i7-8850H 2.6GHZ processor and 64GB of RAM.

6.4.1 Optimization case n. 1: One composite link

For the first optimization case, both sides of L2 are replaced with the composite models. This results in two design parameters (Θ_2, n_{l2}). The optimization software runs 3 simulations in parallel for computational efficiency, and a total of 105 designs is explored. This quantity allows to properly reconstruct the Pareto front with an acceptable density, while keeping the total time required by the process (8h) low. An overview of information about the run is given in Table 6.5.

Table 6.5: Optimization run info - case n. 1

Parameters	Θ_2, n_{l2}
Objective functions	average deviation, energy
Number of designs	105
Computation time	8h

The whole set of explored designs is shown in Figure 6.4. It can be seen how the optimal designs allow for a limited gain in terms of actuation energy, between 1% and 2%. The designs with 10 and 12 layers also allow for a gain in terms of accuracy (as shown in Figure 6.5), while lower numbers of layers cause the system to behave similarly (8) or worse (6) than the reference.

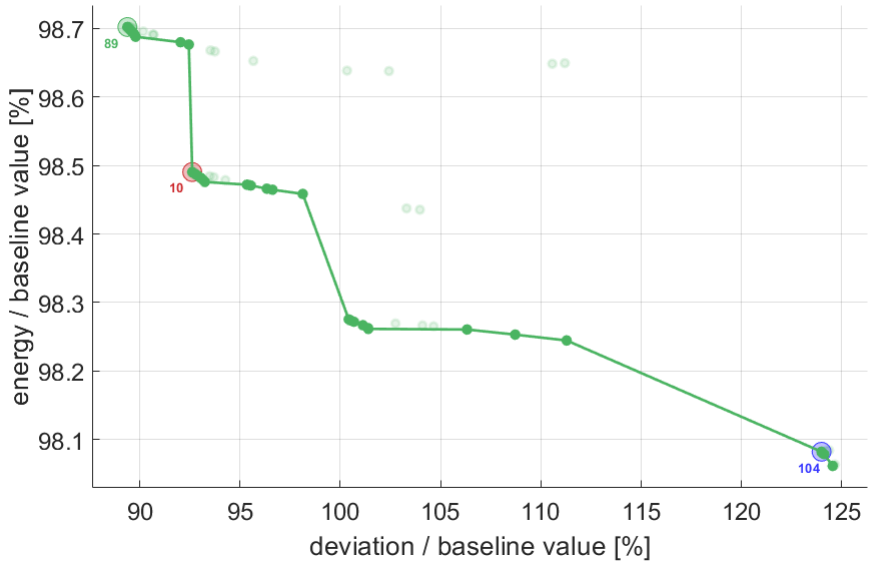


Figure 6.4: Pareto diagram for optimization case 1. The dominated designs are dimmed. 3 representative designs are highlighted whose results are shown in time domain.

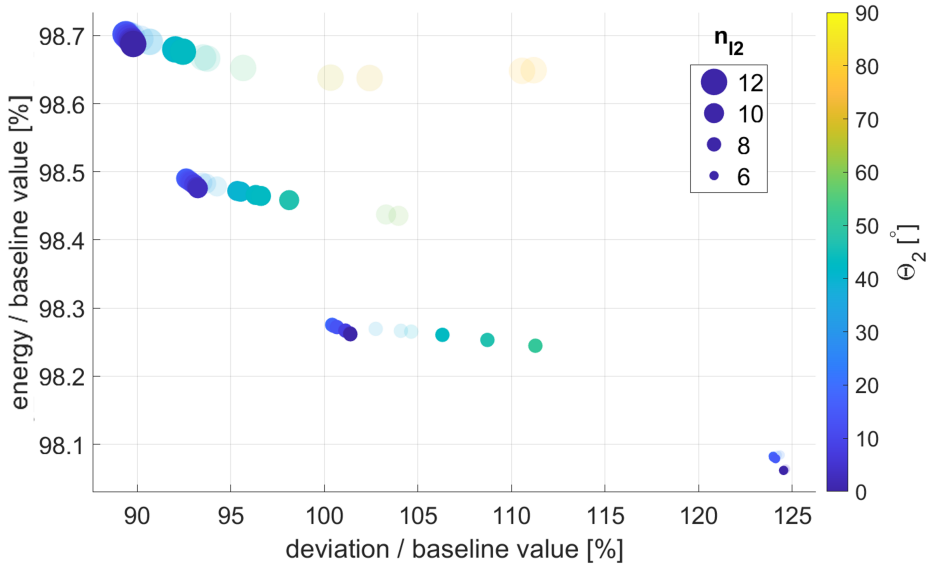


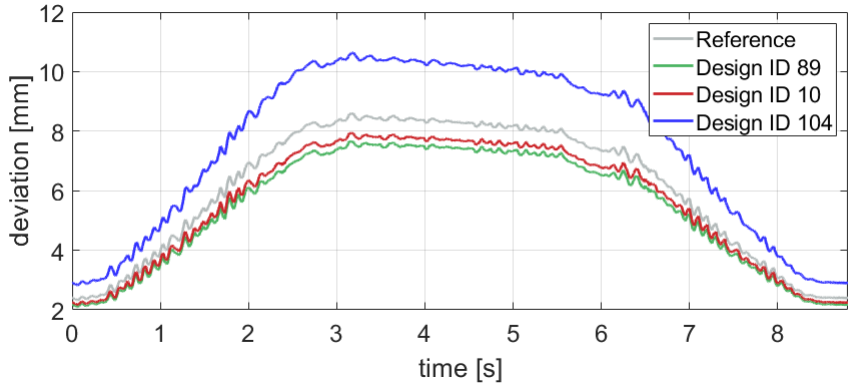
Figure 6.5: Pareto diagram for optimization case 1, with indications of the influence of parameters relative to link 2.

Three representative designs on the Pareto front are selected (highlighted in Figure 6.4) by taking one on each extreme, where one objective function is at its minimum and the other at its maximum, and one in the intermediate part where a trade-off result is obtained. The evolution in time of deviation and instantaneous actuation power (whose integration over time allows to evaluate the overall actuation energy) during the maneuver are shown for the selected designs

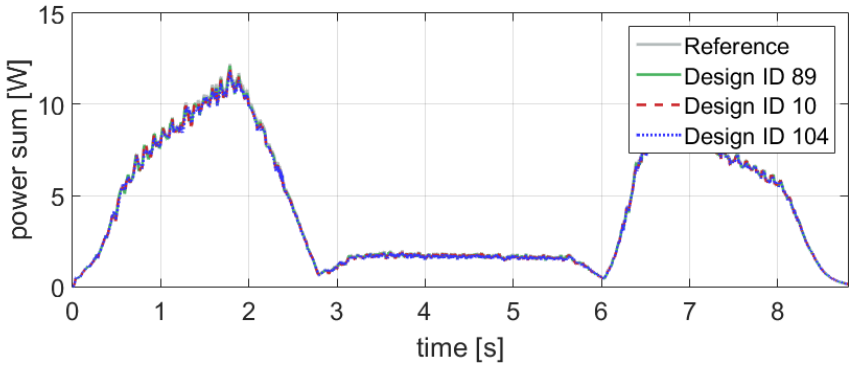
in Figure 6.6. The summary in Table 6.6) shows how the optimum for the fibers angle is always around 20°, and an increase in the number of layers causes a consequent increase in actuation energy and decrease in deviation.

Table 6.6: Representative designs - case n. 1

Design ID	Θ_2	n_{l2}	**avg dev**	**energy**
89	18.2	12	89.39%	98.70%
10	19.6	10	92.62%	98.49%
104	19.6	6	123.99%	98.08%



(a)



(b)

Figure 6.6: Deviation (a) and overall actuation power (b) for three representative designs from optimization case 1. Reference results are relative to the isotropic baseline model.

The effect of design parameters on the objective functions can be quantifiable by calculating the Paerson correlation coefficient ϱ, that is a measure of linear correlation between two sets of data. A value close to 1 or -1 indicates respectively a high positive or negative linear correlation, while a value close to 0 indicates no correlation. The coefficient for two variables $\{x, y\}$, each

with mean values $\{\mu_x, \mu_y\}$ and standard deviations $\{\sigma_x, \sigma_y\}$, is defined as:

$$\varrho(x,y) = \sum_{i=1}^{N} \left(\frac{cov(x,y)}{\sigma_x \sigma_y} \right) = \sum_{i=1}^{N} \left(\frac{(x_i - \mu_i)(x_j - \mu_j)}{\sigma_x \sigma_y} \right) \tag{6.2}$$

For the application case considered, the correlation indexes (Figure 6.7) show how the actuation energy is mainly influenced by the number of layers, while the deviation is influenced by both parameters on a similar scale but with opposite sign.

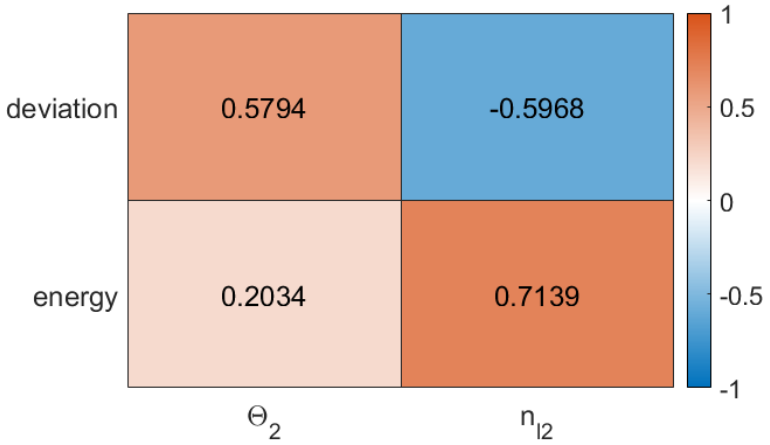


Figure 6.7: Correlations between objective functions and parameters for optimization case 1.

In conclusion, switching only L2 with a composite component has a limited beneficial effect on the performance of the manipulator. An interesting behavior is exhibited for which the optimal direction of the fibers is around 20°. This can be attributed to the need to resist forces on the component in a specific direction during the operation.

6.4.2 Optimization case n. 2: Two composite links

For the second case, both L2 and L3 are replaced with composite components. The number of design parameters is 4, while the other settings are kept the same as in the previous case. The whole optimization run in this case requires a larger time, given the increased size of the model. An overview of information about the run is given in Table 6.7.

Table 6.7: Optimization run info - case n. 2

Parameters	Θ_2, n_{l2}, Θ_3, n_{l3}
Objective functions	average deviation, energy
Number of designs	105
Running time	16h53m

The Pareto front for the explored designs (Figure 6.8) shows how all of optimal designs grant an energy gain between 7% and 10%, much more valuable than what seen in the previous case. While L3 is smaller and has a lower impact on the overall weight of the mechanism, its position further from the axes of the motors causes it to have a much larger influence on the torques needed for actuation. This result proves how the widely employed methodology of lightweighting

alone is not sufficient to optimally design such systems. Considering directly the influence of parameters allows for more optimized designs. Such application is enabled by the use of efficient dynamical models such as the pFMB ones.

For what regards the deviation, it is possible to reduce the error by up to 15% with respect to the baseline. Also in this case the results are satisfactory and better than in the previous case.

The effect of design parameters on the objective functions is interesting to analyze. The actuation energy is mostly influenced by the number of layers of L3, given that as mentioned this component is far from the axes of the motors. It is instead the second link, being the larger and more compliant, that has the wider effect on the deviation.

From Figure 6.9 it can be seen how the optimal values of the angle is around 20° for L2 and between 50° and 70° for L3. It should be noted how the two links require a different design in order to fully take advantage of their anisotropic behavior, given the different loads applied on them during the operation.

The time-domain responses of three reference designs (Figure 6.10) show an improvement in terms of energy with respect to the baseline over the entire maneuver and also in terms of deviation for two of them.

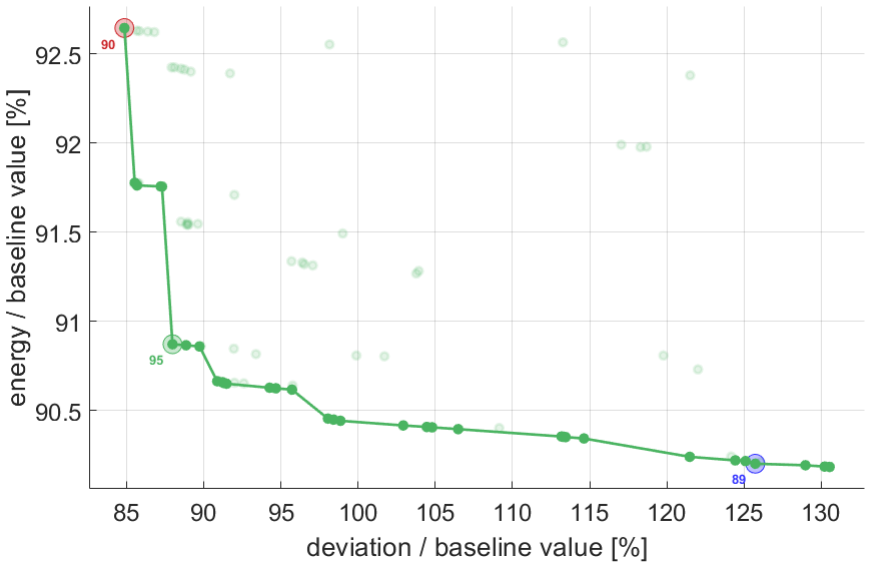
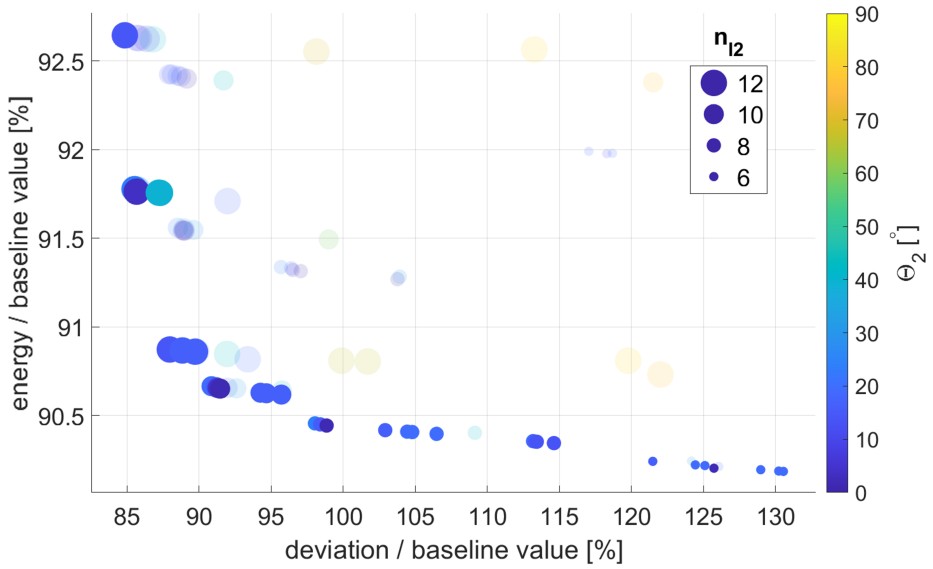
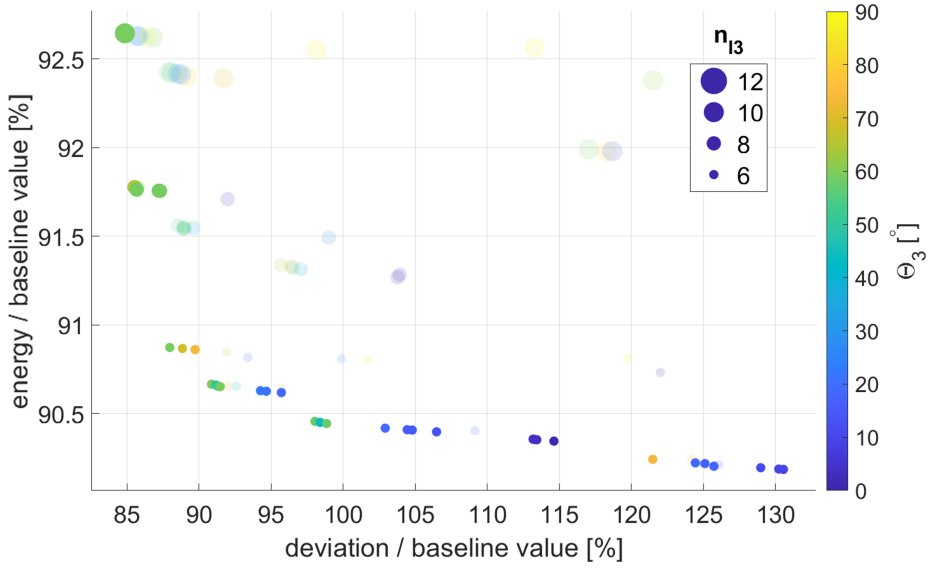


Figure 6.8: Pareto diagram for optimization case 2. The dominated designs are dimmed. 3 representative designs are highlighted whose results are shown in time domain.



(a)

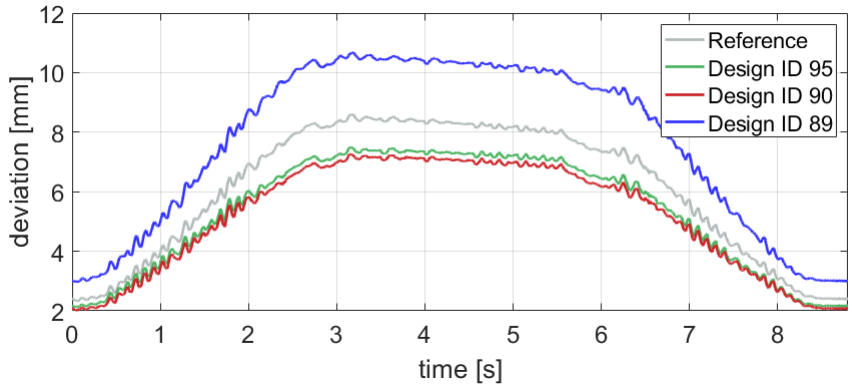


(b)

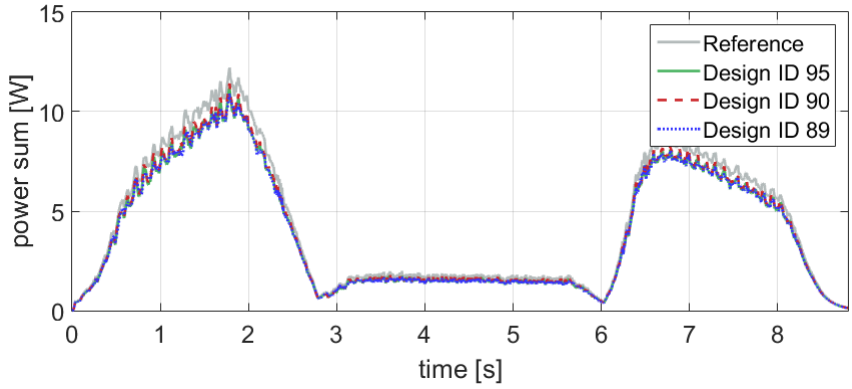
Figure 6.9: Pareto diagram for optimization case 2, with indications of the influence of parameters relative to link 2 (a) and link 3 (b).

Table 6.8: Representative designs - case n. 2

Design ID	Θ_2	n_{l2}	Θ_3	n_{l3}	**avg dev**	**energy**
90	14.4	12	59.4	10	84.86%	92.64%
95	14.2	12	59.4	6	87.98%	90.87%
89	1.8	6	17.4	6	125.72%	90.20%



(a)



(b)

Figure 6.10: Deviation (a) and overall actuation power (b) for three representative designs from optimization case 2. Reference results are relative to the isotropic baseline model.

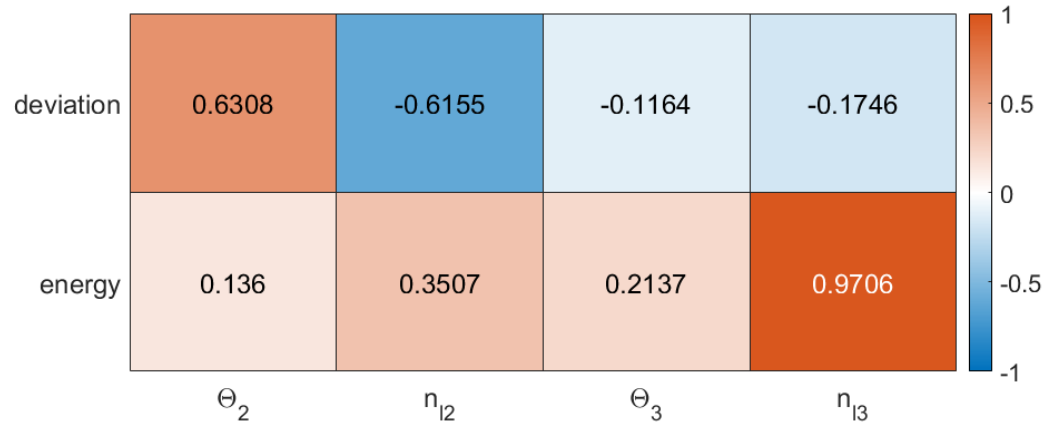


Figure 6.11: Correlations between objective functions and parameters for optimization case 2.

6.5 Conclusions and possible future work

This chapter presented the integration of pFMB models in optimization workflows to enable efficient task-based optimization of material parameters. The workflow for optimization using pFMB models has been presented with considerations on the main steps in which such models are employed. The main advantages, with focus on the main categories of search algorithms, have been presented. These consist in efficient design update while maintaining high accuracy of the models, with the added advantage of efficient and exact computation of derivatives if gradient-based search methods are employed.

The application case of the use of composites to improve the performance of a robotic manipulator was presented. By optimizing angle of the fibers and number of layers, gains in terms of accuracy and actuation energy of up to 15% were achieved. This application showcased both the potential of the modeling methodology to be applied for optimization and of composite materials to be used to improve the performance of robotic manipulators. Furthermore, by employing experimentally validated models as a baseline and for the composite material, the achieved results are representative of a realistic case. It will be of interest in future efforts to further validate the results by testing the optimal designs on a system-level. For the generic optimization workflow employing pFMB, it will be of interest for future applications to explore even more efficient search algorithms, such as gradient-based ones, that allow for quicker or wider explorations.

Considering the optimization application to the robotic manipulator, durability of the components should be kept into account in the choice of designs, together with constraints on actuation torques and maximum error based on real-world applications. It would also be of interest to investigate how the reduction in actuation torque required would enable the use of smaller motors, and to embed the different physical motor models in the optimization workflow in order to account for their inertial effect.

Part III

Virtual sensing

Chapter 7

Estimation methods for dynamic mechanical systems: Overview

7.1 Introduction

This work has so far presented methods to develop accurate and efficient numerical models of mechanical systems, and their use in the design phase. The availability of such models proves fundamental also in the rest of the product life cycle, when one or more physical instances of the system are available. During operations, the models can enable Virtual Sensing applications aimed at e.g. condition monitoring [83, 53]. These rely on the knowledge of the full state of the system to track the evolution of features of interest. Commonly, only limited measurements $\mathbf{y}(t)$ are available. Estimation techniques allow for the extraction of hidden information in the measurements. The result is an estimation of the evolution of states $\mathbf{x}(t)$ and often also of unknown inputs $\mathbf{u}(t)$ and parameters $\mathbf{p}(t)$ of the system.

A large variety of estimation techniques has been proposed in literature. A first distinction can be made between data-driven and model-based approaches. The first class relies on the use of large amounts of data acquired on the systems of interest to reconstruct the relationship between measurements and unknown quantities. In the model-based approach, such relationship is instead modeled through equations keeping into account the knowledge available on the system and the physics underlying its behavior. This second approach allows for a much simpler problem and a more complete set of results, at the cost of having to construct and update the model. For the applications discussed in this work, accurate and efficient numerical models are available from the design phase of the system and can easily be employed also in model-based estimation during operations.

Model-based estimation methods have been traditionally developed for estimation of states in the case of known inputs and parameters [130]. The most straightforward of such methods is the least-square estimation, that consists in identifying the set of states that better matches the measurements in a least-square sense. The accuracy level of measurements and model can be kept into account, resulting in a weighted least-square approach. Furthermore, the solution can be computed recursively only considering a certain window of time for the observations used at each step. The weighted iterative least-square problem yields the Kalman Filter (KF) method [84, 129]. KF algorithms are among the most commonly used model-based estimation methods in engineering. While more advanced methods have been developed based e.g. on Bayesian inference and optimization methods, the appeal of the KF relies in the trade-off capability of

incorporating knowledge on physical and numerical models with their respective confidence levels and the computational efficiency and ease of implementation. For these reasons, the KF has been employed in this work.

In this chapter, an overview on the theoretical background of KF methods is given. In Section 7.2 the classic discrete KF algorithm for linear systems is described, while the extensions to non-linear systems is discussed in Section 7.3. Section 7.4 introduces methods for concurrent estimation of states and unknown quantities. Observability criteria are briefly disussed in Section 7.5, Finally, in Section 7.6 the main choices to be made in the implementation of a KF and limitations of the methodology are discussed.

7.2 Linear Kalman filter

This section presents an overview of the requirements and assumptions for models used in a KF framework, and then analyzes the main steps of the linear KF algorithm.

7.2.1 Model requirements and assumptions

The classical KF [84] is formulated for linear systems whose dynamic behavior is modeled by first order ODEs expressed in a SS representation. For a linear system, the SS representation has been defined in Equation 2.38 and is reported here for clarity:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}_c\mathbf{x}(t) + \mathbf{B}_c\mathbf{u}(t) \\ \mathbf{y} = \mathbf{H}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \end{cases} \tag{7.1}$$

Where not necessary, in the rest of the work the dependency on time will be omitted in notation for clarity purposes.

In practice, such algorithms are numerically implemented on digital machines that require the problem discretization. The SS representation needs to be converted into its discrete-time form:

$$\begin{cases} \mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_k \\ \mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k \end{cases} \tag{7.2}$$

The discrete version of the state-update and input matrices depend on the time-integration scheme employed for the discretization. On the other hand, measurement equations do not change form when discretized, given that no time integration is involved.

7.2.2 Linear discrete Kalman filter

The core assumption made in KF is that the uncertainty on the model and the measurements can be respectively represented by additive zero-mean Gaussian noise $\mathbf{w}_{x,k} \in \mathbb{R}^{2n_{red}}$ and $\mathbf{w}_{y,k} \in \mathbb{R}^{n_m}$. The SS model in Equation 7.2 becomes then:

$$\begin{cases} \mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_k + \mathbf{w}_{x,k} \\ \mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{w}_{y,k} \end{cases} \tag{7.3}$$

Noise levels are characterized by the states covariance matrix $\mathbf{Q}_k = E[\mathbf{w}_{x,k}\mathbf{w}_{x,k}^T] \in \mathbb{R}^{2n_{red}\times 2n_{red}}$ and measurements covariance $\mathbf{R}_k = E[\mathbf{w}_{y,k}\mathbf{w}_{y,k}^T] \in \mathbb{R}^{n_m \times n_m}$, where $E[]$ is the expectation operator. These are commonly assumed as constant and diagonal, for lack of better statistical information.

The analytical optimization problem that is the goal of the KF is solved through a series of steps that can be split into a *prediction* and a *correction* phase [130]. In the prediction phase,

estimates from the previous steps are used to advance the model in time by only employing the numerical model, without knowledge on current measurements. For this reason, quantities estimated in this step are defined *a-priori* estimates (denoted as $\square^-$). In the correction step, knowledge on the measurements is merged with numerical predictions by a weighting based on their uncertainty levels. Quantities estimated in this step are thus defined *a-posteriori* estimates (denoted as $\square^+$).

The goal of the discrete KF is finding the optimal a-posteriori estimate $\mathbf{x}_k^+$ for the true state $\mathbf{x}_k$. Optimality in this scope is defined by the minimization of the expected two-norm of the estimation error:

$$E_k = E[(\mathbf{x}_k^+ - \mathbf{x}_k)^T(\mathbf{x}_k^+ - \mathbf{x}_k)] \tag{7.4}$$

The certainty level on the estimation error is defined through its covariance matrix $\mathbf{P}_k \in \mathbb{R}^{2n_{red} \times 2n_{red}}$. The quantity in Equation 7.4 is equivalent to the trace of $\mathbf{P}_k$, that becomes then the minimization objective for the algorithm.

At each step k, it is assumed that the a-posteriori estimated state $\mathbf{x}_{k-1}^+$ and error covariance $\mathbf{P}_{k-1}^+$ from previous step, the current measurements $\mathbf{y}_k$ and inputs $\mathbf{u}_k$ are available. Starting from these, the algorithm steps are:

- **Prediction**: the a-priori states $\mathbf{q}_k^-$ and measurements $\mathbf{y}_k^-$ are predicted based on the numerical model:

$$\mathbf{x}_k^- = \mathbf{A}\mathbf{x}_{k-1}^+ + \mathbf{B}\mathbf{u}_k \tag{7.5}$$

$$\mathbf{y}_k^- = \mathbf{H}\mathbf{x}_k^- + \mathbf{D}\mathbf{u}_k \tag{7.6}$$

  The states error covariance is also propagated from previous to current time step as:

$$\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}^+\mathbf{A}^T + \mathbf{Q}_k \tag{7.7}$$

- **Correction**: the so-called Kalman gain $\boldsymbol{\mathcal{K}}_k \in \mathbb{R}^{2n_{red} \times n_m}$ is computed as:

$$\boldsymbol{\mathcal{K}}_k = \mathbf{P}_k^-\mathbf{H}^T(\mathbf{H}\mathbf{P}_k^-\mathbf{H}^T + \mathbf{R}_k)^{-1} \tag{7.8}$$

  This term can be intuitively interpreted as the ratio between uncertainties on numerical model and measurements. The a-priori estimates of states are updated by adding a term proportional to the difference between measurements and their predicted values, the so-called innovation term:

$$\mathbf{x}_k^+ = \mathbf{x}_k^- + \boldsymbol{\mathcal{K}}_k(\mathbf{y}_k - \mathbf{y}_k^-) \tag{7.9}$$

  Finally, the states error covariance is also corrected:

$$\mathbf{P}_k^+ = (\mathbf{I} - \boldsymbol{\mathcal{K}}_k\mathbf{H})\mathbf{P}_k^-(\mathbf{I} - \boldsymbol{\mathcal{K}}_k\mathbf{H})^T + \boldsymbol{\mathcal{K}}_k\mathbf{R}_k\boldsymbol{\mathcal{K}}_k^T \tag{7.10}$$

An initial guess is required for the values $\mathbf{x}_0$ and $\mathbf{P}_0$.

7.3 KF for non-linear models

In many applications, the numerical model is non-linear. While the KF guarantees optimality in the sense described in the previous section only for linear systems, it has been adapted in order to use non-linear systems. While optimality is not guaranteed, non-linear versions of the KF have been proven effective in several engineering applications and are commonly practically employed.

One of the most common approaches is the Extended Kalman Filter (EKF) [130]. The EKF has a similar structure with respect to the one of the linear KF, but uses a system linearization to propagate the state error covariance and calculate the Kalman gain. While the EKF loses the optimality guarantee, it has been shown to work properly if small enough time steps are used and the system is regular enough. This approach is then particularly suitable for mildly non-linear systems that allow for a smooth and continuous linearization. For more complicated and highly non-linear systems, particle filters such as the Sigma-Point Kalman Filter (SPKF) [147] allow for better accuracy at the cost of higher computational cost. These are based on applying the non-linear transformations to a set sample points in order to reconstruct the probability distribution without the need for a linearization. Given the properties of the models analyzed in this work, that will be discussed more in detail in the next chapters, the EKF is the approach employed. The rest of this section discusses the formulation of a non-linear SS model, its linearization and the EKF algorithm.

7.3.1 Non-linear State Space

A SS model in the generic non-linear time-continuous form is expressed as follows:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}_c(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t)) \end{cases} \tag{7.11}$$

where $\mathbf{f}_c$ and $\mathbf{h}_c$ are respectively referred to as the continuous time-update and measurement equations. The discrete version is:

$$\begin{cases} \mathbf{x}_k = \mathbf{f}_d(\mathbf{x}_{k-1}, \mathbf{u}_k) \\ \mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \end{cases} \tag{7.12}$$

The non-linear equations can directly be solved to predict the states evolution. The subsequent covariance propagation and correction steps cannot be directly executed on such a system and require adaptations. The EKF in particular consists in employing a linearized version of the system for these steps.

7.3.2 Extended Kalman filter

The EKF algorithm resembles the linear one presented in the previous section, with few modification in order to treat the non-linearity:

- **Prediction**: the a-priori states $\mathbf{q}_k^-$ and measurements $\mathbf{y}_k^-$ predictions are directly obtained from the evaluation of the non-linear model:

$$\mathbf{x}_k^- = \mathbf{f}_d(\mathbf{x}_{k-1}^+, \mathbf{u}_k) \tag{7.13}$$

$$\mathbf{y}_k^- = \mathbf{h}_d(\mathbf{x}_k^-, \mathbf{u}_k) \tag{7.14}$$

  A linearized version of the model is employed for the covariance propagation. This is represented by the Jacobian of the system calculated around the current a-posteriori estimate from the previous step:

$$\mathbf{J}_{k-1} = \left. \frac{\partial \mathbf{f}_d(\mathbf{x}_{k-1}, \mathbf{u}_k)}{\partial \mathbf{x}_{k-1}} \right|_{\mathbf{x}_{k-1}^+, \mathbf{x}_k^-} \tag{7.15}$$

  The states error covariance is then propagated as:

$$\mathbf{P}_k^- = \mathbf{J}_{k-1} \mathbf{P}_{k-1}^+ \mathbf{J}_{k-1}^T + \mathbf{Q}_k \tag{7.16}$$

- **Correction**: Also the measurement equation is linearized by evaluating its Jacobian around the current a-priori estimate:

$$\mathbf{J}_{m,k} = \left.\frac{\partial \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k)}{\partial \mathbf{x}_k}\right|_{\mathbf{x}_k^-}, \tag{7.17}$$

The Kalman gain is then computed as:

$$\boldsymbol{\mathcal{K}}_k = \mathbf{P}_k^- \mathbf{J}_{m,k}^T (\mathbf{J}_{m,k} \mathbf{P}_k^- \mathbf{J}_{m,k}^T + \mathbf{R}_k)^- 1 \tag{7.18}$$

The states correction does not vary and is reported for completeness:

$$\mathbf{x}_k^+ = \mathbf{x}_k^- + \boldsymbol{\mathcal{K}}_k (\mathbf{y}_k - \mathbf{y}_k^-) \tag{7.19}$$

Finally, the states error covariance is also corrected using linearized matrices:

$$\mathbf{P}_k^+ = (\mathbf{I} - \boldsymbol{\mathcal{K}}_k \mathbf{J}_{m,k}) \mathbf{P}_k^- (\mathbf{I} - \boldsymbol{\mathcal{K}}_k \mathbf{J}_{m,k})^T + \boldsymbol{\mathcal{K}}_k \mathbf{R}_k \boldsymbol{\mathcal{K}}_k^T \tag{7.20}$$

The presented EKF algorithm enables estimation on non-linear systems. While not guaranteeing optimality, the filter works properly when a small enough time step Δt is employed, so that the system can be well approximated by the linearization inside the time interval of each step. The other downside of the methodology is the required linearization at each iteration, that can result in a bottleneck for large models if no analytical expression for the Jacobians is available.

7.4 Concurrent estimation of states and unknown quantities

In many practical applications, some of the system's parameters or external inputs can exhibit a high level of uncertainty. Furthermore, such quantities can be difficult to measure. Concurrent estimation of states, inputs and parameters results then helpful in order to achieve good estimation results.

For the generic KF approach, the most common approaches for concurrent estimation are the Dual Kalman Filter (DKF) [89, 79] and the Augmented Kalman Filter (AKF) [98].

DKF consists of two separate filters estimating the states and the unknown quantities in parallel. AKF consists instead of extending, i.e. augmenting, the system state vector with the unknown quantities. This approach allows to keep into account all coupling effects between the estimated quantities, under the condition that the augmented system is observable. Furthermore, it generally results in a gain in computational time [59]. Joint estimation through an augmented EKF is employed in the following chapters. This section gives a brief generic overview of the methodology.

7.4.1 State augmentation

The augmented state is defined by extending the regular system states with the generic unknown quantities $\boldsymbol{\chi} \in \mathbb{R}^{n_{aug}}$ as:

$$\mathbf{x}^* = \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\chi} \end{bmatrix} \tag{7.21}$$

where $\square^*$ denotes a quantity relative to the augmented model. While the evolution in time of standard states is described by the EOMs, a dynamic model for the unknown quantities needs

to be assumed. The most generic of such models employs a zeroth order random walk model [98, 45, 107] where each augmented state is considered as constant with the addition of white Gaussian noise:

$$\dot{\boldsymbol{\chi}}(t) = \mathbf{0} + \mathbf{w}_{\chi}(t) \tag{7.22}$$

In discrete-time, the random walk is represented as:

$$\boldsymbol{\chi}_k = \boldsymbol{\chi}_{k-1} + \mathbf{w}_{\chi,k} \tag{7.23}$$

The random walk modeling approach allows to account for a generic time variation without need for further assumptions or knowledge of the expected behavior.

The uncertainties on unknown quantities is represented by the covariance matrix of the added noise $\mathbf{Q}_{\chi}$, so that the states covariance matrix of the augmented system results in:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{\chi} \end{bmatrix} \tag{7.24}$$

The discrete linear augmented SS results in:

$$\begin{cases} \mathbf{x}_k^* = \mathbf{A}^* \mathbf{x}_{k-1}^* + \mathbf{B}^* \mathbf{u}_k \\ \mathbf{y}_k = \mathbf{H}^* \mathbf{x}_k^* + \mathbf{D}\mathbf{u}_k \end{cases} \tag{7.25}$$

Given the dynamic model assumed for the unknown quantities (Equation 7.23), the augmented state and input matrices result in:

$$\mathbf{A}^* = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \tag{7.26}$$

$$\mathbf{B}^* = \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \tag{7.27}$$

The steps of the KF algorithm previously presented remain the same also in the case of an augmented system.

7.5 Observability

In order for the EKF to be able to correctly estimate the unknown quantities of the system, a proper choice of the measurement set is needed [107, 108, 44]. The two most common scenarios are:

- **Optimal Sensor Placement (OSP)**: the physical system is not sensorized or still being designed, in this case the optimal position and type of sensors has to be evaluated [44, 138, 119];
- **Optimal Sensors Selection (OSS)**: the physical system is already existing and sensorized, and the best subset of sensors to be used for estimation needs to be chosen. Structures are in fact typically over-instrumented to guarantee robustness, but employing a larger-than-required set of measurements in the KF is not always beneficial in terms of accuracy or efficiency [7].

A common approach for sensors choice in both scenarios is the evaluation of some observability metrics [117, 7]. A system is said to be observable if, for some input $\mathbf{u}(t)$, the initial state $\mathbf{x}(t=0)$ can be determined based on the measurements $\mathbf{y}(t)$ in a time interval $t \in [0, T]$ [15, 91]. On

a more intuitive level, observability implies there exists only one combination of initial states and inputs that generate any specific set of measurements. If this is not the case, i.e. the system is unobservable, the KF will not be able to univocally estimate the states (and inputs and parameters in the case of an AKF) that caused the given measurements.

Different observability metrics have been proposed in literature for classic systems theory, such as for example the Kalman observability matrix [130], the Popov-Belevitch-Hautus (PBH) matrix [74, 63], the observability Gramian [91]. All these approaches consist in checking if a matrix generated based on the SS model is full rank. This can be done analytically by considering e.g. how different kinds and positions of sensors impact the rank of such matrix. For practical numerical implementation, the rank check is replaced by the evaluation of the matrix condition number. This second approach allows to implement automated OSP or OSS methodology. The evolution in time of observability during a specific application can also be monitored in order to evaluate specific configurations in which the system could become unobservable. The downside of this approach is that the condition number is sensitive to scaling of the measurements and difficult to interpret.

One alternative approach that allows to also keep into account noise levels on model and measurements is the continuous-time steady-state error covariance matrix [44] or its discrete equivalent [7]. This approach results in a more robust evaluation of the observability level in computational applications, at the cost of a more complicated problem to be solved. In this work, observability is analytically evaluated based on observation of the matrices and their form for different combinations of unknown quantities and measurements. The PBH test is employed for this, and it will be introduced in the next section.

7.5.1 The Popov-Belevitch-Hautus test

The PBH observability test [74, 63] states that a sufficient and necessary condition for the observability of a linear system in the form Equation 7.1 is that the matrix

$$\mathbf{PBH} = \begin{bmatrix} \mathbf{I}s - \mathbf{A}_c \\ \mathbf{H} \end{bmatrix} \tag{7.28}$$

is of full rank for every value of $s \in \mathbb{C}$. The matrix $\mathbf{I}s - \mathbf{A}_c$ is guaranteed to be full rank for all values of s but the eigenvalues of $\mathbf{A}_c$, which are the values on which the analysis focuses. This criterion allows then to verify that each mode of the system is observable. When state augmentation is employed with a random walk model, $\mathbf{A}^*$ has a number of zero-valued rows at least equal to the number of augmented states, thus it has the same number of zero eigenvalues. It is for this value of s that the observability is most critical; non-zero eigenvalues, while being unobservable, remain stable and thus the system is at least detectable [108].

7.6 Discussion on setup and limitations

Practical implementations of the various kinds of KF described in this chapter present challenges and difficulties that are briefly described in this section.

One of the central and most critical parts of the KF is the modeling of measurement noise. The noise contributions are assumed to be additive, zero-mean and white. Such assumptions hold in most cases for measurement noise, where the matrix $\mathbf{R}$ can be taken directly from the used physical sensor data either through constructor data or zero-input acquisitions. The assumption made on model noise brings to the necessity of having accurate models in order to allow reliable estimation results. If a large mismatch between model and physical system is present, the model noise is far from white and zero-mean. Violation of the assumption results in wrong estimates.

The matrix $\mathbf{Q}$ is commonly assumed as constant, and set by trial and error procedure or by experience. In literature some rules of thumb for the choice have been proposed (e.g. [45]) or schemes to adapt the value in time (e.g. [149, 155, 131, 35]). High values of $\mathbf{Q}$ indicate a low confidence on the model, resulting in the filter giving more importance to the data derived from measurements. On the other hand, a lower value of $\mathbf{Q}$ will make the filter favor the predictions coming from the numerical model, considered more accurate. Of particular importance is the setting of the covariance terms relative to augmented states. Given the random walk model, covariance is the only parameter that dictates the evolution in time of the unknown quantities. Setting the zeroth order random walk noise covariance to a low value implies that the associated augmented state is expected to remain constant or vary slowly. A large value permits a large variation of the associated augmented state during each time step. These covariance values depend on the assumed nature of the augmented states, i.e. having a specific expected quasi-static or dynamic behavior.

The above-cited zeroth-order random walk is a powerful modeling tool since it allows to estimate quantities whose dynamic behavior is not known a-priori. On the other hand, if some information is available on such dynamic behavior there is limited possibility to incorporate it in the model, given that the only tunable parameter is the relative covariance. Alternatives have been proposed to incorporate such information [90].

Chapter 8

Estimation of states, inputs and parameters for structural dynamics applications

8.1 Introduction

Parameter identification [105, 60] and tracking [83, 53] have been among the main research interests in structural dynamics over the last years. The system's states are also commonly tracked to have a complete knowledge about the system evolution in time. In most cases, parameters and states are difficult to directly measure. The main limiting factors are, among others, the high cost of the required sensors and mounting inaccessibility, potential interference with operations or, at times, appropriate sensors might not even exist. A common approach to overcome these issues is employing the indirect estimation methods described in Chapter 7. These are typically based on the measurement of a set of responses of the system to known inputs. Often however, measuring these inputs is also not straightforward. Therefore, ideally an output-only procedure should be employed, where the estimation of parameters and states is coupled with the concurrent estimation of inputs.

The previous chapter presented the well known estimation approach based on KF. By employing KF for concurrent estimation (see Section 7.4), several authors jointly estimated states and inputs [45, 108, 119], states and parameters [40, 17, 131, 79] or the three together [107, 48, 94, 80].

When using a KF for parameter estimation, the parameter values are updated at each time step, requiring the model to be updated accordingly. Since in general ROMs do not retain any explicit parametric dependency, it is necessary to recreate the ROM at each step, which could result in a loss of most of the computational efficiency granted by the use of a ROM in the first place. This explicit dependency can however be retained by pMOR techniques such as the one presented in Chapter 3. Employing a pROM inside a KF-based parameter estimation allows to efficiently update the model for each new estimated parameter value, resulting in an overall computationally efficient procedure. This approach has been proposed by Naets et al. [107] for the joint estimation of states, parameters and input in linear structural dynamics applications. Here, the employed pMOR scheme applies a *local* approach [20] based on interpolation of ROMs which have been calculated for different sampled parameter values. By not assuming any particular form for the parametric dependency of the full model, this approach allows to have a general representation for a large variety of parameters. This causes a loss of information about

the kind of parameters considered, and the form of parametric dependency is not fully exploited. The already discussed downsides of the local approach, which result in a large computational overhead, can severely impact the performance of a KF scheme in which the model requires an update at each time step.

This chapter proposes an augmented EKF framework which uses the pMOR approach presented in Chapter 3. The proposed framework enables the efficient joint estimation of states, inputs and parameters. Furthermore, the proposed pMOR approach preserves the structure of the FOM parametric dependency, allowing to distinguish between different types of parameters (e.g. stiffness-related and mass-related) and consequently to analyze their observability individually.

The chapter is structured as follows. In Section 8.2 the EOMs for pROMs are adapted in order to suit the form required in the KF algorithm. The proposed augmented EKF scheme is then described in Section 8.3. In Section 8.4 the influence of the type of measurements and estimated quantities on the system observability is discussed through an observability analysis. Section 8.5 discusses the setup of the filter in terms of parameters chosen by the user. The estimation procedure is finally validated, both numerically in Section 8.6 and experimentally in Section 8.7, considering the application case of the scaled wind turbine blade. Finally, conclusions are given in Section 8.8.

The content of this chapter is partially presented in a publication from the author [32].

8.2 Adaptation of the EOMs for pROMs

The KF algorithms presented in the previous chapter require the models to be first-order, discrete and linear (or linearized). This chapter describes the steps to fit the EOMs of pROMs into such requirements. The models are first expressed through an augmented SS representation and the equations are discretized in time. The linearization terms employed by the EKF are then derived.

8.2.1 Augmented state-space representation

The reduced EOMs in Equation 3.31 are transformed from a second order to a first order form via the state space representation [45] described in Subsection 2.3.4. The augmented state $\mathbf{x}^* \in \mathbb{R}^{2n_{red}+n_{un}+n_p}$ is defined by extending the regular system states with the unknown quantities as:

$$\mathbf{x}^* = \begin{bmatrix} \mathbf{q}_{f_r} \\ \dot{\mathbf{q}}_{f_r} \\ \mathbf{u}^{un} \\ \mathbf{p} \end{bmatrix} \tag{8.1}$$

Here the input is split into the unknown inputs $\mathbf{u}^{un} \in \mathbb{R}^{n_{un}}$ and the known inputs $\mathbf{u}^{kn} \in \mathbb{R}^{n_{kn}}$, with a corresponding split in $\mathbf{S}_{u,r}$, giving $\mathbf{S}_{kn,r} \in \mathbb{R}^{n_{red} \times n_{kn}}$ and $\mathbf{S}_{un,r} \in \mathbb{R}^{n_{red} \times n_{un}}$.

The time evolution of inputs and parameters is represented using the zeroth order random walk model [98, 45, 107]:

$$\dot{\mathbf{u}}^{un}(t) = \mathbf{0} + \mathbf{w}_u(t) \tag{8.2}$$

$$\dot{\mathbf{p}}(t) = \mathbf{0} + \mathbf{w}_p(t) \tag{8.3}$$

The augmented system can then be represented as:

$$\begin{cases} \dot{\mathbf{x}^*}(t) = \mathbf{A}^*(\mathbf{x}^*)\mathbf{x}^*(t) + \mathbf{B}^*(\mathbf{x}^*)\mathbf{u}^{kn}(t) \\ \mathbf{y}(t) = \mathbf{H}^*(\mathbf{x}^*)\mathbf{x}^*(t) + \mathbf{D}^*(\mathbf{x}^*)\mathbf{u}^{kn}(t) \end{cases} \tag{8.4}$$

in which the augmented state and input matrices are:

$$\mathbf{A}^*(\mathbf{x}^*) = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ -\mathbf{M}_{f_r}^{-1}\mathbf{K}_{f_r} & -\mathbf{M}_{f_r}^{-1}\mathbf{C}_{f_r} & \mathbf{M}_{f_r}^{-1}\mathbf{S}_{un,r} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{8.5}$$

$$\mathbf{B}^*(\mathbf{x}^*) = \begin{bmatrix} \mathbf{0} \\ \mathbf{M}_{f_r}^{-1}\mathbf{S}_{kn,r} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{8.6}$$

Note that here and in the following equations the dependency of $\mathbf{K}_{f_r}$, $\mathbf{C}_{f_r}$ and $\mathbf{M}_{f_r}$ on the parameters is not written explicitly, if not needed, for notation clarity. The augmented measurement and feedthrough matrices for the considered types of measurements are:

$$\mathbf{H}_s^* = \begin{bmatrix} \mathbf{N}_s & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \qquad \mathbf{D}_s^* = \begin{bmatrix} \mathbf{0} \end{bmatrix} \tag{8.7}$$

$$\mathbf{H}_d^* = \begin{bmatrix} \mathbf{N}_d & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \qquad \mathbf{D}_d^* = \begin{bmatrix} \mathbf{0} \end{bmatrix} \tag{8.8}$$

$$\mathbf{H}_a^* = \mathbf{N}_d\mathbf{M}_{f_r}^{-1}\begin{bmatrix} -\mathbf{K}_{f_r} & -\mathbf{C}_{f_r} & \mathbf{S}_{kn,r} & \mathbf{0} \end{bmatrix} \qquad \mathbf{D}_a^* = \mathbf{N}_d\mathbf{M}_{f_r}^{-1}\begin{bmatrix} \mathbf{S}_{un,r} \end{bmatrix} \tag{8.9}$$

The augmented model has a nonlinear dependency on the parameters, resulting in general in a nonlinear dependency on the augmented state vector. Because of this, a non-linear KF approach is required. Without loss of generality with respect to other non-linear KF approaches, the discrete version of the EKF is employed in this work. Given the affine dependency of the reduced system matrices on the parameters employed in the pMOR, the linearized system is expected to be smooth and continuous. The next sections describe the time discretization of the system and the linearization procedure.

8.2.2 Discretization and linearization

The discretized version of the EKF employed in this work requires the model in Equation 8.4 to be discretized in time. This work adopts the method described by Risaliti et al. [119] for the discretization and linearization of the state-space equations, in which the integration scheme of choice is the first order BDF.

Using Equation 2.61, the augmented state evolution in time for a time step Δt is discretized as:

$$\mathbf{x}_k^* = \mathbf{x}_{k-1}^* + \Delta t\dot{\mathbf{x}}_k^* \tag{8.10}$$

For the k-th time step, the first equation of Equation 8.4 yields:

$$\dot{\mathbf{x}^*}_k = \mathbf{A}^*(\mathbf{x}_k^*)\mathbf{x}_k^* + \mathbf{B}^*(\mathbf{x}_k)\mathbf{u}_k^{kn} \tag{8.11}$$

From Equation 8.10, the value of $\dot{\mathbf{x}}_k^*$ can be explicitated and inserted into the previous equation, resulting in:

$$\frac{\mathbf{x^*}_k - \mathbf{x^*}_{k-1}}{\Delta t} = \mathbf{A}^*(\mathbf{x}_k^*)\mathbf{x}_k^* + \mathbf{B}^*(\mathbf{x}_k)\mathbf{u}_k^{kn} \tag{8.12}$$

The implicit equation governing the system is then:

$$\mathbf{g}_d(\mathbf{x}_{k-1}^*, \mathbf{x}_k^*, \mathbf{u}_k^{kn}) = \mathbf{x}_{k-1}^* - \mathbf{x}_k^* + \Delta t\left(\mathbf{A}^*\left(\mathbf{x}_k^*\right)\mathbf{x}_k^* + \mathbf{B}^*\left(\mathbf{x}_k^*\right)\mathbf{u}_k^{kn}\right) = \mathbf{0} \tag{8.13}$$

The KF is formulated for explicit SS models, and the EKF linearization requires the extraction of the underlying explicit equation associated with Equation 8.13 as detailed in [119]. This

explicit function exists locally if the corresponding implicit function is continuously differentiable. This property is guaranteed by the adopted affine parameter dependency. The underlying explicit function can be found using a Taylor series expansion around the linearization point $[\mathbf{x}_{k-1}^{*,0}\ \mathbf{x}_{k}^{*,0}]$, truncated at the first order:

$$\mathbf{g}_d(\mathbf{x}_{k-1}^*, \mathbf{x}_k^*, \mathbf{u}_k^{kn}) \approx \mathbf{g}_d^0 + \left.\frac{\partial \mathbf{g}_d}{\partial \mathbf{x}_k^*}\right|_0 (\mathbf{x}_k^* - \mathbf{x}_k^{*,0}) + \left.\frac{\partial \mathbf{g}_d}{\partial \mathbf{x}_{k-1}^*}\right|_0 (\mathbf{x}_{k-1}^* - \mathbf{x}_{k-1}^{*,0}) \approx \mathbf{0} \tag{8.14}$$

where $\mathbf{g}_d^0 = \mathbf{g}_d(\mathbf{x}_{k-1}^{*,0}, \mathbf{x}_k^{*,0}, \mathbf{u}_k^{kn})$. It follows that $\mathbf{x}_k^*$ can be explicitly defined as:

$$\mathbf{f}_d(\mathbf{x}_{k-1}^*, \mathbf{u}_k^{kn}) = \mathbf{x}_k^* = \mathbf{x}_k^{*,0} - \left(\left.\frac{\partial \mathbf{g}_d}{\partial \mathbf{x}_k^*}\right|_0\right)^{-1} \left[\mathbf{g}_d^0 + \left.\frac{\partial \mathbf{g}_d}{\partial \mathbf{x}_{k-1}^*}\right|_0 (\mathbf{x}_{k-1}^* - \mathbf{x}_{k-1}^{*,0})\right] \tag{8.15}$$

from which the Jacobian matrix is computed as:

$$\mathbf{J} = \frac{\partial \mathbf{f}_d(\mathbf{x}_{k-1}^*, \mathbf{u}_k)}{\partial \mathbf{x}_{k-1}^*} = -\left(\left.\frac{\partial \mathbf{g}_d}{\partial \mathbf{x}_k^*}\right|_0\right)^{-1} \left(\left.\frac{\partial \mathbf{g}_d}{\partial \mathbf{x}_{k-1}^*}\right|_0\right) \tag{8.16}$$

The two derivatives required for the evaluation of the Jacobian matrix are:

$$\frac{\partial \mathbf{g}_d}{\partial \mathbf{x}_k^*} = -\mathbf{I} + \Delta t \left[\frac{\partial \mathbf{A}^*(\mathbf{x}_k^*)}{\partial \mathbf{x}_k^*}\mathbf{x}_k^* + \mathbf{A}^*(\mathbf{x}_k^*)\right] + \Delta t \left[\frac{\partial \mathbf{B}^*(\mathbf{x}_k^*)}{\partial \mathbf{x}_k^*}\mathbf{u}_k^{kn}\right] \tag{8.17}$$

$$\frac{\partial \mathbf{g}_d}{\partial \mathbf{x}_{k-1}^*} = \mathbf{I} \tag{8.18}$$

The derivatives of the system matrices $\mathbf{A}^*(\mathbf{x}^*)$ and $\mathbf{B}^*(\mathbf{x}^*)$ are straightforward to compute given the affine dependency employed for the pMOR. Given Equation 8.5 and Equation 8.6, $\mathbf{A}^*(\mathbf{x}^*)$ and $\mathbf{B}^*(\mathbf{x}^*)$ only depend on $\mathbf{p}$, hence only their derivatives with respect to $\mathbf{p}$ are non-zero. As $\mathbf{A}^*(\mathbf{x}^*)$ and $\mathbf{B}^*(\mathbf{x}^*)$ are the result of the multiplication of the $\mathbf{M}_{f_r}^{-1}(\mathbf{p})$, $\mathbf{C}_{f_r}(\mathbf{p})$ and $\mathbf{K}_{f_r}(\mathbf{p})$ matrices, the chain rule for derivation is applied to these terms to obtain the derivatives with respect to $\mathbf{p}$.

The derivative of the stiffness matrix $\mathbf{K}_{f_r}(\mathbf{p})$ depends on the type of parameter considered. If the Young's modulus and density are the parameters of interest, then for the stiffness matrix:

$$\frac{\partial \mathbf{K}_{f_r}(\mathbf{p})}{\partial E} = \mathbf{K}^E \tag{8.19}$$

The derivative of the inverse mass matrix $\mathbf{M}^{r^{-1}}(\mathbf{p})$ is needed and can be calculated by using the property of inverse matrices derivation:

$$\frac{\partial \mathbf{M}_{f_r}^{-1}(\mathbf{p})}{\partial \rho} = -\mathbf{M}_{f_r}^{-1}(\mathbf{p}) \frac{\partial \mathbf{M}_{f_r}(\mathbf{p})}{\partial \rho} \mathbf{M}_{f_r}^{-1}(\mathbf{p}) \qquad \frac{\partial \mathbf{M}_{f_r}(\mathbf{p})}{\partial \rho} = \mathbf{M}^{\rho} \tag{8.20}$$

Given the proportional damping model used, the derivative of $\mathbf{C}_{f_r}(\mathbf{p})$ is a linear combination of the two calculated above.

The analytical derivatives calculation is straightforward for any parameter of interest. This section only showed the ones needed for the validation applications of this and following chapters.

8.3 Augmented extended Kalman filter framework

Chapter 7 presented a generic introduction on EKF (Subsection 7.3.2) and augmented KF (Section 7.4). In this section, the two methodologies are jointly applied to the system obtained in the previous section. The proposed augmented EKF framework is first described, with an overview on the algorithm steps, and then the influence of covariance terms on the performance of the filter itself is discussed.

The discrete system equations of the system considering noise are:

$$\begin{cases} \mathbf{g}_d(\mathbf{x}^*_{k-1}, \mathbf{x}^*_k, \mathbf{u}^{kn}_k) = \mathbf{w}_{x^*,k} \\ \mathbf{y}_k = \mathbf{h}(\mathbf{x}^*_k, \mathbf{u}^{kn}_k) + \mathbf{w}_{y,k} \end{cases} \tag{8.21}$$

The discrete-time version of the augmented EKF, for the proposed implicit augmented SS model, are summarized in the following steps:

Prediction

$$\mathbf{g}_d(\mathbf{x}^{*+}_{k-1}, \mathbf{x}^{*-}_k, \mathbf{u}^{kn}_k) = 0 \tag{8.22}$$

$$\mathbf{J}_{k-1} = \left.\frac{\partial \mathbf{f}_d(\mathbf{x}^*_{k-1}, \mathbf{u}^{kn}_k)}{\partial \mathbf{x}^*_{k-1}}\right|_{\mathbf{x}^{*+}_{k-1}, \mathbf{x}^{*-}_k} \tag{8.23}$$

$$\mathbf{P}^-_k = \mathbf{J}_{k-1}\mathbf{P}^+_{k-1}\mathbf{J}^T_{k-1} + \mathbf{Q}_{k-1} \tag{8.24}$$

Correction

$$\mathbf{J}_{m,k} = \left.\frac{\partial \mathbf{h}(\mathbf{x}^*_k, \mathbf{u}^{kn}_k)}{\partial \mathbf{x}^*_k}\right|_{\mathbf{x}^{*-}_k} \tag{8.25}$$

$$\boldsymbol{\mathcal{K}}_k = \mathbf{P}^-_k \mathbf{J}^T_{m,k}(\mathbf{J}_{m.k}\mathbf{P}^-_k \mathbf{J}^T_{m,k} + \mathbf{R}_k)^{-1} \tag{8.26}$$

$$\mathbf{x}^{*+}_k = \mathbf{x}^{*-}_k + \boldsymbol{\mathcal{K}}_k(\mathbf{y}_k - \mathbf{h}(\mathbf{x}^{*-}_k, \mathbf{u}^{kn}_k)) \tag{8.27}$$

$$\mathbf{P}^+_k = (\mathbf{I} - \boldsymbol{\mathcal{K}}_k\mathbf{J}_{m,k})\mathbf{P}^-_k(\mathbf{I} - \boldsymbol{\mathcal{K}}_k\mathbf{J}_{m,k})^T + \boldsymbol{\mathcal{K}}_k\mathbf{R}_k\boldsymbol{\mathcal{K}}^T_k \tag{8.28}$$

Of particular importance, as already discussed in the previous chapter, is the choice of the matrices $\mathbf{Q}$ and $\mathbf{R}$. In particular, in this work the $\mathbf{R}$ matrix is chosen to be a constant diagonal matrix containing the measurement noise covariance taken directly from the used physical sensor data. The plant noise covariance matrix $\mathbf{Q}$ is also assumed to be constant and diagonal, as the noise is considered to be uncorrelated, and can be split to separate the contributions to the different augmented states as:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_q & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{\dot{q}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_u & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{Q}_p \end{bmatrix} \tag{8.29}$$

Here, $\mathbf{Q}_q$ and $\mathbf{Q}_{\dot{q}}$ represent the uncertainty on the model, while $\mathbf{Q}_u$ and $\mathbf{Q}_p$ represent the uncertainty on the unknown input and parameters, respectively.

The terms of $\mathbf{Q}_u$ are set using the rule of thumb proposed in [45]. For the i-th unknown input u_i, the relative term on the diagonal of $\mathbf{Q}$ is set as:

$$Q_{u_i} = (\Delta t \omega_{u_i} a_{u_i})^2 \tag{8.30}$$

where ω_{u_i} and a_{u_i} are the expected values of frequency and amplitude of the force. As pointed out in [45] and [119], this method aims at keeping the rate of change of the unknown quantity in

a range that allows to correctly track its variation while filtering noise. Following this reasoning, $\mathbf{Q}_p$ is also set using the same rule of thumb:

$$Q_{p_i} = \left(\Delta t \delta_{p_i} a_{p_i}\right)^2 \tag{8.31}$$

In this case a_p is the initial tentative value for the parameter and δ_p is a scaling factor that helps in the setup of the unknown parameter covariance by decoupling it from the absolute value of the parameter itself as well as the integration scheme time step size. In this way, the influence of δ_p on the estimation results can be studied and applied to parameters with different order of magnitudes. The units of δ_p are equivalent to frequency.

The value of δ_p should typically be set to a low value, as the parameter value is usually constant or slowly varying. It should not be set excessively low, as this can slow down the convergence rate to as well as constrain the tracking of any change of the parameter value in time. Given the adopted zeroth order random walk model, a higher augmented state covariance value for the corresponding parameter state allows for a larger change per timestep, lowering the achieved filtering of noise from the estimated augmented states, which is present due to measurement noise propagation in the correction step of the EKF.

Another important choice to be made is the initial guess for the values $\mathbf{x}*_0$ and $\mathbf{P}_0$. It is common practice to assume the structure to be in undeformed and steady conditions at the start of the estimation, so that $\mathbf{q}_{f_r,0} = \mathbf{0}$ and $\dot{\mathbf{q}}_{f_r,0} = \mathbf{0}$. If the assumption is not satisfied, the filter can usually quickly compensate for it. Also the initial input can be assumed to be zero, so that $\mathbf{u}_0^{un} = \mathbf{0}$. The initial parameters values requires a more careful choice given that a zero value would have no physical meaning and furthermore cause numerical instability. In most practical cases, a nominal value for the parameter is known and can be used as an initial guess in $\mathbf{p}_0$. The values chosen for $\mathbf{P}_0$ reflect the confidence in the initial guesses. A high initial error covariance causes the filter to react quickly to initial measurement mismatches. A low value, on the other hand, implies higher initial confidence in the tentative values and will make the filter more inert and less responsive.

8.4 Observability analysis

In order for the EKF to be able to correctly estimate the unknown quantities of the system, a proper choice of the measurement set is needed [107, 108, 44]. This choice can be made by analyzing the system observability, which depends on the system-measurement combination $(\mathbf{A}^*, \mathbf{H}^*)$.

8.4.1 Continuous system linearization

As noted in [107], observability analysis methods are traditionally developed for linear systems, but given the mild non-linearity of structural systems with respect to parameters it is still worth analyzing the properties of the linearized model around representative linearization points. Furthermore, the analysis is done on the time-continuous system for simplicity reasons, but all results translate directly to the discrete case. The continuous linearized model is represented by $\mathbf{J}_c$ and $\mathbf{J}_m$, which are the derivatives of the state and measurement equations of the continuous-time model in Equation 8.4 respectively, with respect to the system state.

The Jacobian of the continuous state equation is defined as:

$$\mathbf{J}_c = \frac{\partial \dot{\mathbf{x}}^*}{\partial \mathbf{x}^*} = \mathbf{A}^*(\mathbf{x}^*) + \frac{\partial \mathbf{A}^*(\mathbf{x}^*)}{\partial \mathbf{x}^*}\mathbf{x}^* + \frac{\partial \mathbf{B}^*(\mathbf{x}^*)}{\partial \mathbf{x}^*}\mathbf{u}^{kn} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ -\mathbf{M}_{f_r}^{-1}\mathbf{K}_{f_r} & -\mathbf{M}_{f_r}^{-1}\mathbf{C}_{f_r} & \mathbf{M}_{f_r}^{-1}\mathbf{S}_{un,r} & \frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial \mathbf{p}} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{8.32}$$

where

$$\frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial \mathbf{p}} = -\frac{\partial \mathbf{M}_{f_r}^{-1}\mathbf{K}_{f_r}}{\partial \mathbf{p}}\mathbf{q}_{f_r} - \frac{\partial \mathbf{M}_{f_r}^{-1}\mathbf{C}_{f_r}}{\partial \mathbf{p}}\dot{\mathbf{q}}_{f_r} + \frac{\partial \mathbf{M}_{f_r}^{-1}}{\partial \mathbf{p}}\mathbf{S}^{un,r}\mathbf{u}^{un} + \frac{\partial \mathbf{M}_{f_r}^{-1}}{\partial \mathbf{p}}\mathbf{S}^{kn,r}\mathbf{u}^{kn} \tag{8.33}$$

Some of the terms in Equation 8.32 and Equation 8.33 become zero for specific choices of parameter types or specific types of input. If only a set of stiffness parameters $\mathbf{p}_s$ is estimated, $\frac{\partial \ddot{\mathbf{q}}}{\partial \mathbf{p}}$ takes the form:

$$\begin{aligned} \frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial \mathbf{p}_s} = -\mathbf{M}_{f_r}^{-1}\frac{\partial \mathbf{K}_{f_r}}{\partial \mathbf{p}_s}\left(\mathbf{q}_{f_r} + \beta\dot{\mathbf{q}}_{f_r}\right) = \\ -\mathbf{M}_{f_r}^{-1}\frac{\partial \mathbf{K}_{f_r}}{\partial \mathbf{p}_s}\mathbf{K}_{f_r}^{-1}\left(\mathbf{S}_{un,r}\mathbf{u}^{un} + \mathbf{S}_{kn,r}\mathbf{u}^{kn} - \mathbf{M}_{f_r}\left(\ddot{\mathbf{q}}_{f_r} + \alpha\dot{\mathbf{q}}_{f_r}\right)\right) \end{aligned} \tag{8.34}$$

If only a set of mass parameters $\mathbf{p}_m$ is to be estimated, $\frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial \mathbf{p}}$ takes the form:

$$\begin{aligned} \frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial \mathbf{p}_m} = -\frac{\partial \mathbf{M}_{f_r}^{-1}}{\partial \mathbf{p}_m}\left(\mathbf{K}_{f_r}\left(\mathbf{q}_{f_r} + \beta\dot{\mathbf{q}}_{f_r}\right) - \mathbf{S}_{kn,r}\mathbf{u}^{kn} - \mathbf{S}_{un,r}\mathbf{u}^{un}\right) = \\ -\mathbf{M}_{f_r}^{-1}\frac{\partial \mathbf{M}_{f_r}}{\partial \mathbf{p}_m}(\ddot{\mathbf{q}}_{f_r} + \alpha\dot{\mathbf{q}}_{f_r}) \end{aligned} \tag{8.35}$$

For a measurement in which the dependency on states and inputs can be written in the form:

$$\mathbf{h}(\mathbf{x}^*, \mathbf{u}^{kn}) = \mathbf{H}^*(\mathbf{x}^*)\mathbf{x}^* + \mathbf{D}^*(\mathbf{x}^*)\mathbf{u}^{kn} \tag{8.36}$$

as it is the case in this work and without loss of generality, the Jacobian results in:

$$\mathbf{J}_m = \frac{\partial \mathbf{h}(\mathbf{x}^*, \mathbf{u}^{kn})}{\partial \mathbf{x}^*} = \mathbf{H}^*(\mathbf{x}^*) + \frac{\partial \mathbf{H}^*(\mathbf{x}^*)}{\partial \mathbf{x}^*}\mathbf{x}^* + \frac{\partial \mathbf{D}^*(\mathbf{x}^*)}{\partial \mathbf{x}^*}\mathbf{u}^{kn} \tag{8.37}$$

In this chapter, strain and acceleration measurements are considered. As can be seen by comparing Equation 2.32 and Equation 2.34, strain and position measurements for structural applications are equivalent in terms of contained information. All results discussing strain measurement apply then seamlessly also to them, and they will not be directly discussed in the remainder of the chapter.

For strain measurements, the linearization results in:

$$\mathbf{J}_{m,s} = \begin{bmatrix} \mathbf{N}_s & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{8.38}$$

For acceleration measurements instead:

$$\mathbf{J}_{m,a} = \mathbf{N}_d\mathbf{\Psi}\begin{bmatrix} -\mathbf{M}_{f_r}^{-1}\mathbf{K}_{f_r} & -\mathbf{M}_{f_r}^{-1}\mathbf{C}_{f_r} & \mathbf{M}_{f_r}^{-1}\mathbf{S}_{un,r} & \frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial \mathbf{p}} \end{bmatrix} \tag{8.39}$$

8.4.2 PBH test for linearized system

The original PBH test in Equation 7.28 can be formulated for the linearized system as:

$$\mathbf{PBH} = \begin{bmatrix} \mathbf{I}s - \mathbf{J}_c \\ \mathbf{J}_m \end{bmatrix} \tag{8.40}$$

The PBH matrix, for $s = 0$ and using strain and acceleration measurements, can be expressed as:

$$\mathbf{PBH} = \begin{bmatrix} \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{M}_{f_r}^{-1}\mathbf{K}_{f_r} & \mathbf{M}_{f_r}^{-1}\mathbf{C}_{f_r} & -\mathbf{M}_{f_r}^{-1}\mathbf{S}_{un,r} & -\frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial \mathbf{p}} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{N}_s\boldsymbol{\Psi} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{N}_d\boldsymbol{\Psi}\mathbf{M}_{f_r}^{-1}\mathbf{K}_{f_r} & -\mathbf{N}_d\boldsymbol{\Psi}\mathbf{M}_{f_r}^{-1}\mathbf{C}_{f_r} & \mathbf{N}_d\boldsymbol{\Psi}\mathbf{M}_{f_r}^{-1}\mathbf{S}_{un,r} & \mathbf{N}_d\boldsymbol{\Psi}\frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial \mathbf{p}} \end{bmatrix} \tag{8.41}$$

8.4.3 Generic observability results

Some conclusions can be drawn directly:

- In order to have full observability, a number of strain measurements at least equal to the total number of unknown quantities (inputs and parameters) should be employed. The rows relative to acceleration measurements in Equation 8.41 are in fact always a linear combination of the second block rows. For any application case then, the use of only acceleration measurements does not allow for an observable system. Acceleration measurements are still important since they add information useful to improve the performances of the estimation procedure
- Parameters related to mass cannot be observed in a static condition (for which $\frac{\partial \ddot{\mathbf{q}}}{\partial \mathbf{p}}$ is zero) because the last block column becomes zero. This follows the practical intuition that in a static case the inertial term is not excited and thus measurements carry no useful information.

Other conclusions can be made in particular cases for which the equations simplify and it is possible to observe linear dependency of the columns of the PBH matrix. In the next subsection the observability for the case of a single parametrized material is discussed.

8.4.4 A particular case

This section considers the case for which one material is parameterized for both Young's modulus and density. In this case the mass, stiffness and damping matrices assume the simple form:

$$\mathbf{K}_{f_r} = E\mathbf{K}_{f_r}^E \tag{8.42}$$

$$\mathbf{M}_{f_r} = \rho\mathbf{M}_{f_r}^\rho \tag{8.43}$$

$$\mathbf{C}_{f_r} = \alpha\rho\mathbf{M}_{f_r}^\rho + \beta E\mathbf{K}_{f_r}^E \tag{8.44}$$

This simplified case yields the following derivatives:

$$\begin{aligned} \frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial E} &= -\frac{1}{\rho}\mathbf{M}_{f_r}^{\rho^{-1}}\mathbf{K}_{f_r}^E\left(\mathbf{q}_{f_r} + \beta\dot{\mathbf{q}}_{f_r}\right) = \\ &= \frac{1}{\rho E}\mathbf{M}_{f_r}^{\rho^{-1}}\left(\mathbf{S}_{kn,r}\mathbf{u}^{kn} + \mathbf{S}_{un,r}\mathbf{u}^{un}\right) - \frac{1}{E}(\ddot{\mathbf{q}}_{f_r} + \alpha\dot{\mathbf{q}}_{f_r}) \end{aligned} \tag{8.45}$$

$$\frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial \rho} = -\frac{E}{\rho^2}\mathbf{M}_{f_r}^{\rho^{-1}}\left(\mathbf{K}_{f_r}^{E}\left(\mathbf{q}_{f_r}+\beta\dot{\mathbf{q}}_{f_r}\right)-\mathbf{S}_{kn,r}\mathbf{u}^{kn}\right) = -\frac{1}{\rho}(\ddot{\mathbf{q}}_{f_r}+\alpha\dot{\mathbf{q}}_{f_r}) \tag{8.46}$$

For $s = 0$, the PBH matrix is:

$$\begin{bmatrix} \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \frac{E}{\rho}\mathbf{M}_{f_r}^{\rho^{-1}}\mathbf{K}_{f_r}^{E} & \left(\alpha+\frac{\beta E}{\rho}\mathbf{M}_{f_r}^{\rho^{-1}}\mathbf{K}_{f_r}^{E}\right) & -\frac{1}{\rho}\mathbf{M}_{f_r}^{\rho^{-1}}\mathbf{S}_{un,r} & -\frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial E} & -\frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial \rho} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{N}_s\mathbf{\Psi} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{N}_d\mathbf{\Psi}\frac{E}{\rho}\mathbf{M}_{f_r}^{\rho^{-1}}\mathbf{K}_{f_r}^{E} & -\mathbf{N}_d\mathbf{\Psi}\left(\alpha+\frac{\beta E}{\rho}\mathbf{M}_{f_r}^{\rho^{-1}}\mathbf{K}_{f_r}^{E}\right) & \mathbf{N}_d\mathbf{\Psi}\frac{1}{\rho}\mathbf{M}_{f_r}^{\rho^{-1}}\mathbf{S}_{un,r} & \mathbf{N}_d\mathbf{\Psi}\frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial E} & \mathbf{N}_d\mathbf{\Psi}\frac{\partial \ddot{\mathbf{q}}_{f_r}}{\partial \rho} \end{bmatrix} \tag{8.47}$$

It can be observed that:

- Given Equation 8.45 and Equation 8.46, it is impossible to simultaneously estimate E and ρ if there is no input being applied to the system. This implies that under a free response condition the effect of mass and inertia is not distinguishable. The last two columns become in fact linearly dependent.
- It is impossible to simultaneously estimate inputs, E and ρ if there is not at least one non-zero known input.
- If E and the input are to be estimated, the system is not observable in static conditions. This implies that it is impossible to estimate both the Young's modulus and a static load, confirming the intuition that the system inversion in this case is not feasible.

8.5 Filter setup

The KF capability of estimating the full states of a system together with unknown inputs and parameters comes at the cost of two main requirements. The first of these is the accuracy of the numerical model with respect to the physical one. The steps required to update a model based on experimental results has been shown in the first part of this work for the blade and manipulator systems. The second requirement is an appropriate choice of the filter setup in terms of sensors used and settings (i.e. covariances, initial parameters values). Such choices are critical and often require an expert user, resulting in one of the most critical point of the approach. This section describes the setup of the filter detailing the choices made. The application case considered is the scaled wind turbine blade discussed in Section 3.5.

8.5.1 Sensor selection

The physical blade is instrumented with strain and acceleration sensors spread over its entire length, as shown in Figure 8.1, therefore also virtual sensors are placed at the same locations. In particular, the sensors used are:

- 10 rectangular strain gage rosettes placed symmetrically on the top and bottom face at 5 different sections along the length of the blade (numbered from 0 to 4 in the remaining part of the paper).
- 4 uniaxial strain gages placed symmetrically on the top and bottom face of sections 2 and 3 along the axis of the blade.
- 10 triaxial accelerometers placed on the top face at 8 sections along the length of the blade (named from A to H in the remaining part of the paper). Their axes are aligned with the global ones.

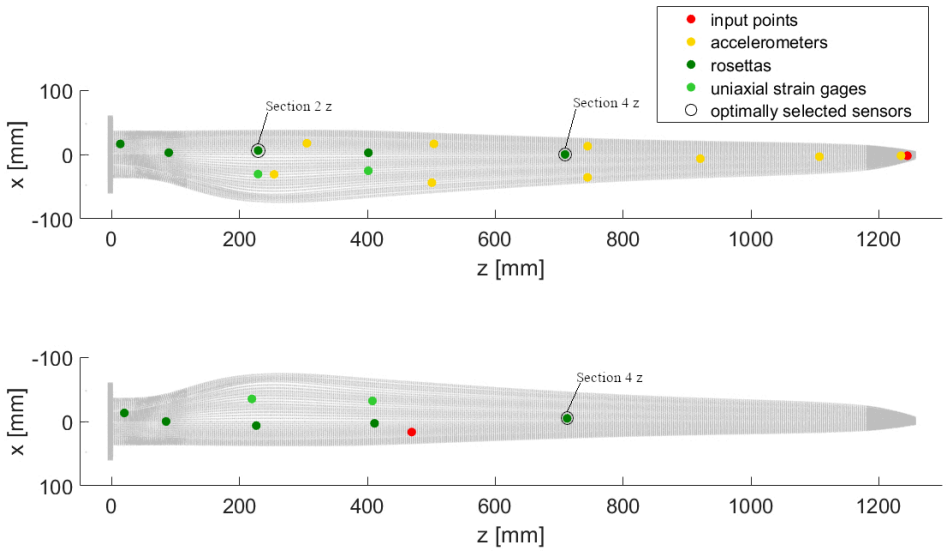


Figure 8.1: Position of sensors and input points on the blade. The 3 sensors labeled are the rosettes of which the z sensors were selected by the OSP procedure.

Only a subset of sensors is used for the estimation, while the remaining ones are used for validation purposes: the estimated values are compared with the simulated ones, for the numerical applications, or with the measured ones, for experimental applications. A minimal number of strain gages is required for observability reasons, as discussed in Section 8.4. The addition of accelerometers to this set could extend the overall bandwidth of the filter. The frequency range investigated in this work is however low enough to allow the use of only strain gages for the estimation, while the accelerations sensors will therefore be used for validation.

The selection of strain gages to use for the estimation has been based on the OSP strategy described in [44]. There the aim is to find the optimal location and orientation of sensors in order to guarantee the best observability for unknown inputs. The strategy starts by screening an initial sensor pool spread over the entire surface of the model, and then subsequently removing the sensors that contribute the least to the observability metric of interest. For the blade setup considered in this work, the OSP strategy has been adapted to start from the pool of existing sensors and act as an OSS method, following the subsequent steps as discussed in the reference.

The OSP procedure has been used to select a set of 3 strain gages, since that is the minimum required number for estimation of the two parameters and one input. The selected sensors are the ones oriented along the axis of the blade in the rosettes on the top face of sections 2 and 4 and on the bottom face of section 4, as shown on Figure 8.1.

The OSP approach targets input estimation and therefore it is only used in this work as a way to select the sensors to use, without any claim of global optimality. The extension of the OSP procedure to consider also observability of parameters is interesting and should be investigated in future efforts. This has been considered out of the scope of this work and is therefore not explored here.

8.5.2 Covariance setup

In the case of experimental data, the measurement noise covariance matrix $\mathbf{R}$ is set using values extracted for each physical sensor. If numerical measurements are used, the noise level is known

and its covariance value can be directly used as the diagonal entries of $\mathbf{R}$. The initial value for $\mathbf{P}$ is set by assuming a low initial error on the states and an initial error on the parameters in the same order of magnitude as the expected parameter values themselves to improve initial convergence.

It is common practice for AKF application to set the covariance of the plant to a low value to let the filter follow the numerical model predictions [45, 107]. In the following application $\mathbf{Q}_q = \varepsilon\mathbf{I}$ and $\mathbf{Q}_{\dot{q}} = \varepsilon\mathbf{I}$, where ε represents the machine precision.

Setting the zeroth order random walk noise covariance to a low value implies that the associated augmented state is expected to remain constant or vary slowly. A large value permits a large variation of the associated augmented state during each time step. These covariance values depend on the assumed nature of the augmented states, i.e. having a specific expected quasi-static or dynamic behavior. In the remaining part of this section the effect of choosing different augmented state noise covariance values on the corresponding estimation results is shown. The values are represented by ω_u or δ_p, which are used to calculate the covariance values by applying Equation 8.30 or Equation 8.31. The following estimation cases are based on numerical data in order to have full control and assure that any variations in the estimated values are only due to the noise present in the supplied numerical measurements. The measurements are generated by a forward simulation using the same integration scheme as in Section 3.5 and then zero-mean white Gaussian noise is added. A 20 Hz sinusoidal load (Figure 8.2) is used for this analysis, without loss of generality.

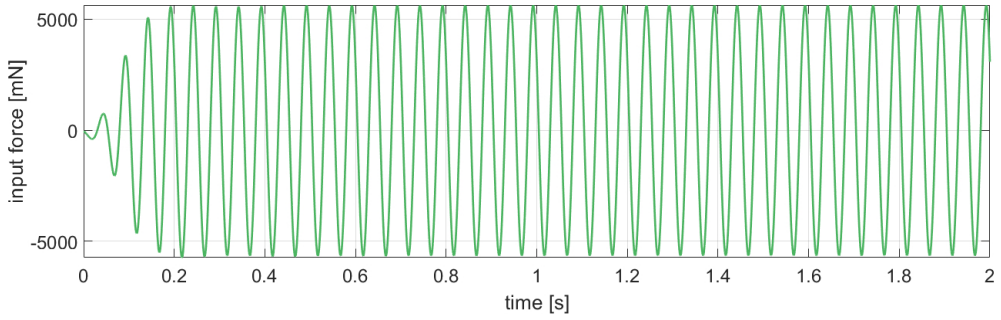


Figure 8.2: 20Hz sine input

The first case considered is the estimation of a constant Young's modulus. The estimated parameter value for different values of δ_E is shown in Figure 8.3. In order to evaluate the estimation performance, the average error is considered along with the maximum and minimum error that define the interval in which the estimated parameter evolves. Ideally the average value for the error should be close to zero and the maximum and minimum interval should be as narrow as possible. A large interval indicates a large variation usually caused by noise. The desired behavior of the filter in this first case is that the estimated parameter value converges and stays as constant as possible. All the δ_E values allow for similar convergence behavior, with a similar and small average error in the last part. It can also be seen that the evolution interval is narrow for δ_E values below a certain threshold ($10^{-2}Hz$) while for higher values it increases, showing more measurement noise leaking into the estimate, as explained in Section 8.3. It can be concluded in this case that a covariance value as low as possible is desired, and that any value below a certain threshold will allow for good results.

A second case is considered where the parameter is degrading in time (Figure 8.4). A low δ_E value constrains the rate of change of the parameter, preventing it from properly tracking the real value. The minimum δ_E value that allows to properly track the evolution of the parameter

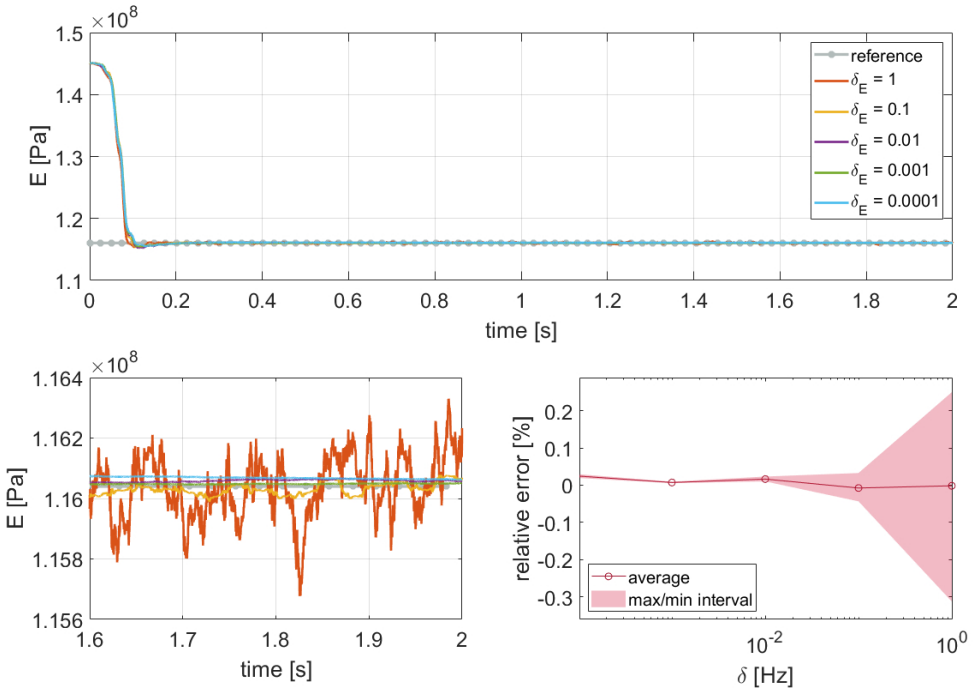


Figure 8.3: Estimated value of E with numerical data for sine load with different values of δ_E. A zoom on the last 0.4s is presented together with the estimation error on the same time interval.

for the presented case is $1Hz$. By further increasing the δ_E value, the average error remains small, while the variation interval gets larger, showing that more measurement noise leaks into the estimated augmented states. In this case a larger δ_E value should be used.

These results indicate that there is no overall optimal covariance value setting and the setting thus has to be made according to the application case and the expected evolution of the augmented state over time. A representative virtual exercise could be carried out to select these settings. In a system identification case, where the model parameters are expected to be constant, δ_E can be set as low as possible. If an abrupt change in parameter values is expected instead, a high value should to be employed at the expense of a more noisy estimate (as explained in Section 8.3).

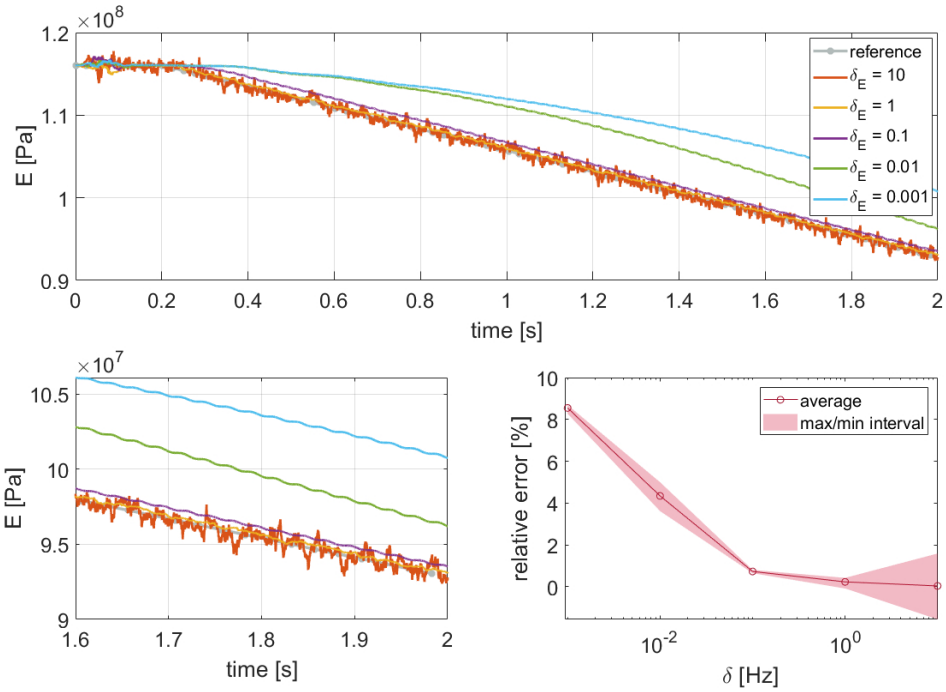


Figure 8.4: Estimated value of degrading E with numerical data for sine load with different values of δ_E. A zoom on the last 0.4s is presented together with the estimation error on the same time interval.

A third case is considered where E and ρ are estimated simultaneously, as shown in Figure 8.5. If the covariance values are not high enough, the estimated parameters tend to converge much slower as compared to the single parameter case. Also here a covariance value for which the average error stabilizes corresponds to $1Hz$. The decreased convergence rate can be explained by the fact that the dynamic stiffness of a linear structural system tends to depend on the ratio between stiffness and mass parameters. Once the filter has reached the correct stiffness/mass ratio (by means of the estimated parameters), the dynamic response of the system using these estimated parameter values is close to the real response, causing the low convergence rate that thus has to be compensated for with higher covariance values of the parameters. This is shown in Figure 8.7. Here, the ratio of the two parameters converges faster than their single values.

When parameters and input are estimated simultaneously, their covariance settings appears to be uncoupled for this specific application, since they only effect their relative augmented state's convergence. As a consequence, the choice of covariance values for parameters, as discussed above, also applies for this case. The results shown in Figure 8.6 are calculated for a fixed $\delta_E = 10^{-3}$ and varying ω_u. As discussed and shown in [107, 109, 118], a small lag is usually present in the estimated input with respect to the measured one, even when the amplitude is correctly estimated. This lag can also be clearly seen in the results of this application when the frequency of the input is set to the known value of 20Hz. This represents a mismatch in the case of input only estimation. When parameters are jointly estimated with the inputs, this phase shift is compensated for by the filter through an estimated parameter value different from the real one. Increasing the estimated input frequency value reduces the observed lag at the expense of more noise. This in turn reduces the error on the parameter estimation, as can be clearly seen in the plot of parameter error with respect to ω_u.

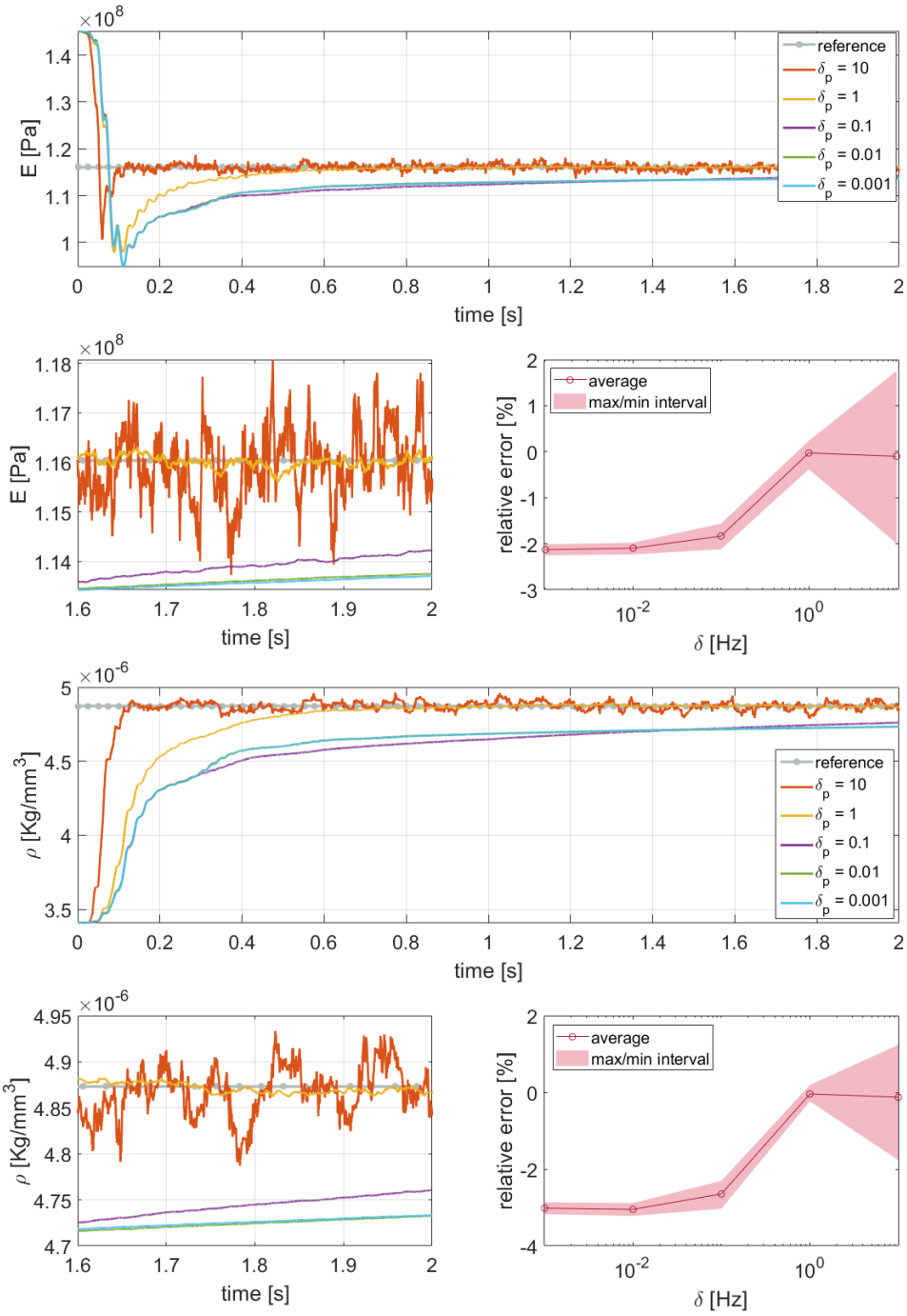


Figure 8.5: Estimated values of E and ρ with numerical data for sine load with different values of $\delta_p = \delta_E = \delta_\rho$. A zoom on the last 0.4s is presented together with the estimation error on the same time interval.

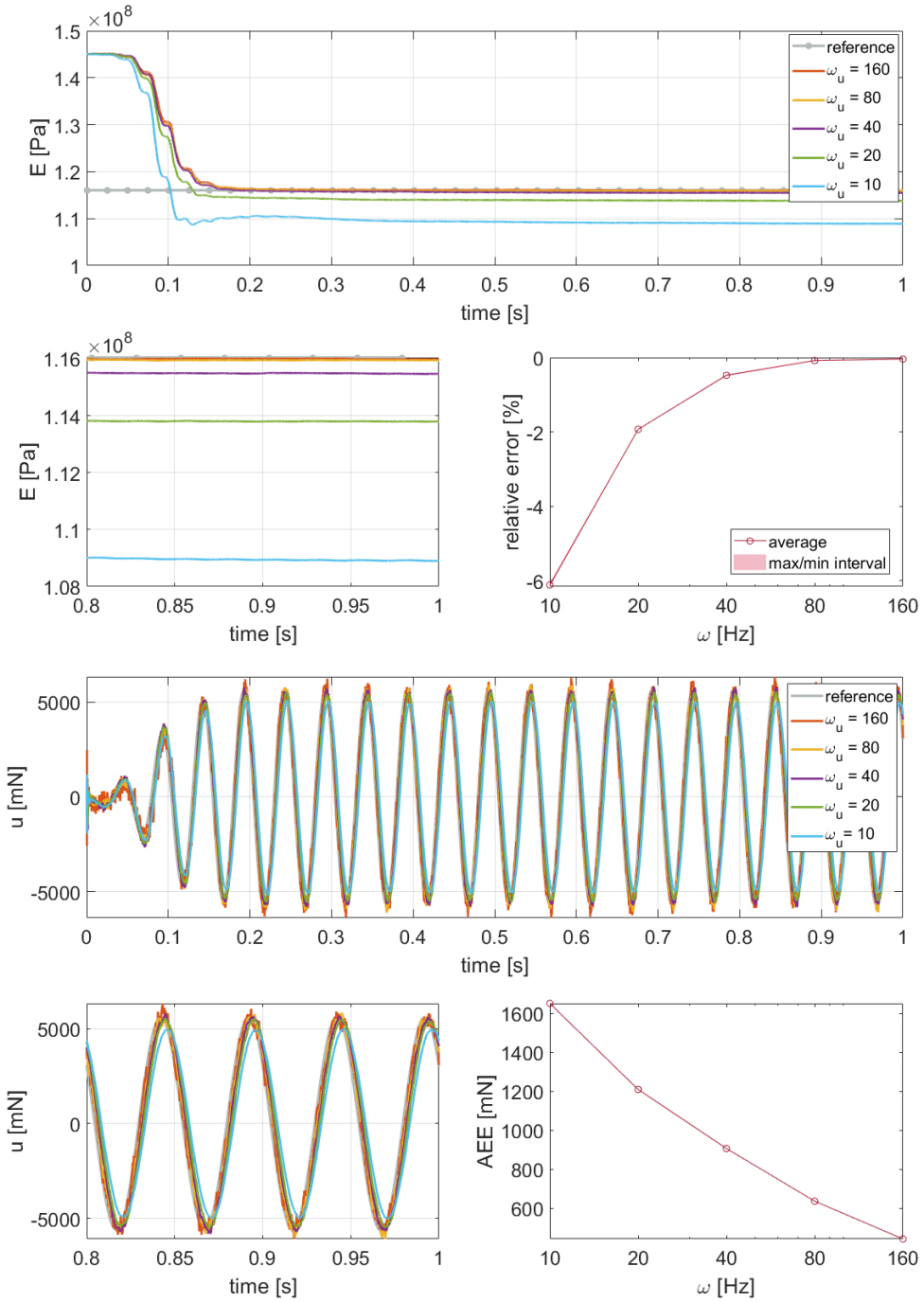


Figure 8.6: Estimated values of E and inputs with numerical data for sine load with different values of ω_u and $\delta_E = 10^{-3}Hz$. A zoom on the last 0.2s is presented together with the estimation error on the same time interval.

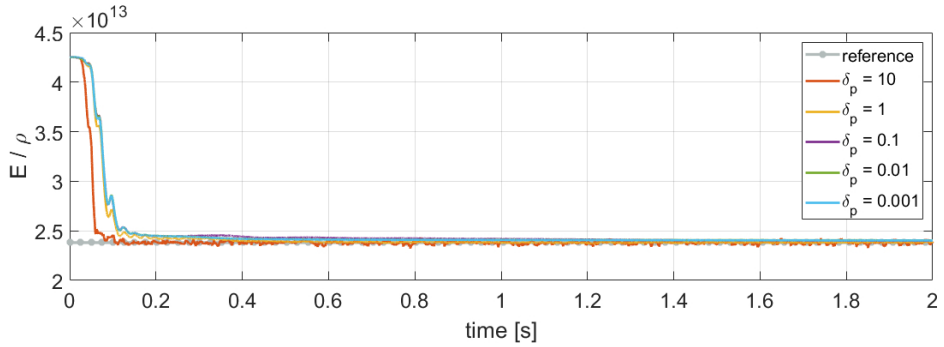


Figure 8.7: Value of the ratio between estimated E and ρ for the case shown in Figure 8.5.

8.6 Numerical validation

A first validation of the estimation procedure is done using numerical data. This allows, by having full control over the system and the measured data, to asses the performance of the filter with the setup defined in the previous section. The numerical approach also allows to consider the case of a parameter degrading with time, for which no experimental data is available.

As the full set of sensors is estimated, the sensors not used as measurements can be compared with their reference simulation values to assess if the filter can correctly estimate the full field of strains and accelerations by using only a limited set of sensors. This evaluation is based on the Averaged Absolute Error (AAE) metric defined in [119] as $\sum_{i=1}^{T} |y_{est,i} - y_{meas,i}|/T$ where $y_{est,i}$ and $y_{meas,i}$ are the estimated and measured signals at the i-th time step, respectively, and T is the number of steps in the considered time interval. This metric is compared with the Maximum Response Amplitude (MRA) for each case in order to compare the average error with the maximum value assumed by the signal. For this numerical application, the sensor estimation results are shown only for the most complex case of parameter and input estimation, as all the others are comparable.

The inputs used for this validation are a static load on the tip, a 20 Hz sinusoidal load (Figure 8.2) and a broadband continuous random signal (Figure 3.9). Data on the inputs is listed in Table 8.1.

Table 8.1: Load case data

Signal type	**Frequency [Hz]**	**Amplitude [N]**
Static	0	14.7
Sine	20	5
Random	0-500	14 (max)

Each application case has been run for every possible load case. In some of the cases the static load has not been used since it makes the system unobservable, as shown in the theoretical observability analysis (Section 8.4). White noise, with an amplitude in the order of $10^{-7} m/m$ and $10^{-3} m/s^2$ for strain and acceleration measurements respectively, is added to the generated numerical measurements. Both simulations and estimations use a time step of $0.1ms$.

An overview of the performance of all the cases in terms of parameter estimation is shown in Table 8.2. This table shows the minimum, maximum and average relative error of the estimated parameters with respect to the reference values in the last 20% of the time interval, for which the estimated values have typically converged to stable values. Analysis of this interval gives an

idea on the convergence to the correct value and how much the parameter varies in time after converging.

Table 8.2: Errors on estimated parameters in last 20% of time interval for numerical measurements

Unknown quantities	**Load**	**Error** [%]			
		Parameter	**min**	**avg**	**max**
E	static	E	0.01	0.01	0.01
	sine	E	0.01	0.02	0.02
	random	E	-0.02	-0.02	-0.02
E, ρ	static	unobservable			
	sine	E	-0.50	-0.22	0.00
		ρ	-0.30	-0.09	0.15
	random	E	-0.23	-0.08	0.16
		ρ	-0.29	-0.07	0.19
degrading E	static	E	-0.06	0.07	0.19
	sine	E	-0.08	0.24	0.43
	random	E	-0.04	0.16	0.38
E, input	static	unobservable			
	sine	E	-0.09	-0.08	-0.07
	random	E	0.18	0.21	0.23
degrading E, input	static	unobservable			
	sine	E	0.07	0.39	0.75
	random	E	0.27	0.56	0.97

8.6.1 Constant parameter estimation

As a first application case, the Young's modulus is estimated. The density and the loads are known. The initial value of the augmented state related to the parameter is set to a value 25% higher than the reference value, and its covariance is calculated by setting $\delta_E = 10^{-3} Hz$. This parameter is observable for all considered load cases.

The evolution of the estimated parameter value over a $1s$ time interval is shown in Figure 8.8. For the static load case the estimated parameter shows a fast convergence to the reference value since the load is applied from the start. For the dynamic cases instead the value converges gradually to the correct one, since the initial input is zero and it gradually increases, allowing the filter to smoothly correct the value of E.

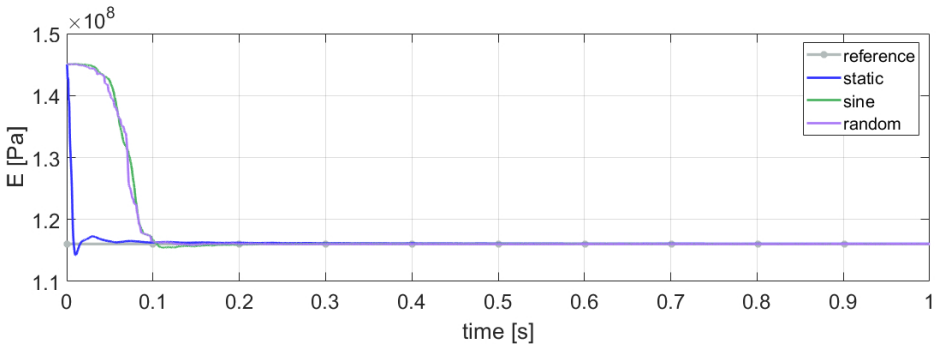


Figure 8.8: Estimated value of E with numerical data, for all load cases, with $\delta_E = 10^{-3} Hz$.

As a second application case, both Young's modulus and density are unknown. The results are shown in Figure 8.9. Only the dynamic load cases can be considered, since the density is unobservable for a static input. The initial errors on the augmented states related to E and ρ are respectively set at 25% and -30%. The observed convergence rate is slower and thus a larger δ_p of $1Hz$ is adopted. The converged parameter values estimated by the filter accurately match the reference ones.

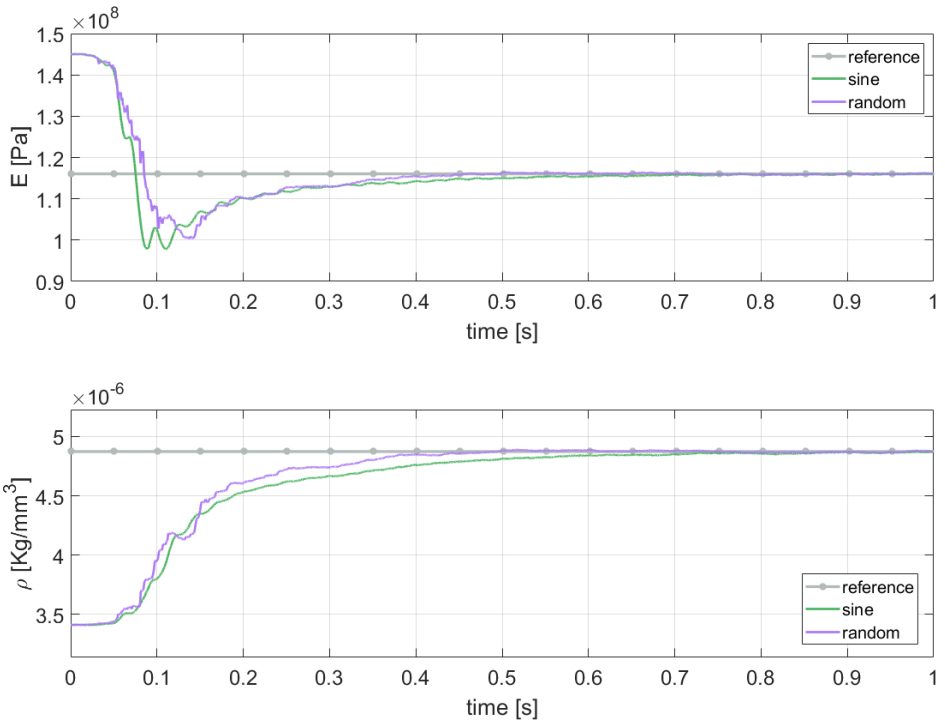


Figure 8.9: Estimated values of E and ρ with numerical data, for sine and random loads, with $\delta_E = 1Hz$, $\delta_\rho = 1Hz$.

8.6.2 Degrading parameter estimation

A fundamental application of the presented methodology is the tracking of a parameter that deteriorates in time. In order to generate numerical data for the deterioration case, a simulation is run using the pROM during which the Young's modulus, starting from the correct value, decreases at a rate of 10%/s. It should be noticed how this type of simulation during which the parameters vary continuously is enabled by the use of the pROM, as it allows to update the model in an efficient way (i.e. without having to recreate the ROM).

The results of the estimation of only E, with $\delta_E = 1Hz$, are shown in Figure 8.10. The large covariance value needed to allow the correct tracking of the parameter variation causes measurement noise to propagate to the estimated parameter. This can be seen as an acceptable solution as the average error remains small.

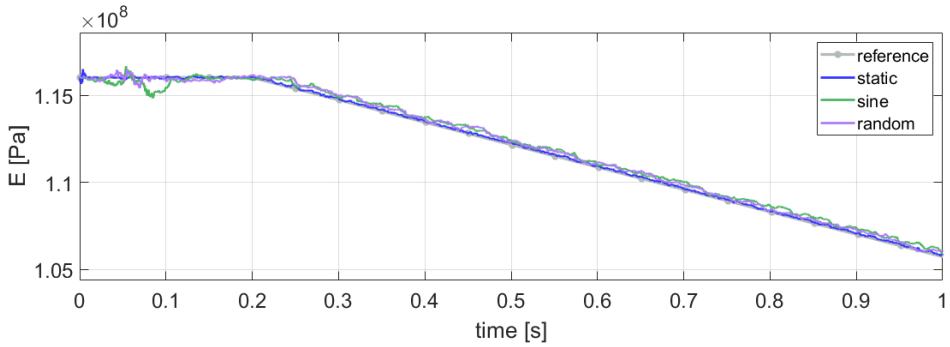


Figure 8.10: Estimated value of degrading E with numerical data, for all load cases, with $\delta_E = 1Hz$.

8.6.3 Parameter and input estimation

As a more complex application case, both the Young's modulus and the input are considered as unknown. In this case the filter has to be able to discern between the influence of a change of parameter values from a time-varying input. The results of the coupled estimation of constant E ($\delta_E = 10^{-3}Hz$) and a dynamic load ($\omega_{u,2} = 80Hz$, $\omega_{u,3} = 2000Hz$) are shown in Figure 8.11. The system is unobservable for the static load case. Both dynamic load cases show excellent results with a correct and fast estimation of the parameters. The estimated input displays some noise given by the necessity for larger related covariance values, as explained in Subsection 8.5.2, to avoid lag. The estimated sensor errors for both strain sensors and accelerometers are shown in Figure 8.13. It can be observed how the AAE is always at least one order of magnitude lower that the MRA. Some exceptions are in the accelerometers with lower MRA, due to accelerations being more sensitive to the noise introduced by the input estimation. The estimation procedure also shows good results for the case where the Young's modulus is varying in time, with an unknown input and known density. The results are shown in Figure 8.12. Here, δ_E is set to 1 Hz and thus more noise is added.

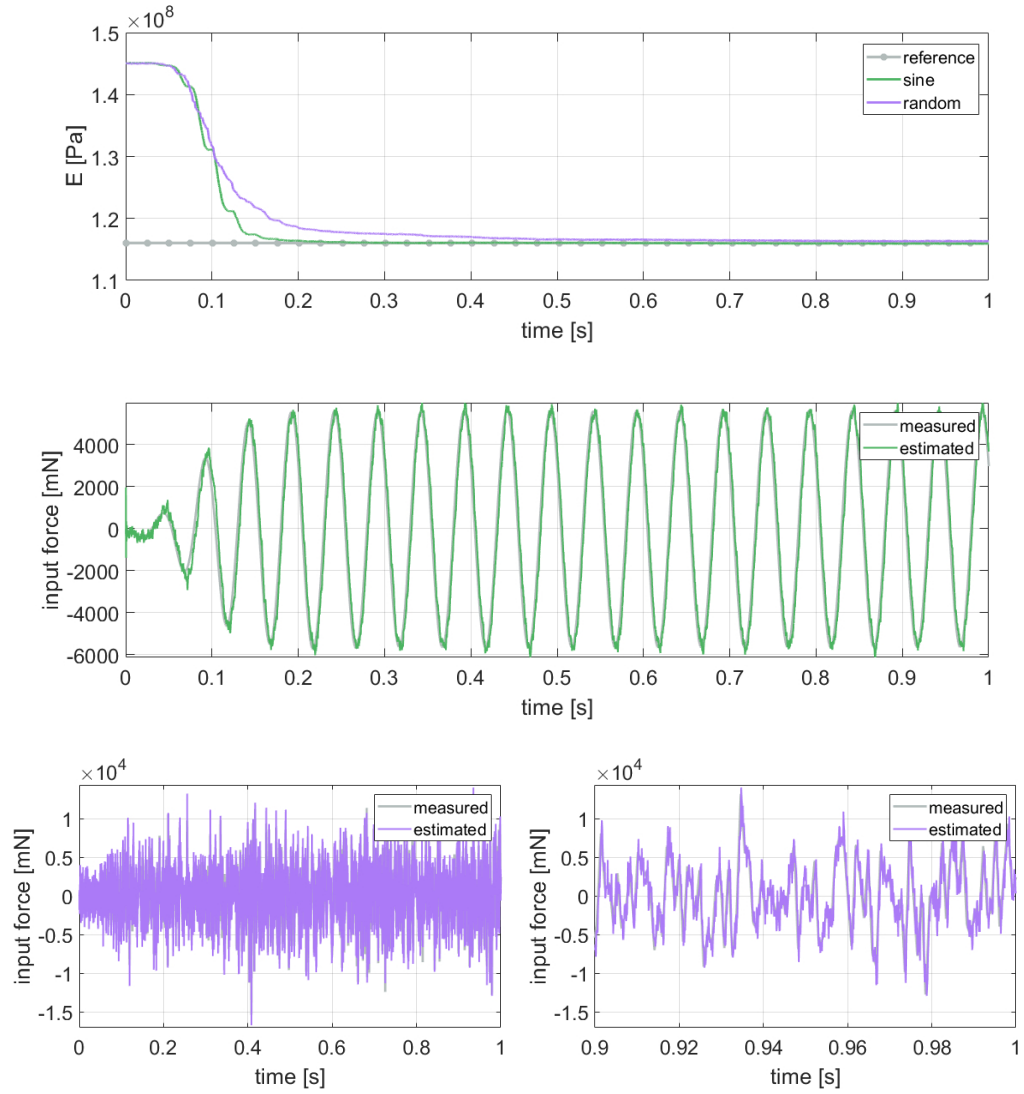


Figure 8.11: Estimated values of E and inputs with numerical data, for the sine and random load cases, with $\delta_E = 10^{-3}Hz$, $\omega_{u,2} = 80Hz$ and $\omega_{u,3} = 2000Hz$. The right figures show a zoom of the 0.9s to 1s range.

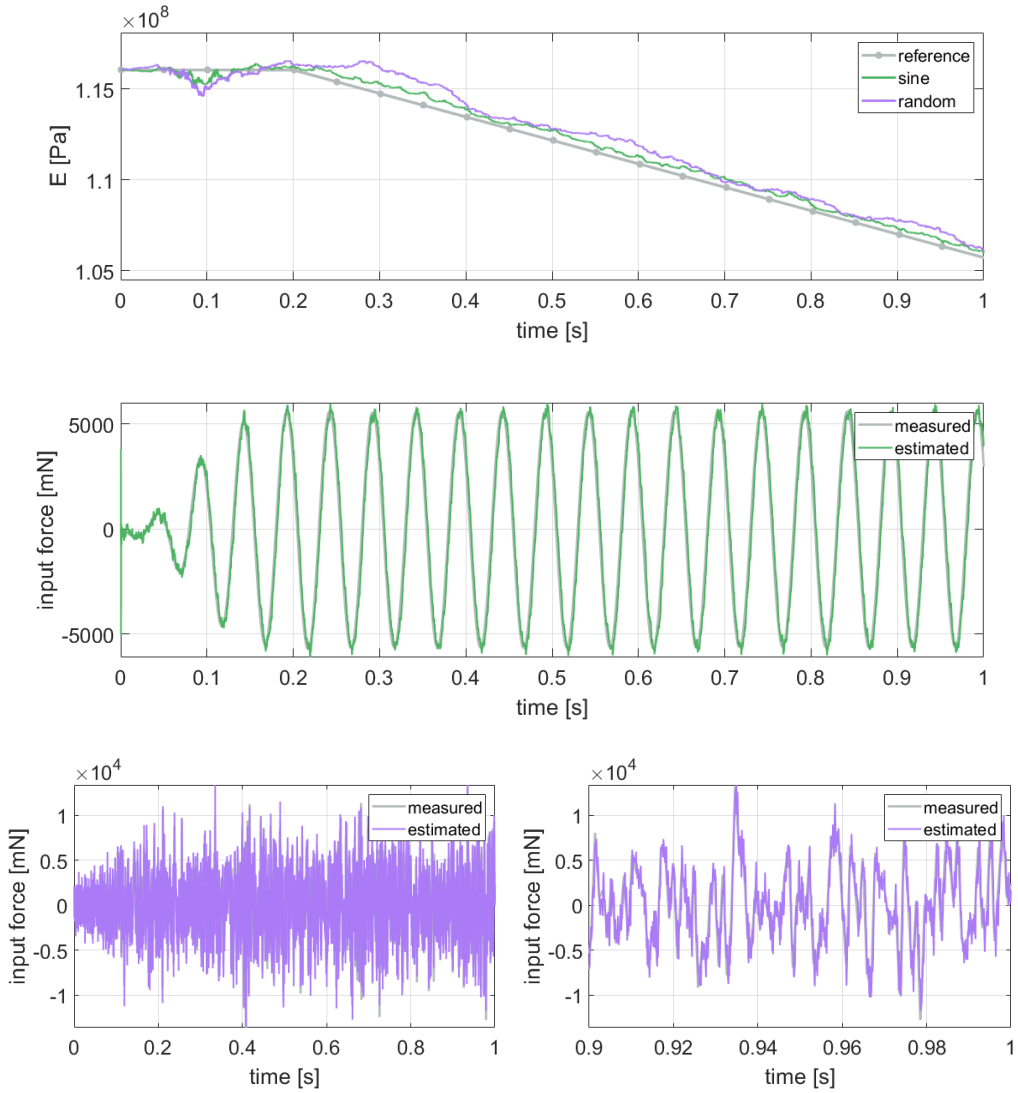
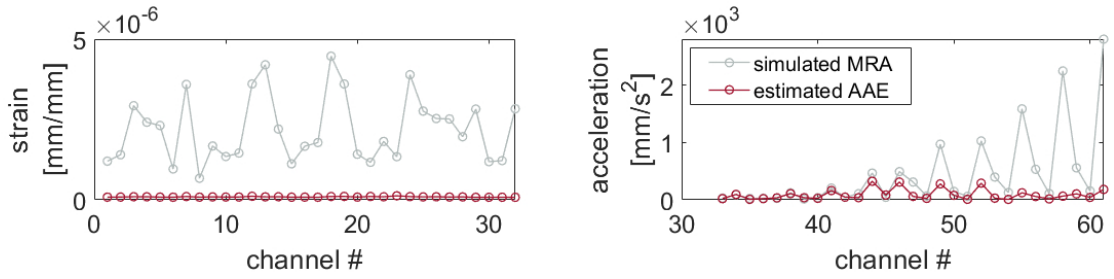
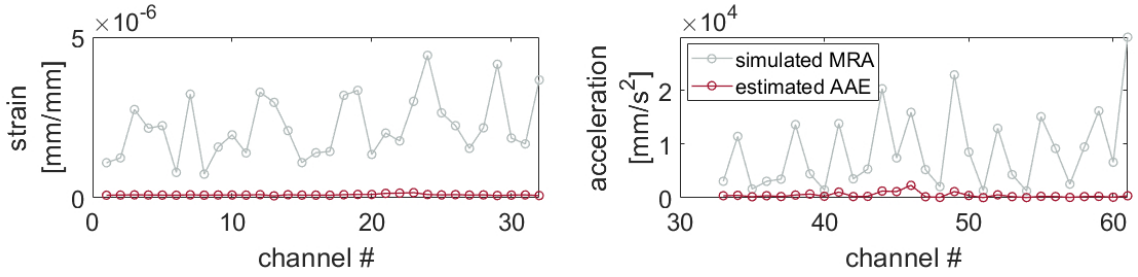


Figure 8.12: Estimated value of degrading E and input with numerical data, for the sine and random load cases, with $\delta_E = 1Hz$, $\omega_{u,2} = 80Hz$ and $\omega_{u,3} = 2000Hz$. The right figures show a zoom of the 0.9s to 1s range.



(a) Sine load case



(b) Random load case

Figure 8.13: Estimated sensors error for the case shown in Figure 8.11.

8.6.4 Parameter and input estimation with non-zero initial conditions

The former results have been obtained with measurements on a model starting at zero initial conditions. In order to demonstrate the robustness of the filter, input and constant Young's modulus are estimated based on a set of measurements with non-zero initial conditions. This is obtained by simulating the system for 2s and discarding the interval from the start up to 1s. The results of the coupled estimation, with the same setting as in the former section, are shown in Figure 8.14. The estimated values show an excellent convergence to the reference ones. The main difference with respect to former cases is in the initial steps, where a less gradual convergence behavior is observed. This can be explained by the fact that the filter is setup with different initial conditions, resulting in a larger initial mismatch that is compensated with quick variations of estimated quantities.

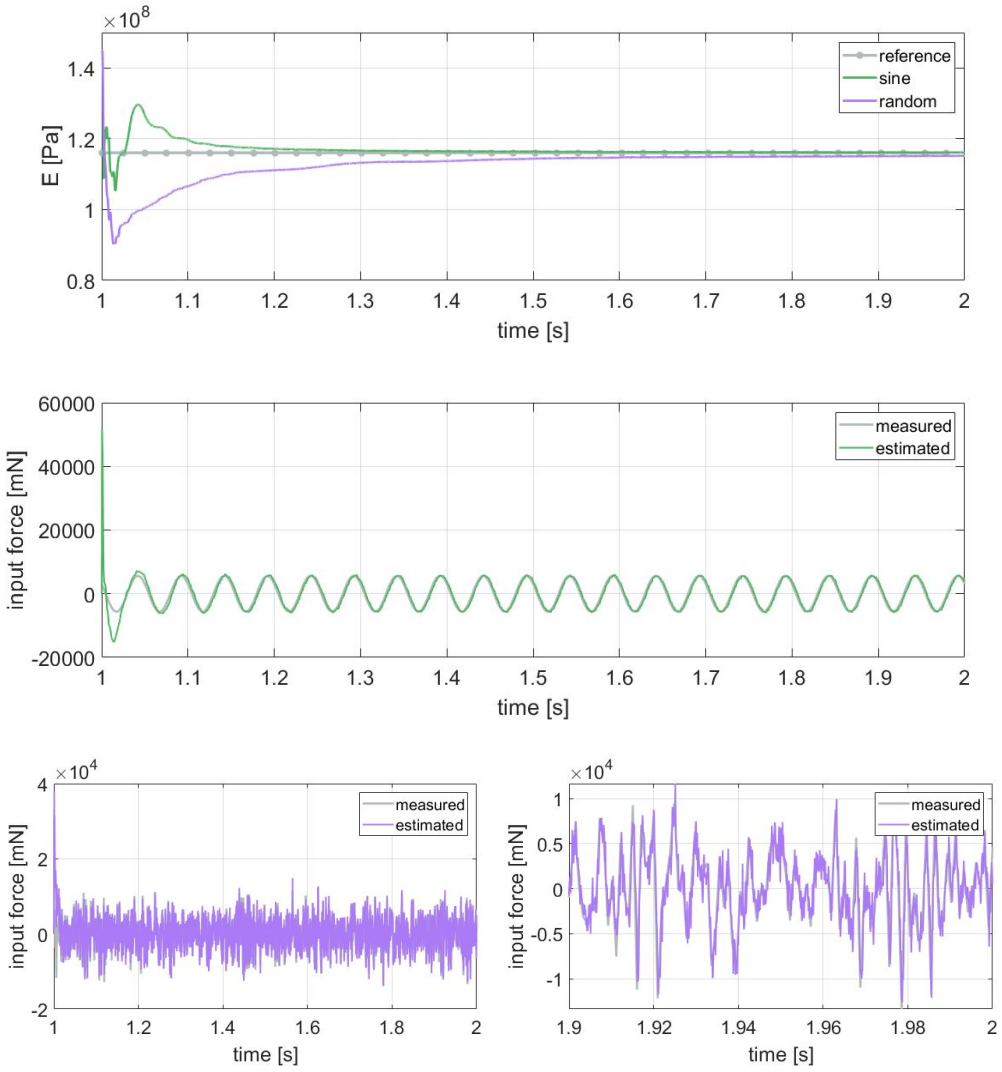


Figure 8.14: Estimated values of E and inputs with numerical data and non-zero initial conditions, for the sine and random load cases, with $\delta_E = 10^{-3}Hz$, $\omega_{u,2} = 80Hz$ and $\omega_{u,3} = 2000Hz$. The right figures show a zoom of the 1.9s to 2s range.

8.7 Experimental validation

In this section, the estimation methodology is validated using experimental data acquired on the experimental setup described in Subsection 3.5.2.

A zero-input acquisition has been done in order to measure the noise level for every sensor. The strain gages and accelerometers as used for data acquisition have noise levels in the order of $10^{-7}m/m$ and $10^{-3}m/s^2$, respectively.

The measurement campaign consisted of separately applying static and dynamic loads to the blade and acquiring data to be used for the estimation. A static load has been applied at the blade tip by suspending a mass of known value $m_s = 1.5Kg$. A 20Hz sine load, following the one used for the numerical validation, has been applied on a point at the middle of the bottom face of the blade by means of an electrodynamic modal shaker. The tip of the shaker was connected to the blade via a mechanical impedance sensor to measure the applied force.

The measurements have been acquired using a sampling period of $1ms$. Since the estimator uses a time step of $0.1ms$, these are linearly interpolated between sampling points in order to match the filter's time step. An overview of the relative errors for the parameter and input estimation using experimental data is shown in Table 8.3 for the different considered load cases. The observed errors are small for most cases, with larger variation ranges when compared with the numerical application cases. The different cases are further discussed below. For each of them, the filter has been setup as described in Subsection 8.5.2.

Table 8.3: Errors on estimated parameters in the last 20% of the considered time interval for experimental measurements

Unknown quantities	**Load**	**Error** [%]			
		Parameter	**min**	**avg**	**max**
E	static	E	0.03	0.04	0.06
	sine	E	1.95	2.00	2.05
E, ρ	static	unobservable			
	sine	E	2.09	3.45	5.01
		ρ	-2.38	-1.00	-0.08
E, input	static	unobservable			
	sine	E	0.46	0.52	0.63

8.7.1 Constant parameter estimation

For the case where the Young's modulus is estimated, with a known density and input, the augmented state covariance value is calculated by choosing $\delta_E = 10^{-3}Hz$. The observed error is close to zero for the static load input and 2% on average for the sine load case. The results are shown in Figure 8.15. The values of the estimated sensors as compared to the measured experimental values display a small error, as shown in Figure 8.16. The main mismatch is present for the sensors mounted closer to the root of the blade, where it is assumed that the boundary conditions have a larger influence on the system dynamics and that the error can thus be attributed to the unmodeled system dynamics related to these boundary conditions. Figure 8.16 includes also the AAE of the simulated measurements generated through a forward simulation with the reference parameter value. These show a similar behavior to estimated ones, proving that the error is not introduced by the filter but it is due to a mismatch in the model.

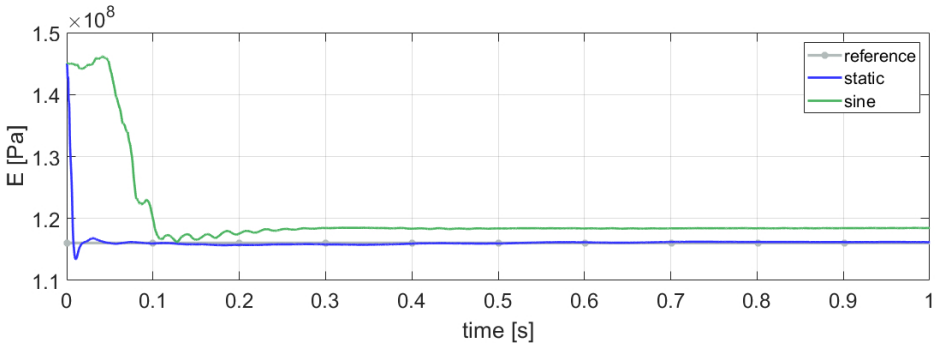
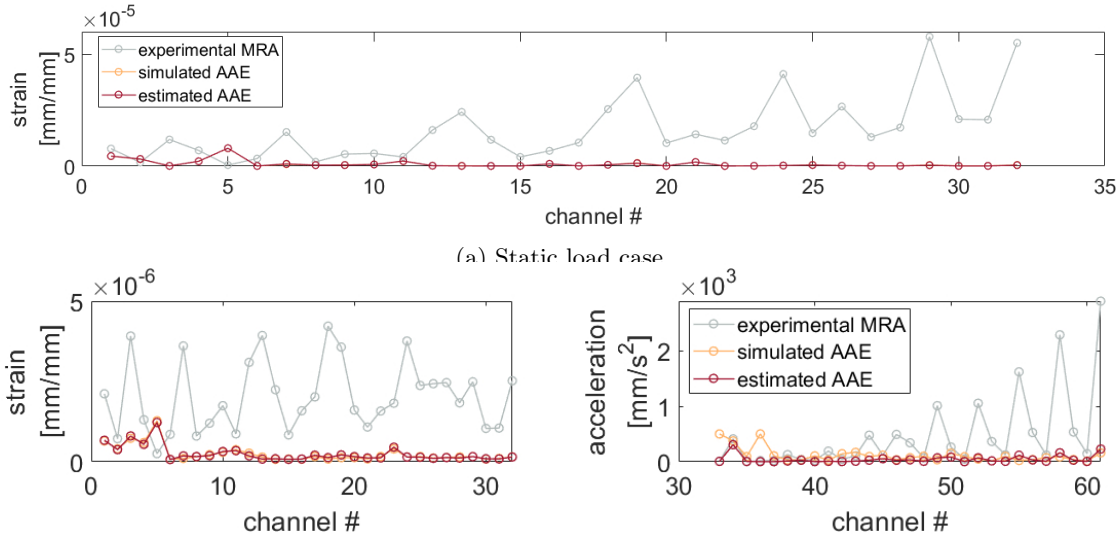


Figure 8.15: Estimated value of E with experimental data, for the static and sine load cases, with $\delta_E = 10^{-3}Hz$.



(a) Static load case

(b) Sine load case

Figure 8.16: Estimated sensor errors for the case shown in Figure 8.15.

For the case where the Young's modulus and density are jointly estimated, with a known input, the augmented state covariance values are calculated by choosing $\delta_p = 1Hz$. The static load case leads to an unobservable system, hence only the sine load case is considered. The observed error has an average value of 1% for density and 3.45% for stiffness, which can be considered acceptable. The results are shown in Figure 8.17. As can be seen in Figure 8.18, the convergence of the ratio between the two parameters is quicker then for the single values. A small error for the full field of sensors can also be observed in this case, as shown in Figure 8.19. The larger required value for $\delta_p = 1Hz$ introduces more noise as well as a low frequency variation of the estimated parameter values. This could be attributed to a small model mismatch. This mismatch is assumed to be filtered out by the smaller value of $\delta_E = 10^{-3}Hz$ in the former case where only the Young's modulus is estimated.

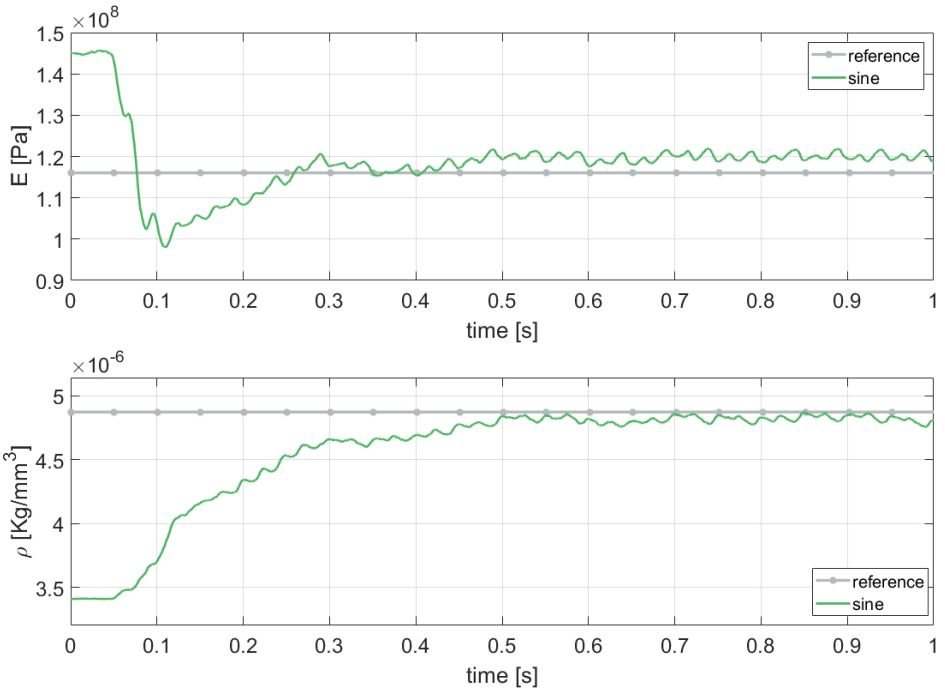


Figure 8.17: Estimated values of E and ρ with experimental data, for the sine load case, with $\delta_E = 1Hz$, $\delta_\rho = 1Hz$.

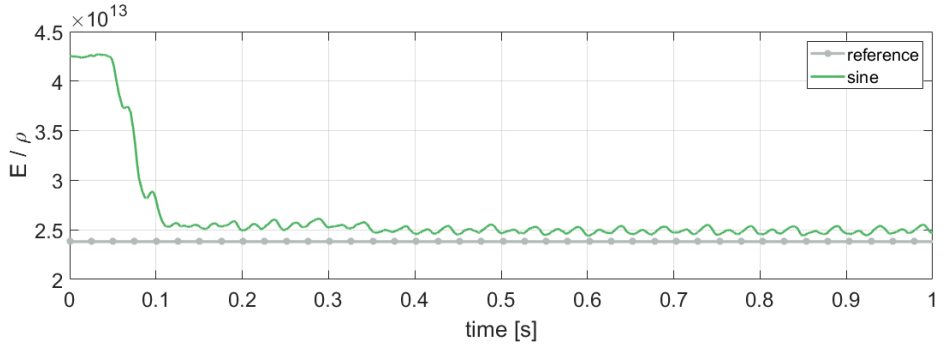


Figure 8.18: Value of the ratio between estimated E and ρ for the case shown in Figure 8.17.

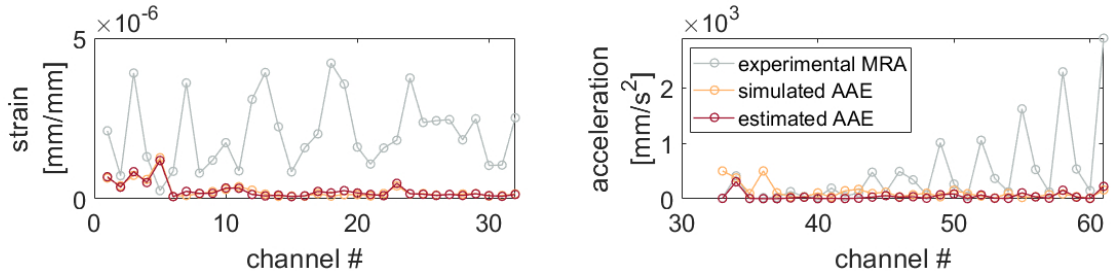


Figure 8.19: Estimated sensors error for the case shown in Figure 8.17.

8.7.2 Parameter and input estimation

For the joint estimation of the Young's modulus and an unknown input, the associated augmented state covariance values are calculated choosing $\delta_E = 10^{-3}Hz$ and $\omega_{u,2} = 80Hz$. As the system is unobservable for a static load, the sine input load case is considered. An average error on the estimated Young's modulus below 1% can be observed, as shown in Figure 8.20. While the full field of strain sensors is accurately estimated, the acceleration estimated sensors exhibit a larger error as compared to the previous cases Figure 8.21. As explained above for the numerical case, this could be attributed to the high frequency load content introduced by the filter due to the larger augmented state covariance values.

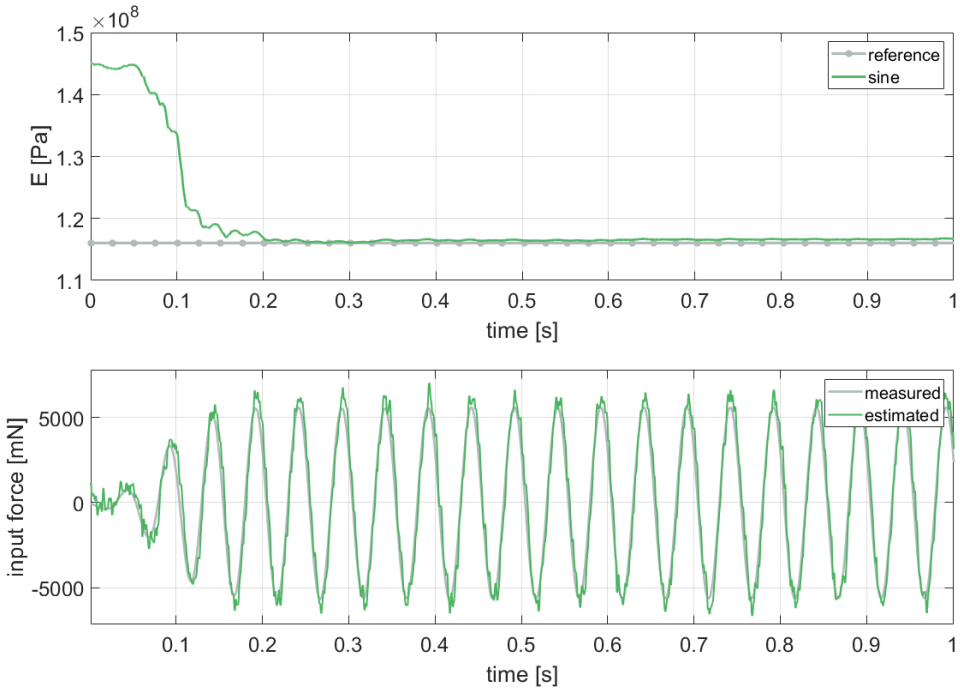


Figure 8.20: Estimated values of E and input with experimental data, for the sine load case, with $\delta_E = 10^{-3}Hz$ and $\omega_{u,2} = 80Hz$.

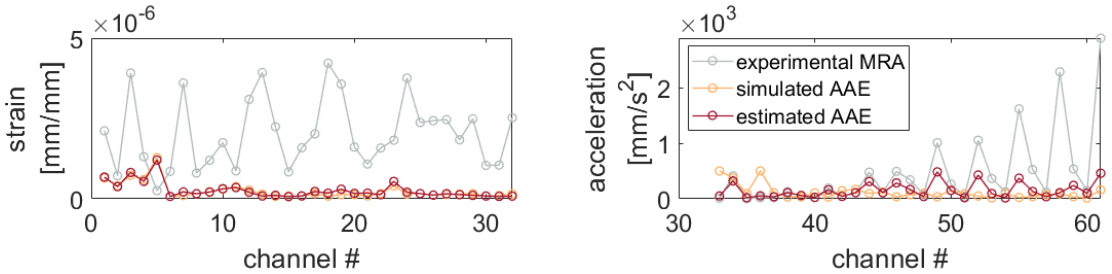


Figure 8.21: Estimated sensors error for the case shown in Figure 8.20.

8.8 Conclusions and possible future work

In this chapter, the integration of the proposed pROM methodology in an augmented EKF has been presented, allowing for the joint estimation of states, inputs and material parameters of an industrial scale structural dynamic system. The integration of the pROM in the EKF algorithm enables the efficient parameter estimation.

An observability analysis showed that in a general case a number of position-level measurements at least equal to the number of unknown inputs and parameters is required. Furthermore, several considerations on the observability of specific combinations of unknown mass and stiffness parameters and inputs have been discussed.

The proposed methodology has been applied to a scaled wind turbine blade. Estimation of parameters, inputs and states on both numerical and experimental data showed that the proposed approach is able to correctly estimate the unknown quantities together with the full field of strains and accelerations using a small set of strain sensors. Practical considerations on the setup of the filter have been discussed, with a focus on the augmented state covariance values; these showed how using an adaptive scheme for the selection of covariance values would be helpful and should be explored in future research.

Future efforts will be aimed at extending the estimation to lumped properties and anisotropic constitutive material parameters, as well as developing an extended OSP approach that considers the augmented states related to the material parameters. The use of accelerometers to improve the bandwidth of the filter should be investigated. It will also be of interest to analyze the computational load of the algorithm in view of real-time applications for condition monitoring.

The estimation of states, inputs and material parameters will be extended to system-level applications in the next chapter.

Chapter 9

Estimation of states, inputs and parameters for Flexible Multibody systems

9.1 Introduction

The estimation framework presented in the previous chapter applies only to structural components. State estimation for FMB systems is of great interest but it has been explored in a much more limited fashion in literature (see [111] for an overview of MB KF-based methodologies). The main reason is that the classical KF and its extensions are formulated for systems modeled through a set of ODEs, while FMB systems are modeled by sets of DAEs (as seen in Chapter 2). The presence of algebraic equality constraints causes the arisal of important issues to be addressed. The exact satisfaction of constraints should in fact be guaranteed, ensuring that kinematic properties of the mechanism are preserved. Constraint violations could also hinder the convergence of non-linear solvers employed in the time integration step. While it is possible to tackle this challenge by operating a split between dependent and independent coordinates, their selection is not always straightforward. The distinction can in fact vary during the simulation, for example in singular kinematic configurations.

Several approaches for constraint enforcement have been summarized in works by Simon et al. [130, 129]. One of these is the Perfect Measurements approach, consisting in modeling the constraint equations as as an additional set of measurements provided to the filter [8]. While simple to implement, this approach requires an accurate tuning of covariance values for the additional measurements and also does not guarantee exact constraints enforcement. Simon et al. also propose constraint elimination methods based on the limiting assumptions of linear constraints and that no change in minimal coordinates occurs. A penalty method approach has been proposed by Risaliti et al. [119] in order to reduce the DAEs to a set of ODEs and increase accuracy by using an implicit numerical scheme for the KF. As for the previous discussed approach, also the penalty method does not allow to exactly enforce the kinematic equality constraints. An interesting approach that has been proposed by Sanjurjo et al. [122] consists in the elimination of constraints by projection. While satisfying constraints exactly, this method has the main downsides of not being able to switch between different independent coordinates sets and only estimating a subset of the system coordinates. This creates an unnatural and somewhat arbitrary hierarchy in the selection of states.

This work employs a formulation recently proposed by Tamarozzi et al. [137], the Augmented MANifold Differential-Algebraic Extended Kalman Filter (AMANDA-EKF). This approach tackles the issues of independent coordinates selection and constraints enforcement by employing projected EOMs and an additional manifold optimization step. The proposed approach results in a general framework that ensures exact constraints enforcement.

The already challenging field of state estimation for MB models acquires further complications when concurrent estimation of unknown quantities is considered. The concurrent estimation of states and inputs has been addressed in literature in e.g. [137, 14, 119, 122]. As noted by Naya et al. [111], the estimation of states and parameters [25, 88, 110] or the three combined [120] has been addressed by a much more limited number of works in literature. This is mainly caused by the difficulty in defining the explicit dependency of measurements and states on parameters, which is needed for the Jacobians computation. The cited works focus on the estimation of inertial, geometrical or lumped parameters that are explicit in the EOMs [25, 120, 110], or employ fitting methods to approximate such dependency [88]. To the author's knowledge, the estimation of parameters relative to flexible components has not been addressed in literature.

This chapter proposes to leverage the explicit dependency on material parameters of pFMB models to enable their estimation. The original AMANDA-EKF methodology is extended by employing pFMB models, resulting in a methodology for the joint estimation of states, inputs and material parameters of flexible components for FMB models.

In Section 9.2 the AMANDA framework and its application to pFMB models is described. Few considerations on observability and use of strain sensors are discussed in Section 9.3. The proposed methodology is validated numerically, on both close-chain and open-chain kinematic models, in Section 9.4. The results of an experimental validation on the Robolink manipulator are then shown in Section 9.5. Finally, conclusions and discussions on limitations and future work are given in Section 9.6.

9.2 The AMANDA-EKF framework

The AMANDA framework has been originally proposed by Tamarozzi et al. [137] as a general estimation methodology for MB systems. Four main targets have been identified:

- **Generality:** the estimation should be possible on all types of MB systems (e.g. rigid and flexible, open and closed-chain kinematics, etc.) and augmented states;
- **Exact constraint satisfaction:** the set of algebraic constraints (Equation 2.46) should be satisfied exactly (on position level) at each estimation step;
- **Full states estimation:** the full state of the system should be estimated without reduction to minimal coordinates;
- **Robustness:** the convergence of numerical integrators at each time step should be facilitated.

Starting from these main goals, the methodology has been defined employing the same base methodologies as in the previous chapters of this work. The GGL formulation (Equation 2.60) of the EOMs is employed and discretized through a first-order BDF (Equation 2.61). An augmented EKF (Subsection 7.3.2) is used for estimation.

The original contribution shows the joint estimation of states and inputs. This work extends the methodology to estimate also material parameters of flexible bodies by employing the pFMB models developed in Chapter 4. The rest of this section will present the AMANDA methodology and its integration with pFMB models.

9.2.1 Augmentation and projection of the EOMs

The FMB EOMs in index-2 form of Equation 2.60 are first augmented in order to account also for the time evolution of unknown inputs and parameters. These are represented, as in the previous chapter, by a zeroth-order random walk model. Furthermore, the EOMs keep into account the parametric dependency of pFMB models. The parametric augmented EOMs result then in:

$$\begin{cases} \dot{\mathbf{q}} - \mathbf{v} - \boldsymbol{\Phi}_q^T(\mathbf{q})\boldsymbol{\mu} = \mathbf{0} \\ \mathbf{M}(\mathbf{q},\mathbf{p})\dot{\mathbf{v}} + \mathbf{C}(\mathbf{p})\mathbf{v} + \mathbf{K}(\mathbf{p})\mathbf{q} + \boldsymbol{\Phi}_q^T(\mathbf{q})\boldsymbol{\lambda} - \mathbf{f}_e(\mathbf{q},\mathbf{v}) - \mathbf{f}_v(\mathbf{q},\mathbf{v},\mathbf{p}) = \mathbf{0} \\ \dot{\mathbf{u}}^{un} = \mathbf{0} \\ \dot{\mathbf{p}} = \mathbf{0} \\ \boldsymbol{\Phi}_q^T(\mathbf{q})\mathbf{v} = \mathbf{0} \\ \boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0} \end{cases} \tag{9.1}$$

These equations are solved in the prediction step of the EKF by using an integration scheme to advance the full states. The direct use of the equations in the correction step would require the evaluation of the Jacobian matrix for the full states of the system, including Lagrange multipliers $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$. The derivatives with respect to Lagrange multipliers typically result in non-trivial and complicated operations. The AMANDA methodology proposes to eliminate the Lagrange multipliers from the EOMs through a nullspace projection [137, 23, 141, 156] and then used the projected EOMs for an exact a-posteriori states correction. The nullspace $\boldsymbol{\mathcal{N}}(\mathbf{q})$ is defined as:

$$\boldsymbol{\mathcal{N}}(\mathbf{q})\boldsymbol{\Phi}_q^T(\mathbf{q}) = \mathbf{0} \tag{9.2}$$

The procedure for the computation of the nullspace is the one proposed by Yen and Petzold [156]. This is based on a coordinates partitioning that allows for cheap and and efficient computation of the nullspace and its derivatives. The constraint equations exhibit no dependency on the parameters considered in this work, both given their nature and thanks to the global reduction basis employed for component-level pROMs. This in turn results in also the nullspace not having any dependency on parameters and not requiring additional derivative terms that would complicate the process.

The nullspace projection enforces the solution of EOMs to lie on the constraint manifold. This in turn results in the elimination of the Lagrange multiplier terms. The projected EOMs used in the Kalman update result then in:

$$\begin{cases} \mathbf{N}(\mathbf{q})\left(\dot{\mathbf{q}} - \mathbf{v}\right) = \mathbf{0} \\ \mathbf{N}(\mathbf{q})\left(\mathbf{M}(\mathbf{q},\mathbf{p})\dot{\mathbf{v}} + \mathbf{C}(\mathbf{p})\mathbf{v} + \mathbf{K}(\mathbf{p})\mathbf{q} - \mathbf{f}_e(\mathbf{q},\mathbf{v}) - \mathbf{f}_v(\mathbf{q},\mathbf{v},\mathbf{p})\right) = \mathbf{0} \\ \dot{\mathbf{u}}^{un} = \mathbf{0} \\ \dot{\mathbf{p}} = \mathbf{0} \\ \boldsymbol{\Phi}_q^T(\mathbf{q})\mathbf{v} = \mathbf{0} \\ \boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0} \end{cases} \tag{9.3}$$

The augmented state can now be defined as:

$$\mathbf{x}^* = \begin{bmatrix} \mathbf{q} \\ \mathbf{v} \\ \mathbf{u}^{un} \\ \mathbf{p} \end{bmatrix} \tag{9.4}$$

In turn, Equation 9.3 can be discretized and expressed as an implicit equation:

$$\mathbf{g}_d(\mathbf{x}_k^*, \dot{\mathbf{x}}_k^*) = \mathbf{0} \tag{9.5}$$

where the first order BDF integration scheme is employed and reported here for clarity:

$$\dot{\mathbf{x}}_k^* = \frac{1}{\Delta t}(\mathbf{x}_k^* - \mathbf{x}_{k-1}^*) \tag{9.6}$$

The Jacobian of the EOMs $\mathbf{J} = \frac{d\mathbf{x}*_k}{d\mathbf{x}*_{k-1}}$ is required for the EKF scheme. This can be retrieved by first computing the total derivative of Equation 9.5 keeping Equation 9.6 into account:

$$\frac{\partial \mathbf{g}_d}{\partial \mathbf{x}_{k-1}^*} = \frac{\partial \mathbf{g}_d}{\partial \mathbf{x}_k^*}\frac{\partial \mathbf{x}_k^*}{\partial \mathbf{x}_{k-1}^*} + \frac{\partial \mathbf{g}_d}{\partial \dot{\mathbf{x}}_k^*}\frac{\partial \dot{\mathbf{x}}_k^*}{\partial \mathbf{x}_k^*}\frac{\partial \mathbf{x}_k^*}{\partial \mathbf{x}_{k-1}^*} + \frac{\partial \mathbf{g}_d}{\partial \dot{\mathbf{x}}_k^*}\frac{\partial \dot{\mathbf{x}}_k^*}{\partial \mathbf{x}_{k-1}^*} = \mathbf{0} \tag{9.7}$$

from which the needed term can be explicitated as:

$$\mathbf{J} = \frac{\partial \dot{\mathbf{x}}_k^*}{\partial \mathbf{x}_{k-1}^*} = \left(\frac{\partial \mathbf{g}_d}{\partial \mathbf{x}_k^*} + \frac{1}{\Delta t}\frac{\partial \mathbf{g}_d}{\partial \dot{\mathbf{x}}_k^*}\right)\frac{1}{\Delta t}\frac{\partial \mathbf{g}_d}{\partial \dot{\mathbf{x}}_k^*} \tag{9.8}$$

Computation of the terms appearing in the former equations require the derivatives of the nullspace with respect to the states, that can be computed using the previously mentioned method. No need for computation of derivatives of the nullspace with respect to parameters arises thanks to the previously mentioned properties of the nullspace. The derivatives of the single components of the EOMs with respect to parameters are instead required. In particular, through the chain rule of derivation this is reduced to assembling the derivatives of the flexible matrices of the components $\mathbf{M}_{f_r}^i$, $\mathbf{C}_{f_r}^i$ and $\mathbf{K}_{f_r}^i$. These can be easily computed as explained in the previous chapter.

9.2.2 Manifold optimization

The prediction step of the AMANDA-EKF for the evaluation of the a-priori states $\mathbf{x}_k^{*-}$ is done using Lagrange multipliers and the order-1 BDF integration scheme [137] (as explained in Chapter 2). In this first step, the constraints are enforced through the solver algorithm. In the correction step instead, the constraints are not imposed. The a-posteriori estimated states $\mathbf{x}_k^{*+}$ could then results in a constraints violation (i.e. $\Phi(\mathbf{q}_k^+) \neq \mathbf{0}$).

The AMANDA framework proposes an additional step, the so-called AMANDA step, to the EKF algorithm. This consists in finding a solution $\hat{\mathbf{x}}_k^{*+}$ that satisfies the constraints and is at the same time as close as possible to $\mathbf{x}_k^{*+}$. The constraints equation only depends on position-level coordinates $\mathbf{q}$. These are then optimized in the main part of the AMANDA step, and velocities are corrected accordingly after the optimal solution is found.

The distance between states is defined in this scope as the Euclidean distance. This translates in a single-objective optimization problem defined as follows:

$$\begin{aligned} \min_{\hat{\mathbf{q}}_k^+} \quad & \|\hat{\mathbf{q}}_k^+ - \mathbf{q}_k^+\| \\ s.t. \quad & \mathbf{\Phi}(\hat{\mathbf{q}}_k^+) = \mathbf{0} \end{aligned} \tag{9.9}$$

An Euclidean gradient descent approach [57] is employed for the solution of the optimization problem. The objective function is defined as:

$$f^{obj} = (\hat{\mathbf{q}}_k^+ - \mathbf{q}_k^+)^2 \tag{9.10}$$

for which the gradient is:

$$\frac{df^{obj}}{d\hat{\mathbf{q}}_k^+} = 2(\hat{\mathbf{q}}_k^+ - \mathbf{q}_k^+) \tag{9.11}$$

The descent step is projected through the nullspace evaluated in the current proposed solution in order to guarantee that the states evolve on the constrains manifold. The correction step then becomes:

$$\boldsymbol{\Delta}(\hat{\mathbf{q}}_k^+) = -\mathbf{N}^T(\hat{\mathbf{q}}_k^+)\mathbf{N}(\hat{\mathbf{q}}_k^+)\frac{df^{obj}}{d\hat{\mathbf{q}}_k^+} \tag{9.12}$$

Finally, at the i-th iteration the solution is updated as:

$$\begin{gathered}\left(\hat{\mathbf{q}}_k^+\right)^{i+1} = \left(\hat{\mathbf{q}}_k^+\right)^i + \boldsymbol{\Delta}^i = \\ \left(\hat{\mathbf{q}}_k^+\right)^i - 2\mathbf{N}^T\left(\hat{\mathbf{q}}_k^+\right)^i \mathbf{N}\left(\hat{\mathbf{q}}_k^+\right)^i \left(\left(\hat{\mathbf{q}}_k^+\right)^i - \mathbf{q}_k^+\right)\end{gathered} \tag{9.13}$$

The a-priori estimate for the current step is employed as the first tentative solution, so that $(\hat{\mathbf{q}}_k^+)^0 = \mathbf{q}_k^-$. The choice aims at guaranteeing that the optimization starts from a configuration lying on the constraints manifold.

The manifold optimization enforces the descent to evolve on the tangent constraint manifold. In the general case, this could still result in a state $(\hat{\mathbf{q}}_k^+)^i$ that does not satisfy the constraints. If required, the proposed solution at each step is then updated by a kinematic assembly procedure [73]. This consists in evaluating and solving the constraint equation $\boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0}$ starting from the current proposed solution.

After the convergence of the optimization problem, velocities $\hat{\mathbf{v}}_k^+$ are computed solving the EOMs with the converged position states. The augmented states are not updated in the AMANDA step.

Finally, the AMANDA step can be summarized with the following operator:

$$\begin{bmatrix}\hat{\mathbf{q}}_k^+ \\ \hat{\mathbf{v}}_k^+\end{bmatrix} = \mathcal{A}(\mathbf{q}_k^+, \mathbf{q}_k^-) \tag{9.14}$$

9.2.3 AMANDA-EKF algorithm

The main benefits of using pFMB models in the AMANDA-EKF framework are summarized in this section, and then the whole algorithm is presented.

The explicit dependency of pFMB models on parameters enables the efficient and exact derivation of the necessary Jacobians. Furthermore, it allows to efficiently execute the update required in the KF correction step after the estimated parameter values are evaluated. Finally, the constant reduction basis and the kinds of parameters considered result in a nullspace not dependent on parameters. This allows to not modify the AMANDA step with respect to the original methodology.

The full AMANDA-EKF algorithm can be summarized in the following steps:

- **Prediction**: the a-priori states $\mathbf{x}_k^{*-}$ and measurements $\mathbf{y}_k^-$ predictions are directly obtained from the evaluation of the model:

$$\mathbf{g}_d(\mathbf{x}_k^*, \mathbf{x}_{k-1}^*, \mathbf{u}_k^{kn}) = \mathbf{0} \tag{9.15}$$

$$\mathbf{y}_k^- = \mathbf{h}_d(\mathbf{x}_k^*, \mathbf{u}_k^{kn}) \tag{9.16}$$

  The current nullspace and its derivatives are calculated, and used to compute the Jacobian of the system around the current a-posteriori estimate from the previous step:

$$\mathbf{J}_{k-1} = \left.\frac{\partial \mathbf{x}_k^*}{\partial \mathbf{x}_{k-1}^*}\right|_{\mathbf{x}_{k-1}^{*+}, \mathbf{x}_k^{*-}} \tag{9.17}$$

The states error covariance is then propagated as:

$$\mathbf{P}_k^- = \mathbf{J}_{k-1}\mathbf{P}_{k-1}^+\mathbf{J}_{k-1}^T + \mathbf{Q}_k \tag{9.18}$$

- **Correction**: Also the measurement equation is linearized by evaluating its Jacobian around the current a-priori estimate:

$$\mathbf{J}_{m,k} = \left.\frac{\partial \mathbf{h}(\mathbf{x}_k^*, \mathbf{u}_k^{kn})}{\partial \mathbf{x}_k^*}\right|_{\mathbf{x}_k^{*-},} \tag{9.19}$$

The Kalman gain is then computed as:

$$\boldsymbol{\mathcal{K}}_k = \mathbf{P}_k^-\mathbf{J}_{m,k}^T(\mathbf{J}_{m,k}\mathbf{P}_k^-\mathbf{J}_{m,k}^T + \mathbf{R}_k^+)^-1 \tag{9.20}$$

The states are then corrected:

$$\mathbf{x}_k^{*+} = \mathbf{x}_k^{*-} + \boldsymbol{\mathcal{K}}_k(\mathbf{y}_k - \mathbf{y}_k^-) \tag{9.21}$$

Finally, the states error covariance is also corrected using linearized matrices:

$$\mathbf{P}_k^+ = (\mathbf{I} - \boldsymbol{\mathcal{K}}_k\mathbf{J}_{m,k})\mathbf{P}_k^-(\mathbf{I} - \boldsymbol{\mathcal{K}}_k\mathbf{J}_{m,k})^T + \boldsymbol{\mathcal{K}}_k\mathbf{R}_k\boldsymbol{\mathcal{K}}_k^T \tag{9.22}$$

- **AMANDA step**: the position states are corrected to satisfy constraints, and velocities are updated accordingly:

$$\begin{bmatrix}\hat{\mathbf{q}}_k^+ \\ \hat{\mathbf{v}}_k^+\end{bmatrix} = \mathcal{A}(\mathbf{q}_k^+, \mathbf{q}_k^-) \tag{9.23}$$

9.3 Practical considerations on sensors selection and handling in system-level estimation

The choice of kinds, numbers and positions of sensors for estimation applications is of paramount importance. The difference between different sensors sets in terms of number and measured quantities has been discussed in the previous chapter for structural estimation applications. This section discusses the sensors choice for estimation on MB systems. The use of strain measurements is discussed in detail with few practical considerations required to employ such measurements when systems evolve under gravity.

9.3.1 Observability

It has been shown in Section 8.4 how the PBH observability criterion allows for an analytical discussion on number and quantity of sensors required for estimation of inputs and parameters. Such analytical approach in the MB would require the linearization of the system and construction of much more complex matrices. This would in turn result in a more complicated problem to be discussed analytically by inspection of such matrices, as done for the structural case. While such effort would be of interest, it does not fall under the main focus of this work. An alternative approach proposed in literature is the numerical analysis of the observability matrices (i.e. [7, 117]). Such approach allows to evaluate how the observability changes during an estimation for a defined set of sensors, but cannot be directly used for generic considerations on sensors choice.

While no rigorous analytical results are available, few considerations can still be made on observability for FMB sensing based on analogies with the results shown in Section 8.4, intuition and experience. A rigorous observability analysis should be the focus of future research efforts.

A first consideration regards sensitivities of sensors with respect to the quantities to be estimated. A sensor on a body could in fact not be influenced by parameters or inputs on a different component. One example of such behavior arises when open-chain kinematics is considered. Also if the system consists of multiple sub-mechanisms, the coupling between them could be weak or only relative to some of the components. The choice of adequate sensors for MB application requires then to first carefully consider which components to sensor.

Regarding the sensor type selection, the first consideration is that the only use of acceleration measurements would also in this case not allow for observability of the augmented states. Such measurements do not in fact contain any information on the static behavior of the system, resulting in only the dynamic behavior being matched and probably a drifting in the estimates (see e.g. [108]). A number of absolute position-level measurements at least equal to the number of augmented states is then still required.

In Section 8.4 it has been discussed how, for structural applications, position and strain measurements are equivalent in terms of contained information. This is not the case anymore when FMB systems are considered. Strain measurements are in fact only dependent on the flexible coordinates of a body, while position measurements depend both on flexible and rigid-body coordinates. Only position-level measurements can then be regarded as absolute and ensure observability. Strain measurements generally result in relative measurements and hence they need to be complemented with additional information in order to guarantee observability.

In some particular application cases, the rigid DOFs of the system are fully defined leaving as independent coordinates only the flexible ones. Such is the case when the rigid DOFs are actuated and the configuration imposed by the actuators at each time step is known. The actuators are in fact commonly modeled as an extra set of time-varying constraints. The manipulator applications described in Chapter 4 are an example of such case, where the time profiles of each revolute driver are known and directly imposed. In these cases the strain measurements can be considered as absolute measurements, given that information on the rigid coordinates is fully available and does not need to be estimated. For such applications then, use of strain measurements alone allows to fulfill the observability requirements.

It is remarked how the considerations presented in this section only represent guidelines arising from experience and intuition. A rigorous observability analysis is one of the most interesting developments to be considered for future efforts.

9.3.2 Estimation of parameters and inputs of systems under gravity employing strain measurements

Particular considerations are required when strain gages are used. Few practical considerations on their use have been discussed in Subsection 4.5.2. The installation of such sensors consists in physically gluing them on the surface of the body, resulting in a complex and delicate process. It is common for mechanisms to remove the interested components, when possible, and only assemble them again when fully sensored. Strain gages tend to exhibit a high sensitivity to temperature, thus the need for a preliminary stabilization phase before every acquisition aimed at minimizing measurements drift. After this, a referencing operation allows to identify the reference strain assumed as the zero. This is commonly done in a zero-input reference configuration. The effect of gravity on this referencing procedure is commonly neglected, given that then in operations the loads applied generate strains with higher orders of magnitude.

For particular applications, e.g. a robotic manipulator performing a maneuver under gravity

and no additional payload, the effect of gravity on measurements cannot be neglected. The strain measurement equation of Equation 2.57 represents an absolute measurement that could be replicated in reality only if the referencing of sensors could be done in an undeformed conditions. This results unfeasible given that it would in most of the applications require the disassembly of the system.

The strain measurement equation can be adapted to account for the reference configuration by adding a bias:

$$h_s(t) = \mathbf{N}_s \mathbf{q}^i_{f_r}(t) - h_s^{ref} = \mathbf{N}_s(\mathbf{q}^i_{f_r}(t) - \mathbf{q}^{i,ref}_{f_r}) \tag{9.24}$$

in which h_s^{ref} is the strain in the reference configuration $\mathbf{q}^{i,ref}_{f_r}$. For a forward simulation, h_s^{ref} can be computed by a static solution (i.e. solving the EOMs by imposing zero velocity and accelerations and reference driver values).

An additional issue arises in parameters are estimated. The reference measurement depends in fact on the parameters of the system (i.e. $h_s^{ref} = h_s^{ref}(\mathbf{p})$) and cannot be computed a-priori of the estimation procedure. Computing the reference strain employing the first tentative value used for the parameters would in fact result in a wrong measurement offset and in turn a wrong estimate of the parameter. This work proposes to employ an adaptive approach that consists in updating the reference value $h_s^{ref}(\mathbf{p})$ at each time step solving the static analysis on the model updated with current estimates. The adaptive strain measurement results in:

$$h_s(t) = \mathbf{N}_s \mathbf{q}^i_{f_r}(t) - h_s^{ref}(\mathbf{p}) = \mathbf{N}_s(\mathbf{q}^i_{f_r}(t) - \mathbf{q}^{i,ref}_{f_r}(\mathbf{p})) \tag{9.25}$$

The main assumption made to enable this approach is that the real parameter does not vary in time, so that the current estimate can be assumed equal to the value during the referencing procedure. If the assumption does not hold, the reference configuration is not known and a different kind of measurements should be employed. It is also assumed that inputs during the referencing procedure are fully known. In most of the cases in fact such procedure is done under gravity load only.

One main drawback is that the system loses observability in the reference configuration. Relative strains will in fact always be zero in this case, for any value of parameters, resulting in a zero sensitivity. For applications in which the system passes from the reference configuration during estimation, the stability of the filter can be hindered. This can be solved by enforcing the parameters to not be updated in the Kalman step if the configuration is close to the unobservable one.

It should be noted how the proposed approach adds a reduced computational effort of computing a static solution at each estimation step. The set of implicit algebraic equations resulting from imposing static conditions can be solved efficiently with any non-linear solver (i.e. Newton-Raphson). Further efficiency is ensured by the possibility of pre-computing the terms of the equations and employing the previous solution as tentative at the start of the algorithms.

9.4 Numerical validation

A first validation of the proposed estimation methodology is done based on numerical data. The two systems presented in Chapter 4 are employed, demonstrating the applicability of the methodology to both open and closed-chain kinematics. Virtual measurements are generated by forward simulations and then polluted with white Gaussian noise, as previously done for structural applications in Chapter 8.

As previously discussed, covariance setups are highly dependent on the application case. The covariance values for augmented states have been selected based on an analysis equivalent to the

one shown in Section 8.5. Given the similar behaviors and conclusions, results for this analysis are not shown in this chapter. The values employed are represented by ω_u or δ_p, which are used to calculate the covariance values by applying Equation 8.30 or Equation 8.31.

The relative adaptive strain measurements formulation presented in Subsection 9.3.2 is employed when parameters constant in time are estimated. For the case in which the parameter is evolving in time, such approach cannot be employed, as previously discussed.

9.4.1 Case study n.1: a four-bar linkage

In a first numerical validation, the system considered is the four-bar linkage mechanism presented in Subsection 4.4.1. The closed-chain kinematic chain implies a strong coupling between components behavior. The states of each body are in fact highly sensitive to states, inputs and parameter of the other components. In practical applications, such sensitivity is desired given that commonly only a reduced set of components is accessible for sensoring. The considered mechanism allows then for interesting estimation applications in which states, inputs and parameters of the entire system are estimated based on measurements on a subset of components.

In the next section the application cases considered are presented. Then the effectiveness of the proposed methodology is numerically validated on different combinations of estimated parameters and inputs.

9.4.1.1 Application cases for estimation

Three applications are considered for validation:

- **Parameter estimation**: the Young's moduli of IC, C and OC are jointly estimated;
- **Parameter and input estimation**: the Young's modulus of C is estimated together with an input on OC;
- **Degrading parameter and input estimation**: the same quantities as the previous application case are estimated, but the Young's modulus of IC is degrading in time.

In every application case, the system is actuated through the same driver profile shown in Figure 4.4. Gravity is acting on the system. For the second and third applications, a sinusoidal load with 15 Hz frequency and 10N amplitude acts on the center of mass of OC in the global z direction (Figure 9.1). Furthermore, in the last application case the Young's modulus of C decreases at a rate of 10% of the initial value per second.

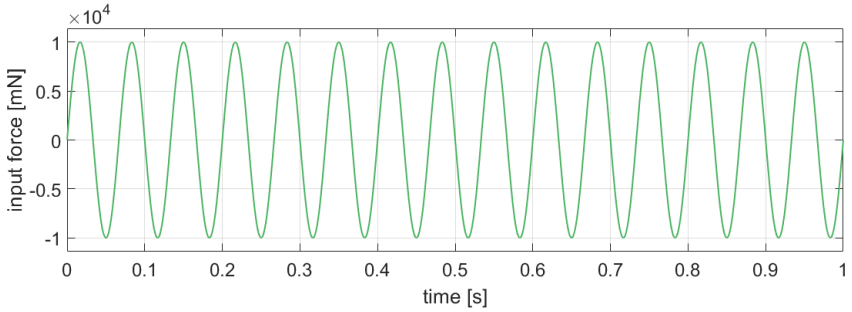


Figure 9.1: 15Hz sine input

For each application, two sensor setups are employed and the estimation performances are compared. In a first ideal case, every components is sensored with one virtual strain gage (*set*

1). A second case is chosen to reflect the practical scenario in which only a part of the system is accessible and sensored. In this second case, only IC is sensored with 3 strain gages (*set 2*).

The sensors employed and the input application points and direction are shown in Figure 9.2.

The numerical measurements used for estimation are generated by simulations using a time step $\Delta t = 1ms$. These are then polluted by adding a white noise in the order of $10^{-7}mm/mm$.

An overview of the performance for all case in terms of parameter estimation is given in Table 9.1.

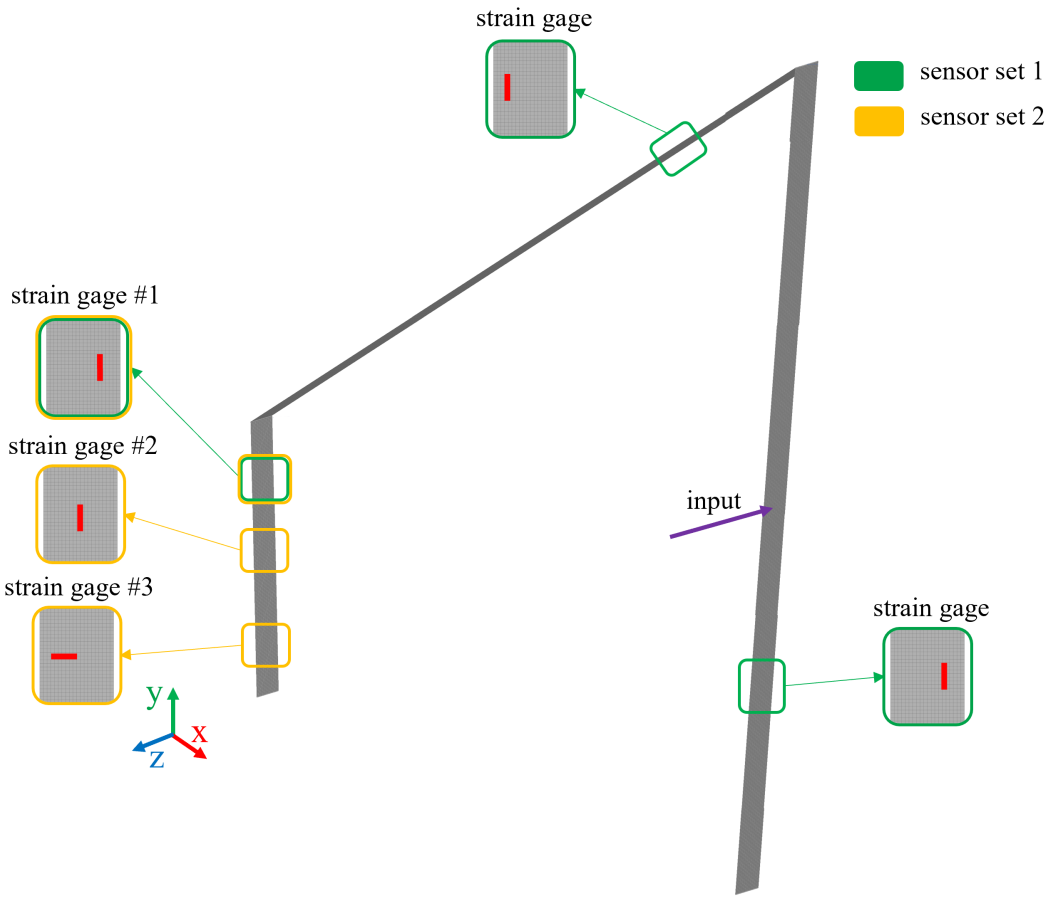


Figure 9.2: Sensors and inputs used for estimation on the four-bar linkage model

Table 9.1: Errors on estimated parameters in last 20% of time interval for the four-bar application case.

Unknown quantities	**Sensors set**	**Error** [%]			
		Parameter	**min**	**avg**	**max**
E_{IC}, E_C, E_{OC}	1	E_{IC}	0.01	0.02	0.03
		E_C	-0.04	-0.05	-0.06
		E_{OC}	0.01	0.03	0.05
	2	E_{IC}	-0.02	-0.02	-0.02
		E_C	-0.07	-0.08	-0.10
		E_{OC}	0.94	0.96	0.98
E_C, input	1	E_C	-0.04	-0.05	-0.07
	2	E_C	0.06	0.07	0.08
degrading E_C, input	1	E_C	0.50	1.11	1.87
	2	E_C	0.97	1.67	2.74

9.4.1.2 Parameter estimation

As a first application case, the Young's moduli of all components are estimated for the actuated mechanism under gravity. The initial values for the parameters are 25% higher than true values, and their covariances are calculated with $\delta_E = 10^{-3}Hz$.

The evolution of estimated parameters and measurements for the ideal sensors set is shown in Figure 9.3. The estimated parameters converge quickly to the correct values, remaining stable after convergence.

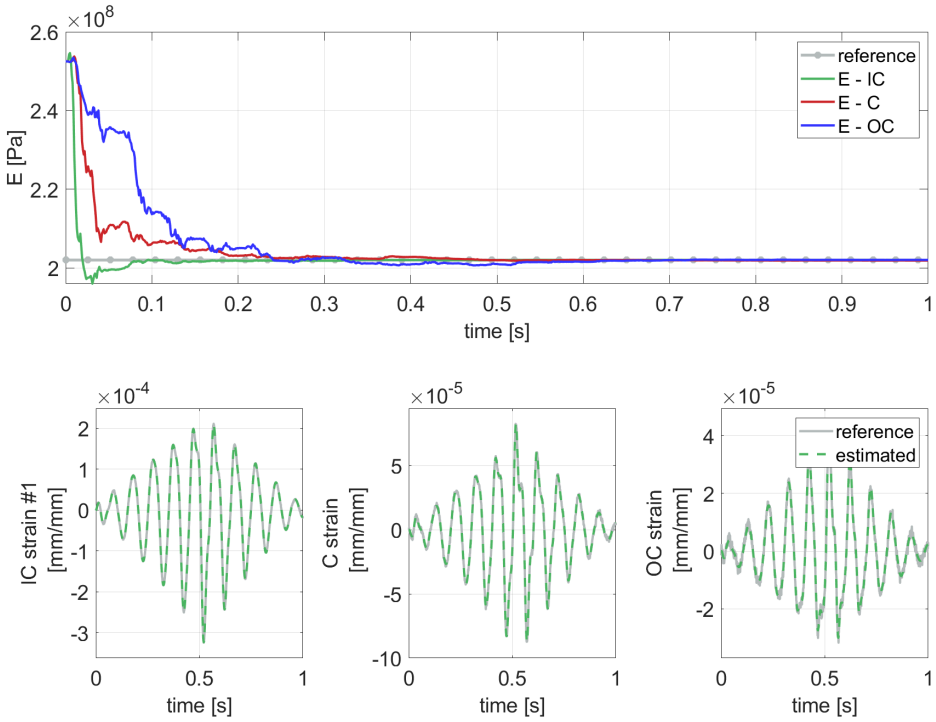


Figure 9.3: Estimated values of E of all components and measurements with numerical data with $\delta_E = 10^{-3}Hz$ using the sensor set 1.

The same set of parameters is also estimated using the second sensor set that only consists on strain measurements on IC. The results are shown in Figure 9.4. Also in this case the parameters converge smoothly to the correct values and stay constant for the rest of the time window. It can be noticed how the Young's modulus of OC is estimated with an error larger than the previous case, while still being in an acceptable range. This behavior can be attributed to the lower sentivity of the IC's deformation on the OC behavior. The reduced loss in accuracy is an acceptable trade-off for the sensing of only one side of the mechanism.

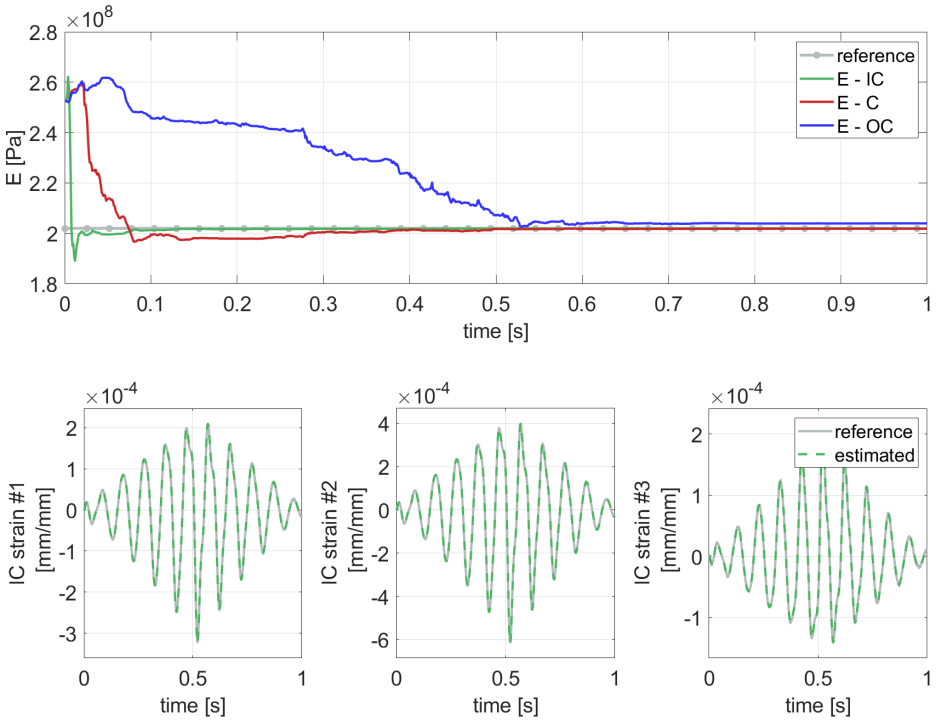


Figure 9.4: Estimated values of E of all components and measurements with numerical data with $\delta_E = 10^{-3}Hz$ using the sensor set 2.

9.4.1.3 Parameter and input estimation

In a second application case, the quantities of interest are the Young's modulus of C and a force applied on OC. This represents a much more complicated case given that the filter has to distinguish between the effects of input and parameter change on different components. The initial value for the parameter is set 25% higher than the true value, and the covariance is set using $\delta_E = 10^{-3}Hz$. The input is instead modeled with $\delta_u = 150Hz$.

The results of the estimation with the ideal sensors setup is shown in Figure 9.5. Both parameter and input are correctly estimated for this case.

The estimation results using sensors only on IC (Figure 9.6) shows the same accuracy in the estimated parameter. The estimated input, on the other hand, is correct in terms of amplitude and frequency but more noisy and with a slight time delay. Equivalently to the previous case, such reduction in filtering quality results from the lower sensitivity between behaviors of IC and OC.

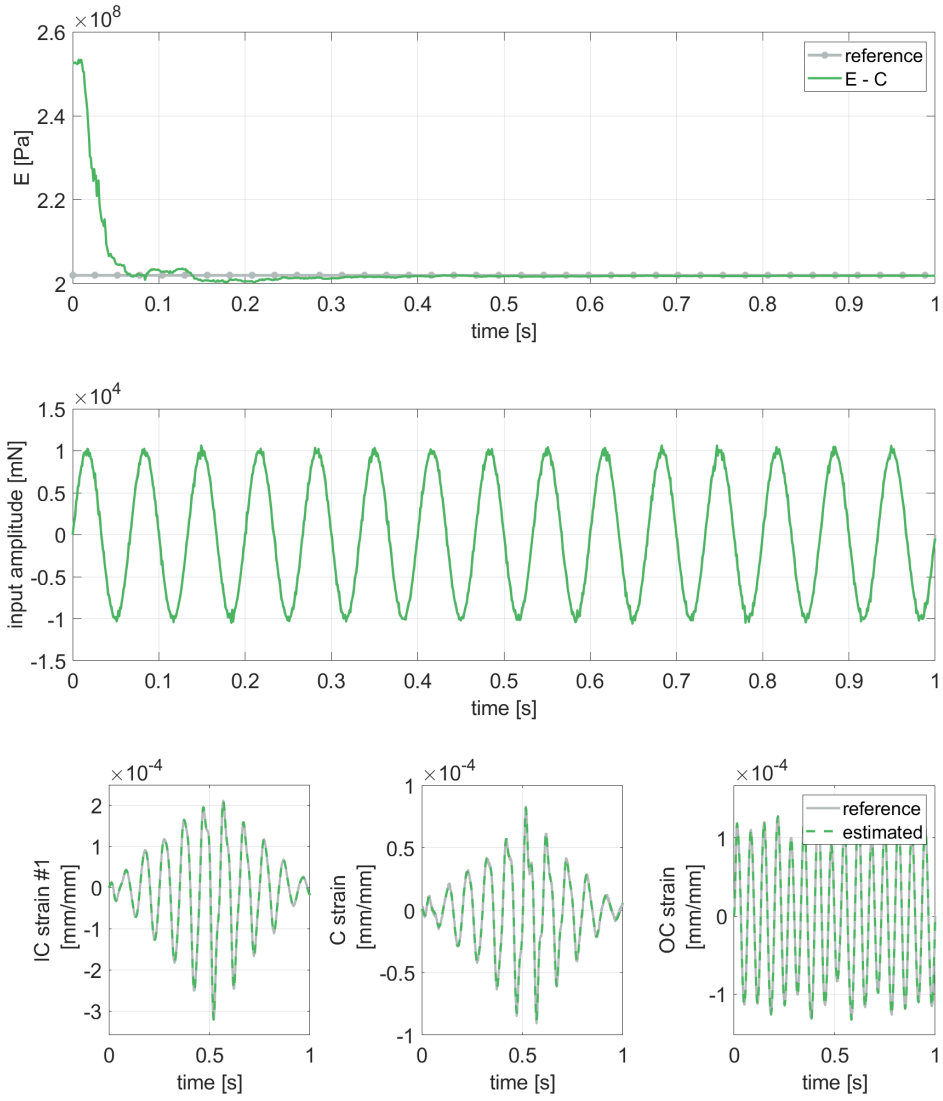


Figure 9.5: Estimated values of E of C, input on OC and measurements with numerical data with $\delta_E = 10^{-3}Hz$ and $\delta_u = 150Hz$ using the sensor set 1.

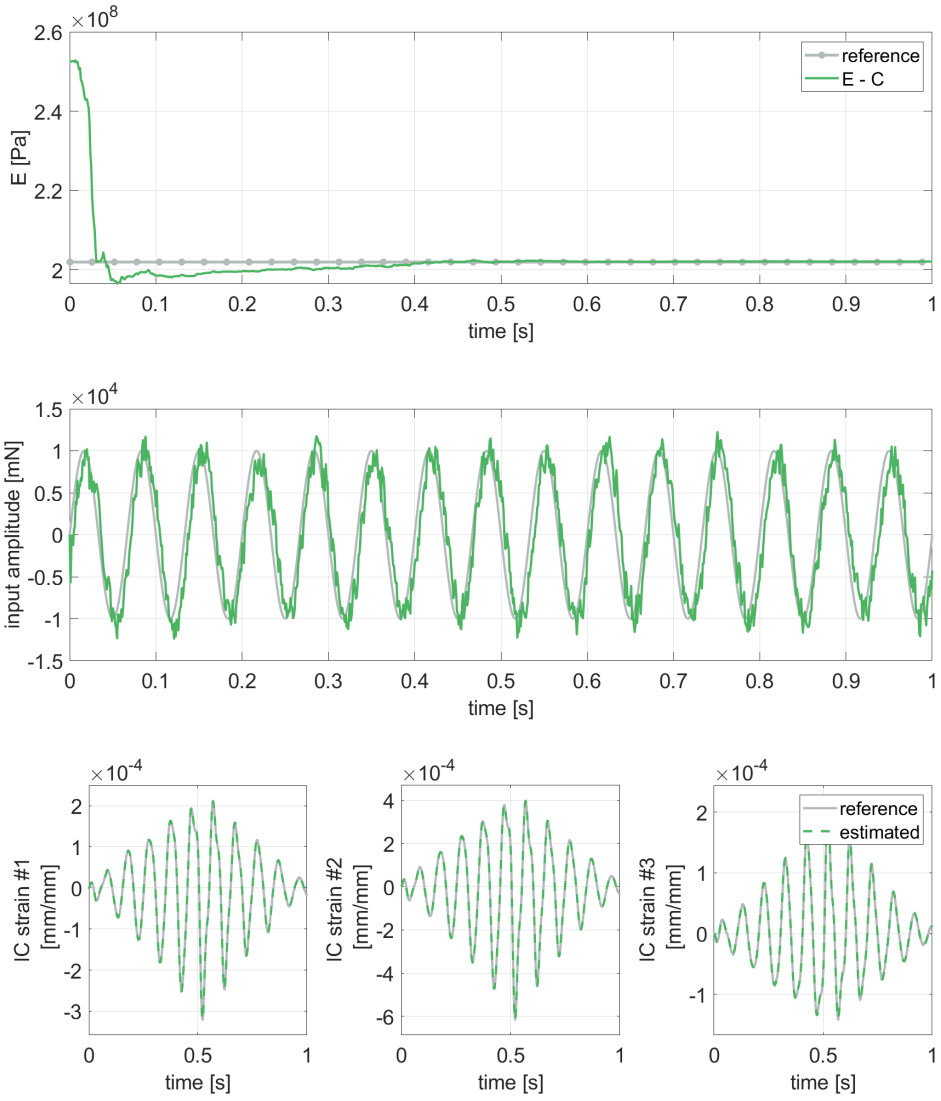


Figure 9.6: Estimated values of E of C, input on OC and measurements with numerical data with $\delta_E = 10^{-3} Hz$ and $\delta_u = 150 Hz$ using the sensor set 2.

9.4.1.4 Degrading parameter and input estimation

A final application case consists in the same estimated quantities, namely Young modulus of C and input on OC, but for a time-varying E. The correct tracking of a deteriorating parameter is of interesting for condition monitoring of the system during operations. The initial value for the parameter is set to match the true value, representing the case for which the systems has been identified and its condition is monitored. The covariances are calculated using $\delta_E = 10^{0} Hz$ and $\delta_u = 150 Hz$.

The results for ideal sensor set (Figure 9.7) and only sensors on IC (Figure 9.8) show an acceptable tracking of the parameter evolution with errors always respectively below 2% and 3%. More measurement noise is present in the estimated quantity given the higher covariance

needed to properly follow the fast evolution in time of the parameter. Also a small time lag is noticeable between the evolution of true and estimated parameters. Furthermore, results for the second sensor set show two zones at the start and end of the time window where the parameter does not vary. This can be attributed to the lower sensitivity of measurements on IC to the quantities of interest, so that when the amplitude of measurements is too small the filter does not modify its estimates.

The inputs are correctly estimated for both cases, showing the same behaviors discussed in the previous section.

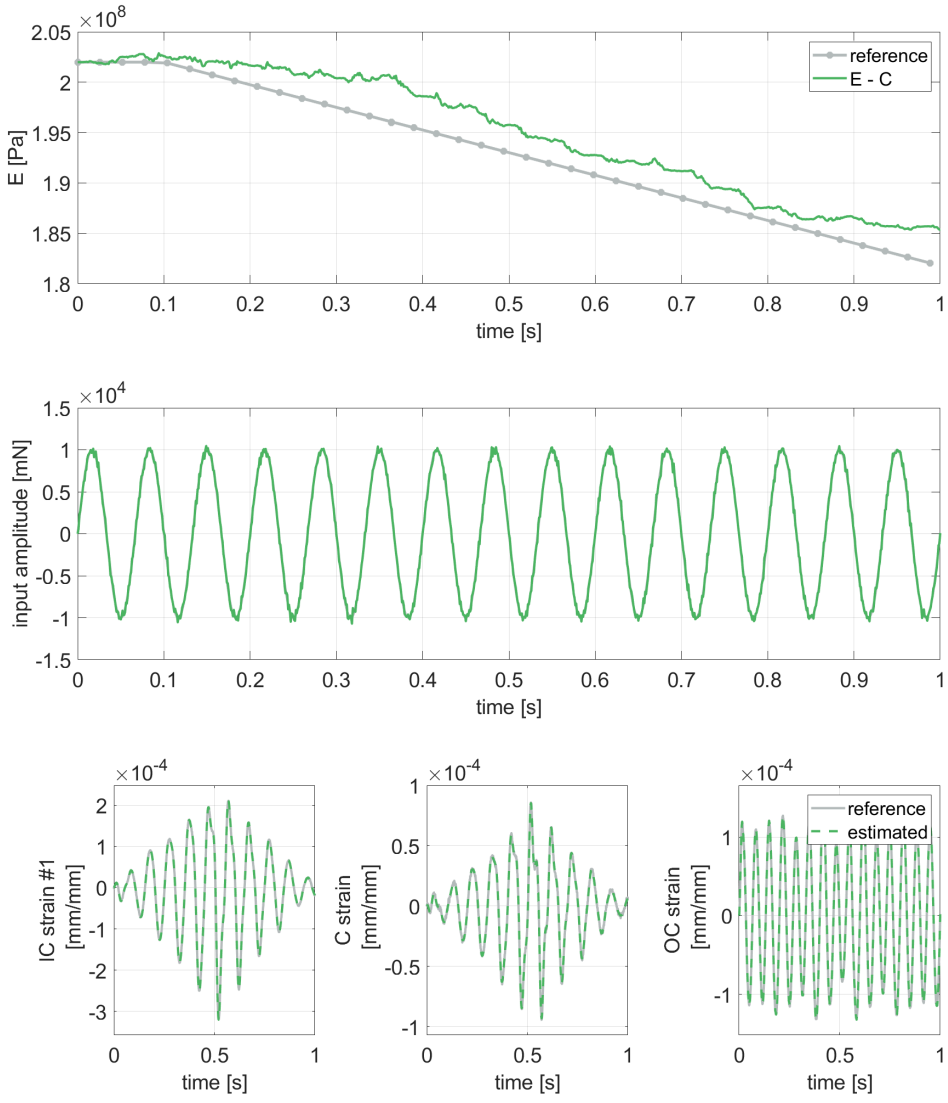


Figure 9.7: Estimated values of degrading E of C, input on OC and measurements with numerical data with $\delta_E = 1Hz$ and $\delta_u = 150Hz$ using the sensor set 1.

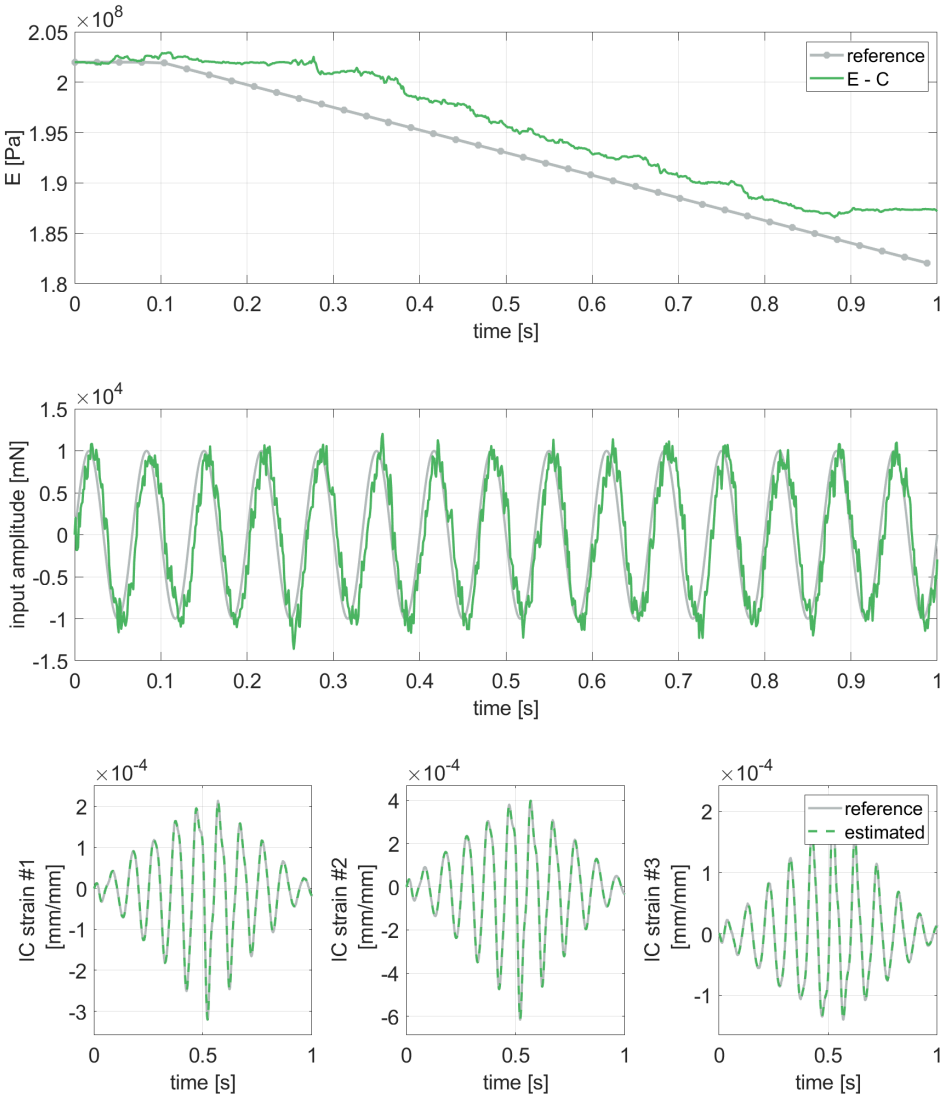


Figure 9.8: Estimated values of degrading E of C, input on OC and measurements with numerical data with $\delta_E = 1Hz$ and $\delta_u = 150Hz$ using the sensor set 2.

9.4.2 Case study n.2: 5 DOF robotic manipulator

A second numerical validation of the proposed system-level estimation methodology is presented considering the Robolink manipulator described in Subsection 4.5.1. The robotic manipulator, being a complex model with many bodies and connection, represents an interesting and realistic application case. The open nature of the manipulator's kinematic chain makes the strains on single links almost independent on the deformations of other components. For this case it is then not possible to estimate parameters of a component by employing strain measurements on a different one, as seen in the previous application.

9.4.2.1 Application cases for estimation

The two maneuvers presented in Subsection 4.5.2 are considered as estimation scenarios. Both are executed under three different conditions to allow for the following estimation applications:

- **Parameter estimation**: the Young's moduli of L2r and L3r are jointly estimated;
- **Degrading parameter estimation**: the same quantities as the previous application case are estimated, but the Young's modulus of L2r is degrading in time while the one of L3r is kept constant.
- **Parameter and input estimation**: the Young's moduli of L2r and L3r are estimated together with an input on the EE;

For each case, gravity acts on the system. For the second application, a degradation of 10% of the initial value in $6s$ is simulated. A 10Hz sinusoidal input is applied to the tip of the end effector after $3s$ from the start of the maneuver for the third application case.

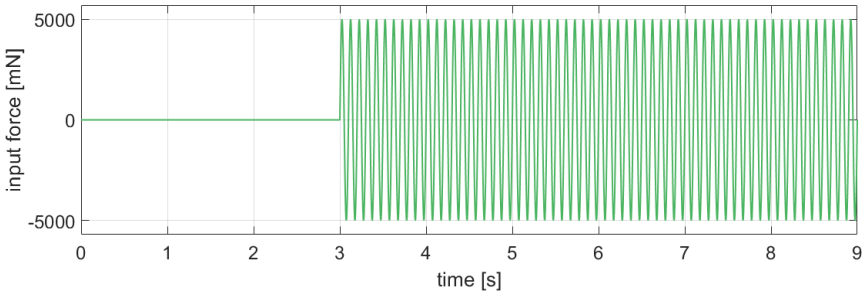


Figure 9.9: 10Hz sine input

The sensors set presented in Chapter 4 is extended by adding an extra virtual strain sensor on each link, in order to guarantee observability in all considered cases. The full set of employed sensors can be seen in Figure 9.10. The relative adaptive strain measurements approach is employed where feasible, as discussed also for the previous application case.

As in the previous section, the numerical measurements used for estimation are generated by simulations using a time step $\Delta t = 1ms$. These are then polluted by adding a white noise in the order of $10^{-7} mm/mm$.

An overview of the performance for all case in terms of parameter estimation is given in Table 9.2.

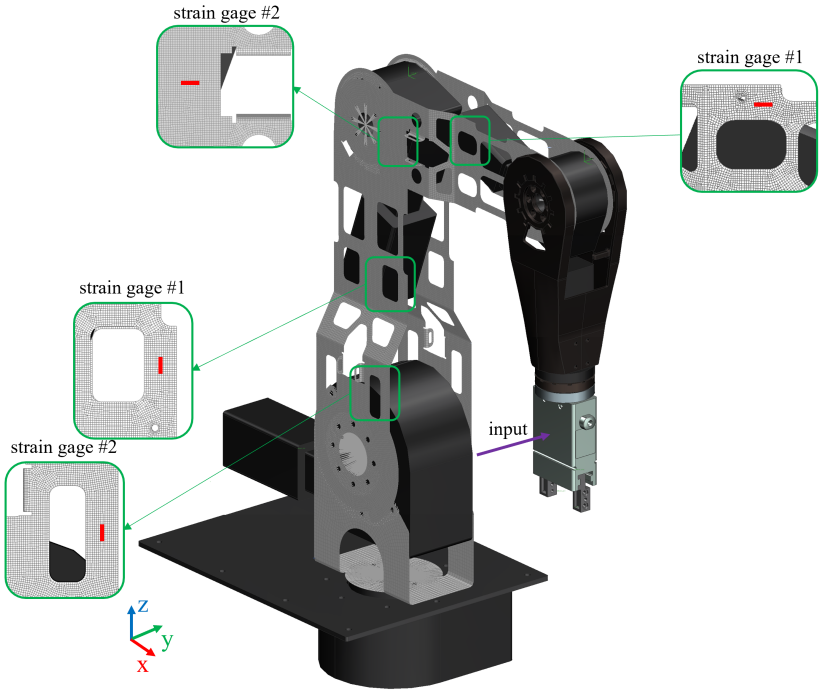


Figure 9.10: Sensors and inputs used for estimation on the Robolink model

Table 9.2: Errors on estimated parameters in last 20% of time interval for the manipulator application case using numerical measurements.

Unknown quantities	**Maneuver**	**Error** [%]			
		Parameter	**min**	**avg**	**max**
E_{2r},E_{3r}	1	E_{2r}	-0.17	-0.04	0.06
		E_{3r}	0.07	0.16	0.27
	2	E_{2r}	-0.18	-0.05	0.04
		E_{3r}	0.07	0.18	0.29
degrading E_{2r},E_{3r}	1	E_{2r}	0.11	0.19	0.28
		E_{3r}	-0.16	-0.04	0.06
	2	E_{2r}	0.07	0.17	0.27
		E_{3r}	-0.03	0.08	0.18
E_{2r}, E_{3r}, input	1	E_{2r}	-0.83	-0.10	0.57
		E_{3r}	-1.32	-0.27	0.88
	2	E_{2r}	-1.38	-0.66	0.14
		E_{3r}	-2.04	-0.89	0.26

9.4.2.2 Parameters estimation

As a first application case, the Young's moduli of L2r and L3r are estimated for the two trajectories executed under gravity. The initial values for the parameters are 25% higher than true values, and their covariances are calculated with $\delta_E = 0.1Hz$.

The evolution of estimated parameters and measurements is shown in Figure 9.11. The convergence of estimated parameters is smooth and quick for both trajectories, remaining constant after the reference value is reached.

In a second application, the Young's modulus of L2r is degrading in time. In this case, both parameters start from the reference value and their covariances are set using $\delta_E = 0.1Hz$. This choice aims at replicating the realistic scenario of an already identified system whose condition is continuously monitored during operations. It can be appreciated in Figure 9.12 how the degradation of the L2r's Young's modulus is properly tracked while the one relative to L3r stays constant.

9.4.2.3 Parameters and inputs estimation

In a second scenario, an input applied to the EE is jointly estimated together with the two Young's moduli. In this case, as discussed in the previous section, a larger number of sensors is required to guarantee observability. Two strain sensors on each link are then used for the estimation. The initial values for the parameters are 25% higher than true values, and covariances are set using $\delta_E = 1Hz$ and $\delta_u = 100Hz$.

Figure 9.13 shows how the input is correctly estimated. Also parameters converge quickly to the reference values, with a larger noise leaking in the estimate due to the required higher covariance setting, as discussed also in Section 8.5.

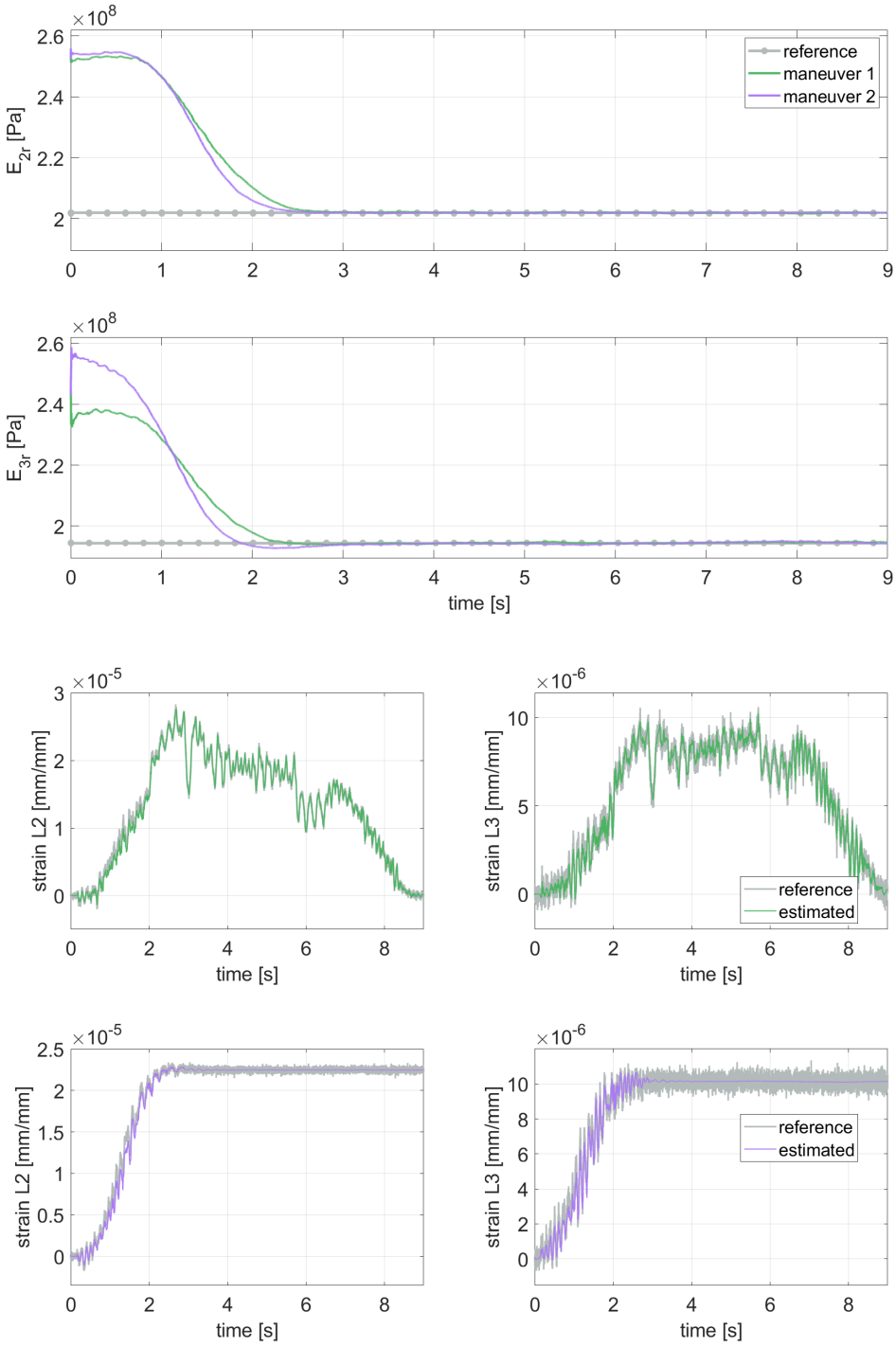


Figure 9.11: Estimated values of E of L2r and L3r and measurements with numerical data with $\delta_E = 0.1Hz$. One strain sensor on each link is employed.

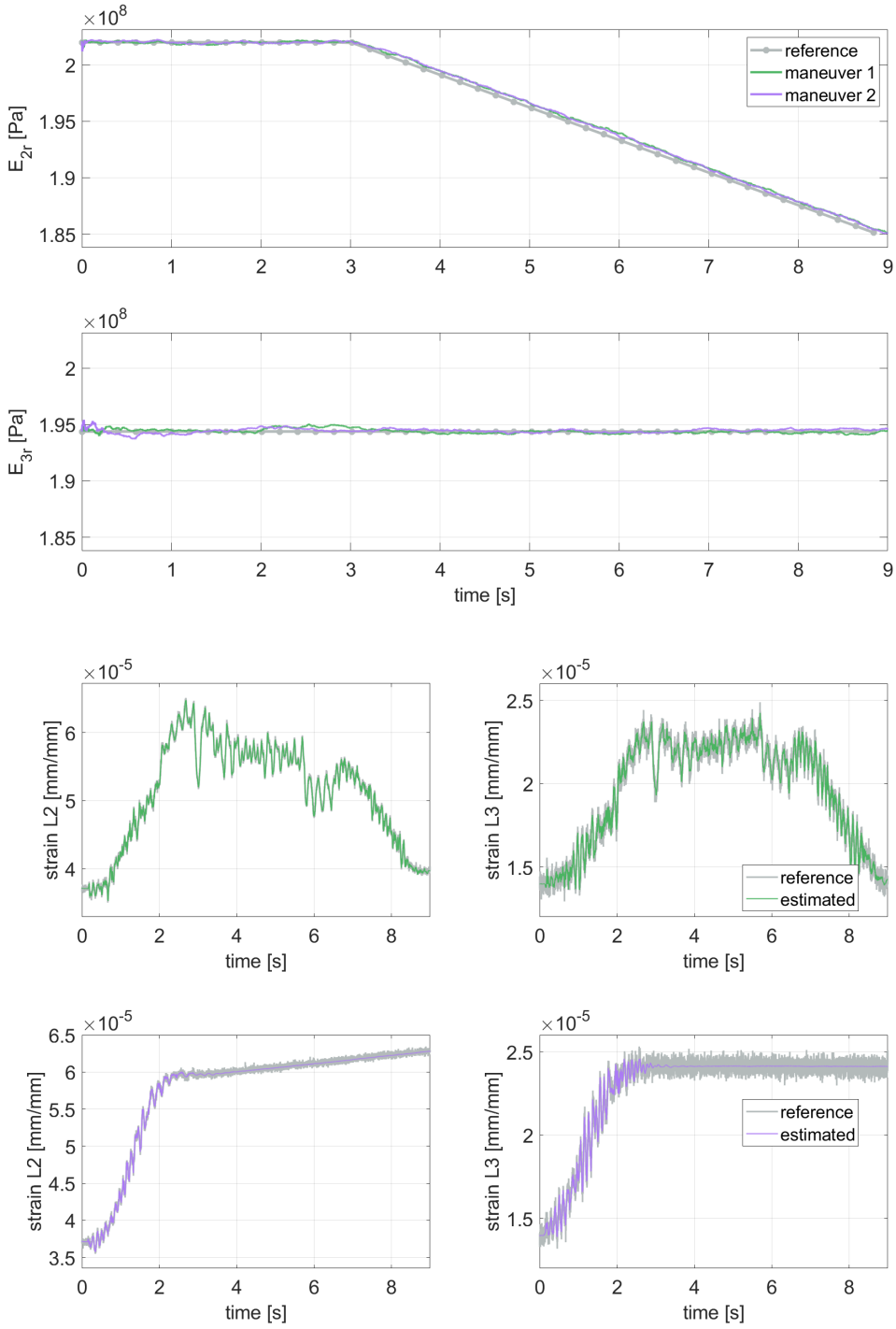


Figure 9.12: Estimated values of degrading E of L2r and L3r and measurements with numerical data with $\delta_E = 0.1Hz$. One strain sensor on each link is employed.

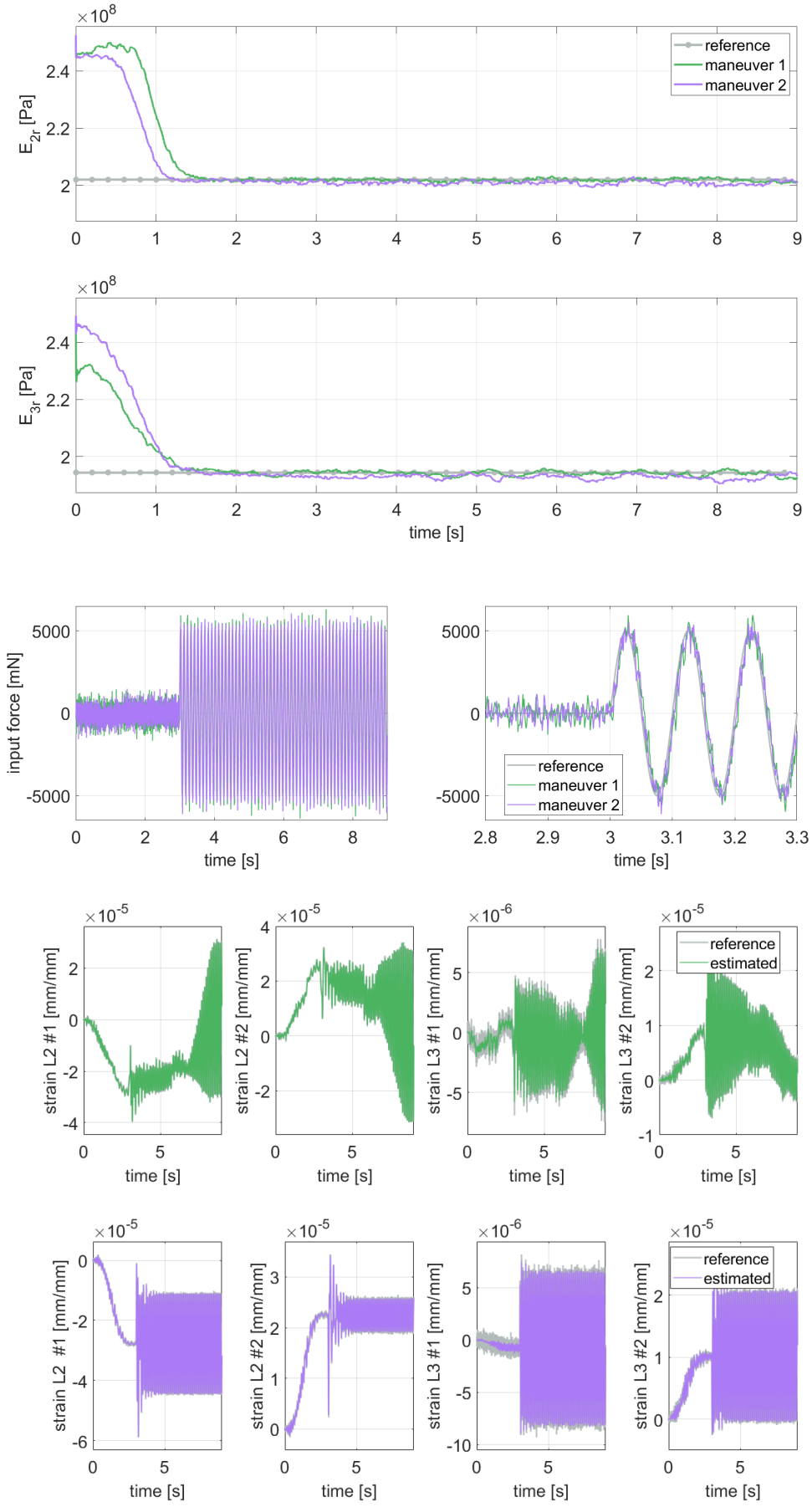


Figure 9.13: Estimated values of E of L2r and L3r, load and measurements with numerical data with $\delta_E = 1Hz$ and $\delta_u = 100Hz$. Two strain sensors on each link are employed.

9.5 Experimental validation

An experimental validation of the methodology is proposed in this chapter. The experimental measurements acquired on the Robolink manipulator are employed. For this case, only the constant parameter application is replicable. No experiments with a degrading parameter or an external input to the system were in fact possible during the experimental campaign. While the proposed results are of great interest in showing potentialities and limits of the estimation approach, it will be interesting in future efforts to also apply it to more complex experimental scenarios.

An overview of the performance for all cases in terms of parameter estimation is given in Table 9.3.

Table 9.3: Errors on estimated parameters in last 20% of time interval for the manipulator application case using experimental measurements.

Unknown quantities	$\boldsymbol{\delta_E}$	**Maneuver**	**Error** [%]		
			min	**avg**	**max**
E_{2r}	$0.1Hz$	1	-2.68	0.11	2.17
		2	-0.55	-0.40	-0.15
E_{2r}	$1Hz$	1	-28.19	-1.49	19.82
		2	-0.79	-0.27	0.59

9.5.1 Application cases for estimation

For experimental data it has been described in Subsection 4.5.2 how the matching between the experimental and virtual strain sensors is accurate enough only on L2r. The parameter linked to this component is then the only one on which the experimental validation focuses. As previously discussed, for open-chain kinematics it is difficult to estimate parameters relative to components not directly sensored. The two maneuvers used also in the previous section are considered. For all cases, the adaptive strain measurements method is employed given that only relative experimental measurements are available.

9.5.2 Parameters estimation

A critical point for the filter setup, as extensively discussed in previous sections, is the setting of covariance values for the augmented states. This choice results of particular importance in the case of experimental data where a mismatch between numerical model and experimental data is present. In this case, the filter tends to compensate for the mismatch by varying the estimated parameter. In mildly dynamic applications such as the considered manipulator case, if the system's static response is correctly modeled the estimate also exhibits a correct average value, with oscillations around it compensating for the wrong dynamic behavior. Such parameter variation can be limited by setting a low value for the parameter's covariance.

The experimental estimation results for a low covariance value ($\delta_E = 0.1Hz$) can be seen in Figure 9.14. For both maneuvers, the parameter converges quickly to a value close to the reference parameter identified through modal testing. The oscillations around the value are then noticeable but limited, with an error always below 3%. The downside of such approach can be seen in the estimated measurements. The filter is in fact able to correctly estimate the static part of the measurements, while damping out most of the dynamics.

In a second possible scenario, the parameter covariance can be set to a higher value ($\delta_E = 1Hz$). In this case, as shown in Figure 9.15, the parameter still converges to the correct value

with an average error of 1.5% but the oscillations are much larger. The measurements are more accurately estimated.

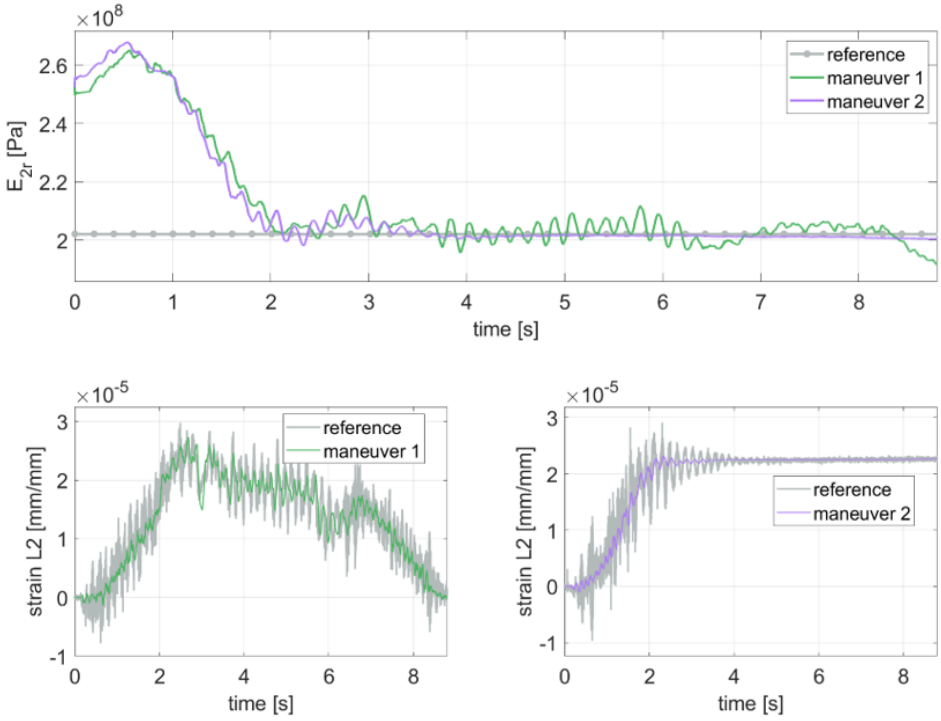


Figure 9.14: Estimated values of E of L2r and measurements with experimental data with $\delta_E = 0.1Hz$. One strain sensor on L2r is employed.

A slight divergence of the estimated parameter can be noticed in the final 0.5s of the maneuver 1 for both cases. In this time window the system is close to the reference configuration, resulting in a loss of observability as previously discussed. While the presented results clearly show the good performance of the filter in the entire maneuver, this behavior demonstrates the need to limit the parameter variation when in such unobservable configuration. This will be one of the most crucial points for future research efforts.

The results shown in this section demonstrate the validity of the proposed application also on experimental data. The main limit of KF approaches, namely the need for an accurate model, is clearly evident from the discussed results. The main assumption in the derivation of the filter is in fact that model noise has zero mean. This implies that if a bias is present between model and real system, the filter cannot guarantee optimality of the estimate. The augmentation to estimate parameters allows to partially curb such issue by allowing the filter to also modify the model itself in order to remove the bias. In the considered case though, the modeling error is caused by phenomena on which the filter cannot directly act. One way to allow for a good estimation is to ensure that at least the static component of the responses is properly matched, as is the case with the manipulator analyzed. In this case then a choice needs to be made between properly estimating the dynamics of the measurements and having low variation on estimated parameters.

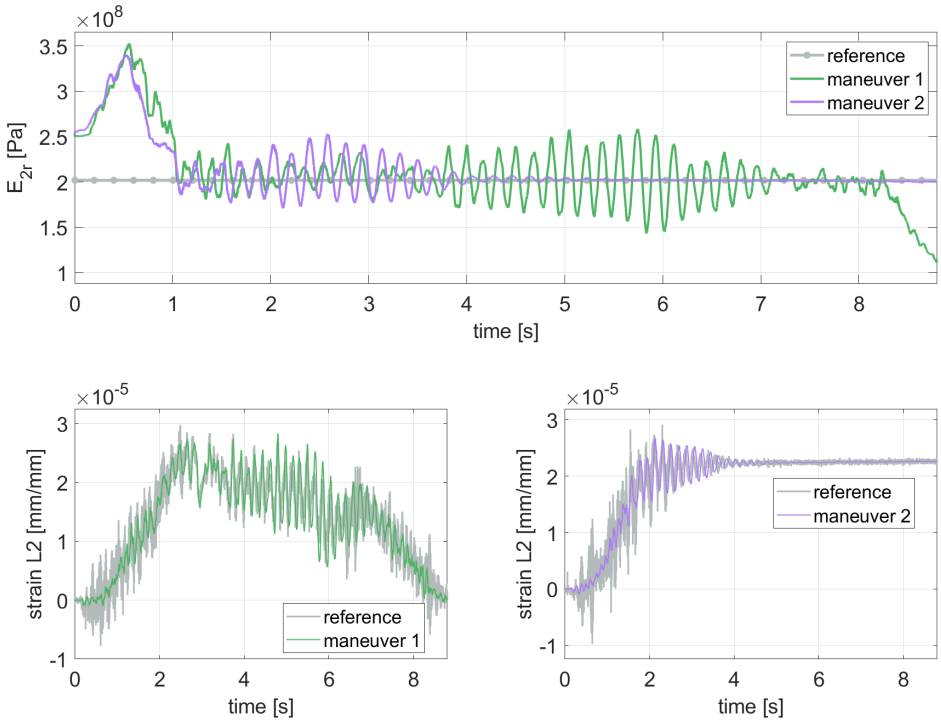


Figure 9.15: Estimated values of E of L2r and measurements with experimental data with $\delta_E = 1Hz$. One strain sensor on L2r is employed.

9.6 Conclusions and possible future work

This chapter presented a methodology for the joint estimation of states, inputs and material parameters of pFMB models. The AMANDA-EKF framework has been introduced, together with its adaptation to work with pFMB models. Practical considerations on the choice of sensors to guarantee observability in the considered cases have been presented. Particular focus has been placed on the use of strain gages on systems evolving under gravity, for which an adaptive measurement equation approach has been proposed. The methodology has been numerically and experimentally validated both on a closed-chain and an open-chain mechanism, showing good results.

Future research efforts proposed for the component-level estimation methodology discussed in Chapter 8 would also benefit the methodology presented in this chapter. Additional efforts should then focus on a rigorous and analytical observability analysis. This would in turn allow to choose proper sensor setups by defining requirements in terms of kinds of sensors, which bodies to sensor and relative positions on the bodies. Furthermore, the use of different kinds of sensors (e.g. position, acceleration, angle) and estimation of different kinds of parameters (i.e. inertial properties, lumped stiffness, etc.) should be investigated. Finally, estimation with relative strain measurements should be robustified in unobservable configurations.

Conclusions

Chapter 10

Overall conclusions and future work

The Industry 4.0 and Smart Manufacturing environments can benefit by adopting lightweight robotic systems for automation of production and many other applications. A good mix of performance enhancement and weight reduction can be achieved by employing advanced materials such as composites. Their use in complex systems further complicates the challenges in most of the parts of the lifetime of such an already complex system. In the design phase, choosing the material parameters that guarantee the best performances is of paramount importance and results in a non-trivial problem. During operation, monitoring the conditions and health states of the system is required in order to prevent damages and adopt efficient maintenance strategies. The described applications can be efficiently tackled by employing model-based techniques. This raises in turn the requirement for numerical models that are at the same time accurate and efficient.

The aforementioned modeling requirements motivated the efforts presented in this work to develop innovative component-level and system-level modeling strategies for multi-material multibody systems. Such models have enabled an efficient design optimization workflow that allowed in turn to show the benefits of using composite components in robotic manipulators. Furthermore, the models have been employed in estimation methodologies that allow to indirectly estimate states, inputs and parameters of components of a system based on a limited set of measurements.

In the following section, the main conclusions derived from the presented work will be summarized. Next, considerations for future research efforts will be presented.

10.1 Overall conclusions

Part I presented the numerical modeling methodologies developed both on component level and system level. The numerical models of the case studies have been introduced together with experimental validation results.

Chapter 2 provided the theoretical background for the modeling methodologies developed in the rest of the work. The constitutive relationships of different families of materials have been presented, with particular focus on Unidirectional Laminates. A brief overview of the linear Finite Elements method has been presented for component-level modeling. The discussion has then been extended to system-level applications considering the Floating Frame of Reference technique for Flexible Multibody modeling. The chapter formulated the equations for common kinds of virtual measurements. Further fundamental details such as first-order reformulation and discretization of the Equations of Motion have been discussed.

Chapter 3 presented a projection-based parametric Model Order Reduction scheme for structural Finite Element models. This exploits the affine dependency of the full model on material parameters and uses a constant global reduction basis to achieve an efficient reduced model that maintains an explicit dependency on material parameters. Two different components have been considered for validation, namely a scaled wind turbine blade in Titanium and the link of a robotic manipulator in Carbon Fiber Reinforced Polymer. The methodology has been validated both in time and frequency domain showing good results. Both models have also been experimentally validated in order to match their physical prototypes.

Chapter 4 proposed a parametric Flexible Multibody formulation that retains an explicit dependency on material parameters of the flexible bodies by employing the component-level parametric modeling technique developed in the previous chapter. The explicit parametric dependency of the various terms in the Equation of Motion has been discussed. The property of having a constant reduction basis results particularly beneficial, avoiding parametric dependencies in the constraint equations. The resulting pFMB models allow an efficient updating of parameters while maintaining a size in the same order as the standard non-parametric models, granting also computational efficiency. Two different systems have been considered for validation, namely a four-bar linkage and a 5 DOF robotic manipulator, with flexible components modeled both using isotropic and composite materials. The methodology was validated showing good results over a wide range of parameters and for different kinds of measurements. Furthermore, the experimental updating of the robotic manipulator has been discussed.

Part II discussed the multiobjective design optimization workflow enabled by the efficient models proposed in the work. The application to a multi-material robotic manipulator is presented.

Chapter 5 provided the theoretical background for multi-objective optimization techniques applied to design of dynamic mechanical systems. Particular focus has been placed on the main search algorithms and on treatment of time-dependent functions. The literature on flexible multibody, robotic manipulators and composite applications has finally been analyzed.

Chapter 6 discussed the use of the parametric Flexible multibody models in a design optimization workflow. Generic considerations on optimization have been initially discussed, with then a focus on multi-objective optimization using zeroth-order search algorithms. The workflow was applied for the choice of optimal material parameters for the links of a robotic manipulator made in composite materials. The experimentally validated manipulator model has been used as a baseline. Employing also an experimentally validated composite material model allowed for a high confidence in the numerical results to be representative of a physical setup. By optimizing angle of the fibers and number of layers, gain in terms of accuracy and actuation energy of up to 15% were achieved. This showed the advantage of employing innovative materials to enable multi-material robotic systems.

Part III presented component-level and system-level Virtual Sensing methodologies for the estimation of states, inputs and parameters. The methodologies have been validated both numerically and experimentally.

Chapter 7 provided the theoretical background on the Kalman Filter in its classical linear version with extensions to non-linear systems and estimation of unknown augmented states. Observability properties and main limitations of the method have been discussed.

Chapter 8 presented the integration of the proposed parametric Reduced Order Models in a Kalman Filter framework for the the joint estimation of states, inputs and material parameters

of structural dynamic components. An observability analysis allowed to define minimal sensor requirements for different combinations of unknown parameters and inputs. The setup of the filter has been analyzed in depth, showing how this phase is crucial for the good performance of the filter and highly application-dependent. Finally, the methodology has been validated both numerically and experimentally on the scaled wind turbine blade showing good results.

Chapter 9 presented a methodology for the estimation of states, input and parameters for Flexible Multibody systems. The proposed parametric Flexible Multibody models have been integrated in the AMANDA framework to enable efficient estimation of material parameters of flexible bodies. Practical considerations on sensors choice have been presented, with particular focus on the use of strain measurements for systems under gravity. The methodology has been validated numerically on the four-bar linkage and the robotic manipulator showing good results. For this second case also an experimental validation has been presented.

10.2 Suggestions for future work

The pMOR methodology has been demonstrated on models with reduced numbers of parameters, typically complex enough for practical applications. It would still be of interest though to investigate how a larger number of parameters influences size and accuracy of the models. Future efforts should also be aimed at identifying affine relationships for other parameters of Finite Element models and, when not possible, at investigating methods to approximate such dependencies.

The potential of the proposed pROM methodology could be further improved by investigating more advanced sampling techniques. While granting a good exploration of the whole parameter set, Latin Hypercube sampling does not exploit any available knowledge on the parametric dependence. Furthermore, the number of samples increases quickly for larger number of parameters. Alternative more efficient approaches should be investigated (i.e. greedy sampling). Information on the parametric dependency could allow to have a finer sampling in the zone of the parameter space where the modal properties vary more quickly and a coarse one where those are less sensitive. Another area of potential improvement is the algorithms for the selection of basis size. The error metric employed could be evaluated with respect to a Full Order Model. The proposed MAC filtering procedure could be refined by also considering the information contained in the Singular Values.

For the system-level modeling methodology, it will be of interest to test the methodology on applications with faster dynamics. The inertia terms play in fact a more prominent role in such cases, requiring an accurate parametric representation.

The experimental validation procedures discussed in this work evidenced wide improvement possibilities. First of all, the composite material should be tested on different fiber angle values, even on a simpler and cheaper model, in order to fully validate the parametric dependency. This would also allow to identify parametric damping behavior. The proportional damping terms have in fact been assumed constant in this work, while they are known from literature to have a dependency on the laminate configuration. On a system level, the limitations and assumptions of the manipulator model have been widely discussed. More accurate results will require modeling of a correct mass distribution for rigid bodies, of the compliant connections and of the cables effects. Furthermore, the experimental setup should ideally guarantee boundary conditions closer to ideal or at least easily identifiable.

An additional experimental campaign should be done on the various optimal designs identified in the optimization phase. This would further validate both the modeling methodology and the performance gains achieved through the use of composite materials. Durability aspects should

also be kept into account as optimization constraints, together with constraints on actuation torques and maximum error based on real-world applications. It would also be of interest to investigate how a reduction in the required actuation torque would enable the use of smaller motors. Finally, the use of parametric models within gradient-based optimization algorithms has been discussed and is considered one of the most promising areas for future methodological research.

For what regards the discussed estimation methodologies, one of the most interesting aspects to investigate is observability. On a component level, the proposed analysis should also keep into account the sensitivity of measurements to parameters that could for example be local to only a portion of the component. For system-level estimation, the development of an analytical observability analysis keeping into account the positions of sensors in the system would be of great interest. Also an Optimal Sensor Placement technique explicitly considering the parameters observability should be investigated.

For what regards the filter setup, automated ways to define covariance values should be investigated. This would allow to make the whole methodology less dependent on the level of expertise of the user. Adaptive methods to vary the covariances during the estimation process are considered promising in this area. Furthermore, alternatives to the random walk model should be employed when information on the expected variation of the unknown quantities is available.

Appendices

Appendix A

Mass matrix of a flexible body in the FFR formulation

The rotational coordinates of the i-th flexible bodies are defined in this work using the Euler parameters;

$$\mathbf{q}^i_\Theta = \begin{bmatrix} e^i_0 & e^i_1 & e^i_2 & e^i_3 \end{bmatrix} \tag{A.1}$$

These can in turn be used to define a set of rotation matrices:

$$\mathbf{E}^i_\Theta(\mathbf{q}^i_\Theta) = \begin{bmatrix} -e^i_1 & e^i_0 & -e^i_3 & e^i_2 \\ -e^i_2 & e^i_3 & e^i_0 & -e^i_1 \\ -e^i_3 & -e^i_2 & e^i_1 & ev_0 \end{bmatrix} \tag{A.2}$$

$$\mathbf{G}^i_\Theta(\mathbf{q}^i_\Theta) = \begin{bmatrix} -e^i_1 & e^i_0 & e^i_3 & -e^i_2 \\ -e^i_2 & -e^i_3 & e^i_0 & e^i_1 \\ -e^i_3 & e^i_2 & -e^i_1 & e^i_0 \end{bmatrix} \tag{A.3}$$

$$\mathbf{A}^i_\Theta(\mathbf{q}^i_\Theta) = \mathbf{E}^i_\Theta(\mathbf{q}^i_\Theta)\mathbf{G}^{i^T}_\Theta(\mathbf{q}^i_\Theta) \tag{A.4}$$

The mass matrix resulting from the FFR formulation is configurations-dependent. In numerical implementations, every entry of the matrix is expressed in terms of a set of mass-invariants that is pre-computed once per flexible body and stored together with the reduced model data itself.

From Chapter 2, the mass matrix of a component is:

$$\mathbf{M}^i(\mathbf{q}^i) = \begin{bmatrix} \mathbf{M}^i_R & \mathbf{M}^i_{R\Theta}(\mathbf{q}^i) & \mathbf{M}^i_{Rf}(\mathbf{q}^i) \\ & \mathbf{M}^i_\Theta(\mathbf{q}^i) & \mathbf{M}^i_{\Theta f}(\mathbf{q}^i) \\ sym & & \mathbf{M}^i_{f_r} \end{bmatrix} \tag{A.5}$$

where the configuration-dependent terms are:

$$\mathbf{M}^i_{R\Theta}(\mathbf{q}^i) = -2\mathbf{E}^i_\Theta \sum_{j=1}^{n^i_{nodes}} m^i_j \mathbf{r}^{i^+}_j \tag{A.6}$$

$$\mathbf{M}^i_{Rf}(\mathbf{q}^i) = \mathbf{A}^i_\Theta \sum_{j=1}^{n^i_{nodes}} m^i_j \mathbf{\Psi}^i_j \tag{A.7}$$

$$\mathbf{M}^i_\Theta(\mathbf{q}^i) = 4\mathbf{G}^{i^T}_\Theta \left(\sum_{j=1}^{n^i_{nodes}} m^i_j \tilde{\mathbf{r}}^{i^T}_j \tilde{\mathbf{r}}^i_j \right) \mathbf{G}^i_\Theta \tag{A.8}$$

$$\mathbf{M}^i_{\Theta f}(\mathbf{q}^i) = -2\mathbf{G}^{i^T}_\Theta \sum_{j=1}^{n^i_{nodes}} m^i_j \tilde{\mathbf{r}}^{i^T}_j \mathbf{\Psi}^i_j + 2\mathbf{q}^i_\Theta \sum_{j=1}^{n^i_{nodes}} m^i_j \mathbf{r}^{i^T}_j \mathbf{\Psi}^i_j \tag{A.9}$$

In the former equations, m^i_j the mass of the $j-th$ node, $\mathbf{r}^i_{0,j}$ and $\mathbf{r}^i_j$ its undeformed and deformed position in the body-attached reference, and $\mathbf{\Psi}^i_j$ its relative row of the reduction basis. The operator $\square^+$ denotes the following matrix associated with a vector:

$$\mathbf{v}^+ = \begin{bmatrix} 0 & \mathbf{v}^T \\ -\mathbf{v} & \tilde{\mathbf{v}} \end{bmatrix} \tag{A.10}$$

A set of mass-invariant terms can be defined:

$$\hat{\mathbf{M}}^i_1 = \sum_{j=1}^{n^i_{nodes}} m^i_j \mathbf{r}^i_{0,j} \tag{A.11}$$

$$\hat{\mathbf{M}}^i_2 = \sum_{j=1}^{n^i_{nodes}} m^i_j \mathbf{\Psi}^i_j \tag{A.12}$$

$$\hat{\mathbf{M}}^i_3 = \sum_{j=1}^{n^i_{nodes}} m^i_j \tilde{\mathbf{r}}^i_{0,j} \tilde{\mathbf{r}}^i_{0,j} \tag{A.13}$$

$$\hat{\mathbf{M}}^i_4 = \sum_{j=1}^{n^i_{nodes}} m^i_j \tilde{\mathbf{r}}^i_{0,j} \tilde{\mathbf{\Psi}}^i_j \tag{A.14}$$

$$\hat{\mathbf{M}}^i_5 = \sum_{j=1}^{n^i_{nodes}} m^i_j \tilde{\mathbf{\Psi}}^i_j \tilde{\mathbf{\Psi}}^i_j \tag{A.15}$$

$$\hat{\mathbf{M}}^i_6 = \sum_{j=1}^{n^i_{nodes}} m^i_j \tilde{\mathbf{\Psi}}^i_j \mathbf{r}^i_{0,j} \tag{A.16}$$

$$\hat{\mathbf{M}}^i_7 = \sum_{j=1}^{n^i_{nodes}} m^i_j \tilde{\mathbf{\Psi}}^i_j \mathbf{\Psi}^i_j \tag{A.17}$$

$$\hat{\mathbf{M}}^i_8 = \sum_{j=1}^{n^i_{nodes}} m^i_j \mathbf{r}^{i^T}_{0,j} \mathbf{\Psi}^i_j \tag{A.18}$$

$$\hat{\mathbf{M}}^i_9 = \sum_{j=1}^{n^i_{nodes}} m^i_j \mathbf{\Psi}^{i^T}_j \mathbf{\Psi}^i_j \tag{A.19}$$

An extra intermediate set of terms dependent only on the flexible coordinates can then be introduced:

$$\hat{\mathbf{D}}_1^i(\mathbf{q}_{f_r}^i) = (\hat{\mathbf{M}}_1^i)^+ + \left((\hat{\mathbf{M}}_2^i \mathbf{q}_{f_r}^i)\right)^+ \tag{A.20}$$

$$\hat{\mathbf{D}}_2^i(\mathbf{q}_{f_r}^i) = -\hat{\mathbf{M}}_3^i - \hat{\mathbf{M}}_4^i \mathbf{q}_{f_r}^i - \left(\hat{\mathbf{M}}_4^i \mathbf{q}_{f_r}^i\right)^T - \hat{\mathbf{M}}_5^i \mathbf{q}_{f_r}^i \mathbf{q}_{f_r}^i \tag{A.21}$$

$$\hat{\mathbf{D}}_3^i(\mathbf{q}_{f_r}^i) = \hat{\mathbf{M}}_6^i + \hat{\mathbf{M}}_7^i \mathbf{q}_{f_r}^i \tag{A.22}$$

$$\hat{\mathbf{D}}_4^i(\mathbf{q}_{f_r}^i) = \hat{\mathbf{M}}_8^i + \mathbf{q}_{f_r}^{i^T} \hat{\mathbf{M}}_9^i \tag{A.23}$$

The terms of the mass matrix can finally be expressed as:

$$\mathbf{M}_{R\Theta}^i(\mathbf{q}^i) = -2\mathbf{E}_\Theta^i(\mathbf{q}_\Theta^i)\hat{\mathbf{D}}_1^i(\mathbf{q}_{f_r}^i) \tag{A.24}$$

$$\mathbf{M}_{Rf}^i(\mathbf{q}^i) = \mathbf{A}_\Theta^i(\mathbf{q}_\Theta^i)\hat{\mathbf{M}}_2^i \tag{A.25}$$

$$\mathbf{M}_\Theta^i(\mathbf{q}^i) = 4\mathbf{G}_\Theta^{i^T}(\mathbf{q}_\Theta^i)\hat{\mathbf{D}}_2^i(\mathbf{q}_{f_r}^i)\mathbf{G}_\Theta^i(\mathbf{q}_\Theta^i) \tag{A.26}$$

$$\mathbf{M}_{\Theta f}^i(\mathbf{q}^i) = -2\mathbf{G}_\Theta^{i^T}(\mathbf{q}_\Theta^i)\hat{\mathbf{D}}_3^i(\mathbf{q}_{f_r}^i) + 2\mathbf{q}_\Theta^i\hat{\mathbf{D}}_4^i(\mathbf{q}_{f_r}^i) \tag{A.27}$$

Bibliography

[1] Siemens Simcenter 3D. https://www.plm.automation.siemens.com/global/en/products/simcenter/, May 2022.

[2] Siemens Simcenter Nastran. https://www.plm.automation.siemens.com/global/en/products/simcenter/simcenter-nastran.html, May 2022.

[3] Siemens Simcenter Testlab. https://www.plm.automation.siemens.com/global/en/products/simcenter/testlab.html, May 2022.

[4] Igus Robolink. https://www.igus.com/robolink/robot-arms, March 2023.

[5] Siemens Simcenter HEEDS. https://plm.sw.siemens.com/en-US/simcenter/integration-solutions/heeds/, April 2023.

[6] S. Abrate and M. Di Sciuva. Equivalent single layer theories for composite and sandwich structures: A review. *Composite Structures*, 179:482–494, 2017.

[7] R. Adduci. *MultiBody model-based virtual sensing*. KU Leuven, 2022.

[8] R. Adduci, M. Vermaut, F. Naets, J. Croes, and W. Desmet. A discrete-time extended kalman filter approach tailored for multibody models: State-input estimation. *Sensors*, 21(13), 2021.

[9] H. Afshari, W. Hare, and S. Tesfamariam. Constrained multi-objective optimization algorithms: Review and comparison with application in reinforced concrete structures. *Applied Soft Computing*, 83:105631, 2019.

[10] K. Agathos, K. E. Tatsis, K. Vlachas, and E. Chatzi. Parametric reduced order models for output-only vibration-based crack detection in shell structures. *Mechanical Systems and Signal Processing*, 162:108051, 2021.

[11] R. Allemang. The modal assurance criterion - twenty years of use and abuse. *Sound & vibration*, 37:14–23, 2003.

[12] J. A. C. Ambrósio, M. A. Neto, and R. P. Leal. Optimization of a complex flexible multibody systems with composite materials. *Multibody System Dynamics*, 18(2):117–144, 2007.

[13] D. Amsallem and C. Farhat. Stabilization of projection-based reduced-order models. *International Journal for Numerical Methods in Engineering*, 91:358–377, 2012.

[14] A. Angeli, W. Desmet, and F. Naets. Deep learning of multibody minimal coordinates for state and input estimation with kalman filtering. *Multibody System Dynamics*, 53(2):205–223, 2021.

[15] M. Anguelova. *Nonlinear observability and identifiability: General theory and a case study of a kinetic model for S. Cerevisiae*. Chalmers University of technology and Gotheborg University, 2004.

[16] K. Asrih. *Flexible multibody dynamics for design optimization and parameter identification*. KU Leuven, 2019.

[17] R. Astroza and A. Alessandri. Effects of model uncertainty in nonlinear structural finite element model updating by numerical simulation of building structures. *Structural Control and Health Monitoring*, 26(3):1–32, 2019.

[18] K. Bathe. *Finite Element Procedures*. Prentice Hall, 2006.

[19] M. Baumann and P. Eberhard. Interpolation-based parametric model order reduction for material removal in elastic multibody systems. *Multibody Syst Dyn*, pages 21–36, 2017.

[20] P. Benner, S. Gugercin, and K. Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, 57(4):483–531, 2015.

[21] J.-M. Berthelot. *Composite Materials: Mechanical Behavior and Structural Analysis*. Springer, New York, 1999.

[22] B. Besselink, U. Tabak, A. Lutowska, N. Van De Wouw, H. Nijmeijer, D. J. Rixen, M. E. Hochstenbach, and W. H. Schilders. A comparison of model reduction techniques from structural dynamics, numerical mathematics and systems and control. *Journal of Sound and Vibration*, 332(19):4403–4422, 2013.

[23] P. Betsch. The discrete null space method for the energy consistent integration of constrained mechanical systems: Part i: Holonomic constraints. *Computer Methods in Applied Mechanics and Engineering*, 194(50):5159–5190, 2005.

[24] S. Bhavikatti and C. Ramakrishnan. Shape optimization of structural systems using finite elements and sequential linear programming. In A. Wexler, editor, *Large Engineering Systems*, pages 224–235. Pergamon, 1977.

[25] E. D. Blanchard, A. Sandu, and C. Sandu. A Polynomial Chaos-Based Kalman Filter Approach for Parameter Estimation of Mechanical Systems. *Journal of Dynamic Systems, Measurement, and Control*, 132(6), 2010.

[26] B. N. Bond and L. Daniel. A piecewise-linear moment-matching approach to parameterized model-order reduction for highly nonlinear systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 26(12):2116–2129, 2007.

[27] W. Bova, E. Lappano, P. G. Catera, and D. Mundo. Development of a parametric model order reduction method for laminated composite structures. *Composite Structures*, 243:112219, 2020.

[28] J. Brownlee. *Clever Algorithms: Nature-Inspired Programming Recipes*. 01 2011.

[29] O. Brüls, E. Lemaire, P. Duysinx, and P. Eberhard. *Optimization of Multibody Systems and Their Structural Components*, pages 49–68. Springer Netherlands, Dordrecht, 2011.

[30] G. Brzhezinski, S. Vettori, E. Di Lorenzo, B. Peeters, E. Chatzi, and F. Cosco. On the dynamic virtualization of a 3d-printed scaled wind turbine blade. In *Proceedings of 40th IMAC Conference*, 2022.

[31] A. Callejo, J. García de Jalón, P. Luque, and D. A. Mántaras. Sensitivity-based, multi-objective design of vehicle suspension systems. *Journal of Computational and Nonlinear Dynamics*, 10(3), 2015.

[32] C. E. Capalbo, D. De Gregoriis, T. Tamarozzi, H. Devriendt, F. Naets, G. Carbone, and D. Mundo. Parameter, input and state estimation for linear structural dynamics using parametric model order reduction and augmented kalman filtering. *Mechanical Systems and Signal Processing*, 185:109799, 2023.

[33] G. Caprino and A. Langella. Optimization of robotic arms made of composite materials for maximum fundamental frequency. *Composite Structures*, 31(1):1–8, 1995.

[34] P. G. Catera. *Development of a CAE-based approach for the concurrent design, manufacturing and testing of hybrid metal-composite spur gears*. University of Calabria, 2019.

[35] M. Cheng and T. C. Becker. Performance of unscented kalman filter for model updating with experimental data. *Earthquake Engineering and Structural Dynamics*, 50(7):1948–1966, 2021.

[36] C. Coello, G. Pulido, and M. Lechuga. Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 8(3):256–279, 2004.

[37] C. A. Coello Coello, D. A. Veldhuizen, and G. B. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problemss*. Springer New York, NY, 2002.

[38] M. Conconi and M. Carricato. A new assessment of singularities of parallel kinematic chains. *IEEE Transactions on Robotics*, 25(4):757–770, 2009.

[39] D. Cook, D. Malkus, M. Plesha, and R. Witt. *Concepts and Applications of Finite Element Analysis*. Wiley, New York, 2001.

[40] A. Corigliano and S. Mariani. Parameter identification in explicit structural dynamics: Performance of the extended kalman filter. *Computer Methods in Applied Mechanics and Engineering*, 193(36-38):3807–3835, 2004.

[41] R. R. craig. A review of time-domain and frequency domain component mode synthesis methods. *Journal of Modal Analysis*, 2, 1987.

[42] R. R. Craig and M. C. C. Bampton. Coupling of substructures for dynamic analyses. *AIAA Journal*, 6 (7), 1968.

[43] E. Creixell-Mediante, J. S. Jensen, F. Naets, J. Brunskog, and M. Larsen. Adaptive parametric model order reduction technique for optimization of vibro-acoustic models: Application to hearing aid design. *Journal of Sound and Vibration*, 424:208–223, 2018.

[44] R. Cumbo, L. Mazzanti, T. Tamarozzi, P. Jiranek, W. Desmet, and F. Naets. Advanced optimal sensor placement for kalman-based multiple-input estimation. *Mechanical Systems and Signal Processing*, 160:107830, 2021.

[45] R. Cumbo, T. Tamarozzi, K. Janssens, and W. Desmet. Kalman-based load identification and full-field estimation analysis on industrial test case. *Mechanical Systems and Signal Processing*, 117:771–785, 2019.

[46] A. De Luca and W. Book. *Robots with Flexible Elements*, pages 287–319. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[47] A. De Santis, B. Siciliano, A. De Luca, and A. Bicchi. An atlas of physical human–robot interaction. *Mechanism and Machine Theory*, 43(3):253–270, 2008.

[48] V. K. Dertimanis, E. N. Chatzi, S. Eftekhar Azam, and C. Papadimitriou. Input-state-parameter estimation of structural systems from limited output information. *Mechanical Systems and Signal Processing*, 126:711–746, 2019.

[49] S. K. Dwivedy and P. Eberhard. Dynamic analysis of flexible manipulators, a literature review. *Mechanism and Machine Theory*, 41(7):749–777, 2006.

[50] M. El-Hofy, M. Helmy, G. Escobar-Palafox, K. Kerrigan, R. Scaife, and H. El-Hofy. Abrasive water jet machining of multidirectional cfrp laminates. *Procedia CIRP*, 68:535–540, 2018.

[51] M. T. M. Emmerich and A. H. Deutz. A tutorial on multiobjective optimization: fundamentals and evolutionary methods. *Natural Computing*, 17(3):585–609, 2018.

[52] H. A. Eschenauer and N. Olhoff. Topology optimization of continuum structures: A review. *Applied Mechanics Reviews*, 54(4):331–390, 2001.

[53] C. R. Farrar and K. Worden. An introduction to structural health monitoring. *Philosophical Transactions of the Royal Society*, 365:303–315, 2007.

[54] Y. Feng, Z. Ji, Y. Gao, H. Zheng, and J. Tan. An energy-saving optimization method for cyclic pick-and-place tasks based on flexible joint configurations. *Robotics and Computer-Integrated Manufacturing*, 67:102037, 2021.

[55] A. Fischer, P. Eberhard, and J. Ambrósio. Parametric flexible multibody model for material removal during turning. *Journal of Computational and Nonlinear Dynamics*, 9(1), 2013.

[56] M. Fischer and P. Eberhard. Application of parametric model reduction with matrix interpolation for simulation of moving loads in elastic multibody systems. *Adv Comput Math*, 41:1049–1072, 2014.

[57] R. Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, Ltd, 2000.

[58] P. Flores. *Concepts and Formulations for Spatial Multibody Dynamics*. Springer Cham, 2015.

[59] B. Forrier. *Virtual torque sensing. A model-based approach for indirect measurement of dynamic operational loads on mechatronic powertrains*. KU Leuven, 2018.

[60] M. Friswell and J. Mottershead. *Finite Element Model Updating in Structural Dynamics*. Springer, Dordrecht, 1995.

[61] M. Gadaleta, M. Pellicciari, and G. Berselli. Optimization of the energy consumption of industrial robots for automatic code generation. *Robotics and Computer-Integrated Manufacturing*, 57:452–464, 2019.

[62] C. Gear, B. Leimkuhler, and G. Gupta. Automatic integration of euler-lagrange equations with constraints. *Journal of Computational and Applied Mathematics*, 12-13:77–90, 1985.

[63] B. K. Ghosh and J. Rosenthal. A generalized popov-belevitch-hautus test of observability. *IEEE Transactions on Automatic Control*, 40(1):176–180, 1995.

[64] R. F. Gibson. *Principles of Composite Material Mechanics*. CRC Press, 2016.

[65] A. Gosavi. *Simulation-Based Optimization*. Springer New York, NY, 2015.

[66] D. E. Grierson and W. H. Pak. Optimal sizing, geometrical and topological design using a genetic algorithm. *Structural optimization*, 6(3):151–159, 1993.

[67] V. Gufler, E. Wehrle, and A. Zwölfer. A review of flexible multibody dynamics for gradient-based design optimization. *Multibody System Dynamics*, pages 379–409, 2021.

[68] M. O. Gulec and S. Ertugrul. Pareto front generation for integrated drive-train and structural optimisation of a robot manipulator conceptual design via nsga-ii. *Advances in Mechanical Engineering*, 15(3), 2023.

[69] R. Guo, C. Li, Y. Niu, and G. Xian. The fatigue performances of carbon fiber reinforced polymer composites – a review. *Journal of Materials Research and Technology*, 21:4773–4789, 2022.

[70] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger. Requirements for safe robots: Measurements, analysis and new insights. *The International Journal of Robotics Research*, 28(11-12):1507–1527, 2009.

[71] E. Hairer, S. Norsett, and G. Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems*, volume 8. 01 1993.

[72] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*, volume 14. 01 1996.

[73] E. Haug. *Computer-Aided Kinematics and Dynamics of Mechanical Systems Volume-I*, volume 1. 01 1989.

[74] M. Hautus. Controllability and observability condition of linear autonomous systems. *Nederlandse Akademie van Wetenschappen. Proceedings. Series A. Indagationes Mathematicae*, 1969.

[75] R. Haynes, J. Cline, B. Shonkwiler, and E. Armanios. On plane stress and plane strain in classical lamination theory. *Composites Science and Technology*, 127:20–27, 2016.

[76] A. Held. On design sensitivities in the structural analysis and optimization of flexible multibody systems. *Multibody System Dynamics*, 54(1):53–74, 2022.

[77] G. Hirzinger, A. Albu-Schaffer, M. Hahnle, I. Schaefer, and N. Sporer. On a new generation of torque controlled light-weight robots. *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, 4:3356–3363 vol.4, 2001.

[78] Y. Hou, C. Liu, and H. Hu. Component-level proper orthogonal decomposition for flexible multibody systems. *Computer Methods in Applied Mechanics and Engineering*, 361:112690, 2020.

[79] Y. Huang, J. Yu, J. Beck, H. Zhu, and H. Li. Novel sparseness-inducing dual kalman filter and its application to tracking time-varying spatially-sparse structural stiffness changes and inputs. *Computer Methods in Applied Mechanics and Engineering*, 372, 2020.

[80] M. Impraimakis and A. W. Smyth. An unscented kalman filter method for real time input-parameter-state estimation. *Mechanical Systems and Signal Processing*, 162(March 2021):108026, 2022.

[81] A. K. Inkulu, M. V. A. R. Bahubalendruni, A. Dara, and S. K. Challenges and opportunities in human robot collaboration context of industry 4.0 - a state of the art review. *Industrial Robot: the international journal of robotics research and application*, 49(2):226–239, 2022.

[82] S. Iwamura, Y. Mizukami, T. Endo, and F. Matsuno. Cable-path optimization method for industrial robot arms. *Robotics and Computer-Integrated Manufacturing*, 73:102245, 2022.

[83] A. K. S. Jardine, D. Lin, and D. Banjevic. A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing*, 20:1483–1510, 2006.

[84] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering, Transactions of the ASME*, 82(1):35–45, 1960.

[85] B.-S. Kang, G.-J. Park, and J. Arora. A review of optimization of structures subjected to transient loads. *Structural and Multidisciplinary Optimization*, 31:81–95, 2006.

[86] B. S. Kang, G. J. Park, and J. S. Arora. Optimization of flexible multibody dynamic systems using the equivalent static load method. *AIAA Journal*, 43(4):846–852, 2005.

[87] I. Karabegovic, E. Karabegovic, M. Mahmic, and E. Husak. Implementation of industry 4.0 and industrial robots in the manufacturing processes. *New Technologies, Development and Application II*, pages 3–14, 2020.

[88] Q. Khadim, M. Kiani-Oshtorjani, S. Jaiswal, M. K. Matikainen, and A. Mikkola. Estimating the characteristic curve of a directional control valve in a combined multibody and hydraulic system using an augmented discrete extended kalman filter. *Sensors*, 21(15), 2021.

[89] H. Khodadadi and H. Jazayeri-Rad. Applying a dual extended kalman filter for the nonlinear state and parameter estimations of a continuous stirred tank reactor. *Computers and Chemical Engineering*, 35(11):2426–2436, 2011.

[90] M. Kirchner, J. Croes, F. Cosco, and W. Desmet. Exploiting input sparsity for joint state/input moving horizon estimation. *Mechanical Systems and Signal Processing*, 101:237–253, 2018.

[91] A. J. Krener and K. Ide. Measures of unobservability. *Proc. Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, page 6401–6406, 2009.

[92] D. Kumar and K. Singh. An approach towards damage free machining of cfrp and gfrp composite material: a review. *Advanced Composite Materials*, 24(sup1):49–63, 2015.

[93] E. Lappano, F. Naets, W. Desmet, D. Mundo, and E. Nijman. A greedy sampling approach for the projection basis construction in parametric model order reduction for structural dynamics models. *Proceedings of ISMA 2016 - International Conference on Noise and Vibration Engineering*, pages 3563–3571, 2016.

[94] Y. Lei, X. Li, J. Huang, and L. Liu. Simultaneous assessment of damage and unknown input for large structural systems by ukf-ui. *Journal of Engineering Mechanics*, 147, 2021.

[95] M. Liang, B. Wang, and T. Yan. Dynamic optimization of robot arm based on flexible multi-body model. *Journal of Mechanical Science and Technology*, 31(8):3747–3754, 2017.

[96] M. Liu and D. Gorman. Formulation of rayleigh damping and its extensions. *Computers & Structures*, 57(2):277–285, 1995.

[97] Y. Liu, J. Chen, J. Liu, and X. Jing. Nonlinear mechanics of flexible cables in space robotic arms subject to complex physical environment. *Nonlinear Dynamics*, 94, 2018.

[98] E. Lourens, E. Reynders, G. De Roeck, G. Degrande, and G. Lombaert. An augmented kalman filter for force identification in structural dynamics. *Mechanical Systems and Signal Processing*, 27(1):446–460, 2012.

[99] Y. Lu. Industry 4.0: A survey on technologies, applications and open research issues. *Journal of Industrial Information Integration*, 6:1–10, 2017.

[100] K. Luo, H. Hu, C. Liu, and Q. Tian. Model order reduction for dynamic simulation of a flexible multibody system via absolute nodal coordinate formulation. *Computer Methods in Applied Mechanics and Engineering*, 324:573–594, 2017.

[101] T. Marauli, H. Gattringer, and A. Müller. Time-optimal path following for non-redundant serial manipulators using an adaptive path-discretization. *Robotica*, page 1–16, 2023.

[102] R. T. Marler and J. S. Arora. Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, 26(6):369–395, 2004.

[103] F. Marques, I. Roupa, M. T. Silva, P. Flores, and H. M. Lankarani. Examination and comparison of different methods to model closed loop kinematic chains using lagrangian formulation with cut joint, clearance joint constraint and elastic joint approaches. *Mechanism and Machine Theory*, 160:104294, 2021.

[104] M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979.

[105] J. Mottershead and M. Friswell. Model updating in structural dynamics: A survey. *Journal of Sound and Vibration*, 167(2):347–375, 1993.

[106] F. Naets. *Development of system-level model reduction techniques for flexible multibody simulation*. KU Leuven, 2013.

[107] F. Naets, J. Croes, and W. Desmet. An online coupled state/input/parameter estimation approach for structural dynamics. *Computer Methods in Applied Mechanics and Engineering*, 283:1167–1188, 2015.

[108] F. Naets, J. Cuadrado, and W. Desmet. Stable force identification in structural dynamics using kalman filtering and dummy-measurements. *Mechanical Systems and Signal Processing*, 50-51:235–248, 2015.

[109] F. Naets, R. Pastorino, J. Cuadrado, and W. Desmet. Online state and input force estimation for multibody models employing extended kalman filtering. *Multibody System Dynamics*, 32(3):317–336, 2014.

[110] F. Naets, S. van Aalst, B. Boulkroune, N. E. Ghouti, and W. Desmet. Design and experimental validation of a stable two-stage estimator for automotive sideslip angle and tire parameters. *IEEE Transactions on Vehicular Technology*, 66(11):9727–9742, 2017.

[111] M. Á. Naya, E. Sanjurjo, A. J. Rodríguez, and J. Cuadrado. Kalman filters based on multibody models: linking simulation and real world. a comprehensive review. *Multibody System Dynamics*, 2023.

[112] M. A. Neto, J. A. Ambrósio, and R. P. Leal. Composite materials in flexible multibody systems. *Computer Methods in Applied Mechanics and Engineering*, 195(50):6860–6873, 2006.

[113] S. Oral and S. Ider. Optimum design of high-speed flexible robotic arms with dynamic behavior constraints. *Computers & Structures*, 65(2):255–259, 1997.

[114] M. Pastor, M. Binda, and T. Harčarik. Modal assurance criterion. *Procedia Engineering*, 48:543–548, 2012.

[115] J. L. J. Pereira, G. A. Oliver, M. B. Francisco, S. S. Cunha, and G. F. Gomes. A review of multi-objective optimization: Methods and algorithms in mechanical engineering problems. *Archives of Computational Methods in Engineering*, 29(4):2285–2308, 2022.

[116] A. Quarteroni, A. Manzoni, and F. Negri. *Reduced basis methods for partial differential equations: An introduction*. Springer, Cham, 2015.

[117] E. Risaliti. *Model based virtual sensors for wheel center loads and full strain field on vehicle suspension components*. KU Leuven, 2019.

[118] E. Risaliti, B. Cornelis, T. Tamarozzi, and W. Desmet. A state-input estimation approach for force identification on an automotive suspension component. *Conference Proceedings of the Society for Experimental Mechanics Series*, 3:359–369, 2016.

[119] E. Risaliti, T. Tamarozzi, M. Vermaut, B. Cornelis, and W. Desmet. Multibody model based estimation of multiple loads and strain field on a vehicle suspension system. *Mechanical Systems and Signal Processing*, 123:1–25, 2019.

[120] A. J. Rodríguez, E. Sanjurjo, R. Pastorino, and M. Ángel Naya. State, parameter and input observers based on multibody models and kalman filters for vehicle dynamics. *Mechanical Systems and Signal Processing*, 155:107544, 2021.

[121] M. Russo. Measuring performance: Metrics for manipulator design, control, and optimization. *Robotics*, 12(1), 2023.

[122] E. Sanjurjo, D. Dopico, A. Luaces, and M. Ángel Naya. State and force observers based on multibody models and the indirect kalman filter. *Mechanical Systems and Signal Processing*, 106:210–228, 2018.

[123] D. Saravanos and J. Lamancusa. Optimum structural design of robotic manipulators with fiber reinforced composite materials. *Computers & Structures*, 36(1):119–132, 1990.

[124] M. Sayahkarajy, Z. Mohamed, A. A. M. Faudzi, and E. Supriyanto. Hybrid vibration and rest-to-rest control of a two-link flexible robotic arm using hinf loop-shaping control design. *Engineering Computations*, 33(2), 2016.

[125] W. H. A. Schilders, H. van der Vorst, and J. Rommes. *Model Order Reduction: Theory, Research Aspects and Applications*. Springer, Berlin, Heidelberg, 2008.

[126] A. A. Shabana. *Dynamics of Multibody Systems*. Cambridge University Press, 2020.

[127] A. M. Shafei and M. H. Korayem. Theoretical and experimental study of dynamic load-carrying capacity for flexible robotic arms in point-to-point motion. *Optimal Control Applications and Methods*, 38(6):963–972, 2017.

[128] S. Sharma and V. Kumar. A comprehensive review on multi-objective optimization techniques: Past, present and future. *Archives of Computational Methods in Engineering*, 29(7):5605–5633, 2022.

[129] D. Simon. *Optimal State Estimation: Kalman, Hinf and Nonlinear Approaches*. Wiley, New York, 2006.

[130] D. Simon. Kalman filtering with state constraints: A survey of linear and nonlinear algorithms. *IET Control Theory and Applications*, 4(8):1303–1318, 2010.

[131] M. Song, R. Astroza, H. Ebrahimian, B. Moaveni, and C. Papadimitriou. Adaptive kalman filters for nonlinear finite element model updating. *Mechanical Systems and Signal Processing*, 143:106837, 2020.

[132] N. Song, H. Peng, and Z. Kan. A hybrid data-driven model order reduction strategy for flexible multibody systems considering impact and friction. *Mechanism and Machine Theory*, 169:104649, 2022.

[133] F. O. Sonmez. Optimum design of composite structures: A literature survey (1969-2009). *Journal of Reinforced Plastics and Composites*, 36(1):3 – 39, 2017.

[134] M. W. Spong. Modeling and control of elastic joint robots. *Journal of Dynamic Systems, Measurement, and Control*, 109(4):310–318, 1987.

[135] G. L. Srinivas and A. Javed. Optimization approaches of industrial serial manipulators to improve energy efficiency: A review. *IOP Conference Series: Materials Science and Engineering*, 912(3):032058, 2020.

[136] T. Tamarozzi, G. H. Heirman, and W. Desmet. An on-line time dependent parametric model order reduction scheme with focus on dynamic stress recovery. *Computer Methods in Applied Mechanics and Engineering*, 268:336–358, 2014.

[137] T. Tamarozzi, P. Jiránek, and D. De Gregoriis. Robust state-input estimation for differential algebraic equations and application to multibody systems. *15th World Congress on Computational Mechanics and 8th Asian Pacific Congress on Computational Mechanics*, 2022.

[138] Y. Tan and L. Zhang. Computational methodologies for optimal sensor placement in structural health monitoring: A review. *Structural Health Monitoring*, 19(4):1287–1308, 2020.

[139] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee. Digital twin in industry: State-of-the-art. *IEEE Transactions on Industrial Informatics*, 15(4):2405–2415, 2019.

[140] R. C. Tech. Sherpa—an efficient and robust optimization/search algorithm, 2008.

[141] Z. Terze, D. Lefeber, and O. Muftic. Null space integration method for constrained multibody systems with no constraint violation. *Multibody System Dynamics*, 6(3):229–243, 2001.

[142] E. Tromme, O. Brüls, and P. Duysinx. Weakly and fully coupled methods for structural optimization of flexible mechanisms. *Multibody System Dynamics*, 38(4):391–417, 2016.

[143] E. Tromme, O. Brüls, J. Emonds-Alt, M. Bruyneel, G. Virlez, and P. Duysinx. Discussion on the optimization problem formulation of flexible components in multibody systems. *Structural and Multidisciplinary Optimization*, 48(6):1189–1206, 2013.

[144] E. Tromme, A. Held, P. Duysinx, and O. Brüls. System-based approaches for structural optimization of flexible mechanisms. *Archives of Computational Methods in Engineering*, 25(3):817–844, 2018.

[145] A. van de Walle, F. Naets, E. Deckers, and W. Desmet. Stability-preserving model order reduction for time-domain simulation of vibro-acoustic fe models. *International Journal for Numerical Methods in Engineering*, 109:889–912, 2017.

[146] H. Van der Auweraer and D. Hartmann. The executable digital twin: merging the digital and the physics worlds. *Proceedings of ISMA 2022- International Conference on Noise and Vibration Engineering*, 2022.

[147] R. Van Der Merwe. *Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*. Oregon Health & Science University, 2004.

[148] S. Vanpaemel. *Sensitivity Analysis of Flexible Multibody Systems. With Applications in Parameter, Topology and Input Optimization*. KU Leuven, 2022.

[149] S. Vettori, E. Di Lorenzo, B. Peeters, M. Luczak, and E. Chatzi. An adaptive-noise augmented kalman filter approach for input-state estimation in structural dynamics. *Mechanical Systems and Signal Processing*, 184:109654, 2023.

[150] F. Vidussi, P. Boscariol, L. Scalera, and A. Gasparetto. Local and trajectory-based indexes for task-related energetic performance optimization of robotic manipulators. *Journal of Mechanisms and Robotics*, 13(2), 2021.

[151] T. M. Wasfy and A. K. Noor. Computational strategies for flexible multibody systems. *Applied Mechanics Reviews*, 56(6):553–613, 2003.

[152] X. Xie, H. Zheng, S. Jonckheere, B. Pluymers, and W. Desmet. A parametric model order reduction technique for inverse viscoelastic material identification. *Computers and Structures*, 212:188–198, 2019.

[153] J. Xu, Y. Yin, J. Paulo Davim, L. Li, M. Ji, N. Geier, and M. Chen. A critical review addressing drilling-induced damage of cfrp composites. *Composite Structures*, 294:115594, 2022.

[154] L. D. Xu, E. L. Xu, and L. Li. Industry 4.0: state of the art and future trends. *International Journal of Production Research*, 56(8):2941–2962, 2018.

[155] Y. Yang, T. Nagayama, and K. Xue. Structure system estimation under seismic excitation with an adaptive extended kalman filter. *Journal of Sound and Vibration*, 489:115690, 2020.

[156] J. Yen and L. R. Petzold. An efficient newton-type iteration for the numerical solution of highly oscillatory constrained multibody dynamic systems. *SIAM Journal on Scientific Computing*, 19(5):1513–1534, 1998.

[157] H. Yin, S. Huang, M. He, and J. Li. An overall structure optimization for a light-weight robotic arm. In *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*, pages 1765–1770, 2016.

[158] H. Yin, J. Liu, and F. Yang. Hybrid structure design of lightweight robotic arms based on carbon fiber reinforced plastic and aluminum alloy. *IEEE Access*, 7:64932–64945, 2019.

[159] H. Yin, F. Yang, and J. Li. Multi-objective optimization design of a carbon fiber reinforced composite upper arm. In *2017 12th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pages 1680–1685, 2017.

[160] C. Yuan, D. Hohlfeld, and T. Bechtold. Design optimization of a miniaturized thermoelectric generator via parametric model order reduction. *Microelectronics Reliability*, 119:114075, 2021.

[161] C. Yuan, S. Kreß, G. Sadashivaiah, E. B. Rudnyi, D. Hohlfeld, and T. Bechtold. Towards efficient design optimization of a miniaturized thermoelectric generator for electrically active implants via model order reduction and submodeling technique. *International Journal for Numerical Methods in Biomedical Engineering*, 36(4):e3311, 2020.

[162] M. Zhang, H. Peng, and N. Song. Semi-analytical sensitivity analysis approach for fully coupled optimization of flexible multibody systems. *Mechanism and Machine Theory*, 159:104256, 2021.

List of publications

Articles in international journals

C.E. Capalbo, D. De Gregoriis, T. Tamarozzi, H. Devriendt, F. Naets, G. Carbone and D. Mundo, *Parameter, input and state estimation for linear structural dynamics using parametric model order reduction and augmented Kalman filtering*, Mechanical Systems and Signal Processing, Volume 185, 2023

A. Rezayat, P. G. Catera, C. E. Capalbo, F. Cosco and D. Mundo, *Numerical and experimental analysis of the transmission error in hybrid metal-composite gears*, Composite Structures, Volume 298, 2022

G. Carbone, M. Ceccarelli, C. E. Capalbo, G. Caroleo and C. Morales-Cruz, *Numerical and experimental performance estimation for a ExoFing - 2 DOFs finger exoskeleton*, Robotica, Volume 40(6), 2022

Articles in international conferences

C. E. Capalbo, D. De Gregoriis, T. Tamarozzi, H. Devriendt, F. Naets, G. Carbone and D. Mundo, *Structural model updating using parametric model order reduction and augmented Kalman filtering*, Proceedings of ISMA 2022 - International Conference on Noise and Vibration Engineering, 2022

C. E. Capalbo, G. Carbone, F. Cosco, A. Galloro and D. Mundo, *Flexible Multibody Modeling for Structural Flexibility Analysis of a Robotic Manipulator*, Advances in Italian Mechanism Science. IFToMM ITALY 2020. Mechanisms and Machine Science, 2021

C. Morales-Cruz, C. E. Capalbo, G. Caroleo, M. Ceccarelli, G. Carbone, *Numerical and Experimental Validation of ExoFing, a Finger Exoskeleton.* New Trends in Mechanism and Machine Science. EuCoMeS 2020. Mechanisms and Machine Science, 2020