Model-free approaches for modeling and control of robotic systems

Achille Melingui

UNIVERSITY OF Lille

Centre de Recherche en Informatique, Signal et Automatique de Lille (CRIStAL)

for the the obtention of the

Habilitation à Diriger des Recherches (HDR)

Specialty : Automatique et Informatique Industrielle

Defended by

Achille Melingui

Model-free approaches for modeling and control of robotic systems

defended on May 25, 2021

Jury :

Reviewers :	Véronique Perdereau	-	Professor, U. Sorbonne
	Abdelaziz Benallegue	-	Professor, UVSQ
	Taha Boukhobza	-	Professor, U. Lorraine
Advisor :	Rochdi Merzouki	-	Professor, UdLille
Examiners :	Mitra Fouladirad	-	Professor, UT Troyes
	Vincent Cocquempot	-	Professor, UdLille

Acknowledgments

The work presented in this manuscript is the synthesis of nine years of research activities carried out at "Centre de Recherche en Informatique, Signal et Automatique de Lille (Cristal)", "Laboratoire d'Electronique, Electrotechnique et Automatique (EEA)", and "Laboratoire de génie électrique, Mécatronique, et Traitement du Signal (LGEMTS)". I would like to thank the successive directors of the Cristal who welcomed me, first as a Ph.D. student and then during my research stays at the Cristal: Philippe Vanheeghe and Olivier Colot. I would also like to thank the directors of the EEA and LGEMTS laboratories who welcomed me, first as a Ph.D. student and then as part of my recruitment as a Lecturer at the University of Yaoundé 1: Bernard Essimbi Zobo, Martin Kom, and Benoît Ndzana.

I would also like to express my gratitude to Véronique Perdereau, Abdelaziz Benallegue, and Taha Boukhobza, respectively, professor at the University of Sorbonne, professor at the University of Versailles-Saint-Quentin-en-Yvelines, and professor at the University of Lorraine. They have accepted the responsibility of reporting on this work. Thank you for the time you devoted to reading the manuscript and for your questions, advice, and scientific exchanges during the presentation.

I would also like to express my gratitude to the examiners who have given me the honor and friendship of participating in this jury, namely Mitra Fouladirad, Professor at Troyes University of Technology, and Vincent Cocquempot, Professor at the University of Lille.

I would also like to express my deepest gratitude to Rochdi Merzouki, Professor at the University of Lille, for his trust in me by agreeing to be the guarantor of this work. Thank you, Rochdi, for your support, friendship, and qualities as a researcher from which I have benefited more than once.

I want to thank Belkacem Ould Bouamama, professor at the University of Lille and director of the research group "Méthodes et Outils pour la Conception Intégrée de Systèmes" (MOCIS), to welcome me to an exciting research group.

This manuscript could not have been written without the help of my colleagues and successive Ph.D. students. May Joseph Jean-Baptiste Mvogo Ahanda, Othman Lakhal, Pushpendra Kumar, vincent Coelen, Abdelkader Belarouci, Inderjeet Singh, Ismail Bensekrane, Audrey Haycinthe Bouyom Bouatchang, Gino Ambroise Jiokou Kouabon, Imrane Mathamat Loufti, Roger Datouo, and Noé paul Ntouba find here the expression of my gratitude and friendship towards them.

My thanks also go to all my friends: Christian, Dora, Thierry, Emmanuel, Bienvenue, Cyrille, Didier, Ghislain,...

My gratitude also goes to Diane, Alice, Sylvain, Jacqueline, désiré, Jean-Marie, Elisabeth, for their continued support.

Finally, I would like to thank my parents, who have always supported me in my choices and my studies in a profound way.

Contents

List of Acronyms

UY1	University of Yaoundé 1.
EEA	Electronique Electrotechnique et Automatique.
LGEMTS	"Laboratoire de génie électrique, Mécatronique, et Traitement du Signal.
CRIStAL	Centre de Recherche en Informatique, Signal et Automatique de Lille.
LAGIS	Laboratoire d'Automatique, d'Informatique et de Signal.
MOCIS	Méthodes et Outils pour la Conception Intégrée de Systèmes.
FKM	Forward Kinematic Model.
IKM	Inverse Kinematic Model.
T2FL	Type-2 Fuzzy Logic.
MF	Membership Function
IT2FL	Interval Type-2 Fuzzy Logic
IT2FNN	Interval Type-2 Fuzzy Neural Network.
APF	Artificial Potential Field.
CF	Command Filters.
CFBC	Command Filter Backstepping.
ADCFB	Adaptive Command Filter Backstepping.
RADCFB	Robust Adaptive Command Filter Backstepping.
SVR	Support Vector Regressors.
SVM	Support Vector Machines

Préambule

Ce document présente mon activité d'enseignement et de recherche pour la période allant de l'obtention de mon D.E.A. en décembre 2009 jusqu'au début de l'année 2021. Il est organisé en deux parties distinctes. La première résume ces activités, couvrant à la fois les aspects de recherche et d'enseignement. La seconde partie est consacrée à mes activités de recherche centrées autour des approches sans modèles pour la modélisation et la commande des systèmes robotiques. Elle détaille mes contributions sur cette thématique et présente les perspectives envisagées. Enfin, après la bibliographie, une sélection d'articles publiés dans des revues est présentée dans les annexes qui concluent le document.

Preface

This document reports my teaching and research activity over the period from completing my Master's degree (Dec. 2009) to the beginning of 2021. It is organized into two distinct parts. The first one summarizes these activities, covering both research and teaching aspects. The second part focuses on my research activities centered around free-approaches for modeling and control of robotic systems. It details my contributions to this topic and presents the prospects envisaged. Finally, after the bibliography, a selection of published journal papers is reported in the appendices that conclude the document.

Chapter 1

Introduction

Extended Curriculum Vitae

1.1 Short Bio

Achille Melingui is a Senior Lecturer at the National Advanced School of Engineering of the University of Yaoundé 1(UY1), Cameroon, since January 2015. He is affiliated to the Laboratoire de Génie Électrique, Mécatronique et Traitement du Signal (LGEMTS) in Yaoundé, and associated with Centre de Recherche en Informatique, Signal et Automatique de Lille (CRIStAL). He is also responsible of the research team *Robotics and vision* of the Laboratory LGEMTS.

In December 2009, he obtained a Master's degree in apply physics option Electronics from the University of Yaoundé 1, Yaoundé, Cameroon. In September 2014, he received the Ph.D. degree in Automatic and industrial computing from the University of Lille in France, and the University of Yaoundé 1 in Cameroon. He was a guest professor at the University of Lille each year since 2016.

His main research interests include autonomous systems and robotics, with particular regard to model free-approaches. His main areas of expertise are soft robots and mobile robots. He published more than 35 papers in international journals and conferences. A summary of the metrics of his research activity is given below.

Summary of metrics

Journal papers	13 papers: 06 papers with an impact factor 3–10 07 papers an impact factor of 1–2 08 as first author or co-first author 06 as corresponding author Plus 03 papers in first round of review.
Conference papers	18
Chapters	04
Other publications	02
Other metrics	298 citations (Google Scholar) h-index: 11 Journal Reviewer: 16 Conference Reviewer: 06 Invited Talks and Seminars: 15
Research projects	04
Gratuated Ph.D. students	02
Gratuated M.Sc.	10
Ongoing PhD thesis	05
Ongoing Master thesis	05

1.2 Personal Data

Name: Achille
Surname: Melingui
Place of Birth: 15 June 1984, Eyene, Cameroon
Citizenship: Cameroonian
Current position: Senior researcher, since January 2015, ENSP, University of Yaoundé 1. 8390 Rue melen, ENSP, Yaoundé, Cameroon.
Tel: +(237) 6 94 70 56 90,
Email: achillemelingui@gmail.com, achille.melingui@univ-lille.fr

1.3 Education

2010-2014 University of Lille and University of Yaoundé 1. Ph.D in Robotics, thesis entitled: Modeling and control of a class of mobile continuum manipulator robots, case of study: Robotino XT.

2007-2009 University of Yaoundé 1, Cameroon. Master of Science, thesis entitled: Autonomous navigation of mobile robots.

2003-2006 University of Yaoundé 1, Cameroon. Bachelor of Science in Physics.

2003 General government technical school, Nkolbisson, Cameroon. Bachelor degree in Electrical Engineering.

1.4 Academic Experience

From Nov.2016	Senior Lecturer at the Department of Electrical and Telecommunications Engineering, ENSP, University of Yaoundé 1.
Jan.2015-Nov.2016	Lecturer at the Department of Electrical and Telecommunications Engineering, ENSP, University of Yaoundé 1.
Oct.2016–Dec.2016	Guest professor at the university of Lille, France.
Jan.2018–Mar.2018	Guest professor at the university of Lille, France.
Jan.2019–Mar.2019	Guest professor at the university of Lille, France.
Jan.2020–Mar.2020	Guest professor at the university of Lille, France.
Sep.2014–Dec.2014	Postdoctoral researcher at the university of Lille, France.

1.5 Participation in research Projects

2019-2022	SPEED Project: Smart Ports Entrepreneurial Ecosystem Development.
2016–2019	VASCO project: Autonomous Vehicles For Port Logistics.
2015–2017	Matrice project: Additive manufacturing for the construction industry.
2009–2014	InTraDE Project: Intelligent Transportation for dynamic Environment.

1.6 Supervision and Mentoring

1.6.1 Gratuated PhD students

- **Joseph Jean-Baptiste Mvogo Ahanda**, Modeling and Control of Flexible Structure Manipulators, defended in 2018.
- **Othman Lakhal**, Contribution to the modelling and control of hyper-redundant robots: application to additive manufacturing in construction, defended in 2018.

1.6.2 Gratuated M.Sc. and Engineer Students

1. **M.Sc. Students**
    - **Leitniz Tchinda Dadjio**, Adaptive neural backstepping control of mobile robots, defended in 2020.
    - **Noe paul Ntouba**, model predictive control of manipulator robots, defended in 2019.
    - **Bienvenue Ayissi Ngah**, Reinforcement learning for inverse kinematics of manipulator robots, defended in 2019.
    - **Yves Mindzié Abessolo**, Model adaptive reference control of manipulator robots, defended in 2019.
    - **Beaudelaire dezo Segning**, Design and control of a mobile robot using fuzzy logic approach, defended in 2019.

- **Salomon Didier Ngan Bakinde** , Design and control of a 5 dof manipulator robot, defended in 2019.
- **Wilfried Clement MBIA**, Concept of virtual obstacles for path tracking of mobile robots, defended in 2019.

2. **Engineer students**

- **Kevine Lehman Mba Ngolle**, Preventive maintenance of synchronous machines: case study of the songloulou hydro electrical power plant, defended in 2020.
- **Bradon Tebid**, Predictive maintenance of DC motors in robotics, defended in 2020.
- **Elisabeth Adidja Owona Ntsama**, Modelling and control of a pasteurization unit: Case of chain 5 of the Cameroonian Brewery Union (UCB), defended in 2019.

1.7 Publications

Submitted Journal Papers

1. A. G. Jiokou Kouabon, **A. Melingui**, J. M. Ahanda, O. Lakhal, M. KOM, and R. Merzouki, " A Learning Framework to Inverse Kinematics of High DOF Redundant Mobile Manipulators," submitted to *Robotics and autonomous systems*. January 2021.

2. A. H. Bouyom Boutchouang, **A. Melingui**, J. J. B.Mvogo Ahanda, O. Lakhal, F. Biya-Motto, and R. Merzouki, " Learning-Based approach to Inverse kinematics of Wheeled Mobile Continuum Manipulators," submitted to *IEEE/ASME Transactions on Mechatronics*, January 2021.

3. **A. Melingui**, J. J.B. Mvogo Ahanda, O. Lakhal, T. Chettibi, and R. Merzouki, " A Learning-Based Approach to Inverse Kinematics of Continuum Manipulators," submitted to *IEEE Robotics and Automation Letters*, February 2021.

Accepted/Published Journal Papers

1. A. H. Bouyom Boutchouang, **A. Melingui**, J. J. B. Mvogo Ahanda, O. Lakhal, F. Biya Motto, and R. Merzouki," Forward Kinematic Modeling of Conical-Shaped Continuum Manipulators," accepted to *Robotica*, 2021.

2. I. Mahamat Loutfi, **A. Melingui**, J. J. B. Mvogo Ahanda, F. Biya Motto, and R. Merzouki," Artificial Potential Field Neurofuzzy Controller for Autonomous Navigation of Mobile Robots," *Proceedings of the Institution of Mechanical Engineers, Par I: Journal of Systems and Control Engineering*, 2020.

3. R. Datouo, J. J.-B. M. Ahanda, **A. Melingui**, F. Biya-Motto, and B. E. Zobo, " Adaptive Fuzzy Finite-time Command Filtered Backstepping Control of Flexible-Joint Robots," *Robotica*, 2020.

4. A. G. Jiokou Kouabon, **A. Melingui**, J. M. Ahanda, O. Lakhal, V. Coelen, M. KOM, and R. Merzouki, "A learning framework to inverse kinematics of high dof redundant manipulators," *Mechanism and Machine Theory*, vol. 153, p. 103978, 2020.

5. I. Bensekrane, P. Kumar, **A. Melingui**, V. Coelen, Y. Amara, T. Chettibi, and R. Merzouki, "Energy planning for autonomous driving of an overactuated road vehicle," I*EEE Transactions on Intelligent Transportation Systems*, 2020.

6. I. Singh, Y. Amara, **A. Melingui**, P. Mani Pathak, and R. Merzouki,"Modeling of continuum manipulators using pythagorean hodograph curves," *Soft robotics*, vol. 5, no. 4, pp. 425–442, 2018.

7. J. J.-B. M. Ahanda, J. B. Mbede, **A. Melingui**, and B. E. Zobo, "Robust adaptive command filtered control of a robotic manipulator with uncertain dynamic and joint space constraints," *Robotica*, vol. 36, no. 5, p. 767, 2018.

8. J. J.-B. M. Ahanda, J. B. Mbede, **A. Melingui**, and B. Essimbi, "Robust adaptive control for robot manipulators: Support vector regression-based command filtered adaptive backstepping approach," *Robotica*, vol. 36, no. 4, pp. 516–534, 2018.

9. **A. Melingui**, J. J.-B. M. Ahanda, O. Lakhal, J. B. Mbede, and R. Merzouki, "Adaptive algorithms for performance improvement of a class of continuum manipulators," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 9, pp. 1531–1541, 2017.

10. O. Lakhal, **A. Melingui**, and R. Merzouki, "Hybrid approach for modeling and solving of kinematics of a compact bionic handling assistant manipulator," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 3, pp. 1326–1335, 2015.

11. **A. Melingui**, O. Lakhal, B. Daachi, J. B. Mbede, and R. Merzouki, "Adaptive neural network control of a compact bionic handling arm," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 6, pp. 2862–2875, 2015.

12. **A. Melingui**, R. Merzouki, and J. Mbede, "Compact bionic handling arm control using neural networks," *Electronics letters*, vol. 50, no. 14, pp. 979–981, 2014.

13. **A. Melingui**, R. Merzouki, J. B. Mbede, and T. Chettibi, "A novel approach to integrate artificial potential field and fuzzy logic into a common framework for robots autonomous navigation," *Proceedings of the Institution of Mechanical Engineers, Par I: Journal of Systems and Control Engineering*, vol. 228, no. 10, pp. 787–801, 2014.

Accepted/Published Conference Papers

1. G. A. JIOKOU K., **A. Melingui**, O. Lakhal, M. KOM, R. Merzouki, "A learning framework to inverse kinematics of redundant manipulators,"Accepted at IFAC, 2020.

2. I. Mahamat Louti, A.H. Bouyom Boutchouang, **A. Melingui**, O. Lakhal, F. Biya Motto, and R. Merzouki, "Learning-Based Approaches for Forward Kinematic Modeling of Continuum Manipulators," Accepted at *IFAC*, 2020.

3. O. Lakhal, **A. Melingui**, G. Dherbomez, and R. Merzouki, "Control of a hyper-redundant robot for quality inspection in additive manufacturing for construction," *in 2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, 2019, pp. 448–453.

4. I. Bensekrane, P. Kumar, **A. Melingui**, V. Coelen, Y. Amara, and R. Merzouki, "Energy planning for unmanned over-actuated road vehicle," *in 2018 IEEE Vehicle Power and Propulsion Conference (VPPC)*. IEEE, 2018, pp. 1–6.

5. I. Singh, Y. Amara, O. Lakhal, **A. Melingui**, and R. Merzouki, "Ph model-based shape reconstruction of heterogeneous continuum closed loop kinematic chain: An application to skipping rope," *in 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 8682–8688.

6. R. Datouo, F. B. Motto, B. E. Zobo, **A. Melingui**, I. Bensekrane, and R. Merzouki, "Optimal motion planning for minimizing energy consumption of wheeled mobile robots," *in 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2017, pp. 2179–2184.

7. **A. Melingui**, J. M. Ahanda, O. Lakhal, J. Mbede et al., "Performance improvement of a class of continuum manipulators via adaptive algorithms," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 4863–4868, 2017.

8. J. J.-B. M. Ahanda, J. B. Mbede, **A. Melingui**, B. E. Zobo, O. Lakhal, and R. Merzouki, "Robust control for robot manipulators: support vector regression based command filtered adaptive backstepping approach," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 8208–8213, 2017.

9. **A. Melingui**, T. Chettibi, R. Merzouki, and J. B. Mbede, "Adaptive navigation of an omni-drive autonomous mobile robot in unstructured dynamic environments," *in 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2013, pp. 1924–1929.

10. **A. Melingui**, C. Escande, N. Benoudjit, R. Merzouki, and J. B. Mbede, "Qualitative approach for forward kinematic modeling of a compact bionic

handling assistant trunk," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 9353–9358, 2014.

11. **A. Melingui**, R. Merzouki, J. B. Mbede, C. Escande, and N. Benoudjit, "Neural networks based approach for inverse kinematic modeling of a compact bionic handling assistant trunk," *in 2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*. IEEE, 2014, pp.1239–1244.

12. **A. Melingui**, R. Merzouki, and J. B. Mbede, "Neuro-fuzzy controller for autonomous navigation of mobile robots," *in 2014 IEEE Conference on Control Applications (CCA)*. IEEE, 2014, pp. 1052–1057.

13. **A. Melingui**, R. Merzouki, J. B. Mbede, C. Escande, B. Daachi, and N. Benoudjit, "Qualitative approach for inverse kinematic modeling of a compact bionic handling assistant trunk," *in 2014 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2014, pp. 754–761.

14. **A. Melingui**, R. Merzouki, and J. Bosco, " Harmonic Functions-based type-2 fuzzy logic controller for autonomous navigation of mobile robots," *Research in Interactive Design (Vol. 4): Mechanics, Design Engineering and Advanced Manufacturing*, p. 71, 2016.

15. **A. Melingui**, R. Merzouki, and J. Bosco. "Forward kinematics modeling of CBHA," *Actes du CARI 2014 (Colloque africain sur la recherche en informatique et mathématiques appliquées)*. (2014).

16. O. Lakhal, A. Melingui, A. Chibani, C. Escande, and R. Merzouki, "Inverse kinematic modeling of a class of continuum bionic handling arm," *in 2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. IEEE, 2014, pp. 1337–1342.

17. O. Lakhal, **A. Melingui**, T. M. Bieze, C. Escande, B. Conrard, and R. Merzouki, "On the kinematic modeling of a class of continuum manipulators," *in 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*. IEEE, 2014, pp. 368–373.

18. J. B. Mbede, **A. Melingui**, B. E. Zobo, R. Merzouki, and B. O. Bouamama, "zslices based type-2 fuzzy motion control for autonomous robotino mobile robot," *in Proceedings of 2012 IEEE/ASME 8th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications*. IEEE, 2012, pp. 63–68.

Chapter 2

Research Activities

2.1 Introduction

This chapter aims to provide a quick and concise overview of the topics covered by my research activity since my master's thesis in 2009 until the end of 2020. It presents the research themes addressed during my master thesis, my Ph.D. thesis, and after my recruitment at the National Advanced School of Engineering of the UY1 in 2015.

2.2 Master Thesis in physics, option Electronics (Dec. 2009)

I did my master's thesis at the Laboratory of Electronique, Electrotechnique et Automatique (EEA) of the UY1 under the supervision of Jean Bosco Mbede, professor at the UY1. The title of my thesis was "Autonomous navigation of mobile robots." I focused mainly on implementing type-2 fuzzy logic (T2FL) for mobile robots' autonomous navigation in congested environments. Parts of the results obtained were accepted at the IEEE International Fuzzy Systems Conference in 2010.

2.3 Ph.D. Student in Automation and industrial computing (Feb. 2010 – Sep. 2014)

My Ph.D. thesis research activity can be roughly grouped into three phases :

1. **Ph.D. student at the EEA Laboratory, UY1 (Feb.2010 – Jun.2011), Cameroon**

   During this period, I focused on deepening the algorithms developed in the Master's thesis, in particular, by applying fuzzy logic to the intelligent management of a roundabout and learning artificial neural networks.

2. **Ph.D. student Visiting at LAGIS, university of Lille (Sep.2011 – Jun.2012), France**

   The second phase took place from September 2011 to June 2012 at the Laboratoire d'Automatique, d'Informatique et de Signal (LAGIS) of the University of Lille under the supervision of Belcacem Ould-Bouamama, professor of the

University of Lille. I joined this laboratory through an Erasmus Mundus mobility program. The main objective was to implement the Master's thesis algorithms on a real mobile platform.

(a) From September 2011 to January 2012, I implemented the type-2 fuzzy logic algorithms on the Robotino mobile platform. The results obtained were published at the IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications in 2012.

(b) From February 2012 to March 2012, I was interested in Bond and Graph modeling. The objective is to get involved in the MOCIS team's themes led by Belcacem Ould-Bouamama.

(c) From May 2012 to December 2012, I was contacted by Professor Rochdi Merzouki for possible modeling of the inverse kinematics of a class of continuum robots named Compact Bionic Handling Assistant (CBHA) with artificial neural networks. Indeed, the system was strongly non-linear, and the forward kinematic equations obtained were mathematically intractable. Therefore, the use of numerical methods or Learning machine tools became necessary. In collaboration with the Ph.D. student, Coralie Escande, we proposed forward and inverse kinematic models using MLP and distal supervised learning, respectively. Some of the results obtained have been published at IFAC 2014 [Melingui *et al.* 2014c], International Joint Conference on Neural Networks in 2014 [Melingui *et al.* 2014g], and IEEE 23rd International Symposium on Industrial Electronics in 2014 [Melingui *et al.* 2014f].

3. **Ph.D. student at LAGIS (Jan.2013 – Sep.2014), France**

The third phase of the thesis work took place from January 2013 to September 2014, notably after a co-tutorship agreement between the University of Lille 1 in France and UY1 in Cameroon. Therefore, my thesis work was under the supervision of Rochdi Merzouki, Professor of the University of Lille, and Jean Bosco Mbede, Professor of the UY1. My thesis's title was **Modelling and control of a class of mobile Continuum manipulator robots, case of study Robotino XT**. It was necessary to control not only the CBHA but also the entire mobile manipulator XT. With the experience gained before, the control of the mobile part was straightforward. We thought about the hybridization of artificial potential fields and type-2 fuzzy logic. The Artificial Potential Field (APF) is quick and easy to implement but faces local minima and robustness problems. Type-2 fuzzy logic is robust, free from local minima, but computationally intensive. The objective was to exploit the best of them, namely the computationally efficient of the APF method and the capability of fuzzy logic methods to achieve navigation behaviors without local minima issues. Part of the results obtained has been published in [Melingui *et al.* 2014e], and [Melingui *et al.* 2013a]. The manipulative part was the most challenging because although the forward and inverse models were developed

[Melingui *et al.* 2014c, Melingui *et al.* 2014g, Melingui *et al.* 2014f], the control was still difficult. Indeed, the platform inherited undesirable effects, such as the pneumatic actuators' hysteresis effect and the memory effect of the component material (polyamide). Therefore, for the same pressure input, different robot configurations could be associated. Consequently, point tracking required on-line correction. We proposed neural control architecture with two correction loops. The first one is related to the correction of the kinetic model of the CBHA, which consisted of overcoming the memory effect. The second, on the other hand, should reduce positioning errors. Parts of the results obtained were published in [Melingui *et al.* 2015, Melingui *et al.* 2014a]. The control of the mobile manipulator was proposed in [Melingui 2014].

2.4 Research topics

Our research interest focuses on autonomous navigation of mobile robots, modelling and control of rigid manipulators, and modelling and control of continuum manipulators. The following sub-sections provide a summary of these research interests.

2.4.1 Autonomous navigation of mobile robots

The issue of autonomous navigation of mobile robots is a topical and widely studied subject. The problem consists - roughly speaking - in bringing a mobile robot from a starting point to a destination point. Therefore, the robot must [Remazeilles 2004]:

- have sensors that give it information about its environment,
- have sensors enabling it to locate itself,
- be able to decide on the actions to be taken to reach the target,
- be able to carry out these actions by controlling one's movements correctly.

Navigation algorithms generally aim to give the robot the necessary perception, decision, and action capabilities to perform the assigned tasks. These algorithms are usually divided into two main classes, depending on whether the problem is solved locally or globally. In global approaches, it is assumed that more or less a priori knowledge of the environment is available. In contrast, it is assumed that no global knowledge is available a priori on the environment for local approaches. Therefore, the robot will evolve and decide on its movements according to the desired goal and the sensory data received as it moves towards the target [Bonin-Font *et al.* 2008].

So far, our contributions in this theme have focused on applying local approaches to mobile robots' autonomous navigation. We have proposed several contributions in this theme, using the artificial potential field technique, fuzzy logic, artificial neural networks, or a combination of these [Melingui *et al.* 2014e, Mbede *et al.* 2012, Melingui *et al.* 2014d, Mahamat Loufti *et al.* 2020]. The basic concepts and the proposed algorithms will be developed in chapter 4.

We have recently been interested in global approaches, mainly through the planning of mobile robots' trajectories with minimization of the energy consumed [Datouo *et al.* 2017].

The prospects in this topic aim at developing pythagorean-hodograph curves-based type-2 fuzzy logic algorithms. The idea is to propose controllers whose profile of the generated velocities (resulting trajectories) would minimize the consumed energy. Because until now, the fuzzy rules established were centered-experts. They depend on the expert's experience on the system to be controlled and understand the treated problematic. A thesis is currently being developed around this problematic. We are also interested in the trajectory tracking of autonomous systems based on the artificial potential field.

2.4.2 Modelling and control of rigid manipulators

Our work on this topic covers both the modeling and control aspects. In modeling, we are interested in modeling the inverse kinematics of redundant manipulators [Kouabon *et al.* 2020a, Jiokou K *et al.* 2020, Kouabon *et al.* 2020b]. The aim is to propose algorithms that can be implemented in real-time and allow maximum exploitation of their characteristics, particularly the exploitation of redundancy for the execution of secondary tasks such as the avoidance of singularities, the avoidance of obstacles, the avoidance of joint limits. A thesis is being completed in this area. The prospects in this topic aim to extend the algorithms proposed to the case of mobile manipulators and the cooperation of several mobile manipulators for the execution of common tasks such as the transport of bulky objects.

The contributions on the control of rigid manipulators are concentrated around the control of flexible joint manipulators. So far, our contributions have focused on deterministic cases [Ahanda *et al.* 2017, Ahanda *et al.* 2018a, Ahanda *et al.* 2018b], i.e., when the robot environment is stable and known. However, when this environment is unknown or subjected to random vibrations, these methods can no longer ensure good tracking accuracy because of unmodelled dynamics resulting from the vibrations. The prospects on this topic are aimed at handling random vibration environments. The proposed algorithms will be developed in chapter 6.

2.4.3 Modelling and control of continuum manipulators

This theme is the continuation of our thesis work. Our contributions to this theme cover both the modeling and control aspects. Several contributions have been proposed covering both modelling [Melingui 2014, Melingui *et al.* 2014c, Melingui *et al.* 2014g, Singh *et al.* 2018b, Lakhal *et al.* 2015, Melingui *et al.* 2019] and control [Melingui *et al.* 2015, Melingui *et al.* 2017b, Melingui *et al.* 2017a]. Our contributions to date have focused on modeling and the proposal of control architectures. The prospects in this area aim to extend the algorithms developed to the case of mobile continuum manipulators [Bouyom Boutchouang *et al.* 2020], the use of deep learning [Bouyom Boutchouang *et al.* 2020], and the reinforcement learning.

The proposed algorithms will be further developed in chapter 5.

2.5 Participation in research projects

This section summarizes my involvement in past and current projects.

- **SPEED Project: Smart Ports Entrepreneurial Ecosystem Development**

  I participated in the European project SPEED (Smart Ports Entrepreneurial Ecosystem Development), within the framework of the societal challenge Port of the Future as a guest researcher (between Jan. 2019 and Mar. 2019). This project aims to integrate new advances in data sciences and Internet of Things (IoT) technologies. In this project, I worked on the energy planning of autonomous vehicles developed in the framework of the European project InTraDE (Intelligent Transportation for Dynamic Environment).

- **VASCO project: Autonomous Vehicles For Port Logistics, 2016–2019**

  This project is of type PSPC (Projet Structurant des Pôles de Compétitivités) of BPI France, with the participation of the companies Gaussin and BA Systèmes and the laboratory LS2N-CNRS. I participated in this project as a guest researcher (between Jan. 2019 and Mar. 2019). The VASCO project aims to develop a concept of autonomous container-carrying vehicles, operating in an unstructured environment, to carry out logistics operations. This project is part of the Port of the Future societal challenge. Within this project's framework, I worked on the estimation of the energy consumption of autonomous container door vehicles. My participation in this project has resulted in the following publications: [Bensekrane *et al.* 2020, Bensekrane *et al.* 2018].

- **Matrice project: Additive manufacturing for the construction industry, 2015–2017**

  ERDF-Region funds funded this project on additive printing in construction. I participated in this project as a guest researcher (between Oct.2016 and Dec.2016). CRIStAL was in charge of developing an integrated concept of mobile robots and continuum manipulators, dedicated to 3D printing in construction, as well as quality control (Figure 4). In this context, I participated in the co-supervision of the thesis of Mr. Othman Lakhal's thesis. This project is part of the societal issue Construction of the future. My participation in this project has resulted in the following publications: [Lakhal *et al.* 2019, Lakhal *et al.* 2015].

- **InTraDE project: Intelligent Transportation for Dynamic Environment, 2009-2015**

This project was co-financed by the INTERREG IV-B northwest Europe (NWE) program. The project aimed to contribute to the socio-economic development of ports located in North-West Europe. I participated in this project as a Ph.D. student (between 2011 and 2014) and as a post-doc (between September 2014 and January 2015). My work was part of the W3Ap action, entitled: Transferability of the European project Intrade. It concerned developing strategies for the autonomous control of Omni-drive intelligent autonomous vehicles, allowing the handling and routing of loads in confined and indoor environments. Thus, the robot RobotinoXT was used as a didactic demonstrator of autonomous transport in a confined space. My participation in this project has resulted in the following publications: [Melingui *et al.* 2014e, Melingui *et al.* 2014a, Melingui *et al.* 2015, Lakhal *et al.* 2015, Melingui *et al.* 2017b, Melingui *et al.* 2013a, Melingui *et al.* 2014d, Melingui *et al.* 2017a, Melingui *et al.* 2014b, Melingui *et al.* 2014f].

2.6 Supervision and Mentoring

2.6.1 Research activies in bref

1. **Posdoctoral fellow (Sep.2014 – Dec. 2014)**

   Hybrid approach for modelling and solving the kinematics of the CBHA

   My postdoctoral research activities began with a postdoctoral contract award from September 2014 to December 2014 at the University of Lille. During this period, I mainly worked with Ph.D. student Othman Lakhal. His thesis work focused on *Contribution to the modeling and control of hyper-redundant robots: application to additive manufacturing in construction*. We contributed, in particular, on the inspection of printed objects. The flexibility of the CBHA continuum manipulator was exploited to inspect hard-to-reach areas. The thesis was defended in November 2018. Part of the results obtained has been published in [Lakhal *et al.* 2019]. The modeling of the CBHA consisted of developing a hybrid approach for the inverse kinematics of continuum manipulators. Initially, a quantitative approach was used to model the CBHA kinematically, inspired by parallel rigid manipulators' modeling. The CBHA was modeled as a vertebrae series. Each vertebra is connected to the next by a flexible link; the latter called an intervertebral being modeled by a 3UPS-1UP (Universal-Prismatic-Spherical) joint. The kinematic models of the CBHA were derived from the inverse kinematic equations (IKE) of each intervertebral. A qualitative approach based on neural networks was then used to provide approximate solutions of the IKEs for real-time implementation. Parts of the results obtained were published in [Lakhal *et al.* 2015, Lakhal *et al.* 2014a].

2. **Lecturer at the UY1 in January 2015, Cameroon**

(a) **Adaptive algorithms for improving the performance of a class of continuum manipulators**

After my recruitment at the UY1, I became interested in improving the adaptive control architecture developed during my Ph.D. thesis. Since neural networks face local minima problems, we proposed a new adaptive control scheme, namely the adaptive support vector regressor controller. This approach exploits optimization learning methods that allow obtaining global solutions to the training problem while keeping small-sized regressors. These features accelerate the convergence of the closed-loop system, thus reducing the execution time.

(b) **Command filtered adaptive backstepping control of flexible joint manipulators**

I supervised the Ph.D. student, Joseph Jean-Baptiste Mvogo Ahanda, who was working on *the modeling and control of flexible manipulators*. The latter was initially under the supervision of Jean Bosco Mbede, professor at the UY1; after his death in March 2017, I had the enormous responsibility of taking over his supervision. The Ph.D. dissertations of several others, including Gino Ambroise Jiokou Kouabon, Audrey Haycinthe Bouyom Bouatchang, Imrane Mathamat Loufti, and Charles Medzo Aba are in progress; they will defend their Ph.D. thesis between September 2021 and March 2022. Regarding Joseph, we were interested in Command filtered adaptive backstepping control of flexible joint manipulators. Two novel control laws were proposed; namely, Support vector regression-based command filtered adaptive backstepping approach and Robust adaptive Command filtered control with uncertain dynamic and joint space constraints. Parts of the results obtained have been published in [Ahanda *et al.* 2018b, Ahanda *et al.* 2018c]. The thesis was defended in November 2018. A novel Ph.D. thesis on this topic that of Charles is in progress where the robot environment is assumed stochastic. Parts of the results obtained are under review in [Medzo Aba *et al.* 2020].

3. **Guest Professor at the University of Lille, Oct.2016 – Dec.2016**,

   Adaptive algorithms for improving the performance of continuum manipulators.

   During my stay, in parallel with the teaching tasks in the MRT Master's program, I implemented the novel control architecture developed earlier on adaptive algorithms for improving the performance of a class of continuum manipulators. Parts of the results obtained were published in [Melingui *et al.* 2017b, Melingui *et al.* 2017a].

4. **Guest Professor at University of Lille, Jan.2018 – Mar.2018**

   Modeling of Continuum Manipulators Using Pythagorean-Hodograph Curves

During my stay, in parallel with the teaching task, I worked with the Ph.D. student Inderjeet Singh, on the modeling of continuum robots using Pythagorean-hodograph curves. Continuum robots have inherent flexibility due to their flexible structure, making it very difficult to control them with high performance. Before developing a control strategy for such robots, it is essential to first reconstruct the robot's behavior by developing a behavioral model. The project consisted of developing a quantitative modeling method based on Pythagorean-hodograph curves. The objective was to obtain a three-dimensional reconstruction of the shape of the continuum manipulator with variable curvature, allowing the calculation of its inverse kinematic model (IKM). Part of the results obtained were published in [Singh *et al.* 2018b, Singh *et al.* 2018a]. The thesis was defended in November 2018.

5. **Guest Professor at University of Lille, Jan.2019 – Mar.2019**

   Contribution to energy planning for the autonomous driving of an over-activated road vehicle

   During my stay, in parallel with the teaching tasks, I worked with the Ph.D. student Ismail Bensekrane on the modeling of energy consumption of over-actuated unmanned road vehicles (URVs) using neuroadaptive fuzzy control architecture. An energy planning strategy for URVs with redundant steering configurations was proposed. In fact, for each segment of the URV's trajectory, indicators on road geometry, actuation redundancy, optimal speed profile, and driving mode were evaluated. The objective was achieved by modeling the energy consumption of the URVs. However, due to the presence of unknown dynamic parameters of the URVs and uncertainties about its interaction with the environment, data-driven-based methods were adopted for the estimation of energy consumption, namely the Adaptive Neuro-Fuzzy Inference System (ANFIS). The ANFIS model was obtained using trained data from the dynamics of a real URV. Some of the results obtained were also published in [Bensekrane *et al.* 2020, Bensekrane *et al.* 2018]. The thesis was defended in July 2019.

6. **Guest Professor at University of Lille, Jan.2020 – Mar.2020**

   (a) **A learning-based approach to inverse kinematics of continuum manipulators**

   During my stay, I focused on developing a general method for deriving the inverse kinematics of continuum manipulators. We proposed a new learning framework that learns and maintains multiple inverse kinematic (IK) solutions of multiple section continuum manipulators. The infinite number of IK solutions has been reduced to a finite number by discretizing the robot's operating space. Besides, the IK problem of multi-section continuum manipulators was transformed into single-section manipulators

by parameterizing the actuation variables of the first (n-1)-sections. The parameterization is performed by clustering the manipulator workspace using the growing neural gas network (GNG) algorithm. The proposed method was computationally efficient and can be applied to any continuum manipulator, regardless of the number of sections. Some of the results obtained are under review in [Melingui *et al.* 2019]. The extension of the approach to the case of mobile continuum manipulator robots, developed by one of my Ph.D. students, is also under review in [Bouyom Boutchouang *et al.* 2020].

(b) **A learning framework for the inverse kinematics of redundant high dof manipulators**

A similar approach was also applied to inverse kinematics of redundant rigid manipulators in collaboration with my Ph.D. student Gino Ambroise Jiokou Kouabon. Indeed, several joint angle vectors can be associated with a given end-effector (EE) pose of a redundant rigid manipulator. However, for a given EE pose, if a set of joint angles is parameterized, the IK problem of redundant manipulators can be reduced to that of non-redundant ones so that the closed-form analytical methods developed for non-redundant manipulators can be applied to obtain the IK solution. In this work, certain redundant manipulator joints are parameterized by clustering the redundant manipulator's workspace and configuration space. Growing Neural Gas Network (GNG) is used for workspace clustering, while a Neighbourhood Function (NF) is introduced in the configuration space clustering. Some of the results obtained have been published in [Kouabon *et al.* 2020a, Jiokou K *et al.* 2020]. The extension of the approach to mobile rigid manipulator robots developed by the same Ph.D. student is under review in [Kouabon *et al.* 2020b].

During this stay, I also investigated the optimal control of mobile continuum manipulators. The idea is to make a mobile continuum manipulator evolve in constrained environments by using each time the configuration of the mobile manipulator (inverse solution), which minimizes the energy consumed. The algorithms are in development.

2.6.2 Gratuated PhD students

Past PhD thesis supervision

- I supervised 50% of the thesis of **Joseph Jean-Baptiste Mvogo Ahanda**, with Bernard Essimbi Zobo, Professor at the UY1, from March 2017 to December 2018. This thesis dealt with the adaptive control of flexible manipulators using mainly the Backstepping approach. This work was concretised by two international conferences [Ahanda *et al.* 2017, Melingui *et al.* 2017a] and three international journals [Melingui *et al.* 2017b, Ahanda *et al.* 2018b, Ahanda *et al.* 2018c]. J. J.B. Mvogo Ahanda is now a lecturer at the Univer-

sity of Bamenda in Cameroon.

- I also supervised 25% of the thesis of **Othman Lakhal**, with Rochdi Merzouki, Professor at the University of Lille, from September 2015 to November 2018. This thesis dealt with the Contribution to the modeling and control of hyper-redundant robots: application to additive manufacturing in construction. This work was concretised by four international conferences [Lakhal *et al.* 2014c, Lakhal *et al.* 2014b, Lakhal *et al.* 2014a, Lakhal *et al.* 2019] and an international journal [Lakhal *et al.* 2015]. Othman Lakhal is now a lecturer at the University of Lille in France.

- I also contributed to the thesis of **Inderjeet Singh**, defended in November 2018, with Rochdi Merzouki, Professor at the university. The thesis dealt with the modeling of continuum robots using Pythagorean hodograph curves. My modest contribution to this work was concretised by an international conference [Singh *et al.* 2018a] and an international journal [Singh *et al.* 2018b]. Inderjeet Singh is now a post-doc at the University of Texas at Arlington in the USA.

- In the same vein, I contributed to the thesis work of **Ismail Bensekrane**, defended in July 2019, with Rochdi Merzouki, Professor at the university. The thesis focused on towards and Energy Planning strategy for Autonomous driving of Over-actuated road vehicles. My modest contribution to this work has been concretised by an international conference [Bensekrane *et al.* 2018] and an international journal [Bensekrane *et al.* 2020]. Ismail Bensekrane is currently serving at the military school in Algiers, Algeria.

Ongoing PhD students

- Since September 2017: I am 50% co-supervising the thesis of **Gino Ambroise Jiokou Kouabon**, with Martin Kom, Professor at the UY1. This thesis deals with the inverse kinematics of redundant manipulators. It aims at proposing a computationally efficient approach for the kinematics of redundant manipulators. The results obtained have been published in an international conference [Jiokou K *et al.* 2020], an international journal [Kouabon *et al.* 2020a], and others are under revision in [Kouabon *et al.* 2020b]. The thesis defense is scheduled for September 2021.

- Since September 2017: I am 50% supervising **Audrey Haycinthe Bouyom Bouatchang**'s thesis, with Fréderic Biya-Motto, Professor at the UY1, which deals with the inverse kinematics of continuum mobile manipulators. The objective of the thesis is to propose a general and computationally efficient approach for the kinematics of mobile continuum manipulators. The results obtained are under review in three international journals [Melingui *et al.* 2019, bou , Bouyom Boutchouang *et al.* 2020]. The thesis defense is scheduled for December 2021.

- Since September 2017: I am 50% co-supervising **Imrane Mahamat Loufti**'s thesis with Fréderic Biya-Motto, Professor at the UY1, which deals with the autonomous navigation of continuous manipulator mobiles. The thesis aims at proposing a computationally efficient navigation strategy for mobile continuum manipulators. The results obtained have been published in an international conference [Mahamat L *et al.* 2020], and others are under review in an international journal [Mahamat Loufti *et al.* 2020]. The thesis defense is scheduled for January 2022.

- Since September 2018: I am 33% supervising **Charles Medzo Aba**'s thesis, with Joseph Jean-Baptiste Mvogo Ahanda and Bernard Essimbi Zobo, respectively, Lecturer at the University of Bamenda and Professor at the UY1. This thesis deals with the control of flexible joint manipulators operating in a stochastic environment. It is a continuation of Joseph's Ph.D. work on the control of flexible joint manipulators. The results obtained are under review in an international journal [Medzo Aba *et al.* 2020]. The thesis defense is scheduled for March 2022.

- Since September 2018: I have been supervising 25% of **Roger Datouo**'s thesis, with Joseph Jean-Baptiste Mvogo Ahanda, Fréderic Biya-Motto, and Bernard Essimbi Zobo, Lecturer at the University of Bamenda and Professors at the UY1, which deals with the control of mobile manipulators with flexible joints. The results obtained were published in an international conference [Datouo *et al.* 2017] and an international journal [Roger *et al.* 2020]. The thesis defense is scheduled for June 2021.

2.6.3 Gratuated M.Sc. and Engineer Students

I also supervised several master's and engineering students at the UY1 :

1. **M.Sc. Students**

   - **Patrick Valery OYONO EBOGO**, Concept of virtual obstacles for path tracking of autonomous vehicles, defended in 2020.
   - **Noe paul Ntouba**, model predictive control of manipulator robots, defended in 2019.
   - **Bienvenue Ayissi Nga**, Reinforcement learning for inverse kinematics of manipulator robots, defended in 2019.
   - **Yves Mindzié Abessolo**, Model adaptive reference control of manipulator robots, defended in 2019.
   - **Beaudelaire dezo Segning**, Design and control of a mobile robot using fuzzy logic approach, defended in 2019.
   - **NGAN BAKINDE Salomon Didier** , Design and control of a 5 dof manipulator robot, defended in 2019.

- **Wilfried Clement MBIA**, Concept of virtual obstacles for path tracking of mobile robots, defended in 2019.

2. **Engineer students**

- **Kevine Lehman Mba Ngolle**, Preventive maintenance of synchronous machines: case study of the songloulou hydro electrical power plant, defended in 2020.
- **Bradon Tebid**, Predictive maintenance of DC motors in robotics, defended in 2020.
- **Elisabeth Adidja OWONA NTSAMA**, Modelling and control of a pasteurization unit: Case of chain 5 of the Cameroonian Brewery Union (UCB), defended in 2019.

2.6.4 Ongoing M.Sc. Students

- **Leitniz Tchinda Dadjio**, Adaptive neural backstepping control of mobile robots.
- **Eddy Jordan Mbetankou**, Adaptive neural sliding mode control of manipulator robots.
- **Brice arthur Kodjouo**, Adaptive neural backstepping control of manipulator robots.
- **MALIK ABDEL MOUAJI NJIKAM**, Adaptive neural backstepping control of mobile manipulator robots.
- **Noe Toukea Ondoua**, Algorithms for integrating photovoltaic energy into the electricity grid.
- **Jean Honore Kadake**, Home Automation using Internet of Things.

2.7 Editorial Service and Committees

2.7.1 Journal Reviewer

- IEEE Transactions on Systems, Man and Cybernetics
- IEEE Transaction on Mechatronics
- IEEE Robotics and Automation Letters
- IEEE Transactions on Industrial Informatics
- Journal of Ambient Intelligence & Humanized Computing (AIHC)
- Part I: Journal of Systems and Control Engineering

- Engineering Applications of Artificial Intelligence
- Robotica
- Robotic Autonomous Systems
- Electronics Letters
- Journal of Intelligent & Robotic Systems
- Advanced Robotics

2.7.2 Conference Reviewer

- IEEE ICRA; IEEE IROS
- IEEE AIM; IFAC WC.

2.7.3 Publications

2.7.4 Publications after my PhD thesis

Submitted Journal Papers

1. A. G. Jiokou Kouabon, **A. Melingui**, J. M. Ahanda, O. Lakhal, M. KOM, and R. Merzouki, " A Learning Framework to Inverse Kinematics of High DOF Redundant Mobile Manipulators," submitted to *Robotics and autonomous systems*. January 2021.

2. A. H. Bouyom Boutchouang, **A. Melingui**, J. J. B.Mvogo Ahanda, O. Lakhal, F. Biya-Motto, and R. Merzouki, " Learning-Based approach to Inverse kinematics of Wheeled Mobile Continuum Manipulators," submitted to *IEEE/ASME Transactions on Mechatronics*, January 2021.

3. **A. Melingui**, J. J.B. Mvogo Ahanda, O. Lakhal, T. Chettibi, and R. Merzouki, " A Learning-Based Approach to Inverse Kinematics of Continuum Manipulators," submitted to *IEEE Robotics and Automation Letters*, February 2021.

Accepted/Published Journal Papers

1. A. H. Bouyom Boutchouang, **A. Melingui**, J. J. B. Mvogo Ahanda, O. Lakhal, F. Biya Motto, and R. Merzouki," Forward Kinematic Modeling of Conical-Shaped Continuum Manipulators," accepted to *Robotica*, 2021.

2. I. Mahamat Loutfi, **A. Melingui**, J. J. B. Mvogo Ahanda, F. Biya Motto, and R. Merzouki," Artificial Potential Field Neurofuzzy Controller for Autonomous Navigation of Mobile Robots," accepted to *Proceedings of the Institution of Mechanical Engineers, Par I: Journal of Systems and Control Engineering*, 2020.

3. R. Datouo, J. J.-B. M. Ahanda, **A. Melingui**, F. Biya-Motto, and B. E. Zobo, " Adaptive Fuzzy Finite-time Command Filtered Backstepping Control of Flexible-Joint Robots," *Robotica*, 2020.

4. A. G. Jiokou Kouabon, **A. Melingui**, J. M. Ahanda, O. Lakhal, V. Coelen, M. KOM, and R. Merzouki, "A learning framework to inverse kinematics of high dof redundant manipulators," *Mechanism and Machine Theory*, vol. 153, p. 103978, 2020.

5. I. Bensekrane, P. Kumar, A. Melingui, V. Coelen, Y. Amara, T. Chettibi, and R. Merzouki, "Energy planning for autonomous driving of an overactuated road vehicle," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

6. I. Singh, Y. Amara, **A. Melingui**, P. Mani Pathak, and R. Merzouki,"Modeling of continuum manipulators using pythagorean hodograph curves," S*oft robotics*, vol. 5, no. 4, pp. 425–442, 2018.

7. J. J.-B. M. Ahanda, J. B. Mbede, **A. Melingui**, and B. E. Zobo, "Robust adaptive command filtered control of a robotic manipulator with uncertain dynamic and joint space constraints," *Robotica*, vol. 36, no. 5, p. 767, 2018.

8. J. J.-B. M. Ahanda, J. B. Mbede, **A. Melingui**, and B. Essimbi, "Robust adaptive control for manipulator robots: Support vector regression-based command filtered adaptive backstepping approach," *Robotica*, vol. 36, no. 4, pp. 516–534, 2018.

9. **A. Melingui**, J. J.-B. M. Ahanda, O. Lakhal, J. B. Mbede, and R. Merzouki, "Adaptive algorithms for performance improvement of a class of continuum manipulators," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 9, pp. 1531–1541, 2017.

10. O. Lakhal, **A. Melingui**, and R. Merzouki, "Hybrid approach for modeling and solving of kinematics of a compact bionic handling assistant manipulator," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 3, pp. 1326–1335, 2015.

Accepted/Published Conference Papers

1. G. A. Jiokou K., **A. Melingui**, O. Lakhal, M. KOM, R. Merzouki, "A learning framework to inverse kinematics of redundant manipulators,"Accepted at *IFAC*, 2020.

2. I. Mahamat Louti, A.H. Bouyom Boutchouang, **A. Melingui**, O. Lakhal, F. Biya Motto, and R. Merzouki, "Learning-Based Approaches for Forward Kinematic Modeling of Continuum Manipulators," Accepted at *IFAC*, 2020.

3. O. Lakhal, **A. Melingui**, G. Dherbomez, and R. Merzouki, "Control of a hyper-redundant robot for quality inspection in additive manufacturing for

construction," *in 2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, 2019, pp. 448–453.

4. I. Bensekrane, P. Kumar, **A. Melingui**, V. Coelen, Y. Amara, and R. Merzouki, "Energy planning for unmanned over-actuated road vehicle," *in 2018 IEEE Vehicle Power and Propulsion Conference (VPPC)*. IEEE, 2018, pp. 1–6.

5. I. Singh, Y. Amara, O. Lakhal, **A. Melingui**, and R. Merzouki, "Ph model-based shape reconstruction of heterogeneous continuum closed loop kinematic chain: An application to skipping rope," *in 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 8682–8688.

6. R. Datouo, F. B. Motto, B. E. Zobo, **A. Melingui**, I. Bensekrane, and R. Merzouki, "Optimal motion planning for minimizing energy consumption of wheeled mobile robots," *in 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2017, pp. 2179–2184.

7. **A. Melingui**, J. M. Ahanda, O. Lakhal, J. Mbede et al., "Performance improvement of a class of continuum manipulators via adaptive algorithms," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 4863–4868, 2017.

8. J. J.-B. M. Ahanda, J. B. Mbede, **A. Melingui**, B. E. Zobo, O. Lakhal, and R. Merzouki, "Robust control for manipulator robots: support vector regression based command filtered adaptive backstepping approach," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 8208–8213, 2017.

2.7.5 Publications during my PhD thesis

Published Journal Papers

1. **A. Melingui**, O. Lakhal, B. Daachi, J. B. Mbede, and R. Merzouki, "Adaptive neural network control of a compact bionic handling arm," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 6, pp. 2862–2875, 2015.

2. **A. Melingui**, R. Merzouki, and J. Mbede, "Compact bionic handling arm control using neural networks," *Electronics letters*, vol. 50, no. 14, pp. 979–981, 2014.

3. **A. Melingui**, R. Merzouki, J. B. Mbede, and T. Chettibi, "A novel approach to integrate artificial potential field and fuzzy logic into a common framework for robots autonomous navigation," *Proceedings of the Institution of Mechanical Engineers, Par I: Journal of Systems and Control Engineering*, vol. 228, no. 10, pp. 787–801, 2014.

Published Conference Papers

1. **A. Melingui**, T. Chettibi, R. Merzouki, and J. B. Mbede, "Adaptive navigation of an omni-drive autonomous mobile robot in unstructured dynamic environments," i*n 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2013, pp. 1924–1929.

2. **A. Melingui**, C. Escande, N. Benoudjit, R. Merzouki, and J. B. Mbede, "Qualitative approach for forward kinematic modeling of a compact bionic handling assistant trunk," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 9353–9358, 2014.

3. **A. Melingui**, R. Merzouki, J. B. Mbede, C. Escande, and N. Benoudjit, "Neural networks based approach for inverse kinematic modeling of a compact bionic handling assistant trunk," *in 2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*. IEEE, 2014, pp.1239–1244.

4. **A. Melingui**, R. Merzouki, and J. B. Mbede, "Neuro-fuzzy controller for autonomous navigation of mobile robots," *in 2014 IEEE Conference on Control Applications (CCA)*. IEEE, 2014, pp. 1052–1057.

5. **A. Melingui**, R. Merzouki, J. B. Mbede, C. Escande, B. Daachi, and N. Benoudjit, "Qualitative approach for inverse kinematic modeling of a compact bionic handling assistant trunk," *in 2014 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2014, pp. 754–761.

6. **A. Melingui**, R. Merzouki, and J. Bosco, " Harmonic Functions-based type-2 fuzzy logic controller for autonomous navigation of mobile robots," *Research in Interactive Design (Vol. 4): Mechanics, Design Engineering and Advanced Manufacturing*, p. 71, 2016.

7. **A. Melingui**, R. Merzouki, and J. Bosco. "Forward kinematics modeling of CBHA, " *Actes du CARI 2014 (Colloque africain sur la recherche en informatique et mathématiques appliquées)*. (2014).

8. O. Lakhal, A. Melingui, A. Chibani, C. Escande, and R. Merzouki, "Inverse kinematic modeling of a class of continuum bionic handling arm," *in 2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. IEEE, 2014, pp. 1337–1342.

9. O. Lakhal, **A. Melingui**, T. M. Bieze, C. Escande, B. Conrard, and R. Merzouki, "On the kinematic modeling of a class of continuum manipulators," *in 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*. IEEE, 2014, pp. 368–373.

10. J. B. Mbede, **A. Melingui**, B. E. Zobo, R. Merzouki, and B. O. Bouamama, "zslices based type-2 fuzzy motion control for autonomous robotino mobile robot," *in Proceedings of 2012 IEEE/ASME 8th IEEE/ASME International*

Conference on Mechatronic and Embedded Systems and Applications. IEEE, 2012, pp. 63–68.

Chapter 3

Teaching Activities

3.1 Introduction

This chapter aims to give a quick and concise overview of the teaching units covered by my teaching activity since I registered for my doctoral thesis in 2010. We present the teaching units dispensed during my doctoral dissertation and after my recruitment to the National Higher School of Engineering of the UY1 in 2015.

3.2 Previous Teachings

My teaching activities dated back to the beginning of my doctoral thesis registration in 2010. I started as a temporary teacher with the Physics Department of the Faculty of Science from February 2010 to June 2011, for an annual hourly volume of 156 hours of practical work. Then, I worked as a temporary teacher from September 2010 to June 2011 in the mathematics and physical sciences department at the National Higher Polytechnic School and the Higher Teacher Training College of the UY1 for an annual hourly volume of 56 hours and 72 hours of tutorials, respectively. The teaching during this period is summarised in Figure 3.1.

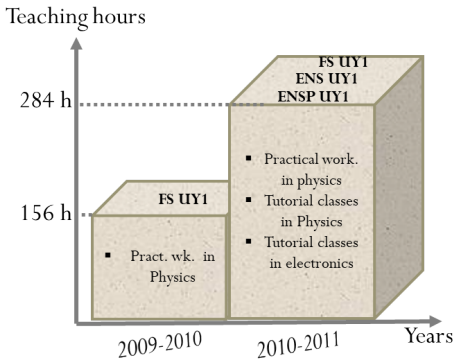

Figure 3.1: Synthesis of teachings 2009-2011

3.3 Current Teachings

Since my recruitment in 2015, my annual teaching costs are growing steadily. Nevertheless, they have stabilized over the last two years. An important aspect concerning

my teaching activities is the evolution of the courses taught over the years, as shown in Figure 3.2. It shows that, in the years immediately following my recruitment, my teaching was more oriented towards electronics and telecommunications. However, I gradually abandoned some of them in favor of the teachings of section 61, a section grouping together the fields of automatic control, industrial computing, and signal processing. For example, the "modern control," "artificial intelligence," and "automatic control" teaching units are part of my research activities, and this enables me to deliver teaching adapted to the evolution of research.

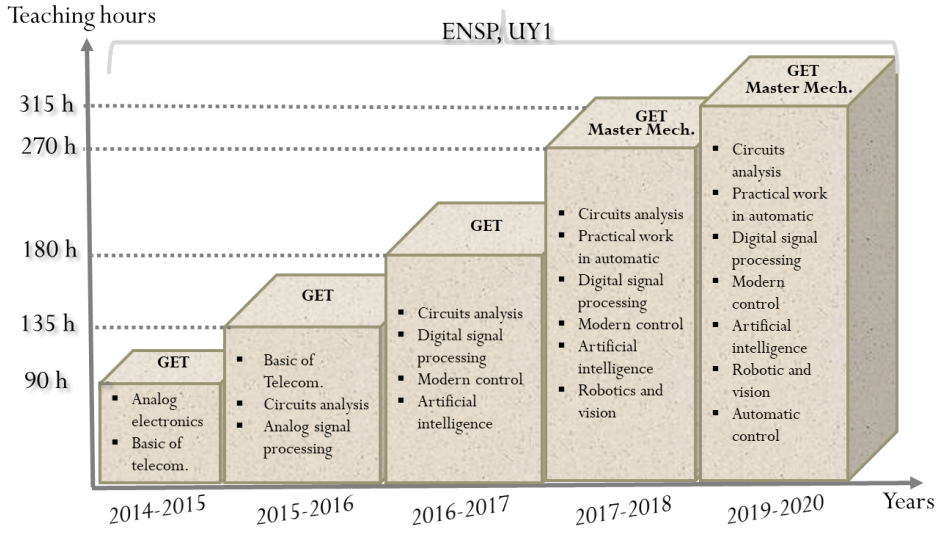

Figure 3.2: Synthesis of teachings since 2015

3.4 Contributions to the development of course materials

3.4.1 Engineering degree

- **Digital signal processing, UY1**

  This course is intended for level 4 engineering students in the Electrical Engineering and Telecommunications Engineering programs. The objective is to make students familiar with the most important methods in DSP. The course makes extensive use of computer-based explanations (MATLAB), examples, and exercises. It introduces to:

  – discrete-time signals and systems in the time domain;

  – discrete-time signal and system representation in the frequency domain;

  – signal and system description in the complex frequency domain;

  – Fourier and fast Fourier transform and to their efficient implementation;

  - Design finite impulse response (FIR) and infinite impulse response (IIR) discrete-time filters.

- **Modern control, UY1**

  This course is intended for level 5 engineering students in the electrical engineering stream. The course's objective is to present control engineering methods using only mathematical tools to bring the student to understand and assimilate the basic concepts of modeling, analysis, and design of control systems. The course first introduces classical control, incredibly linear time-invariant systems, state-space analysis of linear systems, stability analysis, and classical design techniques. Content on modern control covers controller design techniques carried out in state-space (design problems of pole assignment, input-output decoupling, model matching, state observers, and optimal control); discrete-time system adaptive control (MIT rule and model reference), and introduction to fuzzy control.

3.4.2 Master recherche, ENSP

- **Automatic control, UY1**

  This course is intended for Master 2 students in Electrical Engineering. It aims to introduce students to advanced automatic control techniques such as state-space analysis of linear systems, stability analysis, classical design techniques, design techniques performed in state space, adaptive control (MIT rule, model reference, fuzzy control, back-stepping control, and sliding mode control). Students work with the Matlab software as a common platform for control engineers.

- **Artificielle Intelligence, UY1**

  This course is intended for students of Master 2 in Electrical Engineering. The objective is to introduce students to techniques of artificial intelligence with a focus on code production. After a general introduction of artificial intelligence, two artificial intelligence tools are discussed, namely neural networks and fuzzy logic. Regarding neural networks, the course introduces the artificial neuron learning algorithms (Gradient descent, Newton method, Quasi-Newton method, Levenberg-Marquardt algorithm) and deals in depth with two neural network topologies, namely the multilayer perceptron and radial-based networks. Fuzzy logic is limited to traditional fuzzy logic. The fuzzification, fuzzy rule inference, and defuzzification methods are mainly discussed. This theoretical part is followed by practical works on Matlab and implementation on real robots.

- **Robotics and vision, UY1**

  This course is intended for students of Master 2 Mechatronics. The objective is to introduce students to robotics and robotic vision. The course presents

different robotic platforms and then focuses on the two most common classes: wheeled mobile robots and industrial manipulators. Direct DH, modified DH, and inverse kinematics (Paul, Pieper, learning approaches), dynamics (Euler-Lagrange and Newton-Euler formulations), and some classical commands such as PID, PID with gravity compensation and computed torque control are discussed. The vision part introduces monocular and binocular vision. Camera calibration techniques are also covered in hands-on exercises using MATLAB. The course ends with practical lab work on the Scorbot ER robot manipulator (validation of forward and inverse kinematics, pick and place object programming, manual and automatic sorting of objects).

3.4.3 Master international MRT, University of Lille

Artificielle Intelligence

This course is intended for a Master in the international Master of Robotics and Transport (MRT). The goal is to introduce students to artificial intelligence techniques with an emphasis on code creation. After the general introduction to artificial intelligence, two machine learning methods are discussed: neural networks and fuzzy logic. Regarding neural networks, the course introduces artificial neuron learning algorithms (Gradient descent, Newton method, Conjugate gradient, Quasi-Newton method, and Levenberg-Marquardt algorithm) and deals in detail with two neural network topologies, namely multi-layer perceptron and radial-based networks. Fuzzy logic is restricted to conventional fuzzy logic. The fuzzification, fuzzy rule inference, and defuzzification methods are discussed. This theoretical aspect is accompanied by practical work on Matlab and robot implementation.

Chapter 4

Autonomous navigation of mobile robots

4.1 introduction

Mobile robots are becoming increasingly visible within our society today. From manufacturing industries to space exploration, through rescue operations, surveillance missions, education, military interventions, mass production in agriculture, entertainment, etc., mobile robot applications abound today. To successfully achieve their missions, mobile robots are sometimes equipped with high-performance autonomous navigation systems. Therefore, the autonomous navigation of mobile robots becomes a major research topic [Pandey *et al.* 2017, Faisal *et al.* 2013, Hagras 2004, Beom & Cho 1995].

Depending on the information (complete or incomplete) available from the navigation environment, navigation systems can be grouped into two main classes, namely global navigation and local navigation or reactive navigation. In global navigation, the mobile robot has an overview of the navigation environment, with a priori knowledge of the positions of the obstacles and the target to be reached, while in local navigation the mobile robot has only local knowledge of its environment. Yet, given the dynamic nature of mobile robot navigation environments, global navigation approaches can easily become time-consuming. As a result, much research work in mobile robot navigation has been focused on the local navigation method, where computation time is rational [Patle *et al.* 2019, Subbash & Chong 2019, Karray *et al.* 2016, Sanchez *et al.* 2015, Kim & Chwa 2014, Mbede *et al.* 2012].

Among the reactive approaches to navigation, the Fuzzy logic approach has emerged as an effective technique to tackle the uncertainties and imprecisions present in mobile robots' navigation environments. [Sanchez *et al.* 2015, Hagras 2004, Pandey *et al.* 2017, Melingui *et al.* 2014e]. Besides, given a large number of situations that mobile robots face during navigation, it will be difficult and time-consuming for human experts to examine all these situations to find the right decision-making control. Thus, soft computing algorithms such as fuzzy logic become suitable candidates in such situations. For example, the type-1 fuzzy logic method has been widely used for navigation and control of mobile robots [Beom & Cho 1995, Faisal *et al.* 2013, Surmann *et al.* 1995, Parhi 2005, Masmoudi *et al.* 2016, Bakdi *et al.* 2017]. Based on the experts' knowledge, type-1 fuzzy systems have been developed using fuzzy rules and crisp value membership functions. To improve the performance of these systems, learning techniques, such

as genetic algorithms, have also been implemented to tune the parameters of the membership functions [Pratihar *et al.* 1999, Hagras *et al.* 2004]. However, it has been demonstrated that fuzzy type-1 systems cannot fully manage or cope with the linguistic and numerical uncertainties associated with the changing and dynamic unstructured navigation environments of mobile robots because they use the crisp values of the membership functions. Because, once the membership functions of fuzzy type-1 systems are chosen or tuned, the performance of the overall system becomes fixed, and therefore the effects of uncertainties cannot be effectively reduced or eliminated. On the other hand, higher-order fuzzy logic systems (FLS), such as Interval type-2 FLS using Interval type-2 fuzzy sets, have been shown capable of modeling and handling such uncertainties and impressions [Mendel 2019, Sanchez *et al.* 2015, Hagras 2004]. Indeed, this capacity is justified by the fact that an interval type-2 membership function can be seen as possessing an uncountable number of type-1 sets. The third dimension of type-2 fuzzy logic sets and its footprint of uncertainty give them more degrees of freedom sufficient for better uncertainty modeling than type-1 fuzzy sets. In the literature, there are a large number of works that compare the performance of type-1 and type-2 fuzzy systems as in [Sanchez *et al.* 2015, Hagras 2004, Melin *et al.* 2013]. Besides, many optimization algorithms such as the chemical optimization paradigm [Astudillo *et al.* 2013], the particle swarm optimization [Maldonado *et al.* 2013], the hybrid learning algorithm [Méndez & De Los Angeles Hernández 2013], have also been implemented to find optimal parameters for fuzzy membership functions.

Concerning the navigation of mobile robots, there has been much research work using type-2 fuzzy systems. Junratanasiri et al. [Junratanasiri *et al.* 2011] used interval type-2 fuzzy logic for obstacle avoidance and position stabilization of mobile robots navigating in dynamic environments. Huang et al.[Huang *et al.* 2017] proposed an integrated fuzzy logic approach with interval type-2 fuzzy logic for modeling and controlling a two-wheeled mobile inverted pendulum. The objective of the control was to achieve the desired position and direction while maintaining the balance of the system. A reinforcement ant optimized interval type-2 fuzzy logic controller for wheeled mobile-robot wall-following control under reinforcement learning environments was also been proposed in [Juang & Hsu 2009]. Karray et al. [Karray *et al.* 2016] focused on solving the problem of tracking the trajectory of non-holonomic mobile manipulators in the presence of obstacles. An adaptive controller was designed to minimize the effect of model perturbations and uncertainties during trajectory tracking, and two fuzzy controllers were used to keep the robot away from obstacles and reach a virtual target point, respectively. Hagras [Hagras 2004] used interval type-2 FLC to implement the basic navigation behavior and the coordination between these behaviors to produce a type-2 HFLC for two-wheel mobile robots navigating in changing and dynamic unstructured indoor and outdoor environments. However, most of the aforementioned methods are based on a hierarchical architecture and the whole system can become complicated and take a long time to accomplish the different navigation behaviors. As a consequence, it becomes important to have single structures such as fuzzy neural networks which

can at the same time combine various navigation behaviors. In this way, Kim and Chwa [Kim & Chwa 2014] proposed the obstacle avoidance method in the position stabilization for wheeled mobile robots using an interval type-2 fuzzy neural network (IT2FNN). A navigation system using IT2FNN fitting Q-learning has been proposed in [Yi *et al.* 2019] for mobile robot navigation in complex environments. The approximation capabilities of IT2FNN have been specifically used to solve the mapping relationship between the state and the action spaces in Q-learning. In general, the normal type-2 reduction operation used in type-2 FNN systems requires an iterative procedure [Mendel 2017, Liu & Mendel 2011], such as the Karnik-Mendel procedure (K-M) to determine the outputs of the neuro-fuzzy system. However, when the number of fuzzy rules becomes large and the quantity of training data huge, training can take tens or hundreds of hours. In this regard, the normal type-reduction operation has been replaced with weighted bound-set boundaries in [Juang & Juang 2012] to reduce the training time of IT2FNN systems. In the same vein, a simplified interval type-2 fuzzy neural network where the time-consuming K-M iterative procedure is replaced by two design q-factors q_l and q_r has been proposed in [Lin *et al.* 2013]. This is of paramount importance in the autonomous navigation context, where the response time of the navigation systems is critical. Besides, the response time can be significantly reduced if strategies aimed at reducing the size of the fuzzy rule base are considered [Jin 2000, Yam *et al.* 1999, Melingui *et al.* 2014e].

Some autonomous navigation methods, when implemented alone, can lead to poor performance; whereas their combinations, when well-thought-out, can yield exceptional performances. Previously, in the framework of mobile robot navigation, it was proposed an IT2FL system with the attractive and repulsive forces of the artificial potential field (APF) as additional linguistic variables [Melingui *et al.* 2014e, Melingui *et al.* 2013a]. This controller was able to perform basic navigation behaviours with a small fuzzy rule base. The navigation environment was divided into navigation areas, including areas with local minima and no local minima. The APF controller operated virtually in areas with no local minima while the IT2FL controller operated in areas with local minima. Indeed, while navigation in areas with no local minima, the outputs of the APF controller were directly transferred to the output of the IT2FL controller using a "none" membership function (MF). In fact, for a given fuzzy rule, the "none" MF makes it possible to no consider some linguistic variables in the output computing. In this paper, the performance of the former IT2FL controller is improved by replacing the K-M type-reduction operation by two design q-factors q_l and q_r which are learned via back-propagation algorithm.

4.2 Interval type-2 fuzzy logic controller for autonomous navigation of mobile robots

This section focuses on the description of the IT2FL controller developed in [Melingui *et al.* 2014e, Melingui *et al.* 2013a]. We find this description necessary

as the proposed IT2FNN controller is an extension of the latter.

The IT2FL controller developed in [Melingui *et al.* 2014e, Melingui *et al.* 2013a] proposed a hybridization of the fuzzy logic concepts and artificial potential field concepts in a common framework. APF method for navigation of mobile robots is computationally efficient and considers both the problems of obstacle avoidance and trajectory planning, but faces the local minima problem. On the other hand, navigation through the fuzzy logic method can be achieved without issues of local minima but is computationally expensive as the fuzzy rule database increases. Hence, the authors found necessary to combine both methods to exploit the best of them. This hybridization had the objective to exploit the computationally efficient of the APF method and the capability of fuzzy logic methods to achieve navigation behaviours without local minima issues. After noticed that most of local minima situations occur when the potential attractive force and repulsive attractive force are collinear, the navigation environment has been divided into three areas as shown in Fig.4.1, two areas where it is difficult to encounter local minima situations and another one where the robot can encounter local minima situations. The challenge was to associate these methods without using a switch system which generally involves some oscillations at the time of transition from one controller to another. To achieve this, an APF controller has first been designed using the information of different sensors, and its outputs have been used as additional inputs to the IT2FL controller. Thus, during navigation in no local minima areas, the outputs of the APF controller were directly transferred to the output of the IT2FL controller by using the "none" membership function. On the other hand, IT2FL was only used in the local minima areas. The association allows the authors to achieve autonomous navigation of a mobile robot named Robotino with only 15 fuzzy rules as shown in Table 4.1. This number of fuzzy rules is significantly small compared to 40 rules and 128 rules developed for the same platform in [Melingui *et al.* 2013a] and [Oltean *et al.* 2010], respectively.

The following robot' state variables (d_l, d_f, d_r, ΔX, α, F_x, F_y) were used as IT2FL controller inputs and the longitudinal velocity V_x and the lateral velocity V_y as outputs. d_l, d_f, and d_r represent the distance between the robot and the nearest obstacle in the left, front, and right side of the robot, respectively. ΔX the distance between the current position of the robot and the target position and α the orientation of the target relative of the current position of the robot. F_x and F_y represent the longitudinal and lateral artificial forces generated by the APF controller. The linguistic variables d_l, d_f, and d_r were modelled by one rectangular interval type-2 membership function (RIT-2-MF) and two interval type-2 Gaussian membership functions (IT-2-GMF) represented by the labels "*None*" ($\widetilde{None}$), "*Near*" ($\widetilde{N}$), and "*Far*" ($\widetilde{F}$), respectively. The distance between the robot and the goal position was modelled by one RIT-2-MF and two IT-2-GMFs represented by the labels "*None*" ($\widetilde{None}$), "*Near*" ($\widetilde{N}$), and "*Far*" ($\widetilde{F}$), respectively. The orientation of the target relative to the current pose of the robot α, the longitudinal artificial force (F_x) and the lateral artificial force (F_y) were modelled by one RIT-2-MF and three IT-2-GMFs represented by the labels "*None*" ($\widetilde{No}$), "*Negative*" ($\widetilde{NE}$), "*Zero*" ($\widetilde{Z}$) "*Positive*"

($\widetilde{PO}$), respectively. Finally, the outputs, that is, the longitudinal velocity V_x and the lateral velocity V_y were modelled by five IT-2-GMFs represented by the labels "*Fast Backward*" ($\widetilde{FBW}$), "*Backward*" ($\widetilde{BW}$), "*Stop*" ($\widetilde{S}$), "*Forward*" ($\widetilde{FW}$), and "*Fast forward*" ($\widetilde{FFW}$), respectively. The lower and upper membership grades of the RIT-2-MF labeled as "*None*" are equal to one. The use of the longitudinal artificial force (F_x), the lateral artificial force (F_y) as additional input variables, combined with "*None*" MF allow a signification reduction of the number of the fuzzy rules. We refer the reader to [Melingui *et al.* 2014e, Melingui *et al.* 2013a] for more details on the IT2FL controller.

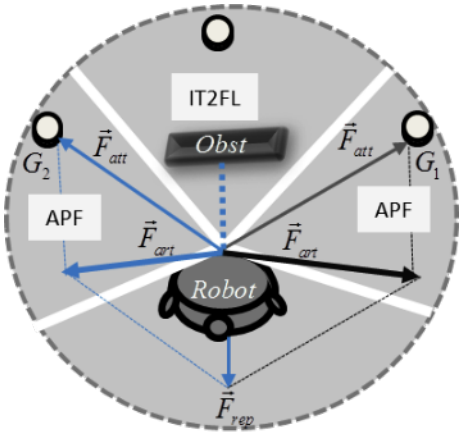

Figure 4.1: Artificial potential force direction when the robot faces with a frontal obstacle in case of a negative ($\widetilde{Ne}$) or positive ($\widetilde{Po}$) orientation of the target point relative to the robot position.

4.3 Artificial Potential Field Neurofuzzy Controller for Autonomous Navigation of Mobile Robots

This work aims to make the IT2FL controller proposed in [Melingui *et al.* 2014e] computationally efficient by replacing the type-reduction operation by two design q-factors q_l and q_r. The idea is to emulate this controller while making it computationally efficient. This section focuses on the development of the proposed IT2FNN controller. The section first presents the IT2FNN architecture and ends with the enhancing of the IT2FNN performance through tuning IT2FNN's parameters using a back-propagation algorithm.

4.3.1 IT2FNN architecture

The topology of the IT2FNN controller consists of four layers as shown in Fig.4.2. The layers I and II form the antecedent part of the IT2FNN while layers III and IV form the consequent part. The function of each layer is detailed hereafter.

Table 4.1: Fuzzy rule base for T2PFL[Melingui *et al.* 2014e]

N^0	d_l	d_f	d_r	Δ_X	α	F_x	F_y	V_x	V_y
1	$None$	$None$	$None$	$None$	$\widetilde{NE}$	$\widetilde{PO}$	$\widetilde{NE}$	$\widetilde{FW}$	$\widetilde{BW}$
2	$None$	$None$	$None$	$None$	$\widetilde{NE}$	$\widetilde{PO}$	$\tilde{Z}$	$\widetilde{FW}$	$\tilde{S}$
3	$None$	$None$	$None$	$None$	$\widetilde{NE}$	$\widetilde{PO}$	$\widetilde{PO}$	$\widetilde{FW}$	$\widetilde{FW}$
4	$None$	$None$	$None$	$None$	$\widetilde{PO}$	$\widetilde{PO}$	$\widetilde{NE}$	$\widetilde{FW}$	$\widetilde{BW}$
5	$None$	$None$	$None$	$None$	$\widetilde{PO}$	$\widetilde{PO}$	$\tilde{Z}$	$\widetilde{FW}$	$\tilde{S}$
6	$None$	$None$	$None$	$None$	$\widetilde{PO}$	$\widetilde{PO}$	$\widetilde{PO}$	$\widetilde{FW}$	$\widetilde{FW}$
7	$\tilde{F}$	$\tilde{F}$	$\tilde{F}$	$None$	$None$	$\tilde{Z}$	$\widetilde{NE}$	$\tilde{S}$	$\widetilde{FBW}$
8	$\tilde{F}$	$\tilde{F}$	$\tilde{F}$	$None$	$None$	$\tilde{Z}$	$\tilde{Z}$	$\tilde{S}$	$\tilde{S}$
9	$\tilde{F}$	$\tilde{F}$	$\tilde{F}$	$None$	$None$	$\tilde{Z}$	$\widetilde{PO}$	$\tilde{S}$	$\widetilde{FFW}$
10	$\tilde{F}$	$\tilde{F}$	$\tilde{N}$	$None$	$\tilde{Z}$	$None$	$None$	$\widetilde{FW}$	$\tilde{S}$
11	$\tilde{N}$	$\tilde{F}$	$\tilde{F}$	$None$	$\tilde{Z}$	$None$	$None$	$\widetilde{FW}$	$\tilde{S}$
12	$\tilde{F}$	$\tilde{N}$	$\tilde{F}$	$\tilde{N}$	$\tilde{Z}$	$None$	$None$	$\tilde{S}$	$\tilde{S}$
13	$\tilde{F}$	$\tilde{N}$	$\tilde{N}$	$\tilde{N}$	$\tilde{Z}$	$None$	$None$	$\tilde{S}$	$\tilde{S}$
14	$\tilde{N}$	$\tilde{N}$	$\tilde{F}$	$\tilde{N}$	$\tilde{Z}$	$None$	$None$	$\tilde{S}$	$\tilde{S}$
15	$\tilde{N}$	$\tilde{N}$	$\tilde{N}$	$\tilde{N}$	$\tilde{Z}$	$None$	$None$	$\tilde{S}$	$\tilde{S}$

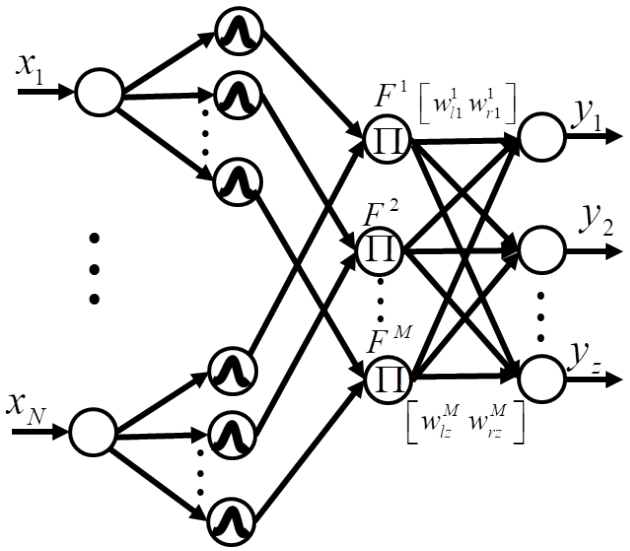

Figure 4.2: Interval type-2 fuzzy controller structure

Layer I (Input layer)

Each node in this layer represents a crisp input variable. This layer receives the current state (x_1, x_2,..., x_n) variable of the robot as input.

Layer II (Fuzzification layer)

This layer is in charge of the fuzzification operation. Each node of this layer is consisted of a Gaussian primary membership function (MF) having a fixed standard deviation σ and an uncertain mean that takes values in $\left[m_{k1}^i, m_{k2}^i\right]$. The upper and lower MFs for an input x_k and a rule i can be written as follows:

$$\overline{u}_{\widetilde{F}_j^i}(x_k) = \begin{cases} N\left(m_{k_1}^i, \sigma_k^i, x_k\right), & x_k < m_{k_1}^i \\ 1 \quad , & m_{k_1}^i \leq x_k \leq m_{k_2}^i \\ N\left(m_{k_2}^i, \sigma_k^i, x_k\right), & x_k > m_{k_2}^i \end{cases} \tag{4.1}$$

$$\underline{u}_{\widetilde{F}_j^i}(x_k) = \begin{cases} N\left(m_{k_2}^i, \sigma_k^i, x_k\right), & x_k \leq \frac{m_{k_1}^i + m_{k_2}^i}{2} \\ N\left(m_{k_1}^i, \sigma_k^i, x_k\right), x_k > \frac{m_{k_1}^i + m_{k_2}^i}{2} \end{cases} \tag{4.2}$$

where $N\left(m_k^i, \sigma_k^i, x_k\right) = \exp\left[-\frac{1}{2}\left(\frac{x_k - m_k^i}{\sigma_k^i}\right)^2\right]$.

Thus, the output of this layer can be expressed as an interval type-1 set $[\overline{u}_{\widetilde{F}_j^i}(x_k), \underline{u}_{\widetilde{F}_j^i}(x_k)]$.

Layer III (Firing layer)

Each node in this layer computes the firing strength. The IT2FNN's F^i firing strength for *ith* rule is a type-1 set interval, expressed as follows:

$$F^i = \left[\underline{f}^i, \overline{f}^i\right] \tag{4.3}$$

where

$$\underline{f}^i = \underline{u}_{\widetilde{F}_1^i}(x_1) * \ldots * \underline{u}_{\widetilde{F}_n^i}(x_n) = \prod_{j=1}^{n} \underline{u}_{\widetilde{F}_1^i}(x_j) \tag{4.4}$$

$$\overline{f}^i = \overline{u}_{\widetilde{F}_1^i}(x_1) * \ldots * \overline{u}_{\widetilde{F}_n^i}(x_n) = \prod_{j=1}^{n} \overline{u}_{\widetilde{F}_1^i}(x_j) \tag{4.5}$$

where $\underline{u}_{\widetilde{F}_1^i}\left(x_j^p\right)$ and $\overline{u}_{\widetilde{F}_1^i}\left(x_j^p\right)$ represent the values of the lower and upper MFs, respectively. $*$ and n refer to the minimum or the product t-norm and the number of inputs to the *ith* rule of the IT2FNN. The IF-THEN rule for IT2FNN is expressed as follows:

$$\begin{aligned} \text{R}^{\text{i}}\text{: IF } x_1 \text{ is } \widetilde{F}_1^i, \text{ and } x_2 \text{ is } \widetilde{F}_2^i, \text{ and } \ldots, \text{and } x_n \text{ is } \widetilde{F}_n^i \\ \text{THEN } y_1 \text{ is } \left[w_{l1}^i\ w_{r1}^i\right], \text{ and}, \ldots, \text{and } y_Z \text{ is } \left[w_{lZ}^i\ w_{rZ}^i\right] \end{aligned} \tag{4.6}$$

Layer IV (Output processing layer)

This layer combines the outputs of the layer II ($\underline{f}^i, \overline{f}^i$) and layer III ($y_i$), the centroid interval set $[w_l^i\, w_r^i]$ of the consequent part [Mendel 2017], and the design q-factors $[q_l, q_r]$ allowing adaptive adjustment of the upper and the lower values without using the K-M iterative procedure to find L and R endpoints [Lin *et al.* 2013]. The use of q-factors significantly reduces the complexity of type-2 FLS and makes the latter suitable for real-time applications such as autonomous navigation of mobile robots given the large size of their rule base. By using q-factors, the left and right endpoints y_l and y_r can be computed as follows:

$$y_l = \frac{(1-q_l)\sum_{i=1}^{M} \underline{f}^i w_l^i + q_l \sum_{i=1}^{M} \overline{f}^i w_l^i}{\sum_{i=1}^{M} \underline{f}^i + \overline{f}^i} \tag{4.7}$$

$$y_r = \frac{(1-q_r)\sum_{i=1}^{M} \overline{f}^i w_r^i + q_r \sum_{i=1}^{M} \underline{f}^i w_r^i}{\sum_{i=1}^{M} \underline{f}^i + \overline{f}^i}. \tag{4.8}$$

Finally, each defuzzified crisp output $y(\vec{x})$ of the IT2FNN is the average of y_l and y_r as follows:

$$y(\vec{x}) = \frac{y_l + y_r}{2}. \tag{4.9}$$

4.3.2 Back-propagation algorithm

This section focuses on training of the IT2FNN through the back-propagation algorithm. The learning process minimizes the following error function:

$$e^p = \frac{1}{2}\left[y(\vec{x}^p) - d^p\right]^2,\ p = 1, ..., P, \tag{4.10}$$

where P is the number of training data $(\vec{x}^p, d^p)$.

Let consider η as the learning rate, the means $m_{k_1}^i$ and m_{k2}^i , the standard deviation σ, the consequent weighting factors w_l^i and w_r^i, and the q-factors q_l and q_r of the ith rule are updated as follows:

$$m_{k1}^i(p+1) = m_{k1}^i(p) - \eta \left.\frac{\partial e^p}{\partial m_{k1}^i}\right|_p, \tag{4.11}$$

$$m_{k2}^i(p+1) = m_{k2}^i(p) - \eta \left.\frac{\partial e^p}{\partial m_{k2}^i}\right|_p, \tag{4.12}$$

$$\sigma_k^i(p+1) = \sigma_k^i(p) - \eta \left.\frac{\partial e^p}{\partial \sigma_k^i}\right|_p, \tag{4.13}$$

$$w_l^i(p+1) = w_l^i(p) - \eta \frac{\partial e^p}{\partial w_l^i}\bigg|_p, \tag{4.14}$$

$$w_r^i(p+1) = w_r^i(p) - \eta \frac{\partial e^p}{\partial w_r^i}\bigg|_p, \tag{4.15}$$

$$q_l^i(p+1) = q_l^i(p) - \eta \frac{\partial e^p}{\partial q_l^i}\bigg|_p, \tag{4.16}$$

$$q_r^i(p+1) = q_r^i(p) - \eta \frac{\partial e^p}{\partial q_r^i}\bigg|_p. \tag{4.17}$$

Equation (4.11) can be rewriting as follows:

$$m_{k1}^i(p+1) = m_{k1}^i(p) - \eta \left[\frac{\partial e^p}{\partial y(\vec{x})} \frac{\partial y(\vec{x})}{\partial y_l} \frac{\partial y_l}{\partial m_{k1}^i} + \frac{\partial e^p}{\partial y(\vec{x})} \frac{\partial y(\vec{x})}{\partial y_r} \frac{\partial y_r}{\partial m_{k1}^i} \right]\bigg|_p. \tag{4.18}$$

From (4.18), (4.9), and (4.10), it follows that

$$\frac{\partial e^p}{\partial y(\vec{x})}\bigg|_p = y(\vec{x}) - d^p, \tag{4.19}$$

$$\frac{\partial y(\vec{x})}{\partial y_l}\bigg|_p = 1/2, \tag{4.20}$$

$$\frac{\partial y(\vec{x})}{\partial y_r}\bigg|_p = 1/2, \tag{4.21}$$

$$\frac{\partial y_l}{\partial m_{k1}^i}\bigg|_p = \left[\frac{\partial y_l}{\partial \overline{u}_{\widetilde{F}_k^i}(x_k)} \frac{\partial \overline{u}_{\widetilde{F}_k^i}(x_k)}{\partial m_{k1}^i} + \frac{\partial y_l}{\partial \underline{u}_{\widetilde{F}_k^i}(x_k)} \frac{\partial \underline{u}_{\widetilde{F}_k^i}(x_k)}{\partial m_{k1}^i} \right], \tag{4.22}$$

$$\frac{\partial y_r}{\partial m_{k1}^i}\bigg|_p = \left[\frac{\partial y_r}{\partial \overline{u}_{\widetilde{F}_k^i}(x_k)} \frac{\partial \overline{u}_{\widetilde{F}_k^i}(x_k)}{\partial m_{k1}^i} + \frac{\partial y_r}{\partial \underline{u}_{\widetilde{F}_k^i}(x_k)} \frac{\partial \underline{u}_{\widetilde{F}_k^i}(x_k)}{\partial m_{k1}^i} \right]. \tag{4.23}$$

From (4.22),(4.23), (4.1), and (4.2), the following relations are derived:

$$\frac{\partial \overline{u}_{\widetilde{F}_k^i}(x_k)}{\partial m_{k1}^i} = \begin{cases} \frac{(x_k - m_{k1}^i) N(m_{k1}^i, \sigma_k^i, x_k)}{{\sigma_k^i}^2}, & x_k < m_{k1}^i \\ 0, & m_{k1}^i \le x_k \le m_{k2}^i \\ 0, & x_k > m_{k2}^i \end{cases} \tag{4.24}$$

$$\frac{\partial \underline{u}_{\widetilde{F}_k^i}(x_k)}{\partial m_{k1}^i} = \begin{cases} 0, & x_k \le \frac{m_{k1}^i + m_{k2}^i}{2} \\ \frac{(x_k - m_{k1}^i) N(m_{k1}^i, \sigma_k^i, x_k)}{{\sigma_k^i}^2}, & x_k > \frac{m_{k1}^i + m_{k2}^i}{2} \end{cases} \tag{4.25}$$

$$\frac{\partial \overline{u}_{\widetilde{F}_k^i}(x_k)}{\partial \sigma_k^i} = \begin{cases} \frac{(x_k - m_{k1}^i)^2 N(m_{k1}^i, \sigma_k^i, x_k)}{{\sigma_k^i}^3}, & x_k < m_{k1}^i \\ 0, & m_{k1}^i \le x_k \le m_{k2}^i \\ \frac{(x_k - m_{k2}^i)^2 N(m_{k2}^i, \sigma_k^i, x_k)}{{\sigma_k^i}^3}, & x_k > m_{k2}^i \end{cases} \tag{4.26}$$

$$\frac{\partial \underline{u}_{\widetilde{F}_k^i}(x_k)}{\partial \sigma_k^i} = \begin{cases} \frac{\left(x_k - m_{k2}^i\right)^2 N\left(m_{k2}^i, \sigma_k^i, x_k\right)}{{\sigma_k^i}^3}, & x_k \leq \frac{m_{k1}^i + m_{k2}^i}{2} \\ \frac{\left(x_k - m_{k1}^i\right)^2 N\left(m_{k1}^i, \sigma_k^i, x_k\right)}{{\sigma_k^i}^3}, & x_k > \frac{m_{k1}^i + m_{k2}^i}{2} \end{cases} \tag{4.27}$$

$$\frac{\partial y_l}{\partial \overline{u}_{\widetilde{F}_k^i}(x_k)} = \frac{\left(\prod_{\substack{j=1 \\ j \neq k}}^{n} \overline{u}_{\widetilde{F}_j^i}\right)\left(q_l w_l^i - y_l\right)}{\sum_{i=1}^{M} \underline{f}^i + \overline{f}^i}. \tag{4.28}$$

Similarly,

$$\frac{\partial y_l}{\partial \underline{u}_{\widetilde{F}_k^i}(x_k)} = \frac{\left(\prod_{\substack{j=1 \\ j \neq k}}^{n} \underline{u}_{\widetilde{F}_j^i}\right)\left(\left(1 - q_l\right) w_l^i - y_l\right)}{\sum_{i=1}^{M} \underline{f}^i + \overline{f}^i}, \tag{4.29}$$

$$\frac{\partial y_r}{\partial \overline{u}_{\widetilde{F}_k^i}(x_k)} = \frac{\left(\prod_{\substack{j=1 \\ j \neq k}}^{n} \overline{u}_{\widetilde{F}_j^i}\right)\left(q_r w_r^i - y_r\right)}{\sum_{i=1}^{M} \underline{f}^i + \overline{f}^i}, \tag{4.30}$$

$$\frac{\partial y_r}{\partial \underline{u}_{\widetilde{F}_k^i}(x_k)} = \frac{\left(\prod_{\substack{j=1 \\ j \neq k}}^{n} \underline{u}_{\widetilde{F}_j^i}\right)\left(\left(1 - q_r\right) w_r^i - y_r\right)}{\sum_{i=1}^{M} \underline{f}^i + \overline{f}^i}. \tag{4.31}$$

If $x_k^p < m_{k_1}^i$, hence substituting (4.28), (4.24), (4.29), (4.25) into (4.22) [let's write this (4.22')] and by substituting (4.30), (4.24), (4.31), (4.25) into (4.23) [let's write this (4.23')], and then substituting (4.19), (4.20), (4.21), (4.22'), and (4.23') into (4.18), we will have:

$$\begin{aligned} & m_{k_1}^i(p+1) = m_{k_1}^i(p) \\ & - \left(\frac{1}{2}\eta \frac{\left(y(\vec{x}^p) - d^p\right)\left(x_k - m_{k_1}^i\right) N\left(m_{k_1}^i, \sigma_k^i, x_k\right)}{{\sigma_k^i}^2} \right. \\ & \left. \times \frac{\left(\prod_{\substack{j=1 \\ j \neq k}}^{n} \overline{u}_{\widetilde{F}_j^i}(x_k)\right)\left[\left(q_l w_l^i + q_r w_r^i\right) - \left(y_l + y_r\right)\right]}{\sum_{i=1}^{M} \underline{f}^i + \overline{f}^i} \right). \end{aligned} \tag{4.32}$$

If $x_k^p \geq m_{k1}^i$ and $x_k^p > \frac{1}{2}\left(m_{k_1}^i + m_{k_2}^i\right)$, by following the same procedure, we get the following:

$$
\begin{aligned}
& m_{k_1}^i(p+1) = m_{k_1}^i(p) \\
& - \left(\frac{1}{2}\eta \frac{(y(\vec{x}^p)-d^p)\left(x_k - m_{k_1}^i\right) N\left(m_{k_1}^i, \sigma_k^i, x_k\right)}{{\sigma_k^i}^2} \right. \\
& \left. \times \frac{\left(\prod_{\substack{j=1 \\ j\neq k}}^{n} \overline{u}_{\widetilde{F}_j^i}(x_k)\right)\left[\left((1-q_l)w_l^i + (1-q_r)w_r^i\right) - (y_l + y_r)\right]}{\sum_{i=1}^{M} \underline{f}^i + \overline{f}^i} \right).
\end{aligned}
\tag{4.33}
$$

The BP equation for $mk2^i$ can be derived by following a procedure similar to the one described above.

If $x_k^p < m_{k_1}^i$ (hence also $x_k^p \leq (m_{k_1}^i + m_{k_2}^i)/2$)

$$
\begin{aligned}
& \sigma_k^i(p+1) = \sigma_k^i(p) - \tfrac{1}{2}\eta \frac{(y(\vec{x}^p)-d^p)}{\left(\sigma_k^i\right)^3} \\
& \times \left[\left(x_k^p - m_{k_1}^i\right)^2 N\left(m_{k_1}^i, \sigma_k^i, x_k^p\right) \right. \\
& \frac{\left(\prod_{\substack{j=1 \\ j\neq k}}^{n} \overline{u}_{\widetilde{F}_j^i}\right)\left(q_l w_l^i + q_r w_r^i - y_l - y_r\right)}{\sum_{i=1}^{M} \bar{f}^i + \underline{f}^i} \\
& + \left(x_k^p - m_{k_2}^i\right)^2 N\left(m_{k_2}^i, \sigma_k^i, x_k^p\right) \\
& \left. \frac{\left(\prod_{\substack{j=1 \\ j\neq k}}^{n} \underline{u}_{\widetilde{F}_j^i}\right)\left((1-q_l)w_l^i + (1-q_r)w_r^i - y_l - y_r\right)}{\sum_{i=1}^{M} \overline{f}^i + \underline{f}^i} \right].
\end{aligned}
\tag{4.34}
$$

If $x_k^p > m_{k_2}^i$

$$
\begin{aligned}
& \sigma_k^i(p+1) = \sigma_k^i(p) - \tfrac{1}{2}\eta \frac{(y(\vec{x}^p)-d^p)}{\left(\sigma_k^i\right)^3} \\
& \times \left[\left(x_k^p - m_{k_2}^i\right)^2 N\left(m_{k_2}^i, \sigma_k^i, x_k^p\right) \right. \\
& \frac{\left(\prod_{\substack{j=1 \\ j\neq k}}^{n} \overline{u}_{\widetilde{F}_j^i}\right)\left(q_l w_l^i + q_r w_r^i - y_l - y_r\right)}{\sum_{i=1}^{M} \bar{f}^i + \underline{f}^i} \\
& + \left(x_k^p - m_{k_1}^i\right)^2 N\left(m_{k_1}^i, \sigma_k^i, x_k^p\right) \\
& \left. \frac{\left(\prod_{\substack{j=1 \\ j\neq k}}^{n} \underline{u}_{\widetilde{F}_j^i}\right)\left((1-q_l)w_l^i + (1-q_r)w_r^i - y_l - y_r\right)}{\sum_{i=1}^{M} \overline{f}^i + \underline{f}^i} \right].
\end{aligned}
\tag{4.35}
$$

The consequent weighting factors can be obtaining similarly. Let's consider w_l^i, the BP equation is expressed as:

$$
\begin{aligned}
& w_l^i(p+1) = w_l^i(p) \\
& \quad -\eta \left[\frac{\partial e^p}{\partial y(\vec{x})} \frac{\partial y(\vec{x})}{\partial y_l} \frac{\partial y_l}{\partial w_l^i} + \frac{\partial e^p}{\partial y(\vec{x})} \frac{\partial y(\vec{x})}{\partial y_r} \frac{\partial y_r}{\partial w_l^i} \right]\Big|_p,
\end{aligned}
\tag{4.36}
$$

where

$$\frac{\partial y_l}{\partial w_l^i\left(x_k^p\right)}=\frac{\left(1-q_l\right)\prod_{j=1}^{n}\underline{u}_{\widetilde{F}_j^i}+q_l\prod_{j=1}^{n}\overline{u}_{\widetilde{F}_j^i}}{\sum_{i=1}^{M}\underline{f}^i+\overline{f}^i}, \tag{4.37}$$

and

$$\frac{\partial y_r}{\partial w_l^i\left(x_k^p\right)}=0. \tag{4.38}$$

By substitution (4.19), (4.20), (4.21) (4.37), and (4.38) into (4.36)

$$\begin{aligned} w_l^i\left(p+1\right)=&w_l^i\left(p\right)-\tfrac{1}{2}\eta\left(y\left(\vec{x}^p\right)-d^p\right)\\ &\times\frac{\left(1-q_l\right)\prod_{j=1}^{n}\underline{u}_{\widetilde{F}_j^i}+q_l\prod_{j=1}^{n}\overline{u}_{\widetilde{F}_j^i}}{\sum_{i=1}^{M}\underline{f}^i+\bar{f}^i}. \end{aligned} \tag{4.39}$$

By following the same procedure, w_r^i is obtained as follows:

$$\begin{aligned} w_r^i\left(p+1\right)=&w_r^i\left(p\right)-\tfrac{1}{2}\eta\left(y\left(\vec{x}^p\right)-d^p\right)\\ &\times\frac{\left(1-q_r\right)\prod_{j=1}^{n}\underline{u}_{\widetilde{F}_j^i}+q_r\prod_{j=1}^{n}\overline{u}_{\widetilde{F}_j^i}}{\sum_{i=1}^{M}\underline{f}^i+\overline{f}^i}. \end{aligned} \tag{4.40}$$

As above, the udpdate of q-factors q_l and q_r can be derived. Let's consider q_l, the BP equation is given as follows:

$$\begin{aligned} q_l\left(p+1\right)=&q_l\left(p\right)\\ &-\eta\left[\frac{\partial e^p}{\partial y(\vec{x})}\frac{\partial y(\vec{x})}{\partial y_l}\frac{\partial y_l}{\partial q_l}+\frac{\partial e^p}{\partial y(\vec{x})}\frac{\partial y(\vec{x})}{\partial y_r}\frac{\partial y_r}{\partial q_l}\right]\Big|_p, \end{aligned} \tag{4.41}$$

where

$$\frac{\partial y_l}{\partial q_l\left(x_k^p\right)}=\frac{-\sum_{i=1}^{M}\underline{f}^i w_l^i+\sum_{i=1}^{M}\overline{f}^i w_l^i}{\sum_{i=1}^{M}\underline{f}^i+\overline{f}^i}, \tag{4.42}$$

and

$$\frac{\partial y_r}{\partial q_l\left(x_k^p\right)}=0. \tag{4.43}$$

By substitution (4.19), (4.20), (4.21) (4.42), and (4.43) into (4.41), we have:

$$\begin{aligned} q_l\left(p+1\right)=&q_l\left(p\right)-\tfrac{1}{2}\eta\left(y\left(\vec{x}^p\right)-d^p\right)\\ &\times\frac{-\sum_{i=1}^{M}\underline{f}^i w_l^i+\sum_{i=1}^{M}\overline{f}^i w_l^i}{\sum_{i=1}^{M}\underline{f}^i+\overline{f}^i}. \end{aligned} \tag{4.44}$$

By following the same procedure, q_r is obtained as follows:

$$\begin{aligned} q_r\left(p+1\right)=&q_r\left(p\right)-\tfrac{1}{2}\eta\left(y\left(\vec{x}^p\right)-d^p\right)\\ &\times\frac{-\sum_{i=1}^{M}\underline{f}^i w_r^i+\sum_{i=1}^{M}\overline{f}^i w_r^i}{\sum_{i=1}^{M}\underline{f}^i+\overline{f}^i}. \end{aligned} \tag{4.45}$$

4.3.3 Application to autonomous navigation of mobile robots

In this section, the IT2FNN controller proposed in the previous section is implemented in the case of the autonomous navigation of mobile robots. The idea being to emulate the controller [Melingui *et al.* 2014e] while making it computationally efficient, the same input variables (d_l, d_f, d_r, ΔX, α, F_x, F_y) and output variables (V_x, V_y) as described in section II are considered.

The control architecture is given in Fig.4.3. The MFs of the linguistic variables relative to distance (d_l, d_f, d_r, ΔX) with the initial setting parameters are depicted in Fig. 4.4. The MFs of the orientation of the target relative of the current position of the robot α, and the longitudinal F_x and lateral F_y artificial forces are depicted in Fig. 4.5 and Fig. 4.6, respectively. Given the number of input variables and the associated MFs, the number of fuzzy rules should be approximately 432 fuzzy rules which is important. However, by taking into account the advantages offered by APF and fuzzy logic approaches, this number can be greatly reduced (15 fuzzy rules, [Melingui *et al.* 2014e]). The fuzzy rules developed in the case of the IT2FL controller in [Melingui *et al.* 2014e] are tuned in this paper by using a back-propagation process via neuro-fuzzy architecture. The standard deviation σ, the uncertain mean that takes values in $\left[m^i_{k1}, m^i_{k2}\right]$, the centroid interval set $[w^i_l\, w^i_r]$ of the consequent part, and the design q-factors $[q_l, q_r]$ are tuned by using the back-propagation equations developed in subsection III-B.

Simple strategies in mobile robot navigation such as goal-reaching, wall-following, corridor-following, U-shaped obstacle avoidance can involve thousands of training data. However, it is difficult for designers to have specific a priori knowledge of all all training samples.Therefore, the proposed IT2FNN is trained directly with fuzzy rules and with some numerical sample data obtained by implemented each simple navigation strategy separately (Target reaching, wall following, corridor following, etc.) using the IT2FL controller developed in [Melingui *et al.* 2014e]. The input variables (d_l, d_f, d_r, ΔX, α, F_x, F_y) and output variables (V_x, V_y) are recorded in each step. This data is then processed to eliminate noise and avoid abrupt changes during the movement of the robot. The goal of the learning process is to first behave like the IT2FL controller regarding obstacle avoidance and target reaching and improve the generalization performance for autonomous navigation of mobile robots. The root-mean-square error between the IT2FNN outputs and the desired outputs obtained at the end of training is small (less than 10^{-2}).

4.3.4 Experiments and results

Two scenarios are considered to validate the proposed IT2FNN controller, three in static environments, and one in a dynamic environment. There are four or five obstacles in each scenario, and the robot should avoid them while moving to the target position. The obstacles are placed so that the path of the mobile robot passes through them and then encounters other obstacles. The scenarios are selected to evaluate whether the proposed IT2FNN can emulate the IT2FL controller proposed

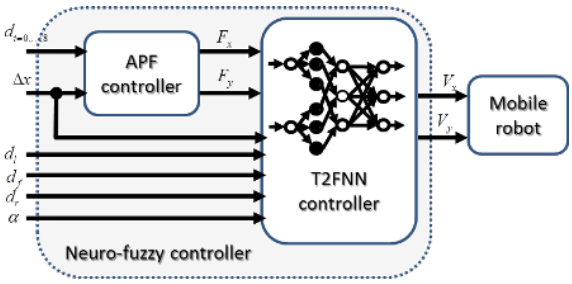

Figure 4.3: IT2FNN control structure

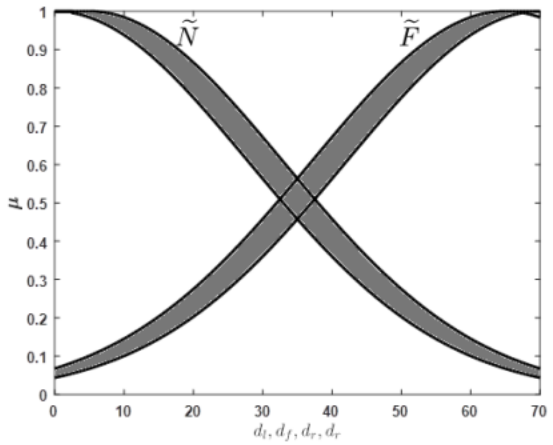

Figure 4.4: The distances between the robot and the nearest obstacles in its left, front, and right sides, and labeled as "*Near*" ($\widetilde{N}$) and "*Far*" ($\widetilde{F}$).

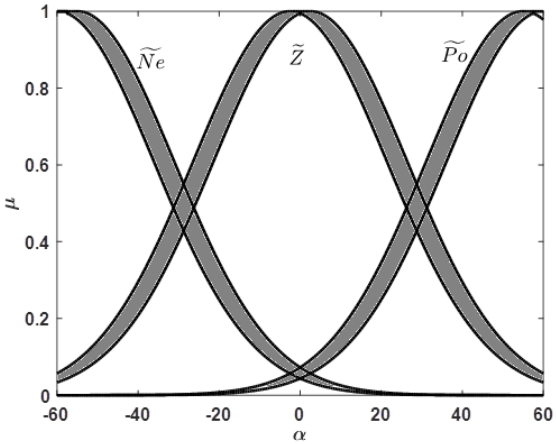

Figure 4.5: The orientation of the target position relative to the current robot position modelled by three interval type-2 Gaussian MFs and labelled as "*Negative*" ($\widetilde{NE}$), "*Zero*" ($\widetilde{Z}$) "*Positive*" ($\widetilde{PO}$).

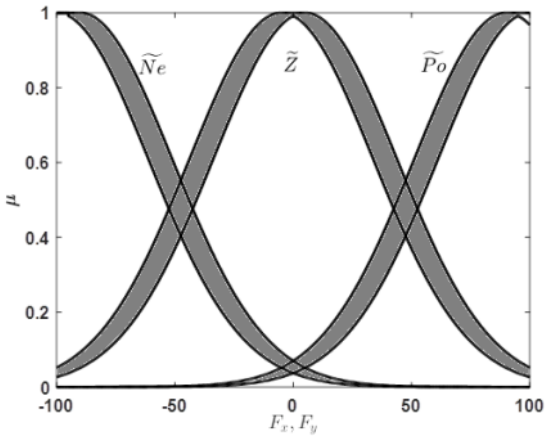

Figure 4.6: The longitudinal artificial force (F_x) and the lateral artificial force (F_y) modeled by three interval type-2 Gaussian MFs and labelled as "*Negative*" ($\widetilde{NE}$), "*Zero*" ($\widetilde{Z}$) "*Positive*" ($\widetilde{PO}$).

in previous work [Melingui *et al.* 2014e] while being computationally efficient. The first scenario attempts to assess the corridor following, obstacle avoidance, and target reaching strategies in the presence of obstacles with regular geometry. The second scenario allows assessing if the issues of local minima are effectively eliminated. An obstacle is placed just after the target point. The corridor following, obstacle avoidance, and target reaching strategies are also evaluated. For each scenario, the proposed controller is compared each time to the previous IT2FL controller [Melingui *et al.* 2014e], to assess whether or not the proposed controller achieves the expected objectives (i.e., both IT2FL emulation and computationally efficient).

The Robotino Mobile robot shown in Fig.4.7 is used as a platform for experimentation. It is a mobile robot formed of three omnidirectional drive units placed from one another at an angle of 120^o. Each drive unit contains a DC motor with an optical shaft encoder, and a 1:16 transmission ratio gearbox. All three drive units are integrated into a steel chassis surrounded by nine infrared distance sensors. There are numerous application programming interfaces available that allow the user to program Robotino in high-level languages such as C++, Matlab, etc. By using the real-time Linux layer, Robotino can be controlled from an external PC via wireless communication. A low-level control system comprising three PID controllers provides the motors with the appropriate currents to track longitudinal and lateral reference velocities.

The experimental results in the case of the first scenario which assesses the strategies of corridor following, obstacle avoidance, and target reaching in presence of obstacles with regular geometry are shown in Figs.4.7, 4.8, and 4.9. Fig.4.7 shows that both IT2FL and IT2FNN can achieve both the target reaching and obstacle avoidance. The robot follows the shorter and smoother path using IT2FNN rather than IT2FL. This implies that the IT2FNN can reach the target position faster than IT2FL. However, the IT2FNN passes closed to some obstacles. Fig.4.8

shows that the longitudinal velocity of IT2FNN is more than IT2FL. Besides, IT2FL has more oscillating behavior, mainly when approaching the target position. In Fig.4.9, we observe almost the same velocity ranges for both controllers, with IT2FL having a more stable velocity than IT2FNN. From the above, we notice that the IT2FL tries to faithfully apply the different fuzzy rules, sometimes leading to some abrupt variations in its trajectory, while the IT2FNN, in addition to applying them, performs some interpolation between these rules (a kind of generalization). Table I shows the moved distances and the time spent by each controller to reach the goal position.

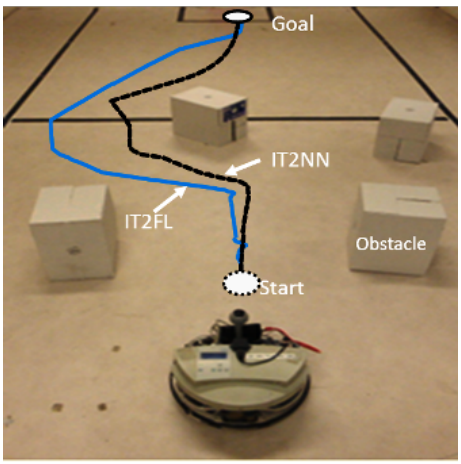

Figure 4.7: Scenario I: The strategies of corridor following, obstacle avoidance, and target reaching strategies in presence of obstacles with regular geometry.

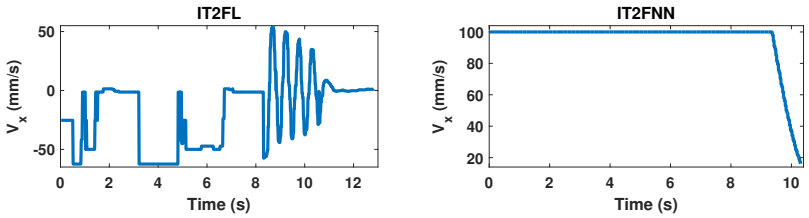

Figure 4.8: Scenario I: Longitudinal velocities provided by IT2FL and IT2FNN controllers

Table 4.2: COMPARISONS OF THE EXPERIMENTAL RESULTS

Exp	Moved distances(m) IT2FL	IT2FNN	Traveled Times (s) IT2FL	IT2FNN
(0,0)–(300,0)	3.35	3.13	14.04	10.30
(0,0)–(250,0)	2.35	2.53	10.85	9.962
(0,0)–(300,-50)	3.38	3.16	14.23	10.89

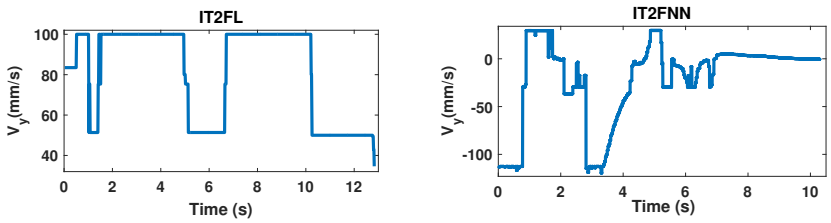

Figure 4.9: Scenario I: Lateral velocities provided by IT2FL and IT2FNN controllers

In the second scenario, an obstacle is placed just after the target position to evaluate the behaviour of the controllers in the presence of local minima situations. This scenario also evaluates the strategies of corridor following, obstacle avoidance, and target reaching. The experimental results obtained are given in Figs.4.10, 4.11, and 4.12. Fig.4.10 shows that both IT2FL and IT2FNN can achieve target reaching and obstacle avoidance while dealing with minima situations. The robot follows the shorter path using IT2FL and a smoother one using IT2FNN. However, the IT2FNN remains faster than IT2FL. The moved distances and the time spent by each controller are provided in Table I. The velocity ranges are the same and the longitudinal velocity of the IT2FNN is again more stable than IT2FL as shown in Figs.4.11. The lateral velocities range are the same as in the previous scenario and both controllers have almost the same behaviours. We observe similar behaviour as in Scenario 1.

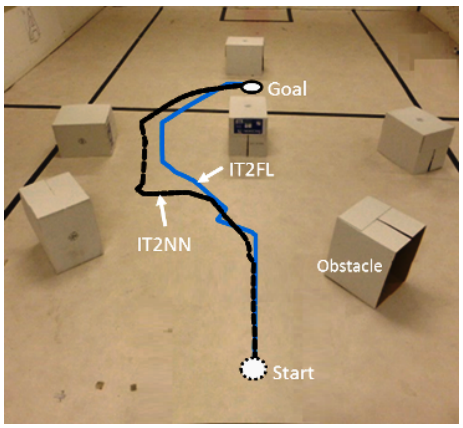

Figure 4.10: Scenario II: The strategies of corridor following, obstacle avoidance, and target reaching in presence local minima situations.

In summary, an analysis of the results obtained in the three scenarios reveals that both controllers can reach a given target position while avoiding the obstacles encountered. We can also say that the expected objectives have been achieved, i.e. to design a controller that is both robust and computationally efficient. Moreover, in addition to the objectives initially defined, we have noticed that the IT2FNN's generalization capacity allows it to better handle the transition phases between

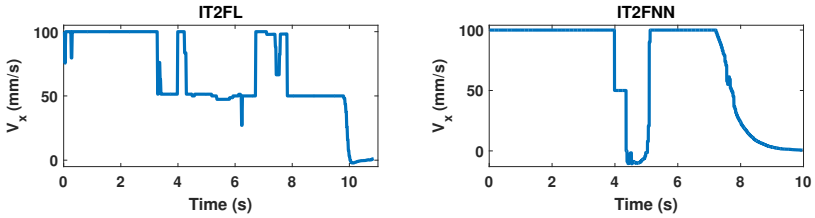

Figure 4.11: Scenario II: Longitudinal velocities provided by IT2FL and IT2FNN controllers

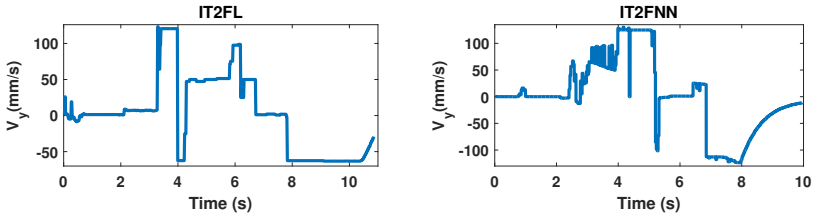

Figure 4.12: Scenario II: Lateral velocities provided by IT2FL and IT2FNN controllers

the different fuzzy rules. A kind of interpolation is thus carried out between the different fuzzy rules. We have also noticed that IT2FL tries to faithfully implement the different fuzzy rules to the detriment of the shape of the generated trajectory (presence of oscillations and abrupt variations). Finally, the interpolation between the different fuzzy rules performed by the IT2FNN sometimes causes the robot to move too close to certain obstacles.

4.4 Conclusion

We have proposed a robust and computationally efficient navigation system for mobile robots using an IT2FNN. The K-M type-reduction operation is replaced by two design q-factors to make the controller computationally efficient, which are learned by the gradient back-propagation algorithm in the IT2FNN structure. As shown in the experimental results, the mobile robot using the IT2FNN controller can achieve both goals and obstacle avoidance in a satisfactory manner, whatever the form of the obstacle encountered. Comparisons of the moved distances and the traveled time indicate that the robot's performance with IT2FNN is better than the IT2FL controller proposed in [Melingui *et al.* 2014e]. That is, the new IT2FNN will guarantee the shorter traveled time and achieve the shorter moved distances than the IT2FL before. Therefore, we can conclude that using IT2FNN for the mobile robot, the new navigation system is a more practical approach and works better than the T2FL controller proposed in [Melingui *et al.* 2014e], as tested in experiments. The fuzzy neural network structure, which can perform multiple tasks simultaneously at a reasonable time, is expected to bring these advantages. We

could expect additional studies involving the IT2FNN for autonomous mobile robot navigation to take place.

Chapter 5

Modelling and control of continuum manipulators

5.1 Introduction

In the recent decade, continuum manipulators have been the subject of intensive researches, mainly thanks to their inherent flexibility. They can bend along their structure and offer agile positioning even in constraint environments. This characteristic make them suitable for various applications such that surgical interventions [Simaan *et al.* 2004], rescue [Casper & Murphy 2003], and exploration [Walker 2013]. The effective exploitation of the characteristics offered by continuum manipulators requires rigorous modeling and the development of appropriate control laws. In the literature, many continuum manipulator platforms such that steerable catheters [Camarillo *et al.* 2008], multi-backbone snake-like robots [Jones & Walker 2006, Rolf & Steil 2014, Melingui *et al.* 2015], concentric tube robots [Dupont *et al.* 2009], exist, and the modelling and control difficulty varies from one platform to another, the most challenging being the pneumatic actuating ones [Jones & Walker 2006, Rolf & Steil 2014, Melingui *et al.* 2015].

Regarding kinematic modeling, several efforts have been invested. From constant curvature approach [Webster & Jones 2010] to the finite element approach [Bieze *et al.* 2018] through variable constant curvature [Mahl *et al.* 2014], data-driven [Melingui *et al.* 2014c] and curve-based approaches [Singh *et al.* 2018b], forward kinematic modelling approaches abound in the literature. However, although all are effective, there are some difficulties and limitations when applying on some continuum platforms. Inverse kinematic modeling has been a long time, one of the challenging problems in modeling and controlling continuum manipulators. The forward kinematic equations obtained being mathematically intractable, most of the researches have oriented to numerical solvers [Godage *et al.* 2015], which can become time-consuming as the number of continuum robots' sections increase. For sake of real-time implementation, some researchers have also interested to data-driven methods [Rolf & Steil 2014, Melingui *et al.* 2015, Giorelli *et al.* 2015]. However, despite all the efforts made, an effective approach applicable to all platforms and whatever the number of sections does not yet exist.

Regarding the control of continuum manipulators, several control strategies have been proposed [Yip & Camarillo 2014, Qi *et al.* 2016, Melingui *et al.* 2017b, Gravagne & Walker 2002, Wang *et al.* 2020] but they generally remain platform dependent.

This chapter focuses on data-driven approaches for modeling and control of continuum manipulators. They are free from assumptions and yield accurate models; the sole drawback is the database building, which becomes more challenging to build as the number of dof increases. It should be noted that, ideally, model-based approaches should be used, as there are already well-structured control laws. Data-based approaches make sense only when the system to be controlled is difficult to model or has un-modeled dynamic parts.

This chapter begins with the forward kinematic modeling, follows with inverse kinematic modeling, and ends with control strategies.

5.2 Forward kinematic modelling

Among the methods proposed to tackle the forward kinematics of the continuum manipulators, a constant curvature bending approximation is commonly used in the modeling process because of its simplification [Webster & Jones 2010, Godage *et al.* 2015, Escande *et al.* 2015]. A complete review of the different modeling approaches yielding equivalent results of constant curvature forward kinematic is provided in [Webster & Jones 2010]. From a modeling simplification point of view, the constant curvature modeling method remains one of the best model-based methods, which is increasingly true when the applications do not require a high degree of accuracy. However, in applications requiring high precision, data-driven approaches can be used. It is important to note that it is more suitable for cylinder-shaped continuous manipulators. Regarding data-driven-based methods, from machine learning to deep learning through reinforcement learning and Gaussian processes, learning architectures abound in the literature. The forward kinematic modeling problem can be seen as a non-linear regression problem, where the solving consists of establishing the non-linear map between configuration space variables (inputs) and workspace variables (outputs). Data-driven approaches generally yield acceptable performance; however, choosing an appropriate learning model remains a challenging task.

The section first presents the constant curvature modeling approach and follows with the data-driven approaches, namely neural networks and deep learning approaches.

5.2.1 Forward kinematic modelling: Constant curvature approach

This section gives a summary of the kinematics of the continuum manipulator based on the shape function [Godage *et al.* 2011b]. Fig. 5.1 shows the kinematic mapping for a continuum manipulator modelled according to the constant curvature approximation. First, we have a robot specific mapping that transforms actuator space variables (q) to configuration space (λ, ϕ, θ). Then, an independent mapping of the robot transforms this configuration space into a workspace.

This section focuses on the most popular continuum manipulators with 3 actuators per section. $q_i = [l_{i1}\ l_{i2}\ l_{i3}]^T$ is an actuator variable vector, where $i = 1, 2, \ldots, N$ is the section number and $j = 1, 2, 3$ the actuator number. Note

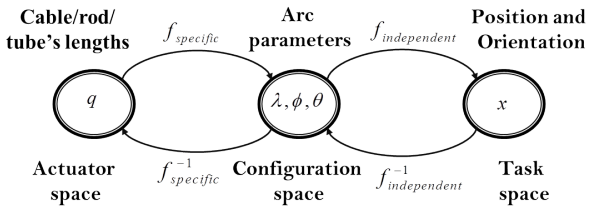

Figure 5.1: Kinematic mapping for a continuum manipulator modelled based on constant-curvature approach.

that the length of any actuator at time t is $L_{i0} + l_{ij}(t)$ with L_{i0} the initial actuator length. λ, ϕ, and θ are the curvature radius, the angle subtended by the bending arc, and the bending plane's angle relative to the $+X$ axis, respectively. These three geometrical parameters (λ, ϕ, θ) can specify the deformation of section i during actuation with respect to local frame $(o_i x_i y_i z_i)$ as shown in Fig. 5.2-b. In terms of multivariate Taylor series, the displacement of an arbitrary point in the continuum manipulator structure can be written as follows: [Godage *et al.* 2011a]:

$$u_f(t, \xi, q) = S(\xi)\, q_f(t, q) \tag{5.1}$$

$$u(t, \xi, q) = u_0(\xi) + S(\xi)\, q_f(t, q) \tag{5.2}$$

where $S \in R^{3\times n}$, $u \in R^{3\times 1}$, and $q_f \in R^{n\times 1}$ are shape matrix, modal position vector, and elastic coordinate vector, respectively. n is the number of terms obtained for a 6^{th} order Taylor approximation [Godage *et al.* 2011a]. The scalar $\xi \in [0, 1]$ defines the points across the neutral axis where $\xi = 0$ corresponds to the base of the section.

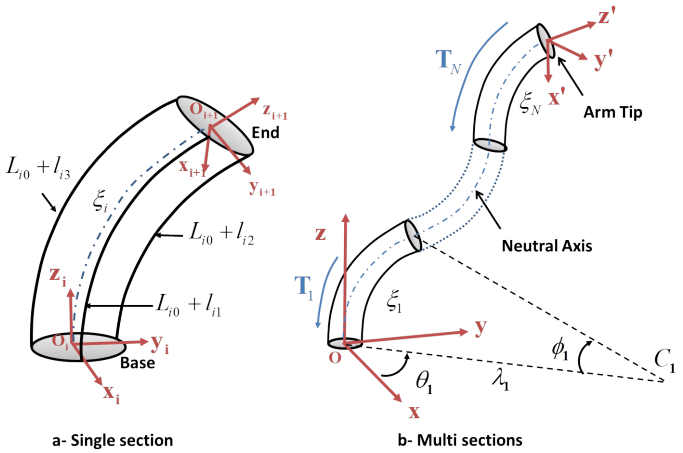

Figure 5.2: Continuum manipulator schematics

Referring to Fig. 5.2-a, for a given section i, a Homogeneous Transformation

Matrix (HTM) ${}^{i}_{i-1}\mathrm{T}$ is expressed as:

$$\begin{aligned} {}^{i}_{i-1}\mathrm{T}\left(\xi_i, q_i\right) &= \mathrm{R}_z\left(\theta_i\right)\mathrm{P}_x\left(\lambda_i\right)\mathrm{R}_y\left(\xi\phi_i\right)\mathrm{P}_x\left(-\lambda_i\right)\mathrm{R}_z^T\left(\theta_i\right) \\ &= \begin{bmatrix} \mathrm{R}\left(\xi_i, q_i\right) & p\left(\xi_i, q_i\right) \\ 0_{1\times 3} & 1 \end{bmatrix} \end{aligned} \tag{5.3}$$

with R_z and R_y being the translation matrices with respect to the Z and Y axes, and P_x is the translation matrix with respect to the X axis [Godage *et al.* 2011b]. The parameters λ_i, ϕ_i, and θ_i are expressed as:

$$\begin{aligned} \lambda_i\left(q_i\right) &= \frac{(3L_{i0}+l_{i1}+l_{i2}+l_{i3})R_i}{2\left(l_{i1}^2+l_{i2}^2+l_{i3}^2-l_{i1}l_{i2}-l_{i1}l_{i3}-l_{i2}l_{i3}\right)^{1/2}} \\ \phi_i\left(q_i\right) &= \frac{2\left(l_{i1}^2+l_{i2}^2+l_{i3}^2-l_{i1}l_{i2}-l_{i1}l_{i3}-l_{i2}l_{i3}\right)^{1/2}}{3R_i} \\ \theta_i\left(q_i\right) &= \tan^{-1}\left\{\frac{\sqrt{3}(l_{i3}-l_{i2})}{l_{i2}+l_{i3}-2l_{i1}}\right\} \end{aligned} . \tag{5.4}$$

As each element of ${}^{i}_{i-1}\mathrm{T}$ is basically a linear or angular displacement, the modal shape function for each segment of ${}^{i}_{i-1}\mathrm{T}$ is obtained from the multivariate extension of the Taylor series for section joint space variables at 0 to obtain the modal transformation matrix (MTM), ${}^{i}_{i-1}\mathrm{T}_\Phi$, as follows

$${}^{i}_{i-1}\mathrm{T}_\Phi\left(\xi_i, q_i\right) = \begin{bmatrix} \Phi_\mathrm{R}\left(\xi_i, q_i\right) & \Phi_p\left(\xi_i, q_i\right) \\ 0_{1\times 3} & 1 \end{bmatrix} \tag{5.5}$$

where Φ_p and Φ_R are modal position vector and modal rotation matrix, respectively. Having obtained the MTM for any i^{th} section, the MTM for all points on the entire continuum manipulator's neutral axis can be deduced relative to the global coordinate frame ($OXYZ$) by using classical coordinate transformation techniques.

$$\begin{aligned} {}^{N}_{0}\mathrm{T}_\Phi\left(\xi, q\right) &= \prod_{k=1}^{N}\left\{{}^{k}_{k-1}\mathrm{T}_\Phi\left(\xi_k, q_k\right)\mathrm{T}_k\right\} \\ &= \begin{bmatrix} \mathrm{R}_\Phi\left(\xi, q\right) & p_\Phi\left(\xi, q\right) \\ 0_{1\times 3} & 1 \end{bmatrix} \end{aligned} \tag{5.6}$$

where $q = \left[q_1^T, q_2^T, \ldots, q_N^T\right]^T$ is the actuator space vector and T_k is any actuator transformation present at the section joints [Godage *et al.* 2011a]. The scalar coefficient vector $\xi = [\xi_1, \xi_2, \ldots, \xi_N]^T$ is evaluated as $\xi = \{\xi_r = 1 \ : \forall r < i, \ \xi_i, \ \xi_r = 0 : \forall r > i\}$.

5.2.2 Forward kinematic modelling: Neural network-based approach

A comparative study made in the case of CBHA robot has shown that the commonly used learning models, namely the multilayer perceptron (MLP), radial based functions (RBF), support vector regression (SVR), and Co-Active adaptive neuro-fuzzy inference system (CANFIS) yield satisfactory performance [Mahamat L *et al.* 2020]. Regarding individual performance, the SVR model achieves the best performance, follows by CANFIS, MLP, and RBF, respectively. However, RBF has the best convergence time, followed by MLP, CANFIS, and SVR, respectively. This section gives

a summary of the modeling process based on neural networks. The description of the CBHA platform is first presented, then follows the construction of the learning database, and the section ends with the presentation of some results obtained.

Description of the CBHA platform

The CBHA robot is a two-section continuum manipulator inspired by the elephant's trunk and made from polyamide materials. It consists of two bending sections, each equipped with three pneumatic actuators (tubes), a wrist axis, and a compliant gripper, as shown in Fig. 5.3. A PID regulator controls the pressure supply in each tube. The elongations of the different tubes are provided by six-wire potentiometers placed along each tube. The elastic deformation of the CBHA results in movements with an infinite number of degrees of freedom. The properties of the polyamides and the pneumatic actuators that compose it make it a challenging platform as well for modeling as for control. It mainly inherits a compliance and memory effect from polyamide materials' properties and a hysteresis effect caused by its pneumatic actuators.

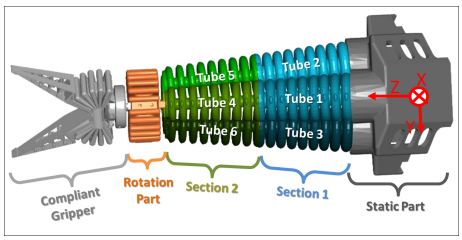

Figure 5.3: The CBHA of the RobotinoXT

Learning database building

The different models' performance can be assessed by comparing the poses provided by each model with those obtained from an optitrack motion capture system [OptiTrack 2018]. An example of experimental setup is depicted in Fig. 5.4. The motion capture system, composed of four cameras, can track a rigid body motion with an accuracy of ± 0.1 mm. The reflective markers are attached to the robot's end-effector, and the latter is moved throughout its workspace. The exploration of the CBHA's workspace is performed by providing pressure into the six CBHA's tubes. The input pressure vectors are obtained by discretizing the pressure variable with a step of 0.5 bar. An input vector $[p_1, p_2, \ldots, p_6]$, with $p_i \in [0.0, 0.5, 1.0, 1.5, 2.0]$ and $i = 1, 2, \ldots, 6$ is provided to CBHA tubes. This results in a sample database of 15625 and a robot operating time of approximately 13 hours. It is worth noting that samples are recorded after a complete mechanical equilibrium of the robot, which takes approximately 3 seconds. This delay is due to the pressure regulation system in the different tubes because the regulation is performed tube by tube.

For each input pressure vector, the robot's end-effector pose and the corresponding wire-potentiometer voltages are recorded.

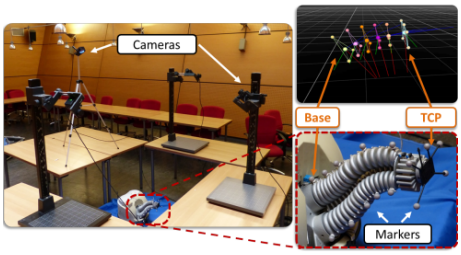

Figure 5.4: OptiTrack Motion Capture System

Results and comments

Table 5.1 lists the results obtained for each model, the mean square error (MSE), the learning time, as well as the parameters of each model are provided. One notes MSE of 5.3481×10^{-5}, 7.0691×10^{-5}, 2.3383×10^{-5}, and 4.5232×10^{-6} for MLP, RBF, CANFIS, and SVR, respectively. The learning times are approximately $01h45mn$, $01h15mn$, $02h30mn$, and $03h41mn$ for MLP, RBF, CANFIS, and SVR, respectively. Euclidean errors resulting after comparison with the poses provided by motion capture system is listed in Table 5.2 and Table 5.3. The maximal absolute Cartesian errors of $4.164mm$, 6.039, 2.79, and 3.638 are noticed for for MLP, RBF, CANFIS, and SVR, respectively, while the maximal orientation errors are 11.604, 15.148, 5.667, and 6.976, respectively.

Table 5.1: Results of MLP, RBF, SVR and CANFIS.

Models	Parameters	MSE	learning time
MLP	28 neurons	$5.3481e^{-5}$	$01h45$
RBF	60 neurons, $\sigma = 0.5$	$7.0691e^{-5}$	$01h15$
CANFIS	15 If-then rules	$2.3383e^{-5}$	$02h30$
SVR	$C = 4000$, $\sigma = 1.2$	$4.5232e^{-6}$	$03h41$

Table 5.2: Absolute value of Euclidean errors with position

Models	$X(mm)$	$Y(mm)$	$Z(mm)$
MLP	2.679	3.219	4.164
RBF	5.960	5.302	6.039
CANFIS	2.033	3.116	3.638
SVR	1.406	1.191	2.796

Table 5.3: Absolute value of Euclidean errors with orientation

Models	$\psi°$	$\theta°$	$\phi°$
MLP	11.604	2.224	10.876
RBF	15.148	3.183	12.096
CANFIS	6.852	1.746	6.976
SVR	4.870	0.856	5.667

5.2.3 Forward kinematic modelling: Deep learning-based approach

Neural networks-based approaches allow obtaining accurate models. However, they suffer from the learning database explosion that wears down the manipulator during data collection. This shortcoming can be solved by associating model and learning-based approaches [Lakhal *et al.* 2015]. Thus, the learning database can be derived from analytical equations to prevent the robot from operating for long periods; and the huge database obtained may be better handled with Deep Neural Networks [bou]. This section gives a summary of the modeling process based on deep learning. It starts with the modeling hypotheses, then follows the inverse kinematic modeling of an inter-module and the derivation of the FKM of an inter-module based on deep learning. The section ends with the database collection process, the optimization of the deep neural network architecture, and the derivation of the FKM of the entire manipulator.

Assumptions

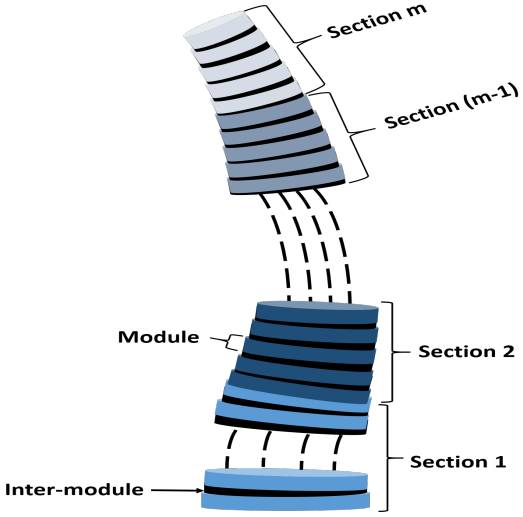

Figure 5.5: Structure of a class of continuum manipulators. The inter-modules are shown in black while the modules are blue.

The structure of a conical shaped continuum manipulator can be represented, as shown in Fig.5.5. Some assumptions are considered to model its kinematic behavior:

1. The manipulator is considered as a series of many sections, each controlled independently.

2. A section is considered as a series of many modules.

3. An inter-module is flexible and non-deformable with k-DoFs mobility, as shown in Fig.5.6.

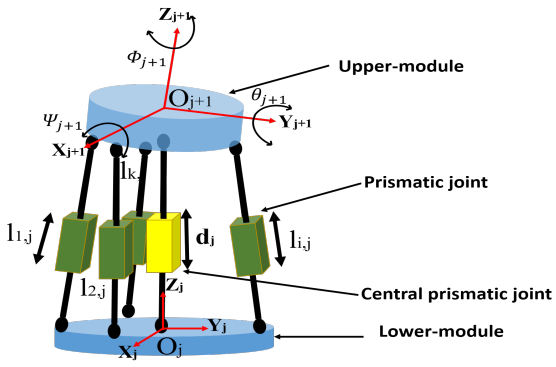

Figure 5.6: Inter-module's paramaters in the general case

Based on these assumptions, an inter-module depicted in Fig.5.6 can be considered as a rigid parallel robot composed of k independent kinematic chains whose ends are connected by two rigid platforms, namely the lower and upper modules. The lower ends of the joints linked to the lower module form a regular polygon, as shown in Fig.5.7. This is also the case for the upper ends linked to the upper-module. These joints have variable lengths and are represented by $l_{i,j}$. They provide the position and orientation of the upper module relative to the lower module. The variable represented by d_j is the distance between the centers of gravity of the lower and upper modules; it is considered as a passive joint. The orientation represented by ψ_j, θ_j and ϕ_j are the roll, pitch, and yaw angles, respectively. The inter-module is then modeled as a parallel robot with k universal-prismatic spheric and one universal-prismatic (kUPS-1UP joints).

Inverse Kinematic Equation of an inter-module

The inverse Kinematic Equations (IKEs) of an inter-module are obtained by calculating the joint variables $l_{i,j}$ of the parallel robot, where i and j are the number of the modules and the number of the universal-prismatic links, respectively. As far as translation movements are concerned, only translation along the z-axis is possible because there are no translations along the x and y axes. The translation along the z-axis is denoted by $Trans(z, \mathrm{d}_j)$. The IKEs can, therefore, be formulated as follows:

$$l_{i,j} = f_i(\psi_j, \theta_j, \phi_j, \mathrm{d}_j) \tag{5.7}$$

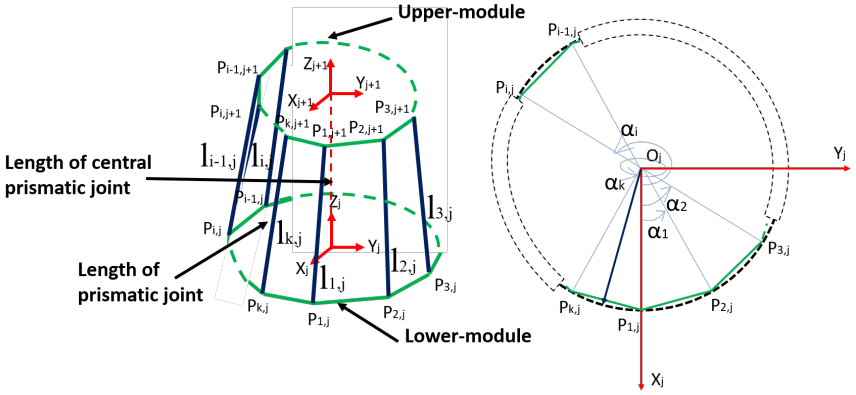

Figure 5.7: Spatial configuration of an inter-module

where $l_{i,j}$ are the lengths of the universal-prismatic links and ψ_j, θ_j and ϕ_j are the roll, pitch, and yaw angles, respectively.

The endpoints of the module j, the center of the circumscribed circle, and the radius of the circumscribed circle are defined by $P_{i,j}$, O_i, and R_j, respectively. The coordinates of the first endpoint $P_{1,j}$ located on x_j-axis and the origin O_i of the reference frame $\Re_j(O_i, x_j, y_j, z_j)$ are given as follows:

$$P_{1,j} \begin{bmatrix} R_j \\ 0 \\ 0 \end{bmatrix} \text{ and } O_i \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

The coordinates of the endpoint $P_{i,j}$ are obtained by performing a rotation $Rot(z_j, \alpha_i)$. Thus, one has

$$O_iP_{i,j} = Rot(O_j, \alpha_i)O_iP_{1,j},\ P_{i,j} \begin{bmatrix} R_j \cos\alpha_i \\ R_j \sin\alpha_i \\ 0 \end{bmatrix} \tag{5.8}$$

with $Rot(O_j, \alpha_i) = \begin{pmatrix} \cos\alpha_i & -\sin\alpha_i & 0 \\ \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 1 \end{pmatrix}$ and $\alpha_i = \frac{2(i-1)\pi}{k}$.

The coordinates of the endpoint $P_{i,j+1}$ relative to the endpoint $P_{i,j}$ in the reference frame $\Re_j(O_i, x_j, y_j, z_j)$ are given as follows:

$$\begin{bmatrix} L_{i,j} \\ 1 \end{bmatrix} = {}^{j}_{j+1}T \begin{bmatrix} P_{i,j+1} \\ 1 \end{bmatrix} - \begin{bmatrix} P_{i,j} \\ 1 \end{bmatrix} \tag{5.9}$$

where ${}^{j}_{j+1}T$ is the transformation matrix of the upper-module relative to the lower-module, and expressed as:

$${}^{j}_{j+1}T = \begin{pmatrix} r^j_{11} & r^j_{12} & r^j_{13} & 0 \\ r^j_{21} & r^j_{22} & r^j_{23} & 0 \\ r^j_{31} & r^j_{32} & r^j_{33} & d_j \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{5.10}$$

with

$$
\begin{aligned}
r^j_{11} &= C\phi_j C\theta_j \\
r^j_{12} &= -S\phi_j C\psi_j + C\phi_j S\theta_j S\psi_j \\
r^j_{13} &= S\phi_j S\psi_j + C\phi_j S\theta_j C\psi_j \\
r^j_{21} &= S\phi_j C\theta_j \\
r^j_{22} &= C\phi_j C\psi_j + S\phi_j S\theta_j S\psi_j \\
r^j_{23} &= -C\phi_j S\psi_j + S\phi_j S\theta_j C\psi_j \\
r^j_{31} &= -S\theta_j \\
r^j_{32} &= C\theta_j S\psi_j \\
r^j_{33} &= C\theta_j C\psi_j
\end{aligned}
$$

The symbols S and C denote the sine and cosine functions, respectively. $L_{i,j}$ is a 3×1 matrix. Finally, the prismatic variable $l_{i,j}$ is equal to the distance between the endpoints $P_{i,j+1}$ and $P_{i,j}$.

$$
l^2_{i,j} = {L_{i,j}}^T L_{i,j} \tag{5.11}
$$

Thus, after introducing (6.2) and (6.11) into (6.4), (6.12) can be rewritten in the format of (6.1)

$$
\begin{aligned}
&l_{i,j} = sqrt(R^2_{j+1} + R^2_j + d^2_j - 2R_{j+1}d_j(\cos\theta_j \sin\psi_j \sin\alpha_i + \sin\theta_j \cos\alpha_i) \\
&-2R_{j+1}R_j(\cos\theta_j \cos\alpha_i \cos(\alpha_i - \phi_j) + \cos\psi_j \sin\alpha_i \sin(\alpha_i - \phi_j) \\
&+ \sin\theta_j \sin\psi_j \sin\alpha_i \cos(\alpha_i - \phi_j)))
\end{aligned} \tag{5.12}
$$

FKM of an inter-module: Deep Neural Networks-based solution

The FKM of an inter-module is achieved by solving the IKEs. This comes down to finding the relation between the joint variables $l_{i,j}$ of an inter-module and the parameters of the upper module ($\psi_j, \theta_j, \phi_j, \mathrm{d}_j$) relative to the lower module frame, and defined as

$$
(\psi_j, \theta_j, \phi_j, \mathrm{d}_j) = f_i^{-1}(l_{i,j}). \tag{5.13}
$$

Equation (5.37) being highly nonlinear, numerical methods, such as Newton-Raphson, least-squares, or symbolic calculation, are generally used to provide an approximate solution. However, for computational time constraints, learning-based methods become an alternative [Lakhal *et al.* 2015]. A deep learning-based solution can be implemented to provide an approximate solution [bou].

Deep learning(DL) methods aim to learn feature hierarchies with features of higher levels of the hierarchy formed by the composition of lower levels' features. Automatically learning features at multiple abstraction levels allow a system to learn complex functions, map the input to output directly from data, without relying entirely on human-made features. This is particularly important for higher-level abstractions, which humans often do not know how to specify explicitly in terms of raw sensory input [Bengio *et al.* 2009]. The architecture of DL adopted in this work is a Deep Neural network (DNN) with many levels of non-linearities allowing it to

represent highly nonlinear regression functions such as those obtained in (5.37). It has the advantage of allowing the initialization of weights in the region close to a good local minimum thanks to pre-training algorithms [Larochelle *et al.* 2009].

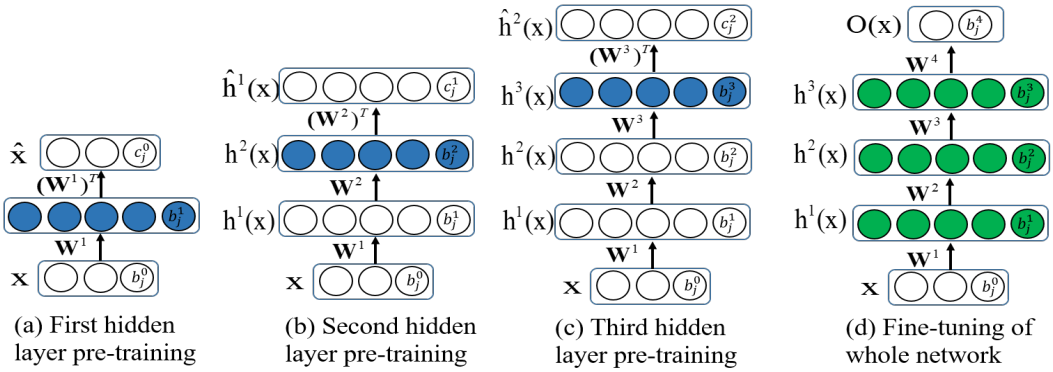

Figure 5.8: Training procedure for DNNs using the autoencoder (AE) algorithm for pre-training. (a), (b) and (c) show the pre-training process of the first, second, and third hidden layers, while (d) shows the fine-tuning of the entire network.

Let $\mathbf{x}$ be the input vector, $h_j^i(\mathbf{x})$ the output of the j^{th} neuron in the i^{th} layer with $i = 0$ the input layer, $i = l + 1$ the output layer, and l the number of hidden layers. The default activation level is determined by the internal bias b_j^i of each unit. The output $h_j^i(\mathbf{x})$ is expressed as follows:

$$h_j^i(\mathbf{x}) = f(a_j^i(\mathbf{x})), with\ a_j^i(\mathbf{x}) = b_j^i + \sum_k W_{jk}^i h_k^{i-1}(\mathbf{x}) \tag{5.14}$$

where W_{jk}^i denotes the weights matrix between the $(i-1)^{th}$ and i^{th} layers, h_k^{i-1} the output of the k^{th} neuron in the $(i-1)^{th}$ layer, and $f(.)$ the activation function with $h^0(\mathbf{x}) = \mathbf{x}$.

The output layer is determined in the same way as follows:

$$\mathbf{O}(\mathbf{x}) = \mathbf{f}(\mathbf{a^{l+1}}(\mathbf{x})), \mathbf{with}\ \mathbf{a^{l+1}}(\mathbf{x}) = \mathbf{b^{l+1}} + \mathbf{W^{l+1}}\mathbf{h^l}(\mathbf{x}) \tag{5.15}$$

The training of DNNs as a regression model consists of an unsupervised pre-training part and a supervised fine-tuning part, as shown in Fig.5.8. In the pre-training part, the training is performed layer-by-layer in an unsupervised greedy fashion to maximize training samples' likelihood. Since the backpropagation algorithm is used, the type of hidden units can be a sigmoid (sigm), a hyperbolic tangent (tanh), or an arctangent (atan). Indeed, these activation functions make the backpropagation algorithm stable. Besides, they have an acceptable degree of smoothness and are easily differentiated [Hornik *et al.* 1989], unlike a ReLU function, which has a differentiation problem that can lead to a dying ReLU problem [Hu *et al.* 2018]. The interested reader is referred to [Larochelle *et al.* 2009] for a detailed description of DNN and autoencoder algorithms.

Database generation

The deep learning approach has been implemented to numerical solve the IKEs (5.37) for real-time implementation and prevent the robot from operating for long periods. This subsection describes the database generation procedure. The algorithm showed in Fig. 5.9 is implemented for this purpose. The principle consists first of all in setting the minimum and maximum limit values of the different parameters θ_j, ψ_j and d_j. These limits are obtained experimentally. The parameters are then discretized within the defined ranges, and the equation (5.37) is evaluated. Finally, the sets of parameter values θ_j, ψ_j and d_j whose values of the inter-modules elongations $l_{i,j}$ are within the limits of the latter are saved. A learning base of 32000 samples has been obtained in the case of the CBHA robot.

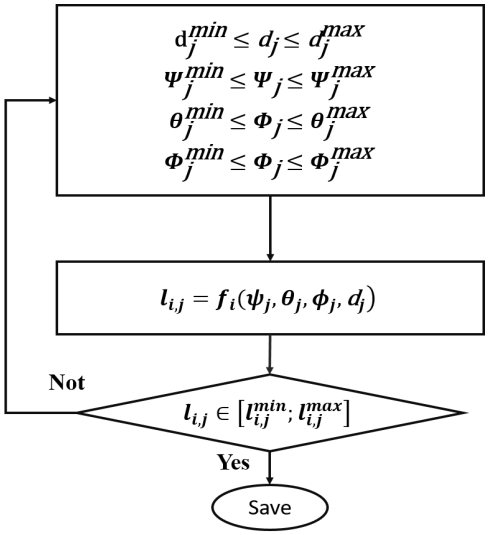

Figure 5.9: Generation algorithm of the learning database

Optimization of DNN parameters

The hyper-parameters of the DNN, namely the number of neurons by hidden layer, the activation function, the learning rate, and the number of epochs, are optimized using particle swarm optimization (PSO) [Poli *et al.* 2007]. Unlike simulated annealing [Dekkers & Aarts 1991] and genetic algorithm [Tanese *et al.* 1989], particle swarm optimization ensures an optimal solution in a short period. Besides, simulated annealing requires a much greater number of investigations to produce optimal solutions [Ethni *et al.* 2009], while genetic algorithm requires a complex coding and decoding process that increases the number of investigations [Savsani *et al.* 2010]. The performances achieved by the different algorithms in the pre-training of Deep Neural Network are listed in Table 5.4. We notice that the PSO algorithm achieves the best performance in terms of computation time. Moreover, the side of the neural network architecture is also acceptable.

Table 5.4: Results of the optimization algorithms for pre-training process

Algorithms	solution finding time in secondes	Performance on validation set	number of neurons
Genetic	1019	2.0346×10^{-5}	281
Particle swarm	803	2.3879×10^{-5}	256
Simulated annealing	1694	4.8105×10^{-5}	265

In the PSO algorithm [Poli *et al.* 2007], the particles are placed in space, and each particle evaluates the objective function according to its current location and the problem to be solved. Then, each particle calculates its displacement in the search space, taking into account its best location and those of its neighboring particles, the history of its own current location, and some random perturbations. Once all the particles have been moved, the calculation is repeated. The particles are handled according to the following equations [Eberhart & Shi 1998]:

$$v_{id} = w \cdot v_{id} + c_1 \cdot rand() \cdot (p_{id} - x_{id}) + c_2 \cdot rand() \cdot (p_{gd} - x_{id}) \tag{5.16}$$

$$x_{id} = x_{id} + v_{id} \tag{5.17}$$

where x_{id} is an element of the i^{th} particle $X^i = \left(x_1^i, x_2^i, \ldots, x_D^i\right)$ with D the dimension of the vector X^i. v_{id} is an element of the rate of the position change (velocity) of the i^{th} particle $V^i = \left(v_1^i, v_2^i, \ldots, v_D^i\right)$. p_{id} is an element of the best previous position of the i^{th} particle $P^i = \left(p_1^i, p_2^i, \ldots, p_D^i\right)$ and p_{gd} the best particle among all the particles in the population. c_1 and c_2 are the acceleration coefficients of v_{id} in the direction of the best p_{id} and p_{gd}. The inertia weight w is employed to control the impact of the previous history of velocities on the current velocity, thereby influencing the trade-off between global (wide-ranging) and local (fine-grained) exploration abilities of the "flying points".

In the case of DNN parameters optimization, the number of neurons per hidden layer, the activation function for each layer, the learning rate, and the number of epochs constitute the particle position vector elements X^i. P^i represents the previous best parameter of DNN, and p_{gd} is the best of DNN parameters among all DNN parameters in the population. V^i is formed of the step size of DNN parameters.

Derivation of Forward Kinematic Model

The FKM is obtained by following the process depicted in Fig. 5.10. The joint variables ($l_{i,j}$) of the inter-modules are obtained from the lengths provided by each tube's length sensors. These joint variables are used as input to the DNNs model, and the end-effector pose is obtained by using the transformation matrices ${}^{j}_{j+1}T$, for $j = 1, ..., N-1$. The composition of homogeneous transformations allows obtaining the end-effector pose as follows:

$$\prod_{j=1}^{N} {}^{j}_{j+1}T = \begin{pmatrix} r_{11}^f & r_{12}^f & r_{13}^f & X_f \\ r_{21}^f & r_{22}^f & r_{23}^f & Y_f \\ r_{31}^f & r_{32}^f & r_{33}^f & Z_f \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{5.18}$$

with

$$\begin{cases} \Psi_f = \text{atan2}(r_{32}^f, r_{33}^f) \\ \Theta_f = \text{atan2}\left(-r_{31}^f, \sqrt{\left(r_{11}^f\right)^2 + \left(r_{21}^f\right)^2}\right) \quad \text{if} \quad \Theta_f \neq \pm\frac{\pi}{2} \\ \Phi_f = \text{atan2}(r_{21}^f, r_{11}^f) \end{cases}$$

and

$$\begin{cases} \Phi_f - sign(\Theta_f)\Psi_f = \text{atan2}(r_{23}^f, r_{13}^f) \\ or \qquad\qquad \text{if} \quad \Theta_f = \pm\frac{\pi}{2} \\ \Phi_f - sign(\Theta_f)\Psi_f = -\text{atan2}(r_{23}^f, r_{13}^f) \end{cases}$$

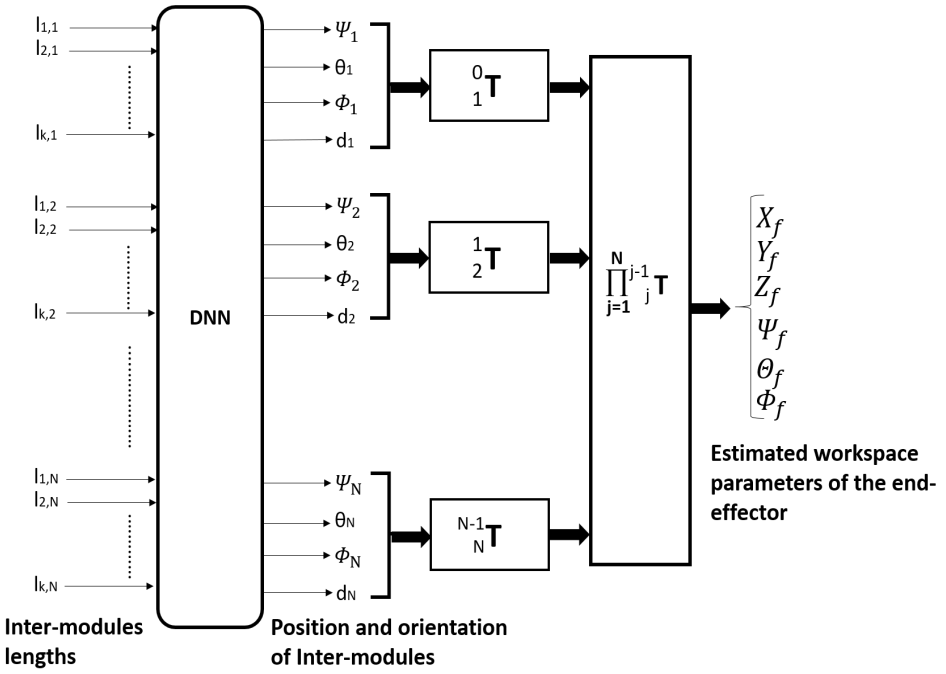

Figure 5.10: Computation of FKM in using the DL

5.3 Inverse kinematic modelling of continuum manipulators

Inverse kinematic (IK) maps for continuum manipulators are generally one-to-many relation problems, i.e., for a given Cartesian pose, an infinite number of configurations can be associated. It is well-known that the merit of redundant robots'

redundancy resolution resides in achieving additional goals, namely the avoidance of physical joint limits, the avoidance of obstacles, and the avoidance of singularity positions. The IKs of continuum manipulators can be solved through analytical or numerical approaches.

Analytical approaches establish a set of equations and preconditions that allow direct mapping of EE poses and configuration space variables. They benefit from being quick and providing all possible solutions for a given EE pose. In the case of rigid manipulators, Pieper's solution [Peiper 1968] can be used for manipulators having a kinematic configuration involving three concurrent articular axes. However, outside these particular cases, it is difficult to solve the IKs of a general redundant mechanism (more than six DOF). Chirikjian et al. [Chirikjian & Burdick 1994] proposed an analytical modal method to address the IK problem of hyper-redundant manipulators. This is based on a collection of intrinsic curve shape functions of the backbone. Neppalli et al. [Neppalli *et al.* 2009] have proposed an analytical approach to solving the IK of multi-section robots in the continuum. Assuming that the endpoints of all sections of the robot continuum are known, a closed-form IK solution of the multi-section continuum was developed. However, the modelling only considers the distance constraints between the robot's continuous sections, while position and orientation are physically coupled. As a result, solutions with combinations of curve parameters that could not be transformed into physically feasible configuration space variables could result.

On the other hand, when the problem of IKs cannot be solved analytically, numerical methods are used. They have the advantage of working irrespective of the number of DOF of the manipulator. The IKs of non-redundant rigid manipulators are generally solved using the inverse of the Jacobian matrix [Hollerbach 1985]. On the other hand, the pseudo-inverse [Klein & Huang 1983] or the extended Jacobian inverse [Klein *et al.* 1995] are applied to the redundant ones. In the case of continuum robots, much effort has also been invested in obtaining fast and robust IK algorithms. Jacobian-based numerical methods have been implemented in [Jones & Walker 2006], [Simaan *et al.* 2004], [Webster *et al.* 2009], and [Mahl *et al.* 2014] for the Air-OCTOR, Snake-like, Concentric-tube, and Bionic Handling Assistant (BHA) continuum robots, respectively. A linear transformation mapping between tendon displacements and beam configuration has been proposed in [Camarillo *et al.* 2008] by implementing minimum and maximum norm algorithms. Godage [Godage *et al.* 2015] proposed a kinematic model for continuum manipulators based on modal shape functions. The method aimed to address some limitations associated with the parametric kinematic curve model [Jones & Walker 2006], namely the difficulty of modeling straight arm poses of continuum sections due to a numerical singularity. The iterative approach, depending on the Jacobian matrix, was used to deduce IKs. However, the Jacobian matrix approaches usually suffer many disadvantages, including processing and high implementation time, local minima, and singularities. Moreover, according to the quality of the initial guess or the distribution of the singularities in the configuration space, poor performance or non-convergence problems can occur. [Guilamo *et al.* 2005].

Machine learning tools have also been explored to solve the IKs of continuum manipulators. A comparison of the performance of a feedforward neural network and a Jacobian method has been proposed in [Giorelli *et al.* 2015] in the framework of IK modeling of nonconstant curvature manipulators. The results revealed that the feedforward neural network was faster and more reliable than the Jacobian approach. In the IKs of the BHA manipulator, a goal babbling approach was used [Rolf & Steil 2014]. However, in these methods, only one IK solution was provided among the redundancy manifolds. A new analysis of self-motions, where the tip of the manipulator remains in a fixed position, has been proposed in [Kapadia & Walker 2013] for two sections planar continuum manipulators. Three types of self-motions are defined, namely: a first is due to the manipulator's extension; a second is due to the manipulator's bending; and a third, where the manipulator is entirely free of motion.

In previous work, a neural network approach via a distal supervised learning architecture was proposed to solve the IK problem of the CBHA manipulator[Melingui *et al.* 2015]. In [Melingui *et al.* 2017b], support vector regressors have also been used for offering ideal and identical solutions with less evaluation time and small-size regressors to speed up the convergence of the closed-loop learning system. A hybrid approach was used in [Lakhal *et al.* 2015] to solve the IKs. The CBHA was considered as a series of vertebrae, where a flexible link connected two successive vertebrae. Pythagorean Hodograph curves have also been used to deal with the IK problem of continuum robots[Singh *et al.* 2018b]. A 3D shape of the continuum manipulator is first reconstructed, and the IKs are then derived. However, in these approaches, only one optimal or non-optimal IK solution can be provided.

This section focuses only on data-driven modeling approaches. It essentially presents two approaches; the first one leads to a single inverse kinematic solution while the second one allows obtaining several inverse kinematic solutions.

5.3.1 Inverse kinematics: distal supervised learning architecture

Direct inverse modeling treats the problem of learning an inverse model as a classical supervised learning problem. The idea is to observe the environment's input/output behavior and train the inverse model directly by reversing the inputs and outputs' roles. Data are provided to the algorithm by sampling in action space and observing the results in sensation space [Jordan & Rumelhart 1992]. Although direct inverse modeling has been shown to be a viable technique [Chiddarwar & Babu 2010], it has two drawbacks that limit its usefulness: First, when a many-to-one mapping characterizes the environment from actions to sensations, the inverse mapping will map more than one image to a given point. The particular manner in which the inconsistency is resolved depends on the form of the cost function. The sum-of-squared error yields an arithmetic average over points that map to the same endpoint (centroid). If the centroid does not belong to the images' manifold, the non-linear many-to-one mappings can yield non-convex inverse images. The second drawback with direct in-

verse modeling is that it is not "goal-directed." The algorithm samples in the action space without regard to particular targets or errors in sensation space. There is no direct way to find an action that corresponds to a specific desired sensation. To overcome the two problems, a Distal Supervised Learning [Jordan & Rumelhart 1992].

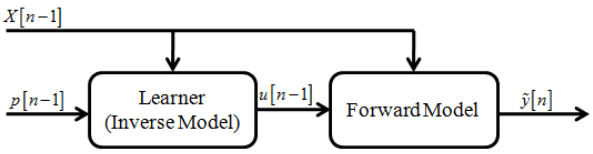

Figure 5.11: Composite learning system [Jordan & Rumelhart 1992]

The distal supervised learning consists of composing a learning system, as depicted in (Fig. 5.11). The current state of the environment is $X[n-1]$. Here, the state does not address issues relating to state representation and state estimation. State representations might involve delayed values of previous actions, or they might involve internal state variables induced as part of the learning procedure [Jordan & Rumelhart 1992]. The intention is $p[n-1]$, the action is $u[n-1]$, and the predicted outcome from the forward model is $\tilde{y}[n]$. We will also refer to the actual outcome as $y[n]$ and the desired output as $y^*[n]$. The forward model is a model that predicts the outcome of the environment, given the current state and the action. The forward model can be learned by applying actions and comparing actual outcomes $y[n]$ with predicted outcomes $\tilde{y}[n]$. The idea of [Jordan & Rumelhart 1992]for solving the inverse problem was to avoid direct inverse modeling entirely. They used the fact that the composition of the inverse and forward models must yield the identity function. They proposed to train a neural network to model the forward kinematics, using this network to train the inverse model indirectly. The composite learning system can be trained by any supervised learning algorithm (back-propagation algorithm, generalized delta learning rule). However, the learning algorithm must not alter the forward model (kept fixing forward weights) while the composite system is being trained. The inverse model will eventually be learned if the training input-output pairs stand for the identity function. In this way, the effect is that only one of the possible solutions is chosen for a given target point. Without additional information about the particular structure of the input-to-action mapping, there is no way of predicting which of the possibly infinite set of inverse models the procedure will find. Moreover, a further virtue of the distal learning approach is the possibility to incorporate additional constraints in the learning procedure.

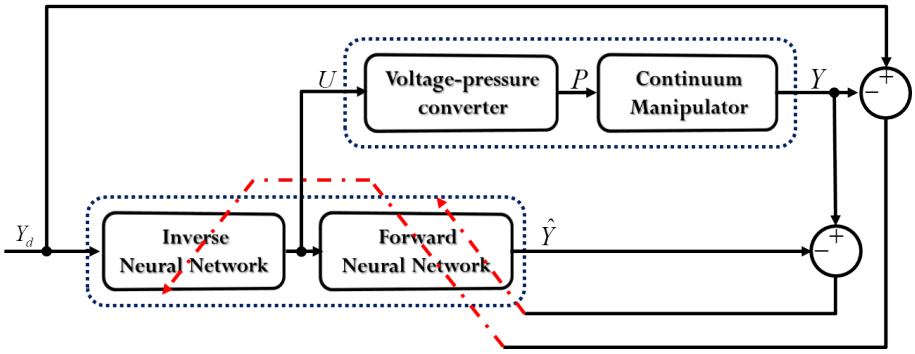

Figure 5.12: Learning architecture for IKM of CBHA

Focusing on the CBHA, the Forward Neural Network (FNN) is first learned to approximate the forward kinematics model of the CBHA (For more details about the forward kinematics model of the CBHA, we refer the reader to [Melingui *et al.* 2013b]). In the second phase, a particular inverse solution is obtained by placing the Inverse Neural Network (INN) and FNN in series, and by replacing the (Voltage-Pressure system + CBHA) with the forward kinematics model that had been trained previously (Fig. 5.12). At this stage, the composite learning system can be trained by any supervised learning algorithm (back-propagation algorithm in this work). However, the learning algorithm must be constrained to avoid altering the forward model (kept fixed forward weight matrices) while the composite system is being trained. For more details about DSL, we refer the reader to ([Jordan & Rumelhart 1992], [Melingui *et al.* 2014b]). $Y_d = [X_d, Y_d, Z_d]^T$ denotes the desired CBHA's tip position, and $Y = [X, Y, Z]^T$ is the real CBHA's tip position. $U = [U_1, U_2, ..., U_6]^T$ is the predicted wire potentiometer voltage, and $\hat{Y} = \left[\hat{X}, \hat{Y}, \hat{Z}\right]^T$ is the predicted CBHA's tip position. The FNN consists of 6 inputs (U), and 3 outputs ($\hat{Y}$), while the INN consists of 3 inputs (Y_d), and 6 outputs (U). The prediction error ($Y - \hat{Y}$) and the performance error ($Y_d - Y$) are respectively used for forward and inverse neural networks learning. A squared penalty term is added to the inverse neural network's objective function to select a particular inverse kinematics function. The cost function yields:

$$J = \frac{1}{2}(Y_d - Y)^T (Y_d - Y) + \lambda \frac{1}{2} \|U\|^2 \tag{5.19}$$

With $\|.\|$ the Euclidean norm. It has shown that [Wu *et al.* 2006], the larger the coefficient λ is, the smaller U becomes. The penalty term λ provides a possibility to effectively control the magnitude of U. Hence, to select a particular inverse solution.

5.3.2 Inverse kinematic with maintaining of multiple IK solutions

A single-section continuum manipulator's ability to reach any point located in the vicinity of its end-effector (EE) can be exploited to maintain multiple IK solutions

of continuum manipulators. The infinite number of IK solutions can be reduced to a finite number by discretizing the robot's actuating space. Besides, the IK problem of multi-section continuum manipulators can be transformed into single-section ones by parameterizing the first $(n-1)$ sections' actuating variables. The parameterization is performed by clustering the manipulator workspace using the growing neural gas (GNG) algorithm. The proposed method is split into two phases: one offline and one online. The offline process or learning phase includes:

- Discretizing the configuration space of the continuum manipulator;
- Building a learning base (q_i,x_i), with q_i the configuration vector and x_i the pose of the EE of the manipulator;
- Clustering the configuration space and workspace of the robot, i.e., forming clusters of all the configurations of the manipulator that lead approximately to the same position of the EE.

For a given EE pose, the online phase includes:

- Selecting the valid equivalent cluster;
- Considering all configuration vectors of the cluster (for multiple IK solutions) or selecting a particular configuration vector randomly or based on a redundancy resolution criterion (minimum bending energy configuration, avoidance of physical actuator limits, obstacle avoidance, or singularity avoidance);
- Keep the values of the configuration variables of the first $(n-1)$-sections of the selected configuration(s) and deduce the values of the last section's configuration variables that lead to the desired pose of the EE.

The Growing Neural Gas (GNG) network [Fritzke 1995] is utilized in workspace clustering. In contrast, a neighborhood function is used for the actuator space clustering [Kumar *et al.* 2010]. Unlike neural gas and self-organizing maps [Martinetz *et al.* 1990, Kohonen 1998], GNGs do not have parameters that alter with time. They will continue to learn, introducing neurons and connections until reaching a given performance criterion [Fritzke 1995]. The neighborhood function is introduced to ensure the conservative characteristic of IK solutions. GNGs have also been successfully implemented in the framework of the IKs of hyper-redundant rigid manipulators in [Kouabon *et al.* 2020a].

The proposed LIKM is illustrated in Fig. 5.13. The learning database consists of samples pairs (x_k, q_k), where x_k and q_k the pose of the manipulator tip and the vector of the actuator variables, respectively. The first step includes clustering the robot's workspace with clustering algorithms such as self-organization maps, neural gas, growing neural gas. The aim is to organize the workspace manipulator into clusters dependent on the similarities of input data. The input vectors in the same cluster are more similar to each other than those belonging to another cluster. This is done using only the Cartesian information of the samples. The Euclidean

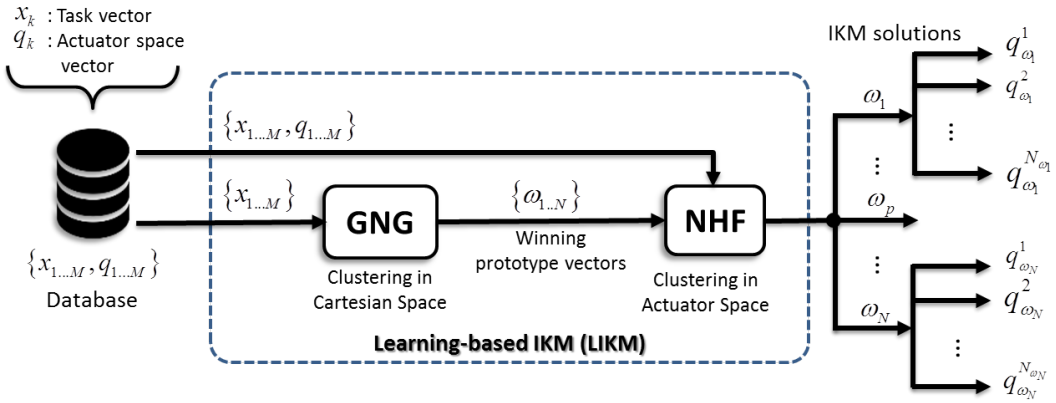

Figure 5.13: Proposed LIKM architecture: for a task pose vector that belongs to w_1 cluster, a set of N_{w_1} inverse kinematic solutions are possible.

distance between the samples is used as a clustering criterion. Clustering in the robot actuator space is the next step. The goal is to construct a pair of (x_k, q_k) while eliminating actuator-too-related space vectors. The neighborhood function is used in this stage to preserve the conservative properties of the IK solutions obtained. [Walter & Schulten 1993].

It should be noted that the generalization capability of neural network architectures ensures belonging for any point in the robot's workspace to a valid cluster. The next step, called redundancy resolution, is to select one particular actuator space vector in the redundancy manifolds where various performance criteria can be used. After the redundancy resolution, the final step consists of determining the end-section parameters by solving the non-redundant forward kinematic equations obtained after parameterizing the variables in the first $(n-1)$ sections.

Clustering in workspace

The IK problem of continuum manipulators usually results in an infinite number of IK solutions. However, some of them are too close to each other to be easily deduced from the others. The discretization of the workspace allows passing from an infinite number of IK solutions to a finite number. However, since only a limited number of workspace positions are considered, the learning database must cover all manipulator workspace regions. Thus, the actuating space's sampling period plays a crucial role in ensuring that redundant configurations are included in the training database. Fortunately, although it is generally challenging to find two separate actuation vectors that correspond to the same position of the manipulator, grouping in the workspace allows redundant actuation vectors to be formed. Another critical parameter is the maximum number of clusters in that it governs the number of redundant solutions. The ability of GNG [Fritzke 1995] to automatically insert new nodes into hidden layers eliminates this critical parameter.

The rapid convergence of GNGs compared to other incremental neural

network architectures such as fuzzy adaptive resonance theory (ARTMAP) [Carpenter *et al.* 1992], SOINN [Furao & Hasegawa 2006], growing cell structures [Fritzke 1994] and their low dependence on GNG's parameter variations make GNGs an ideal architecture for classification. Besides, given the complexity of continuum manipulator platforms, clustering methods that offer acceptable performance with small networks are practical for real-time implementation.

GNGs were widely used for data clustering [Qin & Suganthan 2004, Daszykowski *et al.* 2002, Fatemizadeh *et al.* 2003]. They learn a dynamic graph with a variable number of neurons and connections. The principle is to successively add new neurons to an initially small network by assessing the local statistical measures collected in previous adaptation steps. The topology of the network is generated incrementally using a competitive Hebbian learning algorithm [Martinetz *et al.* 1990].

Let x_k be a Cartesian vector; the algorithm consist of updating the winning prototype vector w_p and its direct topological neighbors w_i with $i \in Q_{w_p}$ and $p \in F_{w_p}$. F_{w_p} and Q_{w_p} represent sets of winning prototype vectors and direct topological neighbors that are connected to w_p by an edge. The update rule is set as follows [Fritzke 1994]:

$$\Delta w_p = \varepsilon_b \left(x_k - w_p\right), \Delta w_i = \varepsilon_n \left(x_k - w_i\right), \forall i \in Q_{w_p} \tag{5.20}$$

where ε_b and ε_n denote the learning rate constants of the winning prototype and its topological neighbors, respectively.

Clustering in actuator space

The objective of clustering in the actuator space is to associate each winning prototype vector with their corresponding actuator space vectors while suppressing nearly identical actuator space vectors. Suppose that the winning vector w_p is associated with a number N_{w_p} of actuator space vectors denoted by $q^j_{w_p}$, $j = 1, 2, ..., N_{w_p}$, as depicted in Fig. 5.13. The incoming input vector q_k is used to create a new actuator space vector if it is not too close to one of the existing vectors or update the existing ones otherwise. A defined threshold K is used to decide whether the incoming actuator space vector is too close to the existing ones or not. Note that the actuator space vectors' association with winning prototype vectors is straightforward since the pair actuator-Cartesian vectors are available. Workspace clustering implicitly implies actuator space clustering. The only task is to remove actuator space vectors that are too close while updating the remaining vectors to avoid large Cartesian errors and to ensure the IK solutions' conservative property.

$$\left\| q^{\beta}_{w_p} - q_k \right\| < K \tag{5.21}$$

where $q^{\beta}_{w_p}$ is the winning actuator space vector.

$$\beta = \arg\min_j \left\| q^j_{w_p} - q_k \right\|, j = 1, 2, \ldots, N_{w_p} \tag{5.22}$$

The actuator space vector is updated using the following competitive rule

$$q_{w_p}^j(t+1) = q_{w_p}^j(t) + \eta h_j \left(q_k - q_{w_p}^j(t) \right) \tag{5.23}$$

where $h_j = \exp\left(-(\beta - j)/2\sigma_t^2\right)$ is a neighbourhood function [Kumar *et al.* 2010]. The latter provides continuous and smooth paths in the actuator space for a given continuous path in the workspace.

Resolution of redundancy

The clustering in the actuator space yields manifold actuator space vectors for a given task space vector. The set $q_{w_p}^j$, $j = 1, 2, \ldots, N_{w_p}$ is the potential IK solutions for a given task space vector x_k. An actuator space vector can be chosen among redundant manifolds following specific criteria such as:

- Minimum variation of the actuator space vector

$$\beta = \arg\min_j \left\| q_{w_p}^j - q_c \right\|, \tag{5.24}$$

  where q_c is the current continuum manipulator actuator space vector;

- Minimum norm of the actuator space vector

$$\beta = \arg\min_j \left\| q_{w_p}^j \right\|. \tag{5.25}$$

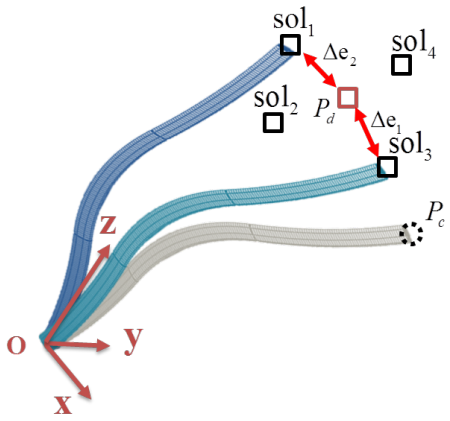

Figure 5.14: Resolution of redundancy in inverse kinematic solution space

Let us consider the example shown in Fig. 5.14, where P_c denotes the position of the robot's current configuration. The robot configuration sol_1, considered as a particular IK solution, is closer to the desired target denoted by P_d. However, given the robot's current configuration, it will require more energy or effort to reach the desired target. The solution sol_3 has slightly larger Cartesian errors than the solution sol_1, but it is closer to the robot's current configuration. It will be called a lazy-arm configuration IK solution. Other redundancy criteria can also be implemented, namely for obstacle avoidance or actuator physical limits avoidance.

Derivation of the IK solution

Typically, any IK solution chosen will result in small Cartesian errors Δe_i. This solution can be used directly as an IK solution for specific applications that do not require very high accuracy (sorting objects, packaging). At this stage, we can follow either a numerical or an analytical approach to reduce or eliminate these remaining errors. For example, the selected Ik solution can be considered as an initial solution for an iterative method, the only drawback being a possible increase in computation time. An analytical method can also be implemented to avoid iterative loops in the learning IK scheme. It is the latter which is developed. The approach consists of parameterizing the variables of the first $(n-1)$-sections of the continuum manipulator using the values of the variables of the first $(n-1)$-sections of the selected IK solution(s). This parameterization then transforms the IKs problem of multi-section continuum manipulators to a single-section IKs one, which can be easily solved. Let T_{EE} be the matrix representing the pose of the EE of the continuum manipulator:

$$T_{EE} = \begin{bmatrix} s_x & n_x & a_x & p_x \\ s_y & n_y & a_y & p_y \\ s_z & n_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.26}$$

The MTMs ${}^{N-1}_{0}\mathrm{T}_{\Phi}(\xi, q)$ being determined from the values of the parameters of the first $(n-1)$-sections of the selected IK solution, the MTM ${}^{N}_{N-1}\mathrm{T}_{\Phi}(\xi, q)$ of the last section of the continuum manipulator can be obtained as follows:

$${}^{N}_{N-1}\mathrm{T}_{\Phi}(\xi, q) = {}^{N-1}_{N-2}\mathrm{T}^{-1}_{\Phi} \cdot \ \cdots \ \cdot \ {}^{2}_{1}\mathrm{T}^{-1}_{\Phi} \cdot \ {}^{1}_{0}\mathrm{T}^{-1}_{\Phi} \cdot \mathrm{T}_{EE}(\xi, q), \tag{5.27}$$

N being the total number of the sections. The method exploits the proximity of the selected IK solutions to the desired position of the manipulator EE, and the ability of a single section continuum manipulator to reach any point located in the vicinity of its EE.

Simulation and Results

The proposed LIKM is implemented on a three-section continuum manipulator with three pneumatic muscle actuators each. All continuum manipulator sections are assumed to be physically identical and share the equivalent values for L_{i0}, R_i, $l_{i,min}$, and $l_{i,max}$ as 70 mm, 50 mm, 0 mm, and 30 mm, respectively. The training database is built from the FKM developed in section II. The actuator variables are varied from 70 mm to 100 mm with a step size of 15mm and lead to a database size of 19683 samples.

The LIKM is obtained via clustering in the workspace and the robot actuating space. The dimensions of the task space and actuator space are R^3 and R^9, respectively. GNG consists of several independent parameters; fortunately, prior tests have shown that only certain parameters have a strong impact on the learning

outcome. Therefore, to empirically select the best model, only a few parameters within a predefined range centered on a search grid were modified. As a result, to empirically select the best model, only a few parameters within a predefined range centered on a search grid were varied. Therefore, the adaptation step λ, the learning rate of best ε_n, the learning rate of neighbors ε_b, and the learning rate of output α were modified throughout the training process.

The resolution and the topology preservation of the GNG models are evaluated after each iteration using the measurement algorithm C [Kaski & Lagus 1996] for good topology preservation of the obtained models. Unlike other measures of the degree of topology preservation (topographic product, topographic function, etc.), C-measure has the advantage of combining both resolution and topology preservation measures. The training is stopped when the stop criterion (a fixed C-measure) is fulfilled, or a maximum of 200 epochs is reached. The following GNG's parameters $\varepsilon_b = 0.24$, $\varepsilon_n = 0.004$, $\alpha = 0.57$, $\alpha_{\max} = 50$, $d = 0.994$, and $\lambda = 100$ have achieved satisfactory performance. The best (lowest) value obtained from the C-measure was 0.042. The training process is conducted on MATLAB software using an Intel Core $i7-2670QMCPU$ at 2.20 GHz. The clustering in the workspace and actuator space took approximately 5 hours and 3 minutes.

An essential advantage of the proposed IK method is its efficiency in computing and maintaining multiple IK solutions. It is free of iterative loops and non-linear optimization algorithms. The following subsections present the simulation results obtained.

Inverse position kinematics

This sub-section is dedicated to IK solutions for a given EE position. Figure 5.15 shows the simulation results for tracking a target point with several IK solutions. A random selection of four configurations is depicted. This is intended to demonstrate the proposed IK method's ability to provide several IK solutions for a given EE position. The simulation results in Figures 5.16-a and 5.16-b show the EE tracking a line-shaped trajectory $[80 - 120t,\ 40,\ 220 + 60t]^T\ (mm)$ with a random selection of IK and a lazy-arm configuration solutions, respectively. Fig. 5.16-c and Fig. 5.16-d show the simulation results for the EE tracking a spiral-shaped trajectory $[90\sin(\pi t)cos(5t),\ 90\sin(\pi t)\sin(5t),\ 215 + 65t]^T\ (mm)$ with a random selection of the IK and lazy-arm configuration solutions, respectively. Tracking of the trajectory is obtained by iteratively sending a series of tracking coordinates along the IK solver path. The IK method converges towards IK solutions without numerical instabilities, mainly due to the features of learning-based approaches and the ability of mode shape functions models to solve IKs without instabilities problems [Godage *et al.* 2015]. The accuracy is approximately 10^{-3}mm.

Inverse kinematics for position with orientation

For some applications (robotic minimally invasive surgery, object inspection), the EE's positioning and orientation are essential. This section presents simulation

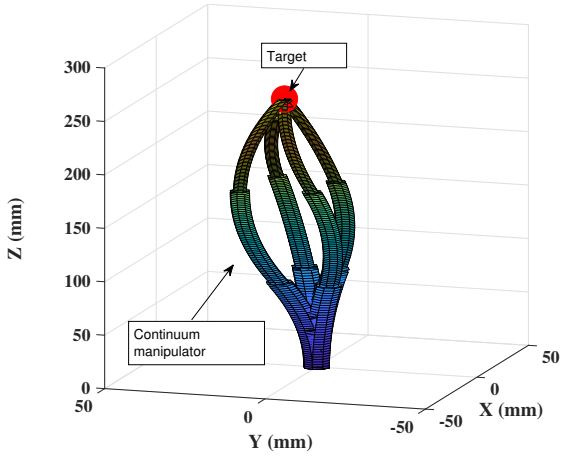

Figure 5.15: Redundancy preservation, a Cartesian point reached with different actuator space vectors.

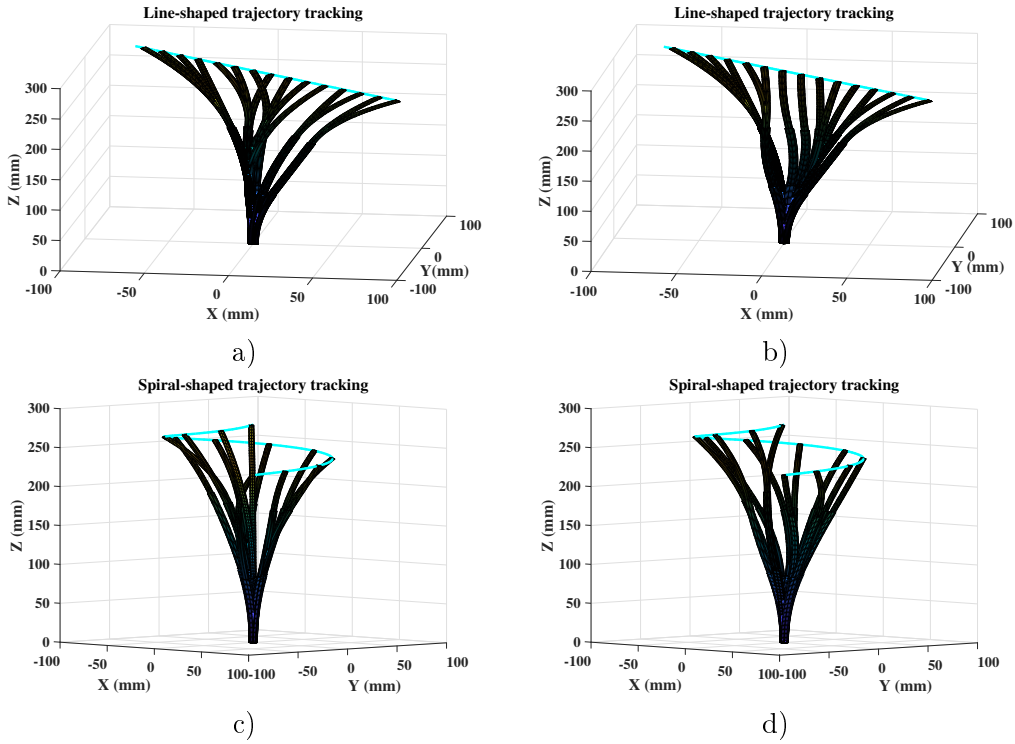

Figure 5.16: Inverse position kinematics: (a) and (b) the EE tracks a line-shaped trajectory with a random selection of IK and lazy-arm configuration solutions, respectively. (c) et (d) the EE tracks a line-shaped trajectory with a random selection of IK and lazy-arm configuration solutions, respectively.

results for tracking a trajectory with an assigned orientation. Fig. 5.17-a and Figure 5.17-b show the simulation results for tracking the above line-shaped trajectory with an orientation vector $[0, 0, \pi/6]$ with a random selection of IK and lazy-arm configuration solutions, respectively. Fig. 5.17-c and Fig. 5.17-d show the simulation results for tracking a spiral-shaped trajectory with an orientation of $[0, \pi/3, \pi]$ with a random selection of IK and lazy-arm configuration solutions, respectively. The IK method also converges towards IK solutions without numerical instabilities. The accuracy of the proposed model, in this case, is approximately 10^{-1}mm.

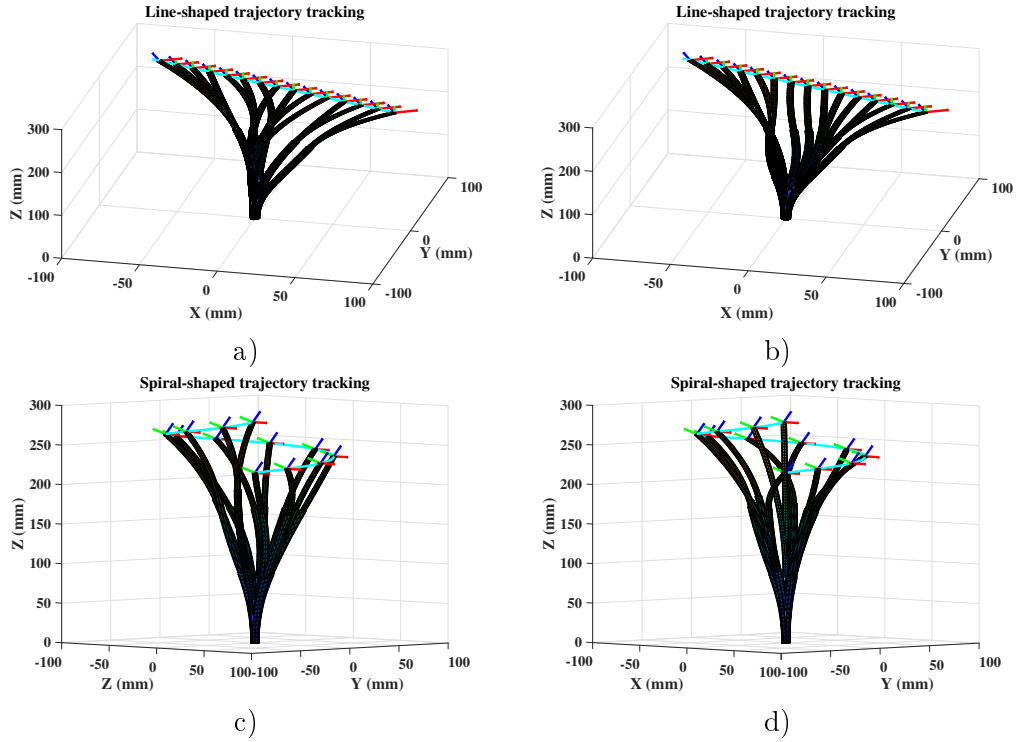

Figure 5.17: Inverse kinematics with position and orientation: (a) and (b) the EE tracks a line-shaped trajectory with an orientation of $[0, 0, \pi/6]$ using a random selection of IK and lazy-arm configuration solutions, respectively. (c) et (d) the EE tracks a spiral-shaped trajectory with an orientation of $[0, \pi/3, \pi]$ using a random selection and lazy-arm configuration solutions, respectively.

Implementation to CBHA manipulator

This section is devoted to the implementation of the proposed method to the CBHA continuum manipulator. The different steps in the implementation of the proposed learning scheme are developed in this section. The section describes the experimental platform, followed by learning the different functions constituting the learning control scheme. The proposed IK approach is validated by following the experimental setup shown in Fig. 5.18. It consists of a motion capture system and a

CBHA robot. Both systems have the same reference frame. The position tracking system comprises eight cameras and can track rigid body movement reliably up to $\pm 0.1mm$. The CBHA robot is controlled by using an adaptive SVR controller developed in [Melingui *et al.* 2017b]. The CBHA robot has poor repeatability because of its nature and actuation mode. The polyamide material that makes up the CBHA introduces a memory effect, and pneumatic actuation introduces a hysteresis effect. Therefore, it is possible to get different EE positions for the same input pressure vector [Melingui *et al.* 2015]. Thus, the sole reliable information on the CBHA EE positioning is the elongations obtained from the wire-potentiometer sensors installed along with each tube L^T. Therefore, since the pressure information does not provide accurate information on the positioning and motion of the CBHA end effector, it is not recommended to establish a direct mapping from pressure space to workspace. The SVR regression technique was used in [Melingui *et al.* 2017b] to approximate the FKM, the CBHA Kinetic model, and the controller. This choice was motivated by the SVR method's ability to yield global solutions with small size regressors. We refer the interested reader to [Melingui *et al.* 2017b] for more details in the control process of the CBHA. The proposed IK method is assessed by comparing LIKM 's generated poses, and those obtained from the motion capture system.

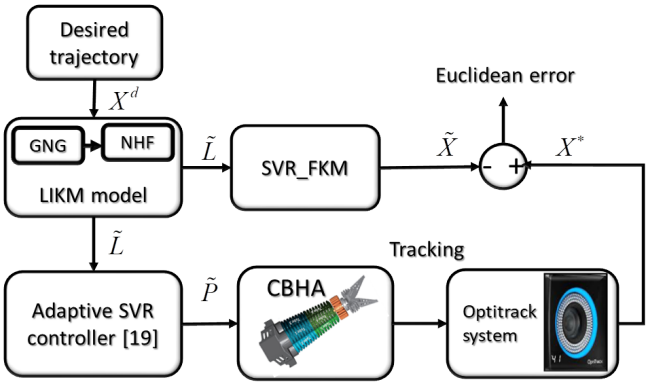

Figure 5.18: Experimental setup where X^d, $\tilde{X}$, and X^* are the desired, estimated, and measured positions, respectively. $\tilde{L}$ and $\tilde{P}$ are the estimated configuration and the pressure vectors, respectively.

Learning-based inverse kinematic model

A database of 15625 samples is collected using the motion tracking system. The exploration is done by supplying the pressure into six CBHA's tubes. The input signals are obtained by discretizing the pressure variable with a step size of 0.5 bar. An input vector $[p_1, p_2, \ldots, p_6]$, with $p_i \in [0.0, 0.5, 1.0, 1.5, 2.0]$ and $i = 1, 2, \ldots, 6$ is supplied to CBHA's tubes. The building of the learning database took approximately 6 hours.

The following GNG's parameters $\varepsilon_b = 0.2$, $\varepsilon_n = 0.006$, $\alpha = 0.5$, $\alpha_{\max} = 50$, $d = 0.995$, and $\lambda = 100$ have achieved satisfactory performance. A GNG neural

network with 400 classes was obtained after 2000 epochs. The best value obtained from the C-measure was 0.041. The clustering in workspace and actuator space took approximately 4 hours and 1 minutes.

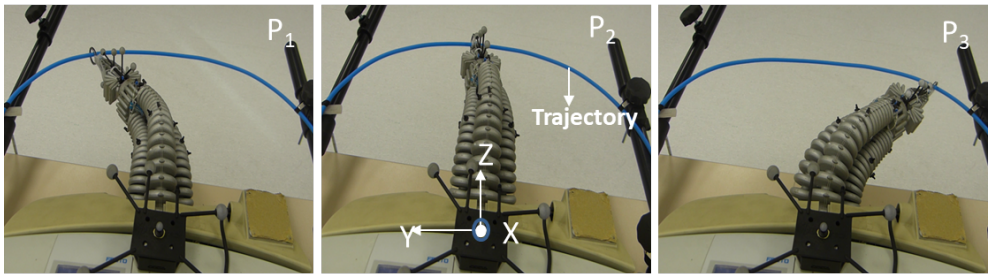

Figure 5.19: Experiment 1: trajectory tracking in unconstrained environment, P_1, P_2, and P_3 denote the robot configuration recording during the trajectory tracking.

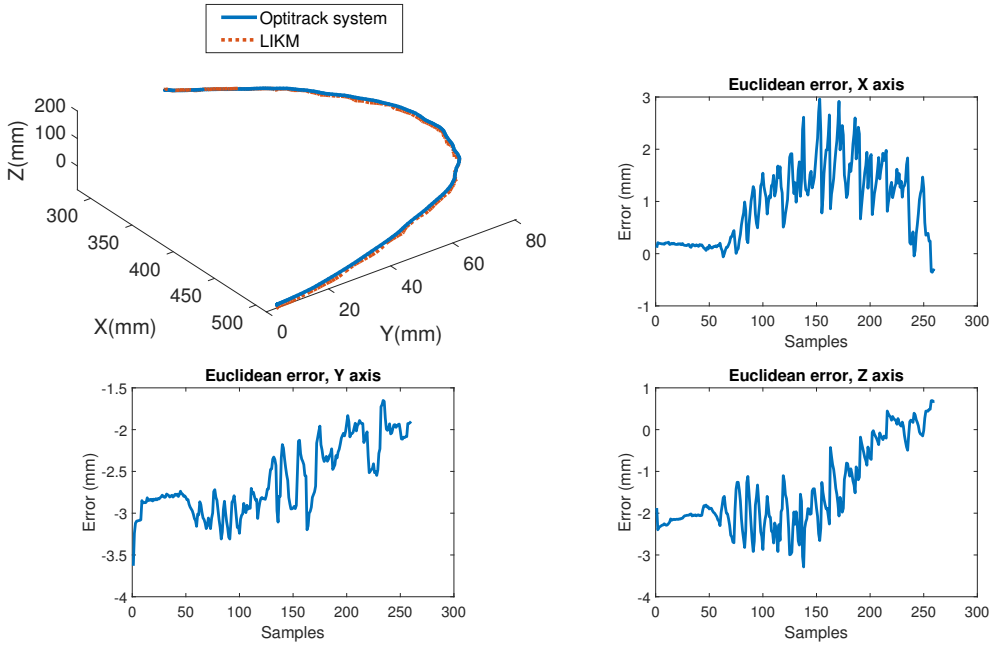

Figure 5.20: Experiment 1: The trajectory tracked by the Optitrack system is plotted in solid curve while that provided by LIKM is plotted in dot curve. Euclidean Cartesian errors in X-axis, Y-axis and Z-axis are also provided.

Experiments and results

The proposed approach is validated through two experiments. The first experiment consists of tracking a circle-shaped trajectory in an unconstrained environment. The second experiment assessed the performance in a constrained environment. The lazy-arm configuration criterion is used to select the IK solution in the first

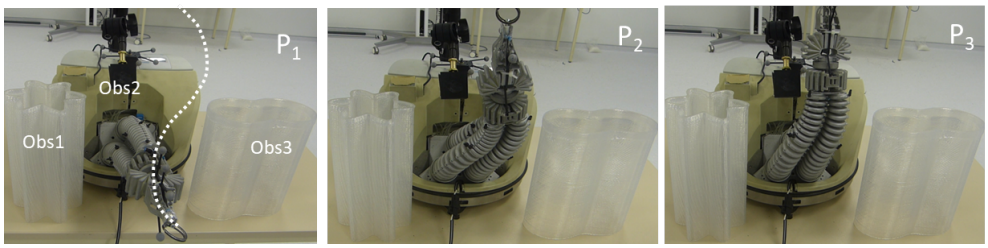

Figure 5.21: Experiment 2: trajectory tracking in environment with three obstacles; P_1, P_2, and P_3 denote the robot configuration recording during the trajectory tracking.

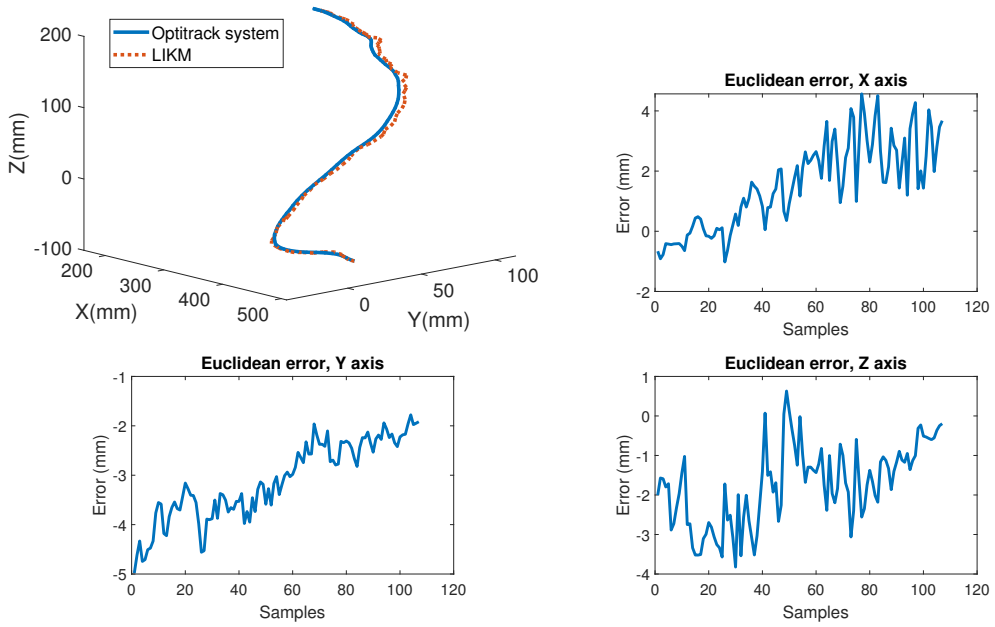

Figure 5.22: Experiment 2: The trajectory tracked by the Optitrack system is plotted in solid curve while that provided by LIKM is plotted in dot curve. Euclidean Cartesian errors in X-axis, Y-axis and Z-axis are also provided.

experiment. The IK solutions that no collide with obstacles and track the desired trajectory are selected in the second experiment. The position and volume occupied by obstacles are supposed to be known.

Fig.5.19 and Fig.5.20 show the experimental results obtained in the first experiment. Fig.5.19 shows a collection of the CBHA configurations recorded during the tracking of the desired trajectory. The measured and the achieved trajectories, and the associated Cartesian errors are shown in Fig.5.20. The plots show that the measured and the achieved trajectories are closed. The error plots show the Euclidean errors between ±3mm with the peaks at the beginning and middle of the trajectory.

The experimental results obtained in the constrained environment are shown in Fig.5.21 and Fig.5.22. A collection of the CBHA configurations is depicted in Fig.5.21. The measured trajectory and that achieved by the EE tip and also the associated errors are represented in Fig.5.22. The trajectory plots show that the two trajectories are closed. The error plots show the Cartesian errors between −5 mm and +4 mm with the peak errors at the beginning and the end of the path.

An analysis of the experimental results obtained shows that the CBHA tip can track a given desired trajectory with Cartesian errors of approximately ±5 mm. These performances remain in the range of the performance achieved by such a continuum manipulators [Melingui *et al.* 2017b, Lakhal *et al.* 2015, **?**]. However, the experimental results provided combine the performance of two subsystems, namely the LIKM and the adaptive SVR controller. The performance achieved is similar to that obtained from the adaptive SVR controller [Melingui *et al.* 2017b], and knowing that the simulation errors are negligible, we can conclude that the adaptive SVR controller caused the remaining errors. The error peaks observed at the beginning and in the middle of the first experiment's path could be justified as follows. The error peak at the beginning of the path would come from one of the adaptive controllers' limitations. The robot is operating near the limit of actuator actuation, and the adaptive controller is unable to provide the pressures required to eliminate these errors. The same phenomenon would be at the origin of the deviations observed in the second experiment's paths. The error peak found in the middle of the first experiment's path would come from gravity since this portion of the trajectory corresponds to the robot's rest position, where the pressures in the tubes are almost zero.

In light of all the above, we can affirm that the experiments' results would be close to those obtained in simulation in the presence of a perfect controller. Therefore, based on the simulations and experiment results, we can conclude that the proposed Ik method allows learning and maintaining the multiple solutions of the IKs of continuum manipulators. While most of the proposed IK algorithms provide a unique IK solution for continuum manipulators [Melingui *et al.* 2015, Rolf & Steil 2014, Giorelli *et al.* 2013], the proposed IK method allows us to learn and maintain multiple IK solutions of multi-section continuum manipulators. It involves no iterative loops and can be extended to any continuum manipulator irrespective of the number of sections. The computing time is close to that of the analytical approaches, approximately 0.029s.

5.4 Data-driving based approach for control of continuum manipulators

This part addresses the position control of continuum manipulators. Their performances in terms of speed limitation and position accuracy are often mediocre compared with rigid body based robots. Regarding continuum manipulators' control, nonadaptive schemes shown poor performance.

Braganza et al. [Braganza *et al.* 2007] implemented a low-level joint controller in a soft extensible manipulator. They utilized neural networks (NN) to compensate for the modeling uncertainties. The contributions proposed in [Anor *et al.* 2011, Yip & Camarillo 2014], have achieved good performance thanks to the type of the robot actuators. Knowing that experimented robots in [Mahl *et al.* 2014, Rolf & Steil 2014, Melingui *et al.* 2015] are pneumatically actuated, and those used in [Anor *et al.* 2011, Yip & Camarillo 2014] are controlled by wire cables. So, the actuation type can affect the global system's performance in a closed-loop; for example, the hysteresis phenomenon still acts in the pneumatic actuation. Besides, the flexible structure and the memory effect of the constitutive material of continuum manipulators cause this non-accuracy in tracking targets. These undesirable effects can strongly affect the positioning of the end effector of the manipulator. It can reach different positions inside the workspace, using the same controlled input of the pneumatic actuators [Melingui *et al.* 2015].

In [Melingui *et al.* 2014a, Melingui *et al.* 2015, Lakhal *et al.* 2015], the authors referred to the need to introduce adaptive algorithms to cope with undesirable effects. They also noted that adaptive control laws are not sufficient as the robot model changes over time. Therefore, they accorded to an update of the robot model and real-time adjustment of the controller's parameters. The authors implemented NN technique [Melingui *et al.* 2014a, Melingui *et al.* 2015, Lakhal *et al.* 2015] for CBHA continuum control. However, the NN technique suffers from local minima issues. Local minima are often poor solutions to the training problem. The probability that a local search method recovers the global minimum is rather small, considering a large number of existing local minima. Besides, the concern to obtain good performance leads mostly to large-size regressors, which are unsuitable for online adaptive control schemes. SVR is a promising alternative; its most important feature is its ability to yield accurate solutions with a short evaluation time and small size regressors, making it suitable for real-time applications. This feature is useful for real-time implementation. Besides, the fact that training an SVM amounts to solve a convex quadratic programming problem means that the solution found is global and that if the solution is not unique, then the set of global solutions is convex [Fletcher 2013]

Thus, this paper proposes an adaptive control scheme, namely an adaptive SVR controller, which gives a better performance in tracking accuracy and rapid convergence for kinematic control of a class of continuum manipulators called CBHA. The paper also attempts to support previous findings [Melingui *et al.* 2014a,

Melingui *et al.* 2015], namely the necessity to include adaptive algorithms in the continuum manipulator control systems to improve the global performance. Thus, this paper shows that using global optimization methods such as SVR does not assure the convergence in tracking. Therefore, adaptive control of the CBHA robot is essential to achieve good performance in the presence of non-linearities and the hysteresis of the pneumatic actuators.

5.4.1 Support vector regressors

Support vector machine (SVM) is a powerful tool to solve pattern classification problems [Vapnik 2013, Smola & Schölkopf 2004] so that the solution is given as a linear combination of some training samples, called the support vectors. Support vector regressors SVR are seen as extensions of SVM. They have shown good generalization ability for various function approximation and time series prediction problems [Lu *et al.* 2017, Shin *et al.* 2009, Sánchez-Fernández *et al.* 2004, Chakrabarty *et al.* 2017]. The SVM/SVR transforms the original problem into quadratic programming (QP) such that QP solvers can be used to obtain global solutions without the issues of the local minima. They are implemented to support the idea that continuum robot performance can be improved through adaptive control algorithms. First, we assume that the large numbers of local minima present in the NN objective function are the main cause of the non-adaptive NN controller's poor performance. After that, we show that even using global optimization approaches (i.e., SVR), non-adaptive control schemes are not sufficient to yield high performance. The SVR algorithms are also used to reduce the computational time as they yield accurate solutions with a short evaluation time and small size regressors. This section gives a brief introduction of multi regression support vector regressors (M-SVR). We show how global optimization is acquired by using the SVR algorithm subsequently.

Single-output SVR

The goal of the linear support vector regression problem with ε-insensitive loss function introduced by Vapnik [Vapnik 2013] is finding a function $f : R^d \to R$ such that:

$$f(x) = \phi(x)\, w + b \tag{5.28}$$

where $x \in R^d$ and $\phi(x) \in R^h$ is a non-linear transformation to a higher dimensional space with $h \geq d$. Given a set of independent and identically distributed samples, i.e.,$\{(x_i, y_i)\}_{i=1}^{n}$ where $x \in R^d$ is an incoming vector and $y \in R$ an observable output. The single output SVR solves this problem by finding the parameters $w \in R^h$ and $b \in R$ that minimize the following objective function

$$\min_{w \in R^h, b \in R} \frac{1}{2}\|w\|^2 + C \sum_{i=0}^{n} L_\varepsilon \left(y_i - f(x_i)\right), \tag{5.29}$$

where $L_\varepsilon = \max\{0, |f(x_i) - y_i| - \varepsilon\}$ is the ε-insensitive error loss function and C is a fixed constant which controls the trade off between the training error and the regularization term. The final solution (w and b) is a linear combination of the training samples in the transformed space that present the absolute errors equal or greater than ε.

Multi-output SVR

In the case of multi-output system, a multidimensional regression problem needs to be generalized. In M dimension outputs case, the regression problem can be seen as several single output case where equation (6.1) becomes:

$$f^j(x) = \phi(x) w^j + b^j,\ j = 1, 2, \ldots, M \tag{5.30}$$

with w^j and b^j the regressor parameters. These regression problems can be solved by minimizing the following objective function:

$$\min_{W,B} \frac{1}{2} \sum_{j=1}^{M} \left\|w^j\right\|^2 + C \sum_{i=1}^{n} L(\mu_i). \tag{5.31}$$

$L(.)$ is the extended Vapnik ε-insensitive loss function based on L_2-norm [Sánchez-Fernández *et al.* 2004] and defined by

$$L(\mu) = \begin{cases} 0, & \mu < \varepsilon \\ \mu^2 - 2\mu\varepsilon + \varepsilon^2, & \mu \geq \varepsilon \end{cases}. \tag{5.32}$$

From the above equations, $\mu_i = \|e_i\|$, $e_i = y_i - (\phi(x_i) W + B)$, $W = \left[w^1, \ldots w^M\right]$, and $B = \left[b^1, \ldots b^M\right]$, the Lagrangian function for the optimization problem (5.31) is expressed as

$$L(W, B) = \frac{1}{2} \sum_{j=1}^{m} \left\|w^j\right\|^2 + C \sum_{i=1}^{n} L(\mu_i) - \sum_{i=1}^{n} \alpha_i \left(\mu_i^2 - e_i^2\right), \tag{5.33}$$

where $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_n]^T$ is a vector consisting of Lagrange multipliers and $e_i = y_i - (\phi(x_i) W + B)$. The Karush-Kuhn-Tucker (KKT) conditions for optimality yield the following linear equations [Sánchez-Fernández *et al.* 2004]:

$$\begin{cases} \nabla_{w^j} L = w^j - \sum\limits_i \alpha_i \phi(x_i) \left(y_{ij} - \phi(x_i) w^j - b^j\right) = 0 \\ \nabla_{b^j} L = -\sum\limits_i \alpha_i \left(y_{ij} - \phi(x_i) w^j - b^j\right) = 0 \\ \nabla_{\alpha_i} L = \mu_i^2 - \|e_i\|^2 = 0 \\ \nabla_{\mu_i} L = 2C(\mu_i - \varepsilon) - 2\alpha_i \mu_i = 0 \end{cases}. \tag{5.34}$$

The equation (6.3) can be expressed as a linear system of equations:

$$\begin{bmatrix} \Phi^T D_\alpha \Phi + I & \Phi^T \alpha \\ \alpha^T \Phi & I^T \alpha \end{bmatrix} \begin{bmatrix} w^j \\ b^j \end{bmatrix} = \begin{bmatrix} \Phi^T D_\alpha y^j \\ \alpha^T y^j \end{bmatrix} \tag{5.35}$$

where $D_\alpha = diag\{\alpha_1, \alpha_2, \ldots, \alpha_n\}$, $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_n]^T$, $j = 1, 2, \ldots, n$, $y^j = [y_{1j}, y_{2j}, \ldots, y_{nj}]^T$, and $\Phi = [\phi(x_1), \phi(x_2), \ldots, \phi(x_n)]$. For simplicity, it is usual to work with the feature space kernel $k(x_i, x_j) = \phi(x_i)\phi(x_j)$ instead of the nonlinear transformation $\phi(.)$ [Smola & Schölkopf 2004]. By using the Representer Theorem [Smola & Schölkopf 2004], which states that the best solution for a learning problem, under fairly general condition, can be expressed as a linear combination of training samples in feature space, i.e.

$$w^j = \sum\nolimits_i \phi(x_i)\theta^j = \Phi^T\theta^j, \tag{5.36}$$

and substituting this equation (6.12) into (6.3), the linear system (6.11) becomes:

$$\begin{bmatrix} K + D_\alpha^{-1} & I \\ \alpha^T K & I^T\alpha \end{bmatrix} \begin{bmatrix} \theta^j \\ b^j \end{bmatrix} = \begin{bmatrix} y^j \\ \alpha^T y^j \end{bmatrix} \tag{5.37}$$

where $(K)_{ij} = k(x_i, x_j)$ denotes the kernel matrix. The system (5.37) is solved using iterative re-weighted least square (IRWLS) introduced in [Pérez-Cruz *et al.* 2001]. The IRWLS procedure can be summarized as follows:

1. Initialize k, Θ^k, B^k to zero, and compute μ_i^k and α_i.
2. Compute the solution to (5.37), and label them as Θ^s and B^s. Then, define an operator $P^k = \begin{bmatrix} \Theta^s - \Theta^k \\ (B^s - B^k)^T \end{bmatrix}$ to evaluate the decrement of each parameter.
3. Get the next step solution $\begin{bmatrix} \Theta^{k+1} \\ (B^{k+1})^T \end{bmatrix} = \begin{bmatrix} \Theta^k \\ (B^k)^T \end{bmatrix} + \eta_k P^k$, where η_k is a step size computing using backtracking algorithm.
4. Calculate μ_i^{k+1} and α_i, and repeat the above step $(2-4)$ until the convergence of the algorithm.

Thus, for a new input vector $x \in R^d$, the output $y \in R^m$ of the regressor is computed as follows:

$$y = \phi^T(x) \cdot \phi^T \cdot \Theta = K_x \cdot \Theta \tag{5.38}$$

where K_x is a vector that contains the kernel of the input vector and the training samples, and $\Theta = [\theta^1, \theta^2, \ldots, \theta^m]$. Note that θ^i which corresponds to $i-th$ output, is a column vector of Θ.

Usually, when we are dealing with a large database, the variable selection becomes essential. It is one of the effective ways to reduce the computational cost while improving the regressor's generalization ability. Thus, the goal is to get the smallest set of variables with generalization ability than the initial set of variables. In this work, a backward variable selection by block deletion [Nagatani *et al.* 2010] is implemented for variable selection. It uses the generalization ability estimated by cross-validation as a selection criterion.

5.4.2 CBHA controller design

Generally, accurate models in the case of rigid robots give the best performance in a closed-loop. However, this is not always the case for continuum manipulators, where, because of the characteristics of composite materials, the robot model can evolve [Melingui *et al.* 2015, Rolf & Steil 2014, Braganza *et al.* 2007]. Such conclusions were made in [Melingui *et al.* 2014a, Melingui *et al.* 2015, Lakhal *et al.* 2015], and in [Mahl *et al.* 2014, Rolf & Steil 2014]. Such model variations are usually treated by integrating adaptive laws in the robot control architectures [Chen *et al.* 2010, Melingui *et al.* 2015, Ge & Wang 2004]. However, it is not always easy to effectively integrate an update step in a control system. It is usually easier to integrate an update step in qualitative models obtained from learning-based approaches [Braganza *et al.* 2007, Rolf & Steil 2014] than quantitative models deriving from analytical equations [Mahl *et al.* 2014]. This work attempts to show that it is difficult to achieve acceptable global performance even with precise local models. Therefore, control systems that integrate adaptive laws can improve performance. Kernel-based learning methods transform the original problem into a quadratic programming (QP) problem such that QP solvers can obtain global solutions, and regression problems can be solved without the issues of the local minima. In addition, it offers the possibility to easily integrate adaptive control algorithms [Long & Nan 2015, He *et al.* 2016b]. That is why SVR regression has been preferred to other regression tools. Therefore, two control schemes are implemented in this work, non-adaptive and adaptive architectures. In the following subsections, the control schemes and the integration procedure of the adaptive algorithm are presented.

non-adaptive and adaptive control architectures

For simplicity's sake, the low-level control is addressed; the high level can be easily extended by following [Melingui *et al.* 2015]. The control schemes for non-adaptive and adaptive controls are represented in Fig. 5.23 and Fig. 5.24, respectively. The adaptive control scheme is derived from the non-adaptive control one by adding an SVR CBHA model. While no adaptive law is not considered in the non-adaptive scheme, the adaptive one integrates two adaptive laws. Knowing that the CBHA model changes over time, the first closed-loop makes its updates. The second one reduces the remaining wire-potentiometer errors by adjusting the SVR controller's parameters. The CBHA model is identified off-line using input-output data. The regressor uses the pressure applied in each tube as input and the voltages of wire-potentiometers as output. The inverse CBHA model (controller) is also trained off-line using the CBHA manipulator's input-output data.

$U_k^d \in R^6$ denotes the desired vector of voltages, and $\Delta P_k \in R^6$ is the vector of pressures. $U_k^m \in R^6$ is the measured vector of voltages, and $U_k^* \in R^6$ is the predicted vector of voltages. The proposed adaptive control scheme guarantees the convergence of voltage errors. Indeed, suppose the CBHA model reproduces the

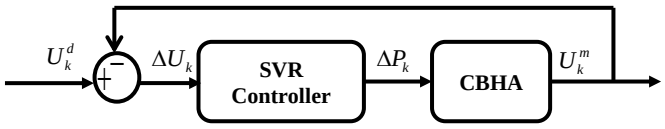

Figure 5.23: non-adaptive control scheme

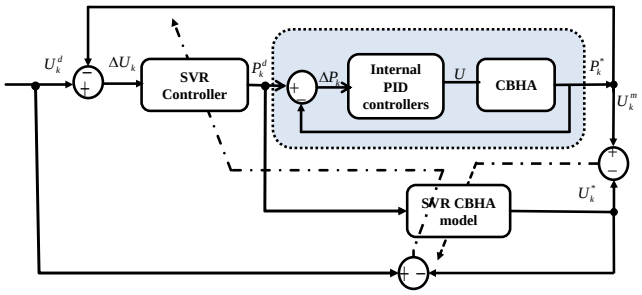

Figure 5.24: Adaptive control scheme

current behavior of the CBHA (i.e., $|U_n^m - U_n^*| \leq \varepsilon$, ε is a small positive number). In that case, the voltage errors can be back-propagated through the CBHA model using the back-propagation algorithm. The CBHA model offers a possibility to translate the voltage errors into controller output.

Integration of the adaptive algorithm

Once the regressors are obtained, the adaptation can be easily made by using (6.20). The latter may be seen as an Adaline NN with the matrix K_x as the input vector, Θ as weight matrix, and y as the output vector. The online adaptation is performed by minimizing the following quadratic cost functions successively at each time step k:

$$\begin{aligned} J_k^1 &= \tfrac{1}{2}(U_k^m - U_k^*)^T (U_k^m - U_k^*)\,, \\ J_k^2 &= \tfrac{1}{2}\left(U_k^d - U_k^*\right)^T \left(U_k^d - U_k^*\right) \end{aligned} \tag{5.39}$$

where J_k^1 and J_k^2 are the cost functions of the CBHA model and controller, respectively. U_k^d is the vector of the desired voltages. By using the gradient-descend method [Polycarpou *et al.* 1992], the weight matrix Θ of (6.20) is updated as follows:

$$\theta_{ij}^{k+1} = \theta_{ij}^k + \eta \frac{\partial J_k^*}{\partial \theta_{ij}^k} \tag{5.40}$$

where η represents the learning rate and θ_{ij} are the elements of the matrix Θ. Depending on the updated model, J_k^* is replaced by either J_k^1 or J_k^2. The partial derivative of the cost function with respect to θ_{ij} is expressed as:

$$\frac{\partial J_k^*}{\theta_{ij}^k} = -\left(U_k^d - U_k^*\right)^T \frac{\partial U_k^*}{\partial \theta_{ij}^k} = -\left(U_k^d - U_k^*\right)^T k_i^* \tag{5.41}$$

where k_i^* is the element of K_x multiplying θ_{ij}.

For input-output data acquisition, the CBHA's posture (wire-potentiometer values) is changing after varying the input pressures inside the bending tubes from the pneumatic actuators. For each desired input pressure, the information provided by wire-potentiometers and pressure sensors is recorded. These desired pressures range from $[0, 1.5]$ bars, where low-level pressure control is performed using PID (Proportional, Integrator, Derivative). Using a step size of 0.5 bar, each tube can be controlled by one of the following pressure values $(0, 0.5, 1, 1.5)$ in bars. With six controlled inputs, a database of $4^6 = 4096$ samples is collected.

For SVR training, the database is scaled to the range $[0, 0.9]$. The backward selection variable by block deletion algorithm [Nagatani & Abe 2007] is performed in the training set. The database is randomly divided, 80% are used for training, and 20% for validation. At the end of the selection phase, the training set is reduced to 1835 samples. The regressors are trained using Gaussian kernel functions:

$$k\left(x, x'\right) = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right) \tag{5.42}$$

where σ is a tunable parameter. Model parameters are optimized using a grid search according to the following $\sigma = \{0.1, 0.1, \ldots, 5\}$, $\varepsilon = \left\{10^{-6}, 11 \cdot 10^{-6} \ldots, 10^{-3}\right\}$, and $C = \{1000, 10 \ldots, 5000\}$. Different combinations of σ, ε, and C parameters are tested and that yielded a better accuracy is retained. The best combinations of the trained regressors in terms of mean square error (MSE) on the test samples yields the values reported in Table 5.5, where ε is the insensitive error, C is a fixed constant which controls the trade off between the training error and the regularization term, and σ is the standard deviation of the kernel function.

Table 5.5: Results achieved by each regressor model on the test samples

Model parameters	C	ε	σ	MSE
CBHA model	4000	51.10^{-6}	1.2	$2.352.10^{-5}$
Controller	4000	91.10^{-6}	1.2	$3.456.10^{-5}$

Stability of the adaptive SVR controller

The goal of the adaptation is to minimize the quadratic cost functions given in (5.39). Let's Consider the following positive definite Lyapunov energy function in discrete form:

$$E_k = \frac{1}{2} e_k^T e_k \tag{5.43}$$

where $e_k(k) = \left(U_k^d - U_k^*\right)$.

The variation of the Lyapunov function yields:

$$\Delta E_k = E_{k+1} - E_k = \frac{1}{2}(e_{k+1}^T e_{k+1} - e_k^T e_k) \tag{5.44}$$

where $e_{k+1} = e_k + \Delta e_k = e_k + \frac{\partial e_k}{\partial \Theta_k}\Delta\Theta_k$. The SVR controller is stable if and only if the CBHA model is stable, because the output errors are back-propagated through the latter.

The equation (5.44) leads

$$\Delta E_k = e_k^T \left[-\eta \frac{\partial U_k^*}{\partial \Theta_k} \left(\frac{\partial U_k^*}{\partial \Theta_k} \right)^T + \frac{1}{2}\eta^2 \left\{ \frac{\partial U_k^*}{\partial \Theta_k} \left(\frac{\partial U_k^*}{\partial \Theta_k} \right)^T \right\}^2 \right] e_k^T. \tag{5.45}$$

The variation ΔE_k is guaranteed to be negative if the learning rate η is chosen according to:

$$0 < \eta < \frac{2}{\max\limits_k \left(\frac{\partial U_k^*}{\partial \Theta_k} \left(\frac{\partial U_k^*}{\partial \Theta_k} \right)^T \right)}. \tag{5.46}$$

Similarly, the SVR controller is stable if the learning rate η_c is chosen according to:

$$0 < \eta_c < \frac{2}{\max\limits_k \left(\frac{\partial U_k^*}{\partial \Delta P_k} \left(\frac{\partial U_k^*}{\partial \Delta P_k} \right)^T \right) \max\limits_k \left(\frac{\partial \Delta P_k}{\partial \Theta_k^c} \left(\frac{\partial \Delta P_k}{\partial \Theta_k^c} \right)^T \right)} \tag{5.47}$$

The index c refers to the controller's parameters. Thus, if η_c is chosen as (5.47), we will have $\Delta E_k < 0$. Noticing that $E_k > 0$, according to the Lyapunov stability theory, it is shown that the training error converges to zero as $t \longrightarrow \infty$.

Experimental results

In the following experiments on the CBHA continuum manipulator, four controllers, namely non-adaptive NN controller, adaptive NN controller, non-adaptive SVR controller, and adaptive SVR controller, are implemented to achieve a common mission of path tracking. The non-adaptive controller's implementation allows whether to support earlier findings [Melingui *et al.* 2015], while that of the adaptive controllers allows assessing improvement in execution time. The NN controllers have been developed in a previous work [Melingui *et al.* 2015]. The structure is almost identical to that depicted in Fig. 5.24; neural network functions replace only SVR functions. The latter is not developed in this paper. We refer the readers to [Melingui *et al.* 2015] for further details. The implementations are conducted in Matlab using an Intel Core i7-2670QM CPU at 2.20GHz. The experimental platform is first presented, followed by the results obtained, and the section ends with a discussion.

To better assess the performance achieved by the controller, the CBHA's end-effector path is represented. Thus, the desired actuator lengths U_k^d and the outputs of the controller U_k^m are transformed into Cartesian positions by using the forward kinematic model detailed in [Melingui *et al.* 2014c]. The other constraint related to the CBHA's control is that only six controlled inputs are used to pilot the tool center from hyper-redundant sections. The desired paths consist of a series of target points that should be reached by the end-effector of the CBHA. Monitoring is caused by

varying the input pressure of each tube. The performance of the manipulator is measured in terms of positioning accuracy. It describes a manipulator continuum's ability to accurately track an arbitrary path in the robot's workspace in a reasonable time.

The path achieved by the non-adaptive NN controller developed in [Melingui *et al.* 2015] and the Cartesian errors of the tracking are presented in Fig. 5.25. The end-effector tracks the desired path with a precision of $10mm$ and an execution time of $300s$. For the non-adaptive SVR controller, the path achieved and the Cartesian errors are given in Fig. 5.26. The end-effector tracks the desired path with a precision of approximately $9mm$ and an execution time of $280s$. The paths achieved by the controllers are closed to the desired path, but the non-adaptive SVR tracking is significant in this case. For adaptive controllers, the performances achieved by NN and SVR controllers are respectively represented in Fig. 5.27 and Fig. 5.28. As expected, the end-effector tracks the desired path with a precision of $5mm$. The execution times are respectively $490s$ and $960s$ for SVR and NN controllers. We notice that the paths obtained by the controllers are even closer to the desired path.

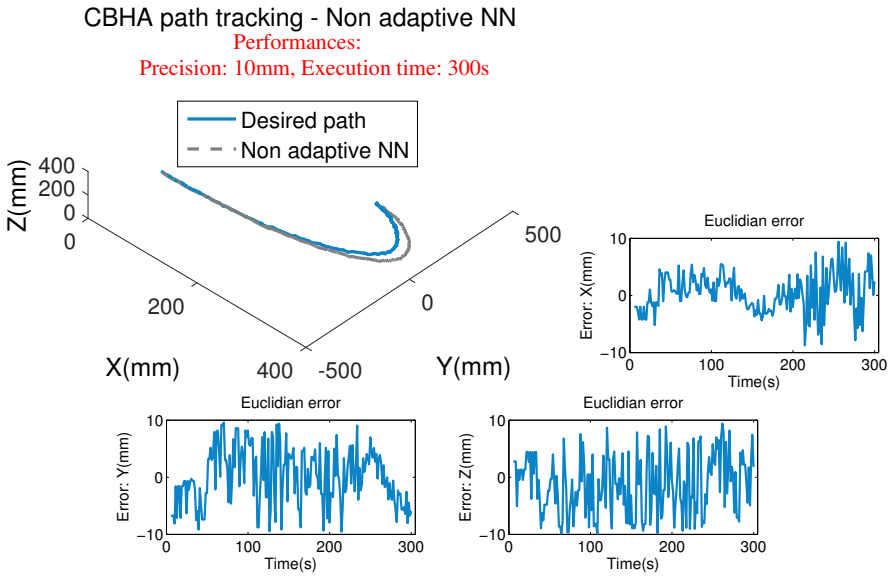

Figure 5.25: CBHA path tracking - Experiment I: Non adaptive NN controller path and associated Euclidean errors with respect to execution time in X, Y, and Z axes.

In the view of the obtained results, it is noticed that the non-adaptive SVR controllers offer relatively better performance of the tracking, comparing to the non-adaptive NN controller for the case of the CBHA. The accuracy of tracking the same path is improved by approximately 5%, and the controllers perform better, nearly continuous paths. Based on the above, we note that the more precise the model is, the better the performance. However, despite using the global optimization algorithm, we note only a slight improvement; the Euclidean errors remain large.

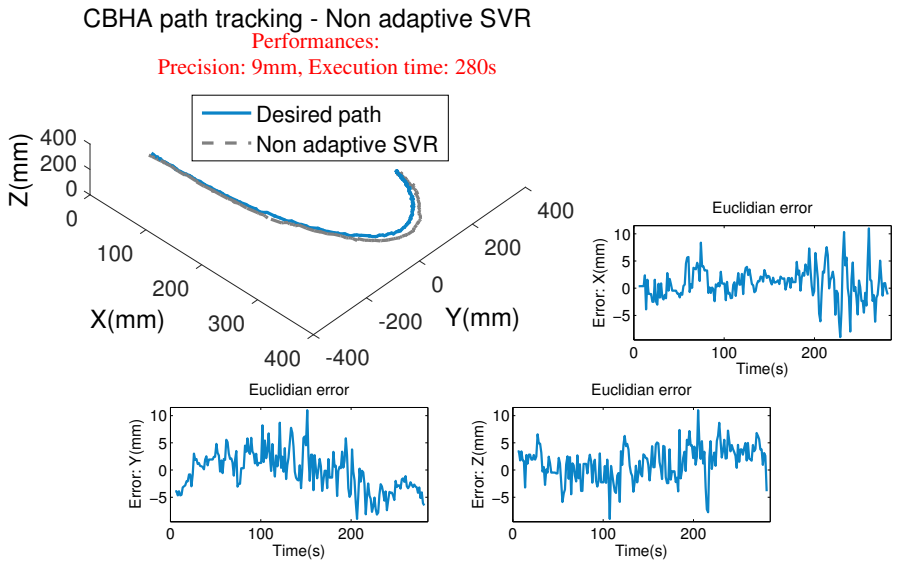

Figure 5.26: CBHA path tracking - Experiment I: Non adaptive SVR controller path and associated Euclidean errors in X, Y, and Z axes.

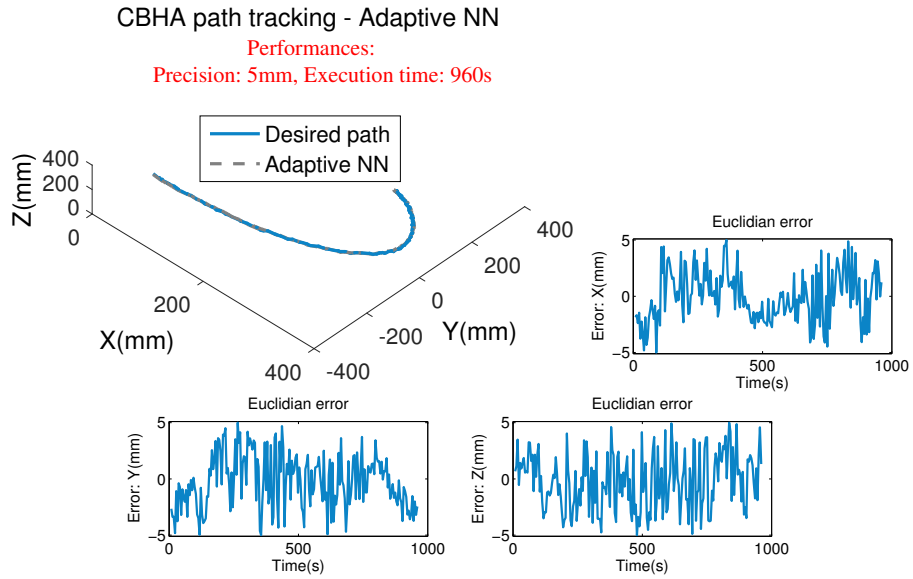

Figure 5.27: CBHA path tracking - Experiment I: Adaptive NN controller path and associated Euclidean errors with respect to execution time in X, Y, and Z axes.

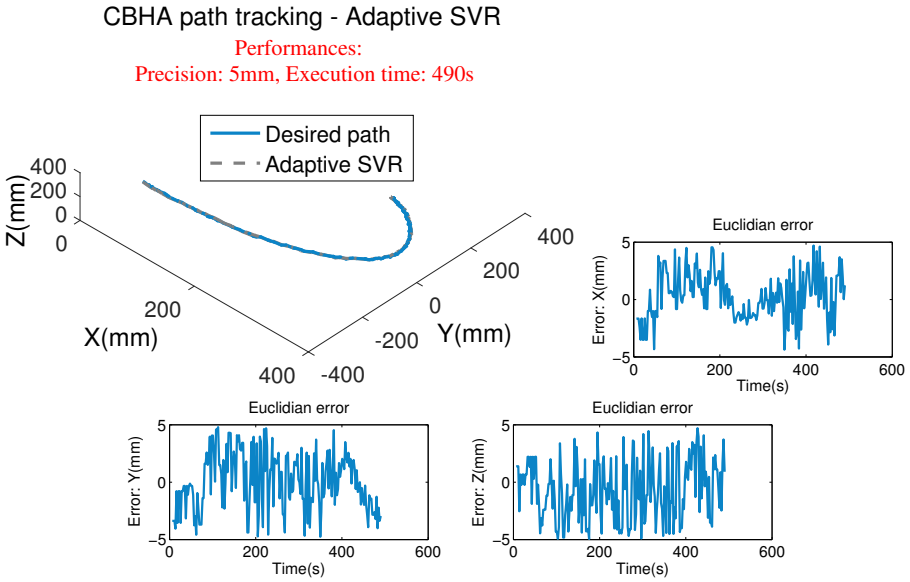

Figure 5.28: CBHA path tracking - Experiment I: Adaptive SVR controller path and associated Euclidean errors with respect to execution time in X, Y, and Z axes.

The performances achieved show that the approximated models' inaccuracy is not the sole cause of the CBHA poor performance.

Thus, we can conclude that an accurate kinematic model is not sufficient to achieve good performance. However, a significant improvement of the robot performance is noted when adaptive control algorithms are integrated. We can also conclude that the association of adaptive algorithms dramatically reduces the influence of the non-desirable effects on the CBHA (memory and hysteresis effects), yielding better performance of the CBHA. Relative to the time of execution, a significant improvement in execution time is noticed. With the same desired accuracy applied to the same trajectory, the adaptive SVR controller is approximately two times faster than the NN controller.

In summary of the CBHA robot results, the adaptive SVR control scheme can track the desired path in real-time with more accuracy and less execution time than the adaptive NN control. The performance achieved can be significantly improved for long execution times. However, it is necessary to find a compromise between the desired precision and the execution time. Compared with NN controllers, SVR controllers are approximately two times faster and easier to implement. In the case of CBHA, the adaptive SVR controller is robust enough to deal with inherent sensory noise, delays during the control, and the varying actuator ranges. The performance achieved by the non-adaptive and adaptive SVR controllers proves the necessity to insert an updating step in the CBHA control scheme. An analysis of recent contributions in the CBHA manipulator provides an additional argument to support the idea that continuum robot performance can be improved through adaptive control algorithms.

5.5 Conclusion

In this chapter, we have presented five contributions. The first one shows a comparative study of four standard learning-based techniques in the framework of the forward kinematic modeling of multi-section continuum manipulators. MLP, RBF, SVR, and CANFIS models were implemented for forward kinematic modeling of a two-section continuum manipulator. The predictions of the different topologies were compared respectively to a high precision motion capture system.

The second one presents a forward kinematic modeling approach for conical shape continuum manipulators. For real-time implementation, the proposed approach includes deep learning-based methods, particularly in solving the resulting inverse kinematic equations. The FKM obtained is computationally inexpensive and can be easily implemented in real-time.

The third one presents a distal supervised learning architecture to solve the inverse kinematics of continuum manipulators. We have demonstrated that the direct supervised learning scheme was not suitable as the learning map can yield multiple inverse functions. Therefore, the inverse kinematic of the continuum manipulator was obtained by first learning the forward kinematic model, and learn the inverse one in a distal supervised learning scheme. However, this learning scheme can only provide one inverse kinematic solution with the possibility to choose its quality.

The fourth contribution presents a learning scheme that can maintain redundant solutions of continuum manipulators h. Redundant solutions have been preserved by clustering the continuum manipulator workspace and actuation space. The GNG was used in the workspace clustering while a neighborhood function was used in the actuation space clustering. The simulations and experiments performed gave satisfactory results concerning the solving of the IKs and redundant solutions.

Finally, the last contribution focuses on controlling a class of continuum manipulators (CBHA) using the kernel-based learning method. Thus, two controllers based on support vector regression have been designed and implemented. We notice that the performance of the controllers' association with the continuum manipulator (CBHA) can be improved by including adaptive control algorithms. The paper also proposes an adaptive controller, which is approximately two times faster than the previous one [Melingui *et al.* 2015].

Chapter 6

Modelling and control of rigid and flexible joint manipulators

6.1 Inverse kinematic modelling of high dof Rigid manipulators

6.1.1 Introduction

Recently, kinematically redundant manipulators have been the subject of active research, mainly thanks to their high flexibility and versatility in executing certain complex tasks. They offer the possibility of simultaneously performing secondary tasks other than the main one, such as joint limit avoidance and obstacle avoidance. These secondary tasks make the solve inverse kinematics (IK) for this class of kinematic structures, an integral part of their real practical application. Typical applications of such systems include collaborative robots [Matthias *et al.* 2011], space robotic arms [Xu *et al.* 2011, Xu *et al.* 2018], dexterous hand [Ott *et al.* 2006], etc.

Kinematically redundant manipulators admit an infinite number of inverse kinematic solutions. Solving their IK remains a real challenge, especially in terms of the computational efficiency of complex kinematic structures to be used in real-time. The methods for solving the IK problem of redundant manipulators can be classified into three groups: analytical or closed-form methods, numerical methods, and hybrid methods, i.e., those that combine the two previous ones.

Regarding analytical methods, they express all the inverse kinematic solutions as functions in terms of the variables pose of the EE. They are computationally efficient and yield all IK solutions for a given EE pose. Peiper [Peiper 1968] proposed a procedure to get IK solutions in closed-form for manipulator robots with three consecutive joints whose axes are parallel or intersect at a single point. Later, Paul [Paul 1981] proposed a more general approach based on the manipulation of homogeneous matrices that could be applied to other types of manipulator robots. In [Chang 1987], a closed-form equation for the inverse kinematics of redundant manipulators was also derived using the Lagrangian multiplier method. The addition of an imaginary number of links [Ivlev & Gräser 1998] to the redundant open kinematic chain has been proposed to transform the redundant kinematic structure into a non-redundant one. The main idea was to complete the system of equations of the redundant kinematic structure with simple geometric equations, which consider the task to be performed or the workspace's properties. A closed-form inverse kinematic solution was derived in [Shimizu *et al.* 2008] from a parameterization of the joint angles. How joint lim-

its affect the inverse solution's feasibility was also explored to develop an analytical method for computing feasible solutions under the joint limits. Other geometric methods have also been developed [Wei *et al.* 2014, Singh & Claassens 2010]. However, the contributions mentioned above are highly configuration-dependent and can be very costly in terms of computation because of the increase in the number of DOFs.

As regards numerical methods, they work regardless of the number of degrees of freedom of the manipulator. The inverse of the Jacobian matrix [Hollerbach 1985] is used to solve IK of non-redundant manipulators, while pseudo-inverse [Klein & Huang 1983] or extended Jacobian inverse [Klein *et al.* 1995] are used for their redundant counterparts. However, Jacobian methods suffer from several shortcomings, including high computation costs and execution time, the existence of local minima, and joint singularities. Besides, calculating the Jacobian inverse becomes more and more expensive as the number of DOFs increases. The gradient projection method was introduced in [Liegeois 1977] to overcome some limitations of Jacobian-based methods. It exploits redundancy to avoid mechanical limits by projecting the homogeneous solution onto the Jacobian matrix's null space. However, this method suffers from joint oscillations. A closed-loop IK [Liegeois 1977] that uses the feedback from the EE pose and a Jacobian transpose to avoid infeasible solutions around singularities was also proposed. A very popular iterative method for solving IKs of serial chain manipulators known as Cyclic Co-ordinated Descent (CCD) has been developed. However, although it is useful, it cannot consider the manipulator's overall constraints and can take unnatural poses.

Other researchers investigated the use of both previous approaches. Several closed-form solutions for inverse kinematic are derived by parameterizing or fixing a set of joint variables. Following that idea, an interesting analytical method based on workspace analysis has been proposed in [Zaplana & Basanez 2018]. The main idea is to reduce redundant manipulators to non-redundant ones by selecting a set of joints, denoted redundant joints, and parameterizing its joint variables. The inverse kinematics of the non-redundant manipulator got is then solved analytically using either Pieper, Paul, or other geometric methods. However, this method is very depending on the number of degrees of redundancy of the manipulator and can be challenging to implement for high DOF redundant manipulators. A hybrid method of performing IK for general 2n + 1 (n is the number of joints) DOF manipulators with a spherical joint at the wrist has been proposed in [Ananthanarayanan & Ordóñez 2015]. The analytical equations were used to determine the first two and last three joint angles, and a numerical technique was used to solve the rest. However, this method can be costly in terms of computation time as the number of elbow joints increases. A segmented geometry method has been proposed in [Mu *et al.* 2018] to solve the IK of hyper-redundant spatial manipulators. According to the hyper-redundant manipulator's geometry, the method segments the manipulator into three sections, namely a shoulder, an elbow, and a wrist. Then, the IK of the manipulator is resolved separately. However, although the computational complexity is simplified, the computation time remains consid-

erable. Besides, the manipulator's segmentation into many segments can result in the loss of some IK solutions, since the inverse kinematics' resolution afterward is made segment by segment.

Machine learning was widely used for solving IK of redundant robotic systems [Raja *et al.* 2019, Kumar *et al.* 2010, Bócsi *et al.* 2011]. Support Vector Regression [Bócsi *et al.* 2011], artificial neural networks [Toshani & Farrokhi 2014, Daachi *et al.* 2012, Yahya *et al.* 2014, de Lope *et al.* 2009], and fuzzy systems [Hu & Woo 2006] have been used to solve IKs of redundant manipulators. Kohonen Self-Organized Map (KSOM) networks have been used in [Kumar *et al.* 2010] to solve the redundancy of a 7 DoFs arm for tracking trajectories with low errors. The advantages of KSOM networks to maintain continuity of the IK solutions with the possibility to select desired Ik solutions from a set of possible IK solutions have been exploited. However, not only do Cartesian errors remain; several iterative loops may be necessary to improve the position accuracy of the selected IK solution.

In this paper, we propose a learning framework that preserves the multiple IK solutions of redundant manipulators. The idea comprises dividing the redundant manipulator's workspace into clusters using clustering algorithms and eliminating some joint angle vectors that are too close to each other in each cluster using a neighborhood function (configuration space clustering). Thus, each cluster's remaining joint angle vectors are potential IK solutions for an input vector that belongs to that cluster. Finally, the full set of all IK solutions or a random IK solution can be provided. A criterion such as lazy arm movement, obstacle avoidance, and joint limit avoidance can be applied to select a particular inverse kinematic solution from the redundancy manifolds. However, for a given EE pose, each selected IK solution leads to some pose errors. In [Kumar *et al.* 2010], an iterative algorithm based on a Jacobian matrix has been proposed to reduce these errors. However, since the selected IK solution is always close to the desired EE pose, the ideal would be to maintain some manipulator's joints fixed and use the rest to eliminate these Cartesian errors. In this work, to completely avoid iterative loop in the derivation of IK solutions, an IK solution is selected depending on the redundancy resolution criterion. The configuration of some joints is maintained fix. Thus, the IK problem of redundant manipulators is reduced to non-redundant ones, and closed-form analytical methods developed for non-redundant manipulators (Pieper, Paul) are applied to obtain the IK solution. Regarding the clustering algorithm, we use the GNG algorithm. Unlike self-organizing maps [Kohonen 1990] and neural gas methods [Martinetz *et al.* 1991], GNGs do not have parameters that change over time and can continue to learn, adding neurons and connections, until a performance criterion is achieved [Fritzke 1994], thus eliminating the crucial parameter of the number of hidden neurons.

The remainder of this section is organized as follows: Sub-section 2 summarizes serial manipulators' forward kinematics. Sub-section 3 focuses on the development of the proposed learning framework for the IK problem of redundant manipulators. We apply the proposed learning framework to a series of redundant manipulators in Sub-section 4 to show the proposed approach's effectiveness. Sub-section 5 presents

the implementation of the proposed Ik method on a KUKA LBR IIWA 7 R800 robot. Finally, some concluding remarks and prospects are drawn in Sub-section 6.

6.1.2 Forward kinematics of redundant manipulators

Among the methods developed to derive the forward kinematics of serial manipulators, the Denavit–Hartenberg (D–H) convention [Hartenberg & Denavit 1955] is generally adopted, thanks to its simplicity. The forward kinematic model (FKM) is essential in establishing the inverse kinematic model (IKM) of redundant manipulators, particularly for generating the learning database in the proposed learning framework. This section describes the forward kinematics of a n-DOF serial manipulator, depicted in Figure6.1. The frame assignments follow the Denavit–Hartenberg (D–H) convention [Hartenberg & Denavit 1955], which enables the representation of every coordinate frame's location to every other.

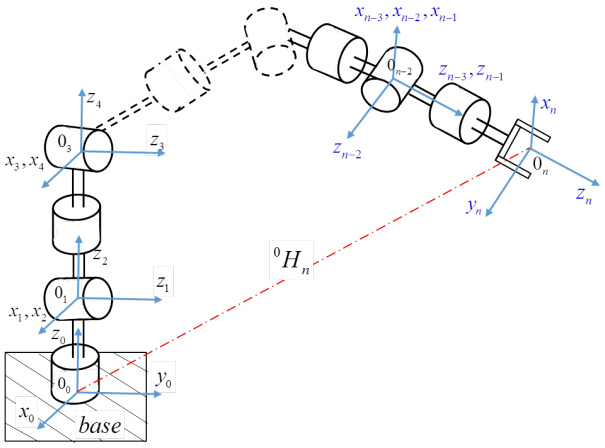

Figure 6.1: n-DOF serial manipulator; ${}^{i-1}H_i(\theta_i)$ with $i = 1, 2, ..., n$, is the homogeneous matrix that represents the coordinate frame of the link i with respect to the frame of the link $i-1$.

Let us consider the robot's kinematic chain of Figure6.1, the forward kinematic can be derived as follows:

$$\begin{aligned} {}^0H_n(\theta) &= {}^0H_1(\theta_1)\,{}^1H_2(\theta_2)\,{}^2H_3(\theta_3)\cdots{}^{n-1}H_n(\theta_n) \\ &= \prod_{i=1}^{n} {}^{i-1}H_i(\theta_i) \end{aligned} \tag{6.1}$$

where n and ${}^0H_n$ represent the total number of DOFs and the homogeneous matrix containing the position and orientation of the EE pose with respect to the reference frame, respectively. θ_i with $i = 1, 2, ..., n$ represents the angle position for each robot joint. The homogeneous matrix ${}^{i-1}H_i$ that transforms the frame attached to link $i-1$ into the frame attached to link i can be expressed as the product of four basic transformations

$${}^{i-1}H_i(\theta_i) = T_{rot_z}(\theta_i)\,T_{trans_x}(a_i)\,T_{trans_z}(d_i)\,T_{rot_x}(\alpha_i)\,, \tag{6.2}$$

where θ_i, a_i, d_i, and α_i are respectively the joint angle, link length, link offset, and link twist associated with the link and joint i. The relation (6.2) can be rewritten as follows:

$$ {}^{i-1}H_i = \begin{bmatrix} C\theta_i & -S\theta_i C\alpha_i & S\theta_i S\alpha_i & a_i C\theta_i \\ S\theta_i & C\theta_i C\alpha_i & -C\theta_i S\alpha_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.3} $$

where C and S refer to the cosine and sine functions, respectively. Equation (6.1) that represents the final transformation from the EE frame to the base frame can be rewritten as follows:

$$ {}^{0}H_n = \prod_{i=1}^{n} {}^{i-1}H_i\left(\theta_i\right) = \begin{bmatrix} R_{3\times3} & P_{3\times1} \\ 0 & 1 \end{bmatrix} \tag{6.4} $$

where $P_{3\times1}$ is the EE position vector and $R_{3\times3}$ is the rotation matrix, which can be reduced to orientations around the three main axes using the Euler or ZYX notation.

6.1.3 Proposed learning framework for inverse kinematics of redundant manipulators

Kinematically redundant manipulators admit an infinite number of inverse kinematic solutions, i.e., for a given EE pose, we can associate several joint angle vectors. Thus, two problems generally emerge, obtaining all the inverse kinematics solutions for a given EE pose and the redundancy resolution, which consists of selecting a randomly IK solution or a particular one, such as satisfy a secondary task for the manipulator (obstacle avoidance, singularity avoidance). Analytical methods are generally computationally efficient and yield all IK solutions for a given EE pose. However, for highly redundant manipulators, the IK problem is not trivial to solve, and closed-form IK solutions become challenging to derive. Numerical methods generally work regardless of the number of degrees of freedom of the manipulator. However, they generally have a high computation cost and execution time as the number of DOF of the manipulator increases. Depending on the application, redundancy resolution criteria are generally applied to derive a particular IK solution. We devote this section to developing the proposed learning framework, which combines the two previous methods to derive the IK solution of redundant manipulators. The section starts with the learning architecture presentation, followed by the clustering in the workspace and configuration space. It ends with the redundancy resolution process.

In general, any selected IK solution results in Cartesian pose errors unless the desired target pose is a learning database sample. The most accurate solution criterion is used in the redundancy resolution process. To reduce the remaining Euclidean distance between the given EE pose and the selected IK solution, an iterative algorithm based on a Jacobian matrix has been proposed in [Kumar *et al.* 2010]. However, not only this algorithm has an iterative loop; the Cartesian errors are not eliminated.

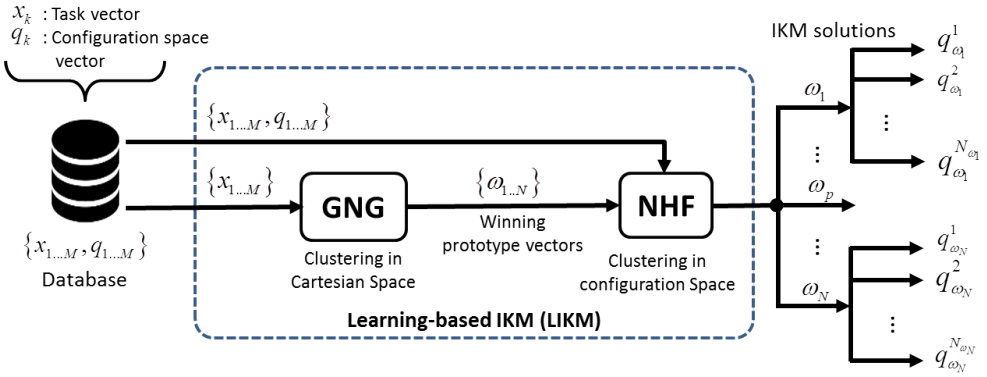

Figure 6.2: Proposed IK learning architecture: for a task pose vector that belongs to w_p cluster, a set of N_{w_p} inverse kinematic solutions are associated.

In this work, to completely eliminate the remaining Cartesian errors while without including iterative loops, some configuration joints of the redundant manipulator are fixed using the selected IK solution's corresponding values. The redundant manipulator is reduced to a non-redundant one. Then, Paul's method is applied to derive the rest of the joint variables. It is worth noting that the joints' selection criteria to be parameterized vary according to the application and the structure of the manipulator. For instance, with anthropomorphic manipulators, the first joint responsible for the entire structure's rotation may remain variable. Two of the following joints $(2, 3, ..., n-3)$ and the three joints of the spherical wrist must also be variable. The other joints can be parameterized by clustering in the workspace and in the configuration space. Depending on the application, other configuration joints may be preferred over others. We only need to make sure that we have the necessary joints for the corrections in positioning (3 joints) and orientation (3 joints) of the end-effector.

To implement Paul's method, if the last three joints of the manipulator form a spherical wrist, as is usually the case, then the problem of inverse kinematics can be decoupled. We first determine the spherical wrist position from the base to the $n-3$-th joint; then, we use the last three joints to determine the EE orientation. This is done by moving the origin of the frame 0_n to the origin of the frame 0_{n-2}. The position of the spherical wrist will be defined by:

$$P_{n-2} = P_n - d_n \cdot {}^0R_n \cdot k \tag{6.5}$$

where d_n is the length of the last link, $k = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$, and ${}^0R_n$ is the rotation matrix of the frame n with respect to the frame 0. P_{n-2} and P_n are the position of the spherical wrist and the EE , respectively. Thus, for the EE position, we have

$$P_n = P_{n-2} + d_n \cdot {}^0R_n \cdot k. \tag{6.6}$$

6.1.4 Application to anthropomorphic manipulators

The proposed learning scheme is applied to three anthropomorphic manipulators: 7-DOFS, 9-DOFs, and 11-DOFS manipulators. The objective is to demonstrate that the proposed method can be applied regardless of the number of DOFs of the manipulator. In each case, the learning database is built from the forward kinematic model of the considered manipulator. The IK solutions are obtained by performing the clustering in the workspace and configuration space. The GNG algorithm comprises several free parameters. However, preliminary tests have shown that only some parameters have a strong influence on overcoming the training. As a result, only a few parameters varied within a pre-defined range based on a search grid to empirically select the best model. The adaptation step λ, the learning rate of best ε_n, the learning rate of neighbours ε_b, and the learning rate of output α were varied during the learning process. The database is normalized in the range $[\,0.1, 0.9]$.

For good topology preservation of GNG models, the resolution and topology preservation are evaluated after each iteration using the C-measure algorithm [Kaski & Lagus 1996]. The training is stopped when the stopping criterion is met or when the maximum epochs of 200 are reached. Among the GNG networks that met the stopping criterion, the network with the best resolution and the best degree of preservation of the input space topology was selected as the best.

In the case of the 7-DOF manipulator, the following GNG's parameters $\varepsilon_b = 0.25$, $\varepsilon_n = 0.003$, $\alpha = 0.55$, $\alpha_{\max} = 50$, $d = 0.995$, and $\lambda = 100$ have achieved satisfactory performance. The learning process performed in MATLAB software using an Intel Core $i7 - 2670QMCPU$ at 2.20 GHz took approximately 16 hours. The clustering in configuration space took approximately 5 minutes. Table 6.1 summarizes the simulation data and results. For a given desired EE pose, there is a slight difference in the calculation times and accuracies. The difference in computation time is expected to come from the size of the cluster matrix, as the size of the cluster matrix increases with the number of dofs. Nevertheless, the computation time remains acceptable regardless of the number of dofs. As for the difference in precision, it is expected to come from the sampling step size, because to reduce the size of the learning bases, the sampling step on the manipulator joints has been increased. Nevertheless, the precision remains acceptable.

Table 6.1: Simulation data and results obtained

Manipulators	Database	Learning time (hours)	Computing time (seconds)	Precision (mm)
7-DOFS	884736	16	0.25	0.05
9-DOFS	1953125	35	0.29	0.08
11-DOFS	4194304	75	0.35	0.1

Anthropomorphic 7-DOF manipulator

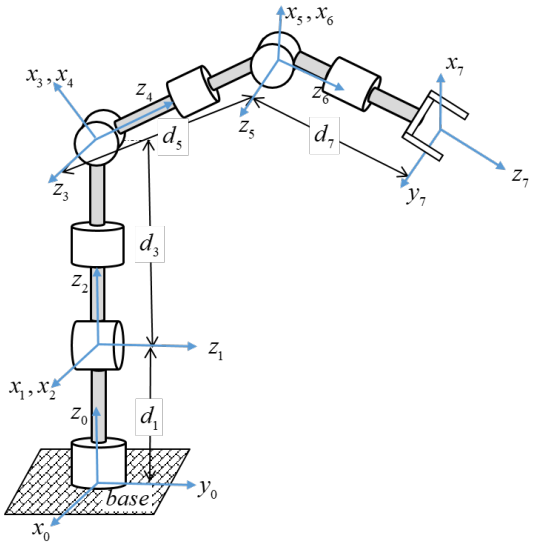

Figure 6.3: Anthropomorphic 7-DOFs manipulator

An anthropomorphic 7-DOF manipulator is shown in Figure 6.3. It is a manipulator robot whose last three joints conform to a spherical wrist. The ranges of input and output spaces are given in Table 6.2.

Table 6.2: Ranges of the input and output spaces

Range of joint angles	Range of Cartesian workspace
$-170° \leq q_1 \leq 170^0$	
$-120° \leq q_2 \leq 120^0$	
$-170° \leq q_3 \leq 170^0$	$-0.845m < x < 0.845m$
$-120° \leq q_4 \leq 120^0$	$-0.840m < y < 0.840m$
$-170° \leq q_5 \leq 170^0$	$-0.350m < z < 1.160m$
$-120° \leq q_6 \leq 120^0$	
$-170° \leq q_7 \leq 170^0$	

For the IK solutions' derivation, the joint θ_2 is parameterized via the clustering in the workspace and configuration space. The remaining joints are calculated by using Paul's method. Note that other elbow joints can be fixed, but the joint θ_2 joint parameter allows us to deduce the other joints easily.

Let U_0 be the matrix defining the position and orientation of the EE defined as

$$U_0 = \begin{bmatrix} s_x & n_x & a_x & p_x \\ s_y & n_y & a_y & p_y \\ s_z & n_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The center of the kneecap located at the origin frame 0_5 and its position denotes ${}^0P_5$ will be defined as follows:

$$P_5 = P_7 - d_7 {}^0R_7 k \tag{6.7}$$

Equation (6.7) can be rewritten in the following form:

$$P_5 = \begin{bmatrix} P_{5x} \\ P_{5y} \\ P_{5z} \end{bmatrix} = \begin{bmatrix} p_x - d_7 a_x \\ p_y - d_7 a_y \\ p_z - d_7 a_z \end{bmatrix}. \tag{6.8}$$

In (6.8), the origin of the frame 0_7 has been translated to origin 0_5, and the position and orientation matrix at the point 0_5 can be expressed as follows:

$$U_w = \begin{bmatrix} s_x^{'} & n_x^{'} & a_x^{'} & p_{5x} \\ s_y^{'} & n_y^{'} & a_y^{'} & p_{5y} \\ s_z^{'} & n_z^{'} & a_z^{'} & p_{5z} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

From the above, the following equalities hold,

$$U_w \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T = {}^0P_5 = {}^0H_5 \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T. \tag{6.9}$$

By multiplying both members of (6.9) by the inverse of ${}^0H_1$, we get

$${}^1H_0 \times {}^0H_5 \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T = {}^1H_0 \times U_w \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T. \tag{6.10}$$

From this equality, we can draw the following system of equations

$$\begin{cases} c_1 p_{5x} + s_1 p_{5y} = -\left(c_2 c_3 s_4 + s_2 c_4\right) d_5 - s_2 d_3 \\ p_{5z} - d_1 = -\left(s_2 c_3 s_4 - c_2 c_4\right) d_5 + c_2 d_3 \\ s_1 p_{5x} - c_1 p_{5y} = s_3 s_4 d_5 \end{cases} \tag{6.11}$$

θ_2 being known, (6.11) can be re-written as follows:

$$\begin{cases} c_1 p_{5x} + s_1 p_{5y} = -s_2\left(c_4 d_5 + d_3\right) - c_2 c_3 s_4 d_5 \\ s_1 p_{5x} - c_1 p_{5y} = s_3 s_4 d_5 \\ p_{5z} - d_1 = -s_2 c_3 s_4 d_5 + c_2 c_4 d_5 + c_2 d_3 \end{cases} \tag{6.12}$$

By squaring each term of this system, we obtain:

$$\begin{cases} \left(c_1 p_{5x}\right)^2 + \left(s_1 p_{5y}\right)^2 + 2 s_1 c_1 p_{5x} p_{5y} = \left(s_2\left(c_4 d_5 + d_3\right)\right)^2 \\ + \left(c_2 c_3 s_4 d_5\right)^2 + 2\left(s_2\left(c_4 d_5 + d_3\right)\right)\left(c_2 c_3 s_4 d_5\right) \\ \left(s_1 p_{5x}\right)^2 + \left(c_1 p_{5y}\right)^2 - 2 s_1 c_1 p_{5x} p_{5y} = \left(s_3 s_4 d_5\right)^2 \\ \left(p_{5z} - d_1\right)^2 = \left(-s_2 c_3 s_4 d_5 + c_2\left(c_4 d_5 + d_3\right)\right)^2 = \left(s_2 c_3 s_4 d_5\right)^2 \\ + \left(c_2\left(c_4 d_5 + d_3\right)\right)^2 - 2 s_2 c_3 s_4 d_5 c_2\left(c_4 d_5 + d_3\right) \end{cases} \tag{6.13}$$

By adding each side of the equation system (6.13), we get:

$$p_{5x}^2 + p_{5y}^2 + \left(p_{5z} - d_1\right)^2 - d_3^2 - d_5^2 = 2 c_4 d_3 d_5 \tag{6.14}$$

with $c_4 = \frac{p_{5x}^2+p_{5y}^2+(p_{5z}-d_1)^2-d_3^2-d_5^2}{2d_3d_5}$.

In this way, $s_4 = \sqrt{1-c_4^2}$ and θ_4 is derived by using the trigonometric relations given in Appendix A.2

$$\theta_4 = atan2\left(s_4, c_4\right). \tag{6.15}$$

Knowing θ_4, we can easily calculate θ_3 since θ_2 is known. The second equation of the system (6.12) gives:

$d_1 + c_2c_4d_5 + c_2d_3 - p_{5z} = s_2c_3s_4d_5$

in that way, $c_3 = \frac{d_1+c_2c_4d_5+c_2d_3-p_{5z}}{s_2s_4d_5}$

and $s_3 = \sqrt{1-c_3^2}$ and θ_3 is derived

$$\theta_3 = a\tan 2\left(s_3, c_3\right). \tag{6.16}$$

Thus, we can calculate θ_1, the third relationship of the system (6.12) yields:

$$s_1p_{5x} - c_1p_{5y} = s_3s_4d_5 \tag{6.17}$$

Equation (6.17) is in the form

$$A_1s_i + A_2c_i = A_3 \tag{6.18}$$

where $A_1 = p_{5x}$, $A_2 = -p_{5y}$, and $A_3 = s_3s_4d_5$. Then, the following expressions can be derived

$s_1 = -\frac{A_1A_3 \pm A_2\sqrt{A_1^2+A_2^2-A_3^2}}{A_1^2+A_2^2}$ and

$c_1 = -\frac{A_2A_3 \pm A_1\sqrt{A_1^2+A_2^2-A_3^2}}{A_1^2+A_2^2}$,

Finally, the joint variable θ_1 is obtained as follows:

$$\theta_1 = a\tan 2\left(s_1, c_1\right). \tag{6.19}$$

Concerning the orientation angles, knowing that ${}^0R_7$ is the rotation matrix of U_0, we have

$${}^0R_7 = \begin{bmatrix} s_x & n_x & a_x \\ s_y & n_y & a_y \\ s_z & n_z & a_z \end{bmatrix}$$

By pre-multiplying both sides of this equation by ${}^4R_0$, we get:

$${}^4R_0\left(\theta_1,\theta_2,\theta_3,\theta_4\right)\cdot {}^0R_7\left(\theta_1,\theta_2,\theta_3,\theta_4,\theta_5,\theta_6,\theta_7\right) =$$

$${}^4R_0\left(\theta_1,\theta_2,\theta_3,\theta_4\right)\cdot \begin{bmatrix} s_x & n_x & a_x \\ s_y & n_y & a_y \\ s_z & n_z & a_z \end{bmatrix} = {}^4R_7\left(\theta_5,\theta_6,\theta_7\right).$$

We can write these equations as:

$$\begin{bmatrix} U & V & W \end{bmatrix} = {}^4R_7\left(\theta_5,\theta_6,\theta_7\right) \tag{6.20}$$

where $U = \begin{bmatrix} U_x & U_y & U_z \end{bmatrix}^T$, $V = \begin{bmatrix} V_x & V_y & V_z \end{bmatrix}^T$, and $W = \begin{bmatrix} W_x & W_y & W_z \end{bmatrix}^T$ with $U_x = (c_4c_3c_2c_1 - c_4s_3s_1 - s_4s_2c_1)\,s_x + (c_4c_3c_2s_1 + c_4s_3c_1 - s_4s_2s_1)\,s_y + (c_4c_3s_2 + s_4c_2)\,s_z$, $U_y = -(s_3c_2c_1 + c_3s_1)\,s_x - (s_3c_2s_1 - c_3c_1)\,s_y - (s_3s_2)\,s_z$, $U_z = (-s_4c_3c_2c_1 + s_4s_3s_1 - c_4s_2c_1)\,s_x - (s_4c_3c_2s_1 + s_4s_3c_1 + c_4s_2s_1)\,s_y$
$-(s_4c_3s_2 - c_4c_2)\,s_z$.

$V_x = (c_4c_3c_2c_1 - c_4s_3s_1 - s_4s_2c_1)\,n_x + (c_4c_3c_2s_1 + c_4s_3c_1 - s_4s_2s_1)\,n_y + (c_4c_3s_2 + s_4c_2)\,n_z$, $V_y = -(s_3c_2c_1 + c_3s_1)\,n_x - (s_3c_2s_1 - c_3c_1)\,n_y - (s_3s_2)\,n_z$, $V_z = (-s_4c_3c_2c_1 + s_4s_3s_1 - c_4s_2c_1)\,n_x - (s_4c_3c_2s_1 + s_4s_3c_1 + c_4s_2s_1)\,n_y$
$-(s_4c_3s_2 - c_4c_2)\,n_z$.

$W_x = (c_4c_3c_2c_1 - c_4s_3s_1 - s_4s_2c_1)\,a_x + (c_4c_3c_2s_1 + c_4s_3c_1 - s_4s_2s_1)\,a_y + (c_4c_3s_2 + s_4c_2)\,a_z$, $W_y = -(s_3c_2c_1 + c_3s_1)\,a_x - (s_3c_2s_1 - c_3c_1)\,a_y - (s_3s_2)\,a_z$, $W_z = (-s_4c_3c_2c_1 + s_4s_3s_1 - c_4s_2c_1)\,a_x - (s_4c_3c_2s_1 + s_4s_3c_1 + c_4s_2s_1)\,a_y$
$-(s_4c_3s_2 - c_4c_2)\,a_z$.

By multiplying each member of (6.20) by ${}^5R_4$, we will get the following equality:

$$\begin{bmatrix} c_5U_x + s_5U_y & c_5V_x + s_5V_y & c_5W_x + s_5W_y \\ U_z & V_z & W_z \\ s_5U_x - c_5U_y & s_5V_x - c_5V_y & s_5W_x - c_5W_y \end{bmatrix} = \begin{bmatrix} c_6c_7 & -c_6s_7 & -s_6 \\ s_6c_7 & -s_6s_7 & c_6 \\ -s_7 & c_7 & 0 \end{bmatrix}.$$

It follows, after identifying both sides of this equality, that

$s_5W_x - c_5W_y = 0$, hence,

$$\theta_5 = a\tan 2\,(-W_y, -W_x) \tag{6.21}$$

we also have the following equality

$$\begin{cases} c_6 = W_z \\ s_6 = -c_5W_x - s_5W_y \end{cases}, \text{ hence}$$

$$\theta_6 = a\tan 2\,(s_6, c_6) \tag{6.22}$$

We also get $\begin{cases} c_7 = s_5V_x - c_5V_y \\ s_7 = -s_5U_x + c_5U_y \end{cases}$, hence

$$\theta_7 = a\tan 2\,(s_7, c_7) \tag{6.23}$$

To illustrate the feasibility of the obtained joint variable expressions, a line path, and a circular path are tracked with different IK solutions. The desired straight-line path is expressed as

$$\begin{cases} -550mm \le x \le 550mm \\ y = ax + b \\ z = ax + c \end{cases} \tag{6.24}$$

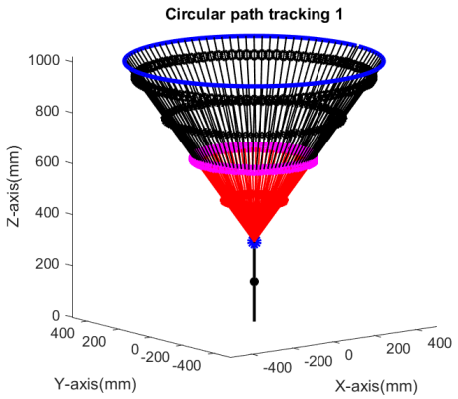

Figure 6.4: A circular path tracking by an anthropomorphic 7-DOF manipulator of with simple configuration vectors

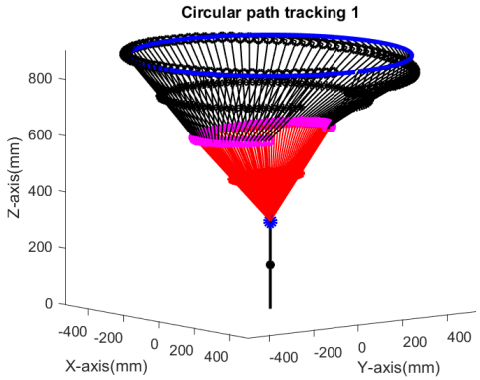

Figure 6.5: A circular path tracking by an anthropomorphic 7-DOF manipulator with complex configuration vectors.

with $a = 0mm$, $b = 350mm$, and $c = 700mm$. A circle centred at $[0\ 0\ 900]^T\ mm$ and parallel to $X - Y$ plane with a radius $R = 450mm$ is used as the desired path for the second scenario.

Figure 6.4 shows the tracking of a circular path with simple configuration vectors, while Figure 6.5 shows the tracking with complex configuration vectors. We can also implement other scenarios, such as obstacle avoidance and joint limit avoidance.

6.1.5 Experiments and Results

This section focuses on the experimental validation of the proposed IK approach. We perform the experiments on a KUKA LBR IIWA 7 R800 robot, shown in Figure 6.6. It is a lightweight collaborative robot weighing around 24 kg with a 7 kg payload and a repeatability of $\pm 0.1mm$. Made of aluminum, it has 7 degrees of freedom, allowing it to be both flexible and accurate. Simultaneously, its powerful sensor technology contributes to its safety and ability to work with humans. The

KUKA LBR IIWA 7 R800 is programmed using the KUKA Sunrise-Workbench. Sunrise Operating System can be interfaced with an external computer using either the Robot Operating System (ROS) or the Fast Research Interface (FRI). The KUKA Sunrise OS controller is programmed into Java so that algorithms can be internally implemented and external sensors interfaced. The section starts with the experimental setup and ends with results obtained.

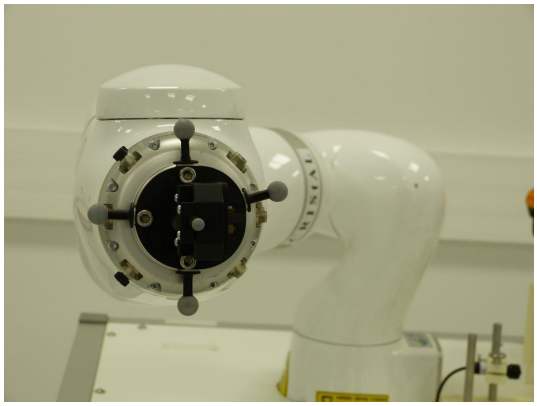

Figure 6.6: KUKA LBR IIWA 7 R800 robot

Experimental setup

The experimental setup used to validate the proposed IK method is depicted in Figure 6.7. It comprises an Optitrack motion capture system and a KUKA LBR IIWA 7 R800 robot. The Optitrack motion capture system composed of 8 cameras can track a rigid body motion with a precision of $\pm 0.1mm$. The KUKA LBR IIWA is controlled from an external PC via the KUKA Sunrise Cabinet robot controller. The proposed IK method's performance is conducted by comparing the forward kinematic model's poses with those obtained from the motion capture system. We attach the rigid body frames in the base and the manipulator's EE by using reflective markers.

Experimental Results

The proposed IK method is validated experimentally through a Lemniscate curve trajectory. The objective is to follow in several ways, a trajectory using the proposed IK method. Since the proposed IK method preserves the multiple IK solutions, the trajectory is tracked using two different configuration vectors. The first solution considered as lazy joint angles uses as the IK solution, the solution whose configuration vector is closer to the current configuration vector of the robot. The second solution uses slightly more complex configuration vectors. The choice of the IK solution to reach the first point of the trajectory is random, and the lazy joint angles are applied to the rest of the trajectory points to make the motion smoother. The estimated trajectory is compared each time with the one obtained from the

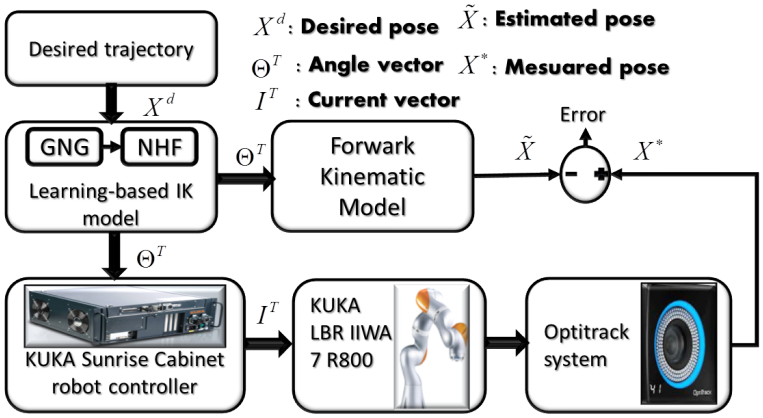

Figure 6.7: Experimental setup for validation of the proposed IK model

Optitrack motion capture system. The different joint angles of the robot are also provided. The equations of the trajectory are given in (6.25).

$$\begin{cases} x = 220 + 100\sin\left(\frac{1}{15}\pi t\right) \\ y = 830 \\ z = 450 + 100\sin\left(\frac{2}{15}\pi t\right) \end{cases}, \ (mm) \tag{6.25}$$

The trajectories estimated and measured, and the associated Cartesian Euclidean errors are shown in Figure 6.8 and Figure 6.10. The associated joint angles are provided in Figure 6.9 and Figure 6.11.

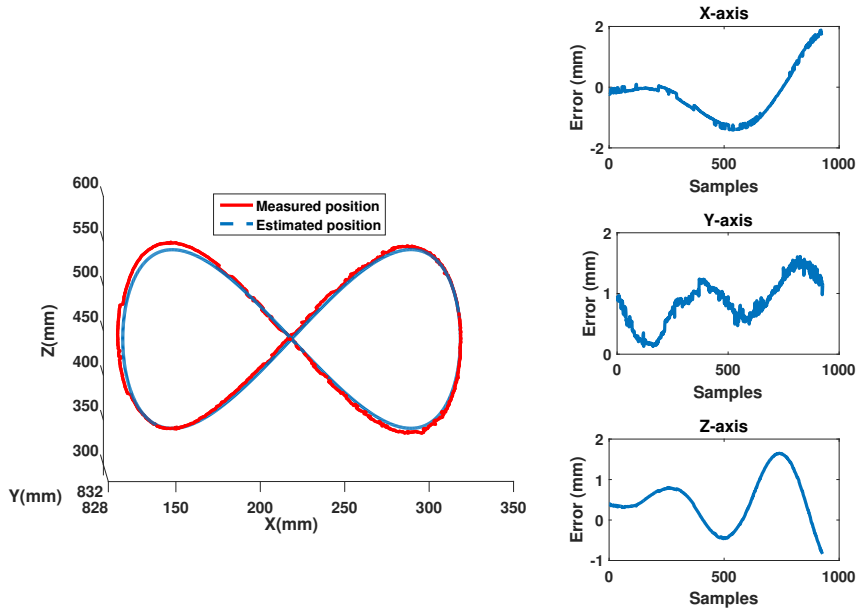

Figure 6.8: Lamniscate curve tracking with lazy joint angles: Estimated and measured trajectories, and associated Cartesian Euclidean errors.

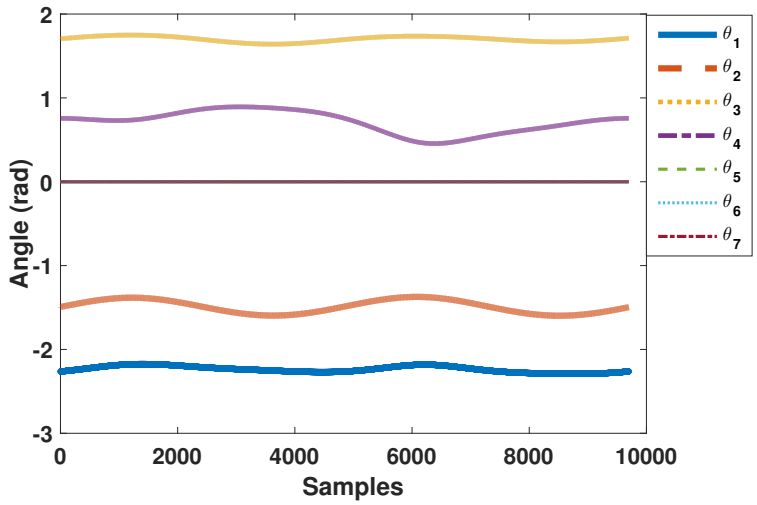

Figure 6.9: Lamniscate curve tracking with lazy joint angles: Associated joint angles

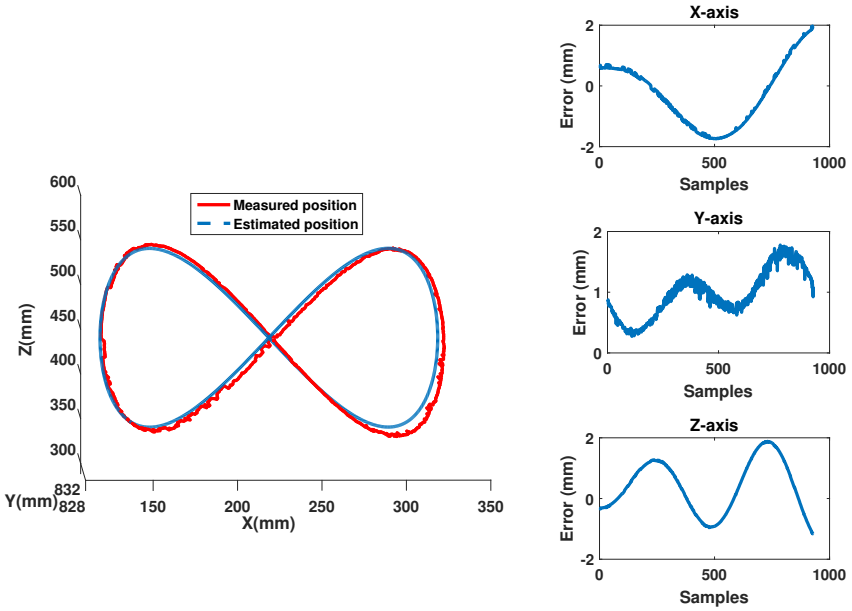

Figure 6.10: Lamniscate curve tracking with complex configuration joint angles: Estimated and measured trajectories, and associated Cartesian Euclidean errors.

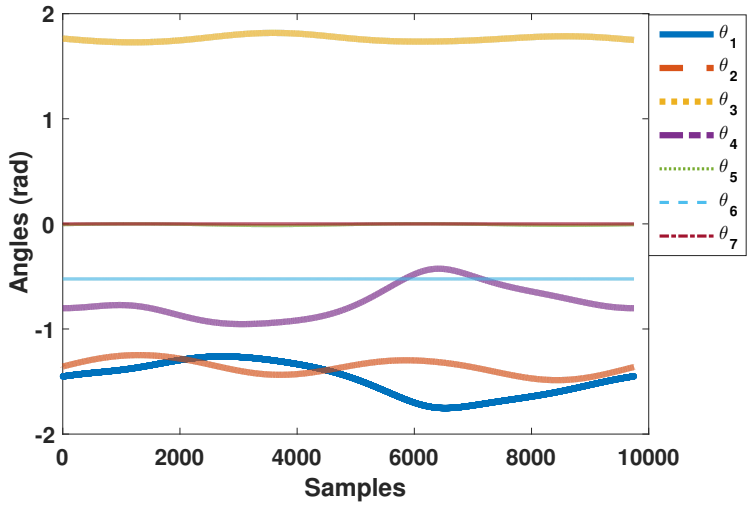

Figure 6.11: Lamniscate curve tracking with complex configuration joint angles: Associated joint angles

As we can see, the estimated and measured trajectories are very close. We can conclude that the performances obtained are acceptable. Looking at the error curves, we note a maximum Euclidean error of approximately 2 mm, which is still acceptable considering the manipulator robot's length and knowing that these errors are cumulative errors of the forward kinematic model and the inverse kinematic model. Although the Optitrack motion tracking system is efficient and accurate, it remains susceptible to the quality of the reflective markers used and the calibration process's quality. Therefore, the remaining Euclidean errors are expected to come from the tracking system.

Regarding the plots of the joint angles, we can observe that all the curves are continuous and regular, which guarantees the conservative property of the inverse kinematic solutions obtained. We also note that the more complex the joint angle configurations are, the greater the errors obtained. Therefore, the KUKA Sunrise Cabinet robot controller could also be another source of errors.

In summary, while the error in simulation remains negligible (0.1 mm), maximum Euclidean errors of the order of 2 mm are observed in experiments. We can therefore conclude, regarding the performance obtained in simulation and experimentation, that the proposed method is not only accurate but also preserves the multiple of IK solutions.

6.2 Contribution to the control of flexible-joint robots

6.2.1 Introduction

In recent past decades, new fields of applications for robot manipulators have emerged, where robots and humans have to share common spaces, especially in

medicine, rescue, and home automation. In this context, the user's safety, the lightness of the structure, and the reduction in size and energy consumption appear as new constraints. In response to these constraints, robot designers have opted for robots with flexible joints. The latter are often actuated using pneumatic actuators or DC motors. However, DC motors are the most used thanks to their low cost, small size, and high controllability. Flexible joint robots also contain elastic elements at the joint level to enable safe physical contact between the robot and human (belt pulley systems, tendon materials). The joint flexibility involves highly non-linearities in the robot dynamic, such as the high coupling between robot links and actuators, under-actuation, dead zone, backlash, and model uncertainties, which introduce serious technical issues to controller design. Therefore, new control challenges emerge:

- Consideration of joint stiffness and full actuator dynamics in the control design.
- Deal with uncertain actuator's and link's dynamic parameters to guaranty the accuracy.
- Deal with non-linearities lead to flexibility, such as hysteresis, death zone, backlash
- Deal with joint constraints to avoid collisions.
- Reduce computational cost for energy-saving, since, for the case of home automation, the manipulator used embedded energy source.
- Ensure fats stabilization to avoid unwanted oscillations.
- Deal with oscillations arising when operating in stochastic environments.

Since the 1980s, many attempts have been made to counter one ore more control mentioned above challenges, and now several methods have been developed. In [Huang & Chen 2004], an adaptive sliding mode control is proposed to overcome the mismatched uncertainties problem. Authors in [Ott *et al.* 2008] proposed a passivity based impedance controller. The approach can control the stiffness and the damping of the FJR without inertia shaping. Kim et al. [Kim & Chung 2015] designed a position controller using a disturbance observer PD control approach. The strategy can eliminate motor-side uncertainties. A position control approach with online gravity compensation is designed in [Sun *et al.* 2017]. In [Kim 2017] sliding mode control approach for FJR is revisited, and the tracking performance is improved. The approach is capable of suppressing residual link vibrations. The energy shaping control method with gravity compensation is revisited in [Yin *et al.* 2018] to perform a position controller. Soltanpour et al. [Soltanpour *et al.* 2019] developed a position controller based on the voltage sliding mode approach. The controller can cope with structured and unstructured

uncertainties without using the FJR model. Another voltage based controller is presented in [Khorashadizadeh & Sadeghijaleh 2018], where motor uncertainties are coped with adaptive fuzzy structure. In [Jin *et al.* 2016, Kim *et al.* 2019], the authors proposed a model-free control approach for friction and disturbance torque compensations.

The backstepping control approach appears as the appropriate method for highly non-linear systems like flexible joint robot manipulator systems [Hwang & Kim 2006, Huang & Chen 2004, Oh & Lee 1999, SU & Stepanenko 1997]. It is a control strategy that consists of stepping back toward the control input of a given system. The design procedure starts at the known-stable subsystem and back-out new controllers that progressively stabilized each outer subsystem. The process is terminated when the final external control is reached. The backstepping control approach is a recursive design technique that aims to compute a control law that stabilizes all the system's subsystem. Thus, the Backstepping control approach can handle a high order system (order $n > 3$), since this kind of system is composed of several subsystems (n subsystems for n-order system). With the backstepping control approach, we have the assurance that each closed-loop signal is stabilized (bounded after a given time). It also offers the possibility to conserve useful non-linearities in contrast to computed torque, sliding mode, or PID control approaches, where certain non-linearities are cancelled to design the controller. Besides, It is also simple to implement robust adaptive controllers with a backstepping control approach. Another advantage is that the Lyapunov function can be easily found since it is constructed recursively, starting with the last subsystems (the one containing the output variable). It is a powerful tool for the design of controllers for non-linear systems in strict-feedback form. However, the standard backstepping algorithm has a known drawback named "explosion of complexity with the order of the system," which increases the computational cost. This is because the control design uses the derivative of the virtual controls of the subsystems. This drawback has limited the use of the backstepping approach for real-time applications or high-order systems control. In 2009, a new backstepping algorithm named *command filtered backstepping* (CFB) was proposed [Farrell *et al.* 2009], where the recursive derivation is avoided by the use of command filters(CF). Besides, filtering errors arising are cancelled using an error compensation mechanism.

We have made several contributions to address this theoretical issue while the previously cited control challenges are solved. The first contribution was to propose an appropriate command filtered backstepping algorithm for flexible joint robot manipulators. In this contribution, support vector regression (SVR) was used to deal with dynamic uncertainties, and a tangent-type robust term was used to cancel tracking and approximation errors. The second contribution was to propose a constrained command filtered backstepping algorithm to deal with joint space constraints. In this contribution, dynamic uncertainties are coped with support vector regression. The third contribution was to propose a finite-time command filtered algorithm to ensure fast stabilization with a settling time managed by the designer.

In this contribution, dynamic uncertainties are coped with using an adaptive fuzzy logic approach.

The remainder of this chapter is organized as follows. Flexible joint robots and their dynamic modeling are first presented. After that, the Standard command filtered algorithm for a non-linear system is presented, and the chapter ends with the presentation of our three contributions.

6.2.2 Flexible joint robots

General presentation

When dealing with robot Kinematic, dynamics, or control design, a common assumption is used to reduce the complexity of the problem, namely "manipulators consist only of rigid bodies (links and motion transmission components)." However, this is an ideal situation that may be considered valid only for slow motion and small interacting forces. In practice, joint flexibility in robot manipulators is present because of the use of compliant transmission elements. It has been demonstrated that if flexibility is not taken into account when considering robot design and control, a degradation of the robot's overall expected performance typically occurs.

Joint flexibility is common in current industrial robots when motion transmission/reduction elements such as belts, long shafts, cables, harmonic drives, or cycloidal gears are used. These components' utility is to relocate the actuators on the robot base, thus improving dynamic efficiency and reducing the moved overall mass. However, in normal robot operation, these components are subjected to forces/torques. They become intrinsically flexible and introduce a time-varying displacement between the actuators' position and the driven links.

Recently, flexible actuation/transmission elements have been deliberately selected in robots intended for physical human-robot interaction. This mechanical compliance guarantees inertial decoupling between the actuator and the link and leads to reducing kinetic energy involved in case of a collision with the human. Fig. 6.12 shows two examples of flexible joint robot manipulators, the *Dexter* robot and the lightweight robot manipulator *DLR LWR-III*.

A difficult trade-off between previously presented safety-oriented mechanical design and the more complex control arising should be found if one wants to preserve rigid robots' same performance. One way to realize this trade-off is to build an efficient, accurate dynamic model.

Dynamic modelling

The usual way to model flexible joint robots is to consider that the flexible transmission elements' deflection is modeled as being concentrated at the robot's joints. This fact significantly reduces the complexity of the associated equations of motion.

For the modeling purpose, we consider that the robot is made up of $n + 1$ rigid bodies, interconnected by n rotary joints, undergoing deflection, and actuated

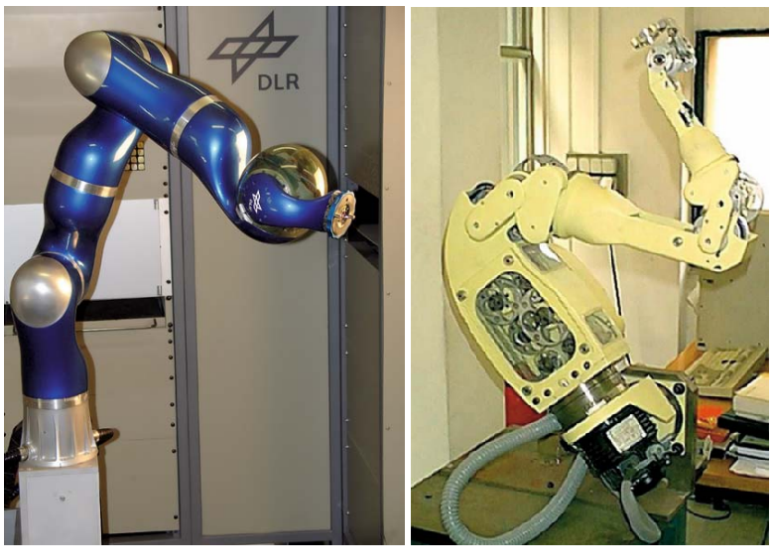

Figure 6.12: (a): The lightweight manipulator DLR LWR-III by the German Aerospace Center, (b): The cable-driven robot Dexter by Scienzia Machinale

by n electrical drives. Three main standard assumptions are made to reduce the complexity of the model.

Assumption 1 *Joint deflections are small, so that flexibility effects are limited to the domain of linear elasticity.*

Assumption 2 *The actuators' rotors are modelled as uniform bodies having their center of mass on the rotation axis.*

Assumption 3 *Each motor is located on the robot arm in a position preceding the driven link.*

Under these assumptions, and using the Denavit-Hartenberg convention and the Euler-Lagrange formalism, the following simplified dynamic model can be derived

$$\begin{array}{lcl} M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) & = & K(N\theta_m - q) \\ J\ddot{\theta}_m + B\dot{\theta}_m + NK(N\theta_m - q) & = & \tau \end{array} \tag{6.26}$$

where $q \in \Re^n$, and $\theta_m \in \Re^n$ are the vector of the link angles and the vector of rotor angular position, respectively. $M(q) \in \Re^{n\times n}$, $C(q,\dot{q}) \in \Re^{n\times n}$, $G(q) \in \Re^n$, and $F(\dot{q}) \in \Re^n$ are the inertia matrix of the links, the Coriolis/centrifugal matrix, the gravity vector and the friction vector, respectively. $N \in \Re^{n\times n}$, $K \in \Re^{n\times n}$, $J \in \Re^{n\times n}$, and $B \in \Re^{n\times n}$ are positive definite diagonal matrix of the gear reduction, the joint stiffness, the actuator inertia and back electromotive force damping, respectively. τ is the torque input on each actuator.

When the electrical part of the actuator is considered, the motor torque τ is produced by the motor current such that $\tau = K_m I_a$, and the dynamic equation of this current is given as follows

$$L\dot{I}_a + RI_a + K_b\dot{\theta}_m = u(t) \tag{6.27}$$

where $I_a \in \Re^n$ is the armature current vector. $L \in \Re^{n\times n}$, $R \in \Re^{n\times n}$, and $K_b \in \Re^{n\times n}$ are the diagonal matrix of electrical inductance, armature resistance, and back electromotive force effects, respectively. $u(t) \in \Re^n$ is the control input voltage.

Equations (6.26) and (6.27) represent the reduced dynamic model of the flexible joint robot actuated by direct current (DC) motors. Control such a non-linear system is a relevant task. The tracking control design is the hardest and most interesting issue in controlling flexible joint robots, and numerous researches have been done in the literature.

6.2.3 Command filtered backstepping algorithm for SISO nonlinear systems

This section presents the modification of the backstepping approach that eliminates the analytic computation of the derivative of virtual control, known as command filtered backstepping control (CFBC). Instead of differentiation, the derivative of virtual controls is computed using command filters. consider the following class of n-th order single-input-single-output (SISO) nonlinear system.

$$\begin{aligned} \dot{x}_i &= f_i(\bar{x}_i) + g_i(\bar{x}_i)x_{i+1} \\ \dot{x}_n &= f_n(x) + g_n(x)u(t) \end{aligned} \tag{6.28}$$

where $i = 1, 2, ...n-1$, $\bar{x}_i = [x_1, ..., x_i]^T$, $x = [x_1, x_2, ..., x_n]^T \in \Re^n$, x_1 the first scalar state, and $u(t)$ the scalar control signal. In this case the function f_i and g_i are supposed to be known. It is assumed that $|g_i(\bar{x}_i)| \geq g_0 \geq 0$.

Let $x_d(t)$ be the desired trajectory with the derivative $\dot{x}_d$ both of which are available and bounded for all $t \geq 0$. The objective of the control design is to found the control signal $u(t)$, which steer $x_1(t)$ from any initial conditions to track the reference $x_d(t)$ and to achieve the stability of the closed-loop system.

The design procedure is given by the following steps.

Step 1: This step aims to stabilize the first subsystem by the use of virtual control α_1.

Define the tracking errors $\tilde{x}_1 = x_1 - x_d$, and $\tilde{x}_2 = x_2 - x_{2c}$, where x_{2c} is the filtered version of the virtual control α_1. The later implies that $\dot{x}_{2c}$ will be the filtered version of the derivative of the virtual control $\dot{\alpha}_1$. To compensate for filtering errors $x_{2c} - \alpha_1$, an auxiliary signal ξ_1 is used. Thus new tracking error signals are needed namely *the compensated tracking* errors: $z_1 = \tilde{x}_1 - \xi_1$, and $z_2 = \tilde{x}_2 - \xi_2$, where ξ_2 is the auxiliary state which will be needed in the second step to deal with the filtering error $x_{3c} - \alpha_2$. Taking the derivative of z_1, yields

$$\begin{aligned} \dot{z}_1 &= \dot{\tilde{x}}_1 - \dot{\xi}_1 \\ &= \dot{x}_1 - \dot{x}_d - \dot{\xi}_1 \\ &= f_1(\bar{x}_1) + g_1(\bar{x}_1)x_2 - \dot{x}_d - \dot{\xi}_1 \\ &= g_1(\bar{x}_1)\tilde{x}_2 + f_1(\bar{x}_1) - \dot{x}_d + g_1(\bar{x}_1)\alpha_1 + g_1(\bar{x}_1)(x_{2c} - \alpha_1) - \dot{\xi}_1. \end{aligned} \tag{6.29}$$

To ensure the convergence $\xi_1 \to 0$, for $t \to \infty$, to cancel the filtering error $x_{2c} - \alpha_1$, and make z_2 appears in (6.29), choose $\dot{\xi}_1$ as

$$\dot{\xi}_1 = -k_1\xi_1 + g_1(\bar{x}_1)(x_{2c} - \alpha_1) + g_1(\bar{x}_1)\xi_2 \tag{6.30}$$

where k_1 is a design positive scalar.

Substituting (6.30) into (6.29) yields

$$\dot{z}_1 = g_1(\bar{x}_1)z_2 + f_1(\bar{x}_1) - \dot{x}_d + k_1\xi_1 + g_1(\bar{x}_1)\alpha_1. \tag{6.31}$$

To found the virtual control law α_1, choose the Lyapunov function as $V_1 = \frac{1}{2}z_1^2$. Taking its derivative one as

$$\begin{aligned} \dot{V}_1 &= z_1\dot{z}_1 \\ &= z_1[g_1(\bar{x}_1)z_2 + f_1(\bar{x}_1) - \dot{x}_d + k_1\xi_1 + g_1(\bar{x}_1)\alpha_1]. \end{aligned} \tag{6.32}$$

To realized $\dot{V}_1 \leq 0$ for $z_2 \to 0$, choose the virtual control as

$$\alpha_1 = \frac{1}{g_1(\bar{x}_1)}[-k_1\tilde{x}_1 + \dot{x}_d - f_1(\bar{x}_1)]. \tag{6.33}$$

Substituting (6.33) into (6.32) it follows

$$\dot{V}_1 = -k_1z_1^2 + g_1(\bar{x}_1)z_1z_2. \tag{6.34}$$

Step 2: In this step, the virtual control α_2 is computed to stabilise the second subsystem, that is to realize $z_2 \to 0$ for $t \to \infty$. For this purpose, define the tracking error $\tilde{x}_3 = x_3 - x_{3c}$, the compensated tracking error $z_3 = \tilde{x}_3 - \xi_3$, where ξ_3 is an auxiliary state which will be used to compensate for the filtering error $x_{4c} - \alpha_3$, with x_{4c} the filtered version of α_3. Taking the derivative of z_2 yields

$$\begin{aligned} \dot{z}_2 &= \dot{x}_2 - \dot{x}_d - \dot{\xi}_2 \\ &= f_2(\bar{x}_2) + g_2(\bar{x}_2)x_3 - \dot{x}_{2c} - \dot{\xi}_2 \\ &= g_2(\bar{x}_2)\tilde{x}_3 + f_3(\bar{x}_2) - \dot{x}_{2c} + g_2(\bar{x}_2)\alpha_2 + g_2(\bar{x}_2)(x_{3c} - \alpha_2) - \dot{\xi}_2. \end{aligned} \tag{6.35}$$

To ensure the convergence $\xi_2 \to 0$, for $t \to \infty$, to cancel the filtering error $x_{3c} - \alpha_2$, and make z_3 appears in (6.35), choose $\dot{\xi}_2$ as

$$\dot{\xi}_2 = -k_2\xi_2 + g_2(\bar{x}_2)(x_{3c} - \alpha_2) + g_2(\bar{x}_2)\xi_3. \tag{6.36}$$

where k_2 is a design positive scalar.

Substituting (6.36) into (6.35) yields

$$\dot{z}_2 = g_2(\bar{x}_2)z_3 + f_2(\bar{x}_2) - \dot{x}_{2c} + k_2\xi_2 + g_2(\bar{x}_2)\alpha_2. \tag{6.37}$$

To found the virtual control law α_2, choose the Lyapunov function as $V_2 = V_1 + \frac{1}{2}z_2^2$. Its derivative is given as

$$\begin{aligned} \dot{V}_2 &= \dot{V}_2 + z_2\dot{z}_2 \\ &= -k_1z_1^2 + z_2[g_2(\bar{x}_2)z_3 + g_1(\bar{x}_1)z_1 + f_2(\bar{x}_2) - \dot{x}_{2c} + k_2\xi_2 + g_2(\bar{x}_2)\alpha_2]. \end{aligned} \tag{6.38}$$

To realized $\dot{V}_2 \leq 0$ for $z_3 \to 0$, choose the virtual control as

$$\alpha_2 = \frac{1}{g_2(\bar{x}_2)}[-k_2\tilde{x}_2 + \dot{x}_{2c} - f_2(\bar{x}_2) - g_1(\bar{x}_1)z_1]. \tag{6.39}$$

Substituting (6.39) into (6.38) it follows

$$\dot{V}_2 = -k_1z_1^2 - k_2z_2^2 + g_2(\bar{x}_2)z_2z_3. \tag{6.40}$$

Step i ($2 \leq i \leq n-1$): in this step, the virtual control α_i is computed to ensure $z_i \to 0$ for $t \to \infty$. Following the same line as in step 2, and defining the Lyapunov fucntion as $V_i = V_{i-1} + \frac{1}{2}z_i^2$ the following virtual control is obtained

$$\alpha_i = \frac{1}{g_i(\bar{x}_i)}[-k_i\tilde{x}_i + \dot{x}_{ic} - f_i(\bar{x}_i) - g_{i-1}(\bar{x}_i)z_{i-1}]. \tag{6.41}$$

where $z_i = \tilde{x}_i - \xi_i$, and ξ chosen as

$$\dot{\xi}_i = -k_i\xi_i + g_i(\bar{x}_i)(x_{i+1,c} - \alpha_i) + g_i(\bar{x}_i)\xi_{i+1}. \tag{6.42}$$

After simple mathematical rearrangements, the derivative of the Lyapunov function is as follows

$$\dot{V}_i = \sum_{j=1}^{i} -k_iz_i^2 + g_i(\bar{x}_i)z_iz_{i+1}. \tag{6.43}$$

Step n: This is the final step, where the control law $u(t)$ is computed. The procedure is similar to that of step i. By choosing the Lyapunov function as $V_n = V_{n-1} + \frac{1}{2}z_n^2$, the control law is obtained as

$$u(t) = \alpha_n \tag{6.44}$$

with $\tilde{x}_n = x_n - x_{nc}$, $z_n = \tilde{x}_n$, and $\xi_n = 0$. The derivative of the Lyapunov function is obtained as follows

$$\dot{V}_n = \sum_{j=1}^{n} -k_nz_n^2. \tag{6.45}$$

Based on the Lyapunov theory, it is demonstrated in [Farrell *et al.* 2009] that the control law $u(t)$ achieves the same stability properties as the one obtained by using the standard backstepping algorithm.

For $i = 1, ...n-1$ the state space implementation of command filters (CF) are chosen as

$$\begin{aligned} \dot{y}_{i1} &= \omega_n y_{i2} \\ \dot{y}_{i2} &= -2\zeta\omega_n y_{i2} - \omega_n(y_{i1} - \alpha_i) \end{aligned} \tag{6.46}$$

with $x_{i+1,c}(t) = y_{i1}$, and $x_{i+1,c}(t) = \omega_n y_{i2}$ the output of each filter. Th filter initial conditions are set as $y_{i1}(0) = \alpha_i(0)$, and $y_{i2}(0) = 0$. $\omega_n \geq k_{i+1}$ and $\zeta \in (0, 1]$ are filter design parameters.

As we will see in the next section, the command filtered backstepping algorithm needs fundamental modifications to control flexible joints robot manipulators. Our first contribution to the domain of command filtered backstepping control (CFBC) for flexible joint robots was this concern.

6.2.4 Robust adaptive control for robot manipulators: Support vector regression-based command filtered adaptive backstepping approach [Ahanda *et al.* 2018a]

Highlight of the contribution

One of the drawbacks of backstepping control approaches is their sensitivity to model uncertainties and external disturbances. This noise sensitivity is more a relevant problem in the CFBC approach due to command filters. There is, therefore, a need to encounter uncertain dynamics and external disturbances by introducing an efficient and robust adaptive architecture in the command filtered backstepping algorithm. Moreover, to simplify the closed-loop robotic systems' stability analysis, the skew symmetry property of the matrix $\dot{M}(q, \dot{q}) - 2C(q, \dot{q})$ is usually used. However, the fact that the command filtered backstepping algorithm uses an error compensation mechanism to compensate for filtering errors $(x_{i+1,c} - \alpha_i)$, renders problematic the use of the skew-symmetric property and leads to more complex stability analysis. Based on the drawbacks above, a new command filtered backstepping algorithm is proposed to control flexible joint robots. The proposed control approach's effectiveness leads to the introduction of robust adaptive architecture and a new error compensation mechanism.

Adaptive control is a group of methods that provide a systematic approach for automatically adjusting controllers to achieve or maintain a desired level of control performance when the plant's parameters are unknown or change in time. Adaptive law is provided in the control system such that the required performance of the testing system is achieved. In robot manipulators, function approximation technique is used as an adaptive algorithm to learn an unknown or uncertain dynamics. For instance, in [Huang & Chen 2004], an adaptive sliding controller is proposed for a single-link flexible-joint robot subjected to mismatch uncertainties. Chien et al. [Chien & Huang 2007a] proposed an adaptive control using Fourier's series dynamic uncertainties approximation for flexible-joint robot manipulator subjected to time-varying uncertainties. M. M. Fateh [Fateh & Khorashadizadeh 2016] proposed a robust control by adaptive fuzzy estimation of uncertainties. W. He et al.

[He *et al.* 2016a] proposed a neural network control of a robotic manipulator subjected to input dead zone and output constraint, where adaptive neural networks are used to approximate the dead zone and the unknown model. In these valuable contributions, the stability of the system is proven by assuming low approximation errors. However, in a nonlinear system like flexible joint robots, this assumption is not always realizable.

It has been shown in [Smola & Schölkopf 2004] that support vector regression (SVR) generates global solutions, unlike other function approximations techniques such as Fourier series and neural networks. This is because SVR transforms the regression problem into quadratic programming (QP) one, such that a global solution can be obtained using QP-solvers. Therefore SVR is a solution to improve regression accuracy. Moreover, SVR offers the possibility to reduce the training database by a good selection of the so-called support-vectors. This data-based reduction is another advantage over the Neural network or Fourier series approach when coming with adaptive law for online computational cost reduction. SVR is, therefore, a reasonable choice for the design of adaptive architecture. This adaptive architecture is designed using the tangent-type robust term to deal with filtering and approximation errors and external disturbances.

Control design

In this part we consider that the model of the robot is given by 6.26 and 6.27. Let define the state space representation as:

$$\begin{aligned} e &= \dot{q} - v, \\ x_1 &= \theta_m, \\ x_2 &= \dot{\theta}_m, \\ x_3 &= I_a, \end{aligned} \tag{6.47}$$

with $v = \dot{q}_{des} - k_0\tilde{q}$ and $\tilde{q} = q - q_{des}$. k_0, q_{des}, and $\dot{q}_{des}$ are a positive diagonal matrix, the desired link position and its derivative, respectively. In this part, the actual and the nominal parameters are supposed to be different (model uncertainties). The state space representation of the system becomes

$$\begin{aligned} &M(q)\dot{e} = -[M(q)\dot{v} + C(q,\dot{q})(v+e) + G(q) + Kq] \\ &+KNx_1 + \Psi_1(q,\dot{q},x_1,t) \\ &\dot{x}_1 = x_2 \\ &\dot{x}_2 = K_{mJ}x_3 - B_J x_2 - NK_J(Nx_1 - q) + \Psi_2(q,x_1,x_2,x_3,t) \\ &\dot{x}_3 = L^{-1}u - R_L x_3 - K_{bL}x_2 + \Psi_3(u,x_2,x_3,t), \end{aligned} \tag{6.48}$$

where,

$$\begin{aligned}
&\Psi_1(q,\dot{q},x_1,t) = -M(q)\hat{M}^{-1}(q)[\hat{K}(Nx_1-q) - d_1 \\
&-\hat{C}(q,\dot{q})\dot{q} - \hat{G}(q)] + [K(Nx_1-q) - C(q,\dot{q})\dot{q} - G(q)] \\
&\Psi_2(q,x_1,x_2,x_3,t) = (B_J - \hat{B}_J)x_2 \\
&+(K_J - \hat{K}_J)N(Nx_1-q) + d_2 \\
&\Psi_3(u,x_2,x_3,t) = (\hat{L}^{-1} - L^{-1})u + (R_L - \hat{R}_L)x_3 \\
&+(K_{bL} - \hat{K}_{bL})x_2 + d_3,
\end{aligned} \tag{6.49}$$

Ψ_i represent the dynamic uncertainties of the robot manipulator, d_i are the external disturbances. The matrix $\hat{*}$ represents the actual value of the matrix $*$. Ψ_i are unknown functions, which will be approximated using SVR.

Define the compensated tracking error as: $z_1 = e - \xi_1$, $z_2 = \tilde{x}_1 - \xi_2$, $z_3 = \tilde{x}_2 - \xi_3$, and $z_4 = \tilde{x}_3$, where, $e = \dot{q} - v$, $\tilde{x}_1 = x_1 - x_{1c}$, $\tilde{x}_1 = x_2 - x_{2c}$, and $\tilde{x}_3 = x_3 - x_{3c}$. Following the command filtered backstepping procedure, the following are found: The new error compensation mechanism,

$$\begin{aligned}
&M(q)\dot{\xi}_1 = -(C(q,\dot{q}) + k_1)\xi_1 + (KN)(x_{1c} - \alpha_1) + KN\xi_2, \\
&\dot{\xi}_2 = -k_2\xi_2 + (x_{2c} - \alpha_2) + \xi_3, \\
&\dot{\xi}_3 = -k_3\xi_3 + K_{mJ}(x_{3c} - \alpha_3).
\end{aligned} \tag{6.50}$$

The control law,

$$\begin{aligned}
&\alpha_1 = -((Kr)^{-1})[k_1 e - (M(q)\dot{v} + C(q,\dot{q})v + G(q) + Kq) \\
&-\hat{w}_1^T\phi(q,\dot{q},x_1,t) - \delta_1 \tanh\left({}^{z_1}/_{\varepsilon_0}\right)], \\
&\alpha_2 = -(k_2\tilde{x}_1 + KNz_1 - \dot{x}_{1c}), \\
&\alpha_3 = -({K_{mJ}}^{-1})[k_3\tilde{x}_2 - (B_J x_2 + NK_J(x_1 - q)) - z_2 - \dot{x}_{2c} \\
&-\hat{w}_2^T\phi(q,x_1,x_2,x_3,t) - \delta_2 \tanh\left({}^{z_3}/_{\varepsilon_0}\right)], \\
&u = -L[k_4\tilde{x}_3 - (R_L x_3 + K_{bL}x_2) - K_{mL}z_3 - \dot{x}_{3c} \\
&-\hat{w}_3^T\phi(x_2,x_3,u,t) - \delta_3 \tanh\left({}^{z_4}/_{\varepsilon_0}\right)].
\end{aligned} \tag{6.51}$$

with the update law

$$\begin{aligned}
\dot{\hat{w}}_1 &= \Gamma_1\left(\phi(q,\dot{q},x_1,t){z_1}^T - \sigma_1\hat{w}_1\right) \\
\dot{\hat{w}}_2 &= \Gamma_2\left(\phi(q,x_1,x_2,x_3,t){z_3}^T - \sigma_2\hat{w}_2\right) \\
\dot{\hat{w}}_3 &= \Gamma_3\left(\phi(x_2,x_3,u,t){z_4}^T - \sigma_3\hat{w}_3\right),
\end{aligned} \tag{6.52}$$

where Γ_i is positive diagonal matrix, and σ_i is a small positive constant, called $\sigma-$modification coefficient.

It can be demonstrated that the control law 6.51 achieved asymptotic stability of the closed-loop system and a good disturbances rejection. Moreover, the error compensation mechanism 6.50 facilitates the demonstration of the closed-loop stability while ensuring the compensation of filtering errors.

To demonstrate the proposed approach's effectiveness, let conduct a case study using a 3-link flexible joint robot manipulator actuated by DC motors. The reference trajectories are chosen as: $q_1^{des}(t) = 0.9t + 0.2t^2$, $q_2^{des}(t) = 1 + 0.5cos(2t)$,

$q_3^{des}(t) = 1 + 0.6(sin(t))^2$. We consider that, due to the load variation during the robot operations, the actual mass of the third link ($\hat{m}_3$) is different from its nominal mass (m_3), and it can be computed as $\hat{m}_3 = m_3 + a\%m_3$, where $a\%$ is the uncertainty's percentage. That is, we consider that the robotic system is perturbed only by $\Psi_1(q, \dot{q}, x_1, t)$ uncertainty. We choose three values of $a\%$, namely 13%, 15%, and 18%. We compare the simulation results of the adaptive command filtered backstepping (ADCFB) and the robust adaptive command filtered backstepping (RADCFB) for each value. The results are shown in Fig.6.13. It is observed that the ADCFB performs good tracking performances up to 70% of m_3 uncertainty, while there is a degradation of the ADCFB tracking performances for m_3 uncertainty up to $18,3\%$. As shown in Fig.6.14, the proposed approach (RADCFB) performs good tracking accuracy without additional control effort. Thus there is an improvement of tracking performances without additional control effort.

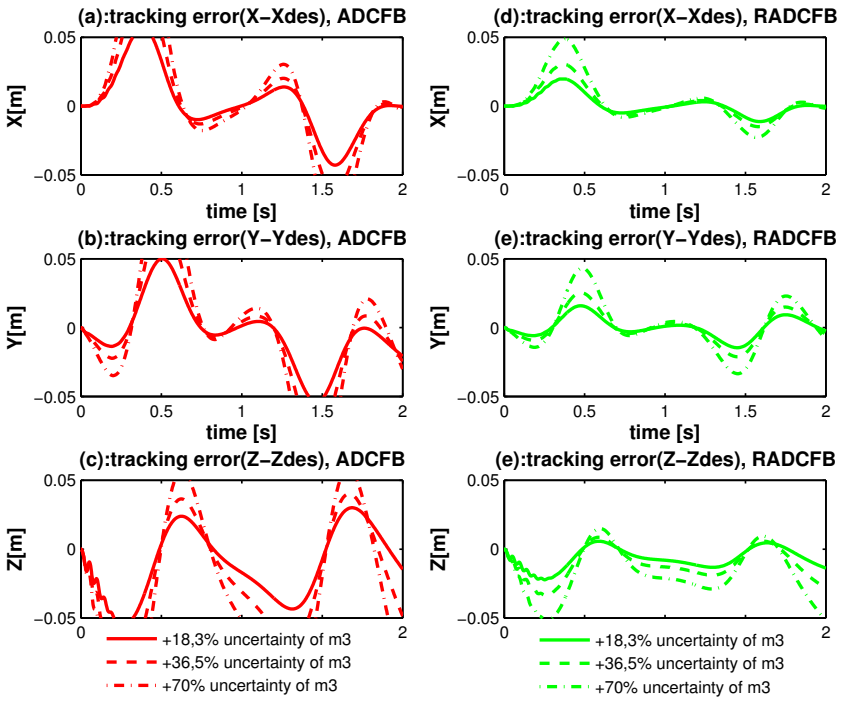

Figure 6.13: Comparison of tracking performances between ADCFB and RADCFB under tree cases of m_3 uncertainties

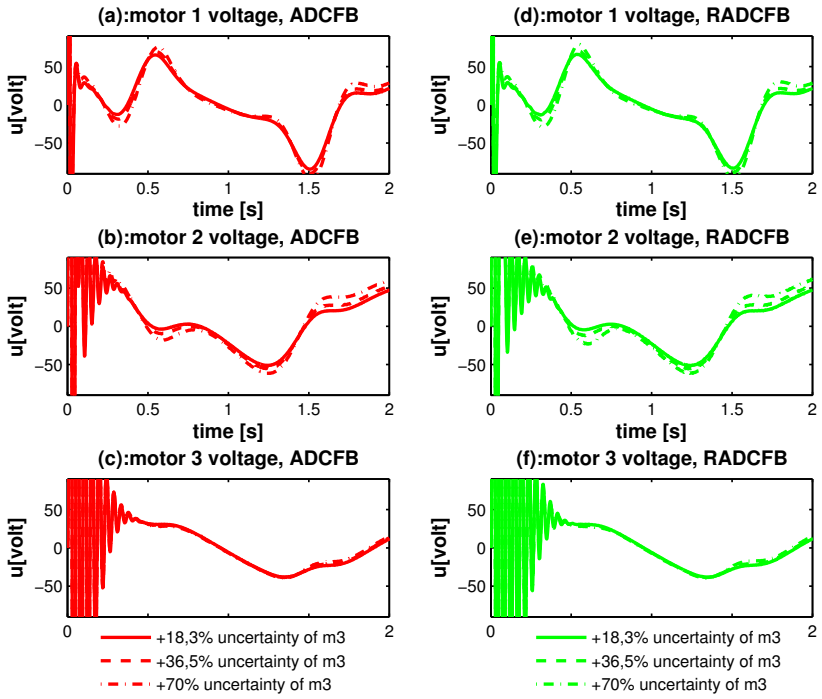

Figure 6.14: Comparison of output voltages between ADCFB and RADCFB under tree cases of m_3 uncertainties

In the context of human-robot collaboration, one way to ensure human safety is to constraint the robot joint angle set. Found a control law which deals with joint constraints while keeping good accuracy is a relevant task. The following section aims to address the issue.

6.2.5 Robust adaptive command filtered control of a robotic manipulator with uncertain dynamic and joint space constraints [Ahanda *et al.* 2018b]

Highlight of the contribution

In practice, constraints are ubiquitous in any physical system and can be manifested as physical stoppages, saturation, and performance and safety specifications, among others. It is also known that when the constraints are violated during the system's operation, it may result in performance degradation or system damage. There is, therefore, a need to consider both dynamic uncertainties and constraints in the control of robot manipulators.

The rigorous handling of uncertainties and constraints has become an important research topic in these past few years, and various methods have been proposed. In [Chien & Huang 2007b, Bürger & Guay 2010] the notion of set invariance control is used to handle constraints. The method consists of switching between a

nominal controller in the interior of the admissible set and the intervention control at the boundary, using the idea of barrier certificates to ensure invariance. Predictive control [Grüne & Pannek 2007, Magni *et al.* 2009] is another well-known method to control constrained non-linear systems. The technique considers the problem within an optimization framework inherently suitable for handling constraints, by solving an on-line finite horizon open-loop optimal control problem, subject to the system dynamics and constraints. In [Ding *et al.* 2017], an estimation technique is used to estimate the unknown force-related constraints in the case of the opening door problem. The approach is based on manipulator joint positions measurements instead of force/torque sensors, which is expensive or can be affected by computational delays and physical conditions such as temperature, humidity, or radiation. The estimation technique uses the motor current and the link-side and motor-side encoders. Other notable methods include reference governor based control [Bemporad 1998, Gilbert & Kolmanovsky 2002], extremum seeking control [Dehaan & Guay 2005], non-overshooting control [Krstic & Bement 2006], adaptive variable structure control [Su *et al.* 1995] and error transformation [Do 2010]. However, all proposed approaches are difficult to implement with a command filtered backstepping algorithm. On of the difficulty leads to the presence of an error compensation mechanism. It was recently shown that by using a barrier Lyapunov function, constraint demands could be met [Tee & Ge 2011, Niu & Zhao 2013]. This is because the barrier Lyapunov function grows to infinity whenever it approaches a defined limit. Thus, if the barrier Lyapunov function is kept bounded, the defined constraints will be met. This approach seems closed to the backstepping philosophy since Lyapunov's barrier can be part of Lyapunov's overall backstepping. However, if the control system is of order n, the algorithm presented in [Tee & Ge 2011, Niu & Zhao 2013], will need the n-*th* derivative of the desired trajectory. This fact can be computationally and technically impossible in high order systems ($n \geq 3$) like flexible joint robot manipulators. Because, for such systems, the controls are constructed in the task space. The transformation from task space to joint space (inverse kinematic) cannot compute more than three order's derivative of the desired joint angle.

This part's contribution leads to the use of a command filter to overcome the drawback and satisfy practical needs. To achieve constraint requirements, we constraint the first compensated tracking error z_1 using the barrier Lyapunov function.

Control design

In this part we also consider that nominal and actual parameters are different, that is the robot's model can be represented as follows:

$$\begin{aligned} &M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = K(Nq_m - q) + \Psi_l(q,\dot{q},t) \\ &J\ddot{\theta}_m + B\dot{\theta}_m + NK(N\theta_m - q) = K_m I_a + \Psi_m(q,\theta_m,t) \\ &L\dot{I}_a + RI_a + K_b\dot{\theta}_m = u + \Psi_i(q,I_a,t), \end{aligned} \tag{6.53}$$

where

$$\begin{aligned}
&\Psi_1(q,\dot{q},\theta_m,t) = -M(q)\hat{M}(q)^{-1}\left[\hat{C}(q,\dot{q})\dot{q} + \hat{G}(q) + \hat{K}(N\theta_m - q) + f_{dis}(t)\right] \\
&+ [C(q,\dot{q})\dot{q} + G(q) + K(N\theta_m - q)] \\
&\Psi_2(q,\theta_m,\dot{\theta}_m,I_a,t) = -J\hat{J}^{-1}\left[\hat{B}\dot{\theta}_m + \hat{K}_m I_a + N\hat{K}(N\theta_m - q) + f_m(t)\right] \\
&+ \left[B\dot{\theta}_m + K_m I_a + NK(N\theta_m - q)\right] \\
&\Psi_3(q,I_a,t) = -L\hat{L}^{-1}\left[\hat{R}I_a + \hat{K}_b\dot{q}_m + f_i(t)\right] + \left[RI_a + K_b\dot{\theta}_m\right].
\end{aligned} \tag{6.54}$$

Let the state space variables be:

$$\begin{aligned}
x_1 &= q, \\
x_2 &= \dot{q}, \\
x_3 &= \theta_m, \\
x_4 &= \dot{\theta}_m, \\
x_5 &= I_a.
\end{aligned} \tag{6.55}$$

Substituting these variables in 6.53 yields

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= D(q)^{-1}\left[K\left(Nx_3 - x_1\right) - G(x_1) - C(x_1,x_2)x_2 + \Psi_1\left(x_1,x_2,x_3,t\right)\right] \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= J^{-1}\left[K_m x_5 - Bx_4 - rK\left(Nx_3 - x_1\right) + \Psi_2\left(x_1,x_3,x_4,x_5,t\right)\right] \\
\dot{x}_5 &= L^{-1}\left[u - Rx_5 - K_b x_4 + \Psi_3\left(x_4,x_5,t\right)\right].
\end{aligned} \tag{6.56}$$

The main objective is to satisfy the constraint requirements $|x_1| \leq k$, where k is a vector of constraint values. If the control law $u(t)$ ensure that z_1, and ξ_1 are bounded, then, based on the fact that $\tilde{x}_1 = z_1 + \xi_1$, it follows that $\tilde{x}_1$ is bounded. In addition, for a given bounded desired link angle x_{1d}, the state x_1 will be bounded. To ensure the boundedness of z_1, a barrier Lyapunov function is used, and for the boundedness of ξ_1, we proposed a robust adaptive support vector regression. Define a constant vector b such that $|z_1| \leq b \leq k$, and a vector Z as

$$Z = [\frac{z_{11}}{{b_1}^2 - {z_{11}}^2}\frac{z_{12}}{{b_2}^2 - {z_{12}}^2}\cdots\frac{z_{1n}}{{b_n}^2 - {z_{1n}}^2}]^T, \tag{6.57}$$

where b_i are the elements of the vector b, and z_{1i} the elements of the vector z_1.

Following the command filtered backstepping algorithm, it results to:
The control laws:

$$\begin{aligned}
&\alpha_1 = \dot{x}_{1d} - k_1\tilde{x}_1 \\
&\alpha_2 = (KN)^{-1}[M(x_1)\dot{x}_{2c} - k_2\tilde{x}_2 - Z + Kx_1 + G(x_1) + C(x_1,x_2)x_{2c} - \hat{w}_1^T\phi_1 - \delta_1 tanh(\tfrac{z_2}{\varepsilon_0})] \\
&\alpha_3 = \dot{x}_{3c} - k_3\tilde{x}_3 - KNz_2 \\
&\alpha_4 = {K_m}^{-1}J[\dot{x}_{4c} - k_4\tilde{x}_4 + J^{-1}(Bx_4 + NK(Nx_3 - x_1) - z_3 - \hat{w}_2^T\phi_2 - \delta_2 tanh(\tfrac{z_4}{\varepsilon_0}))] \\
&u = L[\dot{x}_{5c} - k_5 z_5 - L^{-1}K_m z_4 + L^{-1}(Rx_5 + K_b x_4 - \hat{w}_3^T\phi_3 - \delta_3 tanh(\tfrac{z_5}{\varepsilon_0}))],
\end{aligned} \tag{6.58}$$

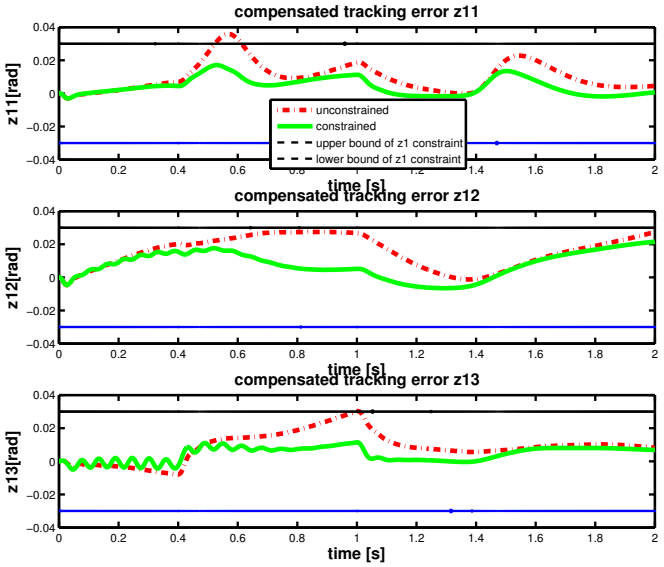

Figure 6.15: Compensated tracking error z_1 for constrained (green line) and unconstrained (red line) controllers

The update law:

$$\begin{aligned}\dot{\hat{w}}_1 &= \Gamma_1(\phi_1 {z_2}^T - \sigma_1 \hat{w}_1)\\ \dot{\hat{w}}_2 &= \Gamma_2(\phi_2 {z_4}^T - \sigma_2 \hat{w}_2)\\ \dot{\hat{w}}_3 &= \Gamma_3(\phi_3 {z_5}^T - \sigma_3 \hat{w}_3),\end{aligned} \tag{6.59}$$

The error compensation mechanism:

$$\begin{aligned}\dot{\xi}_1 &= -k_1\xi_1 + (x_{2c} - \alpha_1) + \xi_2\\ \dot{\xi}_2 &= -M(x_1)^{-1}[C(x_1, x_2) + k_2]\xi_2 + M(x_1)^{-1}KN\,(x_{3c} - \alpha_2) + M(x_1)^{-1}KN\xi_3\\ \dot{\xi}_3 &= -k_3\xi_3 + (x_{4c} - \alpha_3) + \xi_4\\ \dot{\xi}_4 &= -k_4\xi_4 + J^{-1}K_m\,(x_{5c} - \alpha_4)\,.\end{aligned} \tag{6.60}$$

To found out the virtual control law α_1, the first Lyapunov function is chosen as $V_1 = \frac{1}{2}\sum_{i=1}^{n} ln(\frac{{b_i}^2}{{b_i}^2 - {z_{1i}}^2})$ (which is a barrier Lyapunov function). Thus if $z_{1i} \to b_i$, V_1 grows to the infinity, which means that the system will be stopped (velocity tends to zeros).

To illustrate the proposed approach's superiority, let conduct a case study using a 3-link flexible joint robot manipulator actuated by DC motors. A comparative simulation between the constrained (proposed approach) and the unconstrained command filtered backstepping approaches is shown in Fig. 6.15, where the task is to track the reference $x_d = 1 - cos(\pi t)$, while satisfying the constraints $|z_{1i}| \leq 0.03$, $i = 1, ..., n$. It is shown that the proposed control satisfy the constraints while the unconstrained command filtered backstepping violet the constraints during and after

the transient phase ($t \geq 1s$). As shown in Fig. 6.16, after the transient phase, the proposed control approach performs the same control voltage as the unconstrained control law after the transient phase. These two results mean that the proposed approach satisfies the constraint requirements without additional control effort.

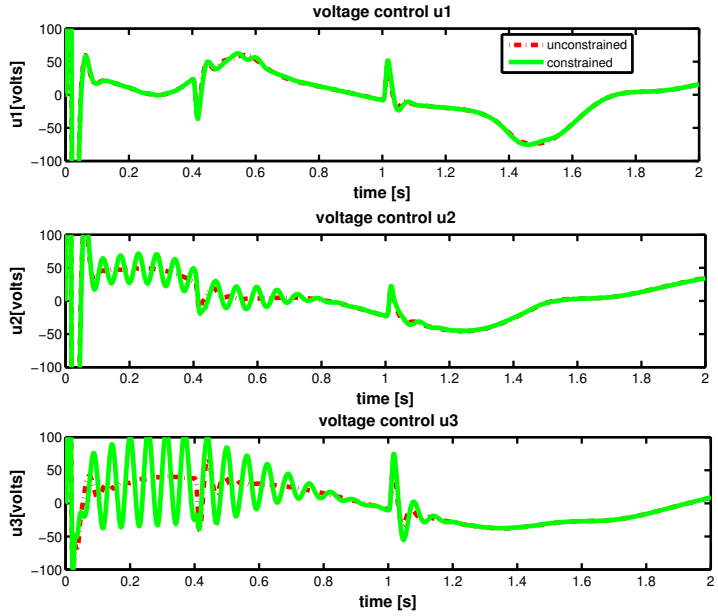

Figure 6.16: Voltage input of constrained (green line) and unconstrained controllers (red line).

The proposed two control approaches satisfy the performance requirements after a long transient phase (for instance, in Fig.6.15, the transient phase ends after $1s$). In the context of human-robot collaboration, fast stabilization can be useful to ensure safety. For instance, when the system goes to random oscillations, a control law with rapid stabilization can preserve tracking accuracy and, therefore, the human's security. The contribution of the next part goes to the adaptive finite-time command filtered backstepping control.

6.2.6 Adaptive Fuzzy Finite-time Command Filtered Backstepping Control of Flexible Joint Robots [Roger *et al.* 2020]

Highlight of the contribution

Command filtered backstepping control produces satisfactory performances, for the control uncertain high order nonlinear systems. It can be easily combined with a robust adaptive architecture, as demonstrated in the previous sections. Presently, new challenges for the command filtered backstepping control approaches are related to the following technical issues:

- How to cope with uncertainties to protect command filters (CF) while reducing significantly computational burden;

- How to ensure fast filtering, while guaranteeing the filter errors compensation;
- How to ensure fast stabilization of the closed-loop system despite the presence of inevitable filtering errors and model uncertainties.

Several approaches have been proposed to address one, two, or all the aforementioned technical issues. The authors usually use a finite-time control approach [Bhat & Bernstein 2000, Xia *et al.* 2018] to ensure fast filtering or fast stabilization, while uncertainties are generally coped using neural networks or fuzzy logic systems (FLSs) [C. Bing & Chong 2017, M. Li & Guangdeng 2019]. Yu et al. [Yu *et al.* 2017] proposed a fuzzy finite-time command filtered control for uncertain single input single output (SISO) system under input saturation. The authors developed new virtual control and error compensation mechanism. Han et al. [Han *et al.* 2018] developed a finite-time adaptive fuzzy command filtering controller for induction motors under input saturation. In [Yu *et al.* 2005, Zhao *et al.* 2010, Galicki 2015, Nguyen *et al.* 2019] different finite-time controllers are proposed for rigid robot manipulators. However, all the control approaches above cannot be useful for finite-time control of flexible joint robots. The reason is that joint flexibility and actuators dynamics render difficulty in the demonstration of the finite-time stability of the closed-loop system. To overcome this theoretical limitation, we use the positiveness and the boundedness of the inertia matrix, and the skew symmetry property of the matrix $\dot{M}(q) - 2C(d, \dot{q})\dot{q}$. Besides reducing the computational burden, an adaptive fuzzy controller is designed where only one adaptive parameter is required to be tuned online. Note that for the SVR adaptive laws developed above, the number of adaptive parameters is equal to the number of support vectors. Thus the new approach is an improvement of the two previous ones in terms of computational duration.

The following Properties and Lemmas are necessary to found out the control law.

Property 1 (Boundedness of inertia matrix) *The inertia matrix is bounded in the sense that*

$$\lambda_{min}\left(M\left(x\right)\right)\|x\|^2 \leq x^T M(x) x \leq \lambda_{max}\left(M\left(x\right)\right)\|x\|^2. \tag{6.61}$$

Property 2 (Skew Symmetry) *The inertia matrix and the centripetal Coriolis matrix have the following property:*

$$x^T\left(\dot{M}(q) - 2C(q, \dot{q})\right)x = 0, \forall x \in \Re^n \tag{6.62}$$

where $\dot{M}(q)$ is the time derivative of the inertia matrix.

Lemma 1 *[Zhao et al. 2010] The finite-time command filter (FTCF) is defined as:*

$$\begin{array}{rcl} \dot{y}_{i1} & = & v_{i1} \\ v_{i1} & = & -R_1|y_{i1} - \alpha_i|^{\frac{1}{2}} tanh((y_{i1} - \alpha_i)/\varepsilon_i) + y_{i2} \\ \dot{y}_{i2} & = & -R_2 tanh((y_{i1} - v_{i1})/\varepsilon_i) \end{array} \tag{6.63}$$

where R_1, R_2 and ε_i are design parameters, α_i the input, $x_{ic} = y_{i1}$ and $\dot{x}_{ic} = v_{i1}$ are the output of the FTCF.

Lemma 2 *[Yu et al. 2005] For any numbers $c_1 > 0$, $c_2 > 0$, $\gamma \in [0, 1]$, an extended Lyapunov condition of finite-time stability is defined as $\dot{V}(x)+c_1V(x)+c_2V^{\gamma}(x) \leq 0$, where the settling time is*

$$T \leq \frac{1}{c_1(1-\gamma)} ln(\frac{c_1V(x_0)^{1-\gamma} + c_2}{c_2}) \tag{6.64}$$

with $V(x_0)$ the initial value of $V(x)$.

Lemma 3 *[Yu et al. 2005] For $a_1 > 0$, $a_2 > 0$ and $0 < \gamma < 1$, the following inequality holds*

$$(a_1 + a_2)^{\gamma} \leq {a_1}^{\gamma} + {a_2}^{\gamma}. \tag{6.65}$$

Control design

In this part, we consider that the robot model is given as in 6.26, with $N = I_{n\times n}$. Let the state variable be as: $x_1 = \theta_m$, $x_2 = \dot{\theta}_m$, $\tilde{q} = q - q^{des}$. Then it follows

$$\begin{array}{rcl} M(q)\,\dot{e} & = & Kx_1 - C(q,\dot{q})\,e - M(q)\,\dot{v} \\ & & +f_1\left(q,\dot{q},q^{des}\right) \\ \dot{x}_1 & = & x_2 \\ \dot{x}_2 & = & J^{-1}u + f_2(q,x_1,x_2) \end{array} \tag{6.66}$$

where

$$\begin{array}{rcl} f_1\left(q,\dot{q},q^{des}\right) & = & -[C(q,\dot{q})\,v + G(q) + F(\dot{q}) + Kq] \\ f_2(q,x_1,x_2) & = & -J^{-1}\left[Bx_2 + K(x_1 - q)\right]. \end{array} \tag{6.67}$$

The objective is to design an adaptive fuzzy finite-time command filtered backstepping control law u for the n-link FJR system (6.66). The derivation of u has to take into account unknown non-linear functions f_1 and f_2, such that the output q can track the desired trajectory q^{des} in finite-time. Besides, all the closed-loop system signals have to converge to a neighborhood close to zero in finite-time.

To ensure compensation of finite-time filter errors, we have proposed the following mechanism:

$$\begin{array}{rcl} M(q)\dot{\xi}_1 & = & -(k_1 + C(q,\dot{q}))\xi_1 + K(x_{1c} - \alpha_1) \\ & & +K\xi_2 - \lambda_1\nu_1 \\ \dot{\xi}_2 & = & -k_2\xi_2 + (x_{2c} - \alpha_2) - K\xi_1 - \lambda_2\nu_2 \end{array} \tag{6.68}$$

where k_i is a positive definite diagonal matrix, λ_i is a positive constant, and ν_i is a vector defined as $\nu_i = [tanh(\xi_{i1}/a_i), tanh(\xi_{i2}/a_i), ..., tanh(\xi_{in}/a_i)]^T$, with $a_i > 0$.

To guarantee the finite-time convergence of the closed-loop signals, the virtual controls α_1 and α_2, the control law u and the update law $\hat{\theta}$ are chosen as

$$\begin{aligned}
\alpha_1 &= K^{-1}[-k_1 e + M(q)\dot{v} - \tfrac{1}{2}z_1 \\
&\quad -\tfrac{1}{\gamma_1}\Xi_{z_1}(z_1^T M(q) z_1)^{\frac{1+\beta}{2}} - (\tfrac{\hat{\theta} S_1^T S_1}{2h_1^2})z_1] \\
\alpha_2 &= -k_2\tilde{x}_1 - Ke + \dot{x}_{1c} - \gamma_2 \Lambda_{z_2}|z_2|^{\beta} \\
u &= J[-k_3 z_3 - z_2 + \dot{x}_{2c} - \tfrac{1}{2}z_3 - \gamma_3\Lambda_{z_3}|z_3|^{\beta} \\
&\quad -(\tfrac{\hat{\theta} S_2^T S_2}{2h_2^2})z_3] \\
\dot{\hat{\theta}} &= r(\tfrac{z_1^T z_1}{2h_1^2} S_1^T S_1 + \tfrac{z_3^T z_3}{2h_2^2} S_2^T S_2) - \sigma\hat{\theta}
\end{aligned} \tag{6.69}$$

where $\sigma, r, \gamma_1, \gamma_2, \gamma_3, h_1, h_2$, and $\beta \in [0,1]$ are a positive constants. Λ_{z_i} is a diagonal matrix defined as $\Lambda_{z_i} = diag([sign(z_{i1}), sign(z_{i2}), ..., sign(z_{in})])$, with $i = 2, 3$. Ξ_{z_1} is a vector defined as $\Xi_{z_1} = [\frac{sign(z_{11})}{|z_{11}|+\varepsilon_e}, \frac{sign(z_{12})}{|z_{12}|+\varepsilon_e}, ..., \frac{sign(z_{1n})}{|z_{1n}|+\varepsilon_e}]^T$, with $\varepsilon_e > 0$.

Remark 1 *Achieve finite-time stability of the closed-loop system is the main difficulty in the finite-time control of nonlinear system, and particularly when the system is order $n > 1$. For the case of command filtered backstepping control of FJR system ($n \geq 2$), this difficulty increase with the presence of errors compensation mechanism. To overcome this difficulty we have introduce the terms $M(q)\dot{v} - \frac{1}{\gamma_1}\Xi_{z_1}[z_1^T M(q) z_1|^{\frac{1+\beta}{2}}$, and $-\gamma_i\Lambda_{z_i}|z_i|^{\beta}$, $i = \{2, 3\}$, in the previous presented command filtered backstepping algorithm flexible for joint robots.*

Trough Lyapunov stability, it can be proved that the proposed control law ensure the boundedness of the tracking error $\tilde{q}_j$ in finite-time T_1 such that

$-\frac{\chi_0}{k_{01j}} \leq \tilde{q}_j(t) \leq \frac{\chi_0}{k_{01j}} + [\tilde{q}_j(0) - \frac{\chi_0}{k_{01j}}]exp(-k_{01j}T_1)$

where

$\chi_0 = max\{\sqrt{\frac{\eta_0}{k_0}}, \sqrt{2 \times (\frac{\eta_0}{2\lambda_0})^{\frac{4}{3}}}\} + max\{\sqrt{\frac{c_0}{a_0}}, \sqrt{2 \times (\frac{c_0}{2b_0})^{\frac{2}{1+\beta}}}\}$,

$T_1 = max\{max(T_{0i}) + \frac{2}{k_0}ln(\frac{k_0\sqrt{V_\xi(0)}+\lambda_0}{\lambda_0}), T\}$,

$a_0 = min\{(2k_{21} - \lambda_2), (2k_{22} - \lambda_2), ..., (2k_{2n} - \lambda_2), 2k_{31}, 2k_{32}, ..., 2k_{3n}, \frac{\sigma}{2}, 1\}$, $b_0 = min\{\frac{\varpi_0}{\gamma_1} \times 2^{\frac{1+\beta}{2}}, \gamma_2 \times 2^{\frac{1+\beta}{2}}, \gamma_3 \times 2^{\frac{1+\beta}{2}}, \sigma^{\frac{1+\beta}{2}}\}$, $k_0 = min\{2k_{1j}, 2k_{2j}\}$, $\lambda_0 = min\{\sqrt{2}(\lambda_1 - \mu_1 K_j), \sqrt{2}(\lambda_2 - \mu_2)\}$, j=1,2,...,n.

Remark 2 *From the obtained result, it is observed that the settling time T_1, and the bounds $-\frac{\chi_0}{k_{01j}}$ and $[\tilde{q}_j(0) - \frac{\chi_0}{k_{01j}}]exp(-k_{01j}T_1) + \frac{\chi_0}{k_{01j}}$, of the tracking error $\tilde{q}_j(t)$, can become smaller with a good choice of the design parameters, that is the larger values for γ_1, k_{01j}, k_{1j}, k_{2j} and k_{3j}, and the smaller values for γ_2, γ_3, λ_2, and σ. Thus in contrast to the previous developed control approaches, the proposed controller can not only deal with model uncertainty, but can also ensure finite-time convergence of the closed-loop signals, and finite-time filter errors compensation.*

To show the control approach's effectiveness, let run a case study with a 2-link flexible joint robot. We compare the proposed approach's tracking performance with the command filtred backstepping control approach developed by Ling et al. [Ling *et al.* 2019]. Simulation results are presented in Figs. 6.17 and 6.18. Fig. 6.17 shows the trajectories of the angular position q, the desired angular position

q^{des}, and the tracking error $\tilde{q}$ for the two controllers. It is shown clearly that the proposed control approach not only has a faster convergence rate but also achieves better tracking error, with a settling time $T_1 < 2s$ and the tracking error bounded as $|\tilde{q}| \leq 4 \times 10^{-6} rad$. Fig. 6.18 shows the control effort for the two controllers. The proposed approach does not require additional control efforts to achieve a better tracking effect. This observation means that the proposed controller performs the best tracking performance with no additional control effort.

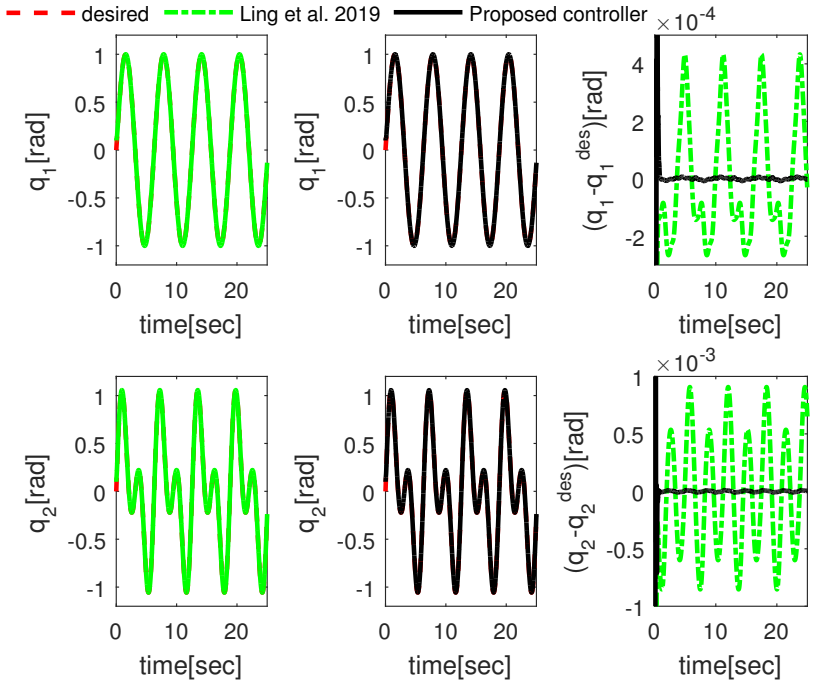

Figure 6.17: Case f_1 and f_2 considered known: Angular position q and tracking error $\tilde{q}$, for link 1 and link 2.

6.3 Conclusion

This chapter presents our contributions to the topics of inverse kinematic modeling of redundant manipulators and control of flexible joint manipulator robots. Regarding the inverse kinematic modeling of redundant manipulators, we present a general approach to derive the inverse kinematics of high dof rigid manipulators. The proposed method has been validated through simulations and experiments. Concerning the control of flexible joint manipulators, three contributions have been presented. In the first one, Command filtered backstepping algorithm has been applied to the case of flexible joint manipulator control, and support vector regressors (SVR) were used to deal with dynamic uncertainties. A tangent-type robust term was also used to cancel tracking and approximation errors. The second contribution implemented

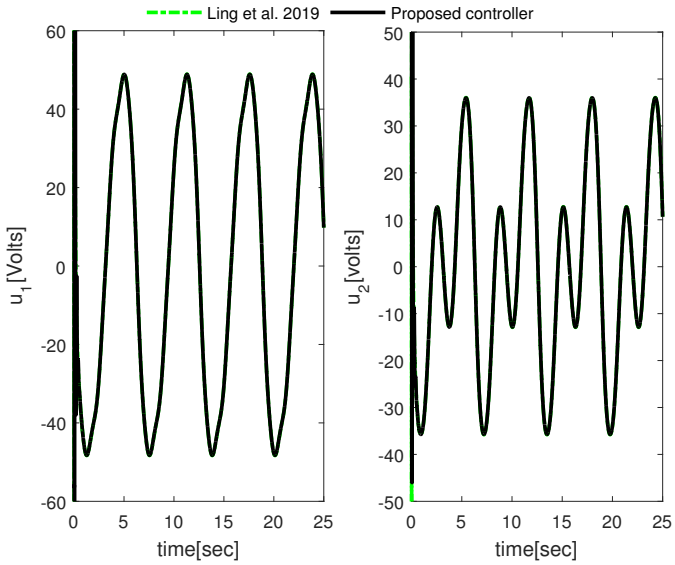

Figure 6.18: Case f_1 and f_2 considered known: The demanded control voltage u.

the same control approach to the case of joint space constraints. The last one implements the finite-time Command filtered algorithm to ensure fast stabilization.

Chapter 7

Conclusion and future work

In work presented above, we have presented control strategies for the autonomous navigation of mobile robots. We have also presented learning architectures for the modeling and control of continuum manipulators and a learning architecture that maintains multiple inverse kinematic solutions for high dof rigid manipulators. Finally, we have proposed new control laws for the control of flexible joint manipulators. Our prospective study is a continuation of these works and aims at improving the existing ones. Our research team recently interested in Machine Learning algorithms (artificial neural networks and machine vectors support), artificial intelligence (Fuzzy logic, artificial potential field), and backstepping control. Our main short-term objective is to reinforce our research team of the UY1, which currently has only two confirmed researchers. In this way, four Ph.D. theses are being finalized. Besides, our research team focuses on new algorithms:

- Deep neural networks for modeling and control of robotic systems, a Ph.D. thesis and a master thesis are in progress;
- Reinforcement learning for robot control, a master on the inverse kinematics of manipulator robots via reinforcement learning was completed; -
- Development PH-curves-based type-2 fuzzy logic algorithms for efficient robot motion planning, and a Ph.D. thesis is in progress;
- Siding mode control, a master thesis is in progress.

7.1 Autonomous navigation of mobile robots

Concerning the autonomous navigation of mobile robots, we have been interested in navigation strategies to move a robot intelligently from one point to another. However, strategies to reduce energy consumption have received little or no attention. Our strategies for minimizing the energy consumed have remained too dependent on the robot model's accuracy developed [Datouo *et al.* 2017, Bensekrane *et al.* 2020]. However, since an ideal model does not exist, with the research team's experience on PH curve modeling, our first prospective aims to focus on the development of PH-curves-based type-2 fuzzy logic algorithms. Fuzzy controllers whose profile of the generated velocities directly leads to the minimization of the consumed energy. This theme's second prospective study focuses on vision-based navigation, particularly on a possible coupling between deep learning networks and fuzzy logic.

7.2 Modelling and control of continuum robots

Chapter 5 highlights our contributions on the topic of modeling and control of continuum manipulators. The different learning architectures and control strategies proposed were validated both in simulation and experimentally, demonstrating their feasibility. We have succeeded in fulfilling many constraints existing in a classical control of a mechatronic system. However, the system dynamics or the interaction with the external environment were taken into account only to a limited extent. Thus, the first prospect is the validation of the algorithms' extension to the case of mobile continuum manipulators [Bouyom Boutchouang *et al.* 2020], and those in progress on kinematic modeling based on minimizing the energy of continuum manipulator mobiles. The next one aims at proposing a modeling methodology integrating both the dynamics of the continuum robot and those of its actuators. The logical next step will be to develop control laws for position and force control. However, this perspective is not the only envisaged. Indeed, most of the flexible robots currently under development are characterized by a very complex mechanical structure. Modeling problems will become more and more recurrent. To this end, very recently, we have been interested in learning architectures dealing with large databases [bou].

7.3 Kinematic modelling of high dof rigid manipulators

Regarding manipulators with rigid joints, much remains to be explored. After the validation of the extension proposed for mobile manipulators [Kouabon *et al.* 2020b], the objective is to focus on the kinematic modeling of multi-systems. Indeed, most of the robots currently under development are characterized by a complex, often anthropomorphic structure. Moreover, they are designed to be able to interact with humans in an advanced way. Therefore, these robots are called upon to carry out increasingly advanced missions where the need for cooperation can hardly be overlooked. In the future, we wish to move towards the kinematic modeling of systems with two or more mobile manipulators to carry out tasks requiring coordination (cooperative payload transport).

7.4 Control of flexible joint robots

Concerning the control of flexible joint robots, we have extended the strategy of reducing the complexity of backstepping (command filters), previously applicable to simple systems, to manipulators with flexible joints. Constraints on control effort were subsequently integrated, and the convergence of the control architecture was improved by offering finite-time control. However, the proposed algorithms were only validated in simulation. Consequently, the logical next step in this theme will be to move towards implementation in real systems. Algorithms developed in particular for the control of the horse leg are currently being implemented.

The next prospective study is an extension of the developed algorithms. Indeed, most control approaches developed for flexible joint robots are focused on deterministic cases, i.e., when the robot environment is stable and known. However, when this environment is unknown or subject to random vibrations, these methods can no longer ensure good tracking accuracy due to the un-modeled dynamics resulting from the vibrations.

When a robot manipulator is used in air or sea transport, it is subjected to random vibrations that are difficult to model. It is necessary to know how the random disturbances are modeled to propose an effective control method in this context. The first attempts in considering random vibration phenomena were first to consider the noise produced as white noise. Later, the idea was to consider them as the Wiener process's formal derivative process to deduce the Itô type stochastic differential equation systems. By following this idea, several stochastic modeling and controller have been proposed [M.-Y. Cui & Xie 2014, Z. Wu & Shii 2012]. The second idea was to view the produced disturbances as colored noise. Many modeling and controller approaches have been developed for random benchmark and automobile suspension systems [Z. Wu & Cui 2017]. For a class of electrically driven manipulators, presently, only Cui et al. [Cui & Wu 2019] have proposed a modeling approach and a controller design for the case of a random vibration environment. The authors have modeled random disturbances as torque or voltage disturbed colored noises. They have designed a vectorial backstepping controller to deal with joint flexibility and random uncertainties. However, the proposed control law is too complicated and can be time-consuming for real-time applications due to the repeated derivation of virtual control signals. The proposed control approach does not reduce the effect of disturbances on the tracking performances through robust terms.

We will propose a less complex and more robust control strategy for electrically driven flexible joint manipulators working in random vibration environments in future work. The main idea is to use command filters with new error compensation mechanisms to deal with random disturbances to reduce the control structure's complexity. Robust terms will also be used to protect command filters and mitigate the effect of disturbances. Therefore, the tracking error's mean square will converge to an arbitrarily small neighborhood close to zero.

Appendices

Appendix A

Publications attached to the manuscript

A.1 [Melingui *et al.* 2017b]

A. Melingui, J. J.-B. M. Ahanda, O. Lakhal, J. B. Mbede, and R. Merzouki, "Adaptive algorithms for performance improvement of a class of continuum manipulators," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 9, pp. 1531–1541, 2017.

Abstract: This paper addresses the position control of continuum manipulators. Their performances in terms of speed limitation and position accuracy are often mediocre compared with rigid body based robots. In regards to continuum manipulators control, nonadaptive kinematic schemes were shown poor performance in terms of tracking position accuracy, and existing adaptive schemes were time-consuming. This paper presents a novel adaptive control scheme, namely the adaptive support vector regressor controller. The proposed approach exploits the optimization learning methods which yield global solutions of the training problem while keeping small size regressors. These characteristics make it possible to accelerate the convergence of the closed-loop system, thus reducing the execution time. The experimental results obtained using the compact bionic handling assistant robot demonstrate that nonadaptive kinematic architectures even in the presence of accurate learning models are not robust enough to deal with these challenging platforms and that adaptive control schemes can significantly improve the performance.

A.2 [Lakhal *et al.* 2015]

O. Lakhal, **A. Melingui**, and R. Merzouki, "Hybrid approach for modeling and solving of kinematics of a compact bionic handling assistant manipulator," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 3, pp. 1326–1335, 2015.

Abstract: This paper deals with a methodology for real-time solving of a complex kinematics of a class of continuum manipulators, namely the compact bionic handling assistant (CBHA). First, a quantitative approach is used to model kinematically the CBHA inspired from the modeling of parallel rigid manipulators. For this case, the CBHA is modeled as a series of vertebrae, where each vertebra is connected to the next one through a flexible link. The latter named an intervertebra is modeled by three universal-prismatic-spherical and one universal-prismatic joints.

The kinematic models of the CBHA are derived from the inverse kinematic equations (IKE) of each intervertebra. A qualitative approach based on neural networks is used to provide approximated solutions of the IKE for real-time implementation. Thus, the combination of the advantages of quantitative and qualitative approaches allows proposing a hybrid methodology for accurate modelling and solving the kinematics of this class of continuum robots. A set of experiments is conducted using a CBHA in order to evaluate the level of efficiency of the proposed hybrid approach.

A.3 [Kouabon *et al.* 2020a]

A. G. Jiokou Kouabon, **A. Melingui**, J. M. Ahanda, O. Lakhal, V. Coelen, M. KOM, and R. Merzouki, "A learning framework to inverse kinematics of high dof redundant manipulators," *Mechanism and Machine Theory*, vol. 153, p. 103978, 2020.

Abstract: This paper proposes a learning framework for solving the inverse kinematics (IK) problem of high DOF redundant manipulators. These have several possible combinations to get the end effector (EE) pose. Therefore, for a given EE pose, several joint angle vectors can be associated. However, for a given EE pose, if a set of joint angles is parameterized, the IK problem of redundant manipulators can be reduced to that of non-redundant ones, such that the closed-form analytical methods developed for non-redundant manipulators can be applied to obtain the IK solution. In this paper, some redundant manipulator's joints are parameterized through workspace clustering and configuration space clustering of the redundant manipulator. The growing neural gas network (GNG) is used for workspace clustering while a neighborhood function (NF) is introduced in configuration space clustering. The results obtained by performing a series of simulations and experiments on redundant manipulators show the effectiveness of the proposed approach.

A.4 [Singh *et al.* 2018b]

I. Singh, Y. Amara, **A. Melingui**, P. Mani Pathak, and R. Merzouki,"Modeling of continuum manipulators using pythagorean hodograph curves," *Soft robotics*, vol. 5, no. 4, pp. 425–442, 2018.

Abstract: Research on continuum manipulators is increasingly developing in the context of bionic robotics because of their many advantages over conventional rigid manipulators. Due to their soft structure, they have inherent flexibility, which makes it a huge challenge to control them with high performances. Before elaborating a control strategy of such robots, it is essential to reconstruct first the behavior of the robot through development of an approximate behavioral model. This can be kinematic or dynamic depending on the conditions of operation of the robot itself. Kinematically, two types of modeling methods exist to describe the robot behavior; quantitative methods describe a model-based method, and qualitative methods describe a learning-based method. In kinematic modeling of continuum manipula-

tor, the assumption of constant curvature is often considered to simplify the model formulation. In this work, a quantitative modeling method is proposed, based on the Pythagorean hodograph (PH) curves. The aim is to obtain a three-dimensional reconstruction of the shape of the continuum manipulator with variable curvature, allowing the calculation of its inverse kinematic model (IKM). It is noticed that the performances of the PH-based kinematic modeling of continuum manipulators are considerable regarding position accuracy, shape reconstruction, and time/cost of the model calculation, than other kinematic modeling methods, for two cases: free load manipulation and variable load manipulation. This modeling method is applied to the compact bionic handling assistant (CBHA) manipulator for validation. The results are compared with other IKMs developed in case of CBHA manipulator.

A.5 [Bensekrane *et al.* 2020]

I. Bensekrane, P. Kumar, **A. Melingui**, V. Coelen, Y. Amara, T. Chettibi, and R. Merzouki, "Energy planning for autonomous driving of an overactuated road vehicle," I*EEE Transactions on Intelligent Transportation Systems*, 2020.

Abstract: In this work, an energy planning strategy is proposed for over-actuated unmanned road vehicles (URVs) having redundant steering configurations. In fact, indicators on the road geometry, the actuation redundancy, the optimal velocity profile, and the driving mode are evaluated for each segment of the URV's trajectory. To reach this objective, a power consumption estimation model is developed for the URV. Due to the presence of unknown dynamic parameters of the URV and uncertainties about its interaction with the environment, an artificial intelligence (AI) technique, based on data-learning qualitative method, is used for the power consumption estimation, namely Adaptive Neuro Fuzzy Inference System (ANFIS). The ANFIS model is obtained using trained data from a Real URV dynamics. Then, an energy digraph is built with all feasible configurations taking into account the kinematic and dynamic constraints based on a 3D grid map setup, according to velocity, arc-length, and driving mode. In this weighted directed graph, the edges describe the consumed energy by the URV along a segment of a trajectory. The vertices describe the start and end points of each segment. Subsequently, an optimization algorithm is applied on the digraph to get a global optimal solution combining driving mode, power consumption, and velocity profile of the URV. The obtained results are compared with the dynamic programming method for global offline optimization. Finally, the obtained simulation and experimental results, applied on RobuCar URV, highlight the effectiveness of the proposed energy planning.

A.6 [Ahanda *et al.* 2017]

J. J.-B. M. Ahanda, J. B. Mbede, **A. Melingui**, and B. Essimbi, "Robust adaptive control for robot manipulators: Support vector regression-based command filtered

adaptive backstepping approach," *Robotica*, vol. 36, no. 4, pp. 516–534, 2018.

Abstract: This study derives a robust adaptive control of electrically driven robot manipulators using a support vector regression (SVR)-based command filtered adaptive backstepping approach. The robot system is supposed to be subject to model uncertainties, neglected dynamics, and external disturbances. The command filtered backstepping algorithm is extended to the case of the robot manipulators. A robust term is added to the common adaptive SVR algorithm, to mitigate the effects of the SVR approximation error in the path tracking performance. The stability analysis of the closed loop system using the Lyapunov theory permits to highlight adaptation laws and to prove that all the signals in the closed loop system are bounded. Simulations show the effectiveness of the proposed control strategy.

A.7 [Ahanda *et al.* 2018b]

J. J.-B. M. Ahanda, J. B. Mbede, **A. Melingui**, and B. E. Zobo, "Robust adaptive command filtered control of a robotic manipulator with uncertain dynamic and joint space constraints," *Robotica*, vol. 36, no. 5, p. 767, 2018.

Abstract: The problem of robust adaptive control of a robotic manipulator subjected to uncertain dynamics and joint space constraints is addressed in this paper. Command filters are used to overcome the time derivatives of virtual control, thus reducing the need for desired trajectory differentiations. A barrier Lyapunov function is used to deal with the joint space constraints. A robust adaptive support vector regression architecture is used to reduce filtering errors, approximation errors and handle dynamic uncertainties. The stability analysis of the closed-loop system using the Lyapunov theory permits to highlight adaptation laws and to prove that all signals of the closed-loop system are bounded. Simulations show the effectiveness of the proposed control strategy.

A.8 [Roger *et al.* 2020]

R. Datouo, J. J.-B. M. Ahanda, **A. Melingui**, F. Biya-Motto, and B. E. Zobo, " Adaptive Fuzzy Finite-time Command Filtered Backstepping Control of Flexible-Joint Robots," *Robotica*, 2020.

Abstract: The problem of finite-time tracking control for n-link flexible-joint robot manipulators is addressed. An adaptive fuzzy finite-time command-filtered backstepping control scheme is presented to solve the following problems: "explosion of terms" problem, finite-time stabilization of the closed-loop system, and the reduction of computational cost. To this end, new virtual adaptive control signals and new finite-time error compensation mechanism are constructed using inherent properties of robot manipulator systems. Based on the Lyapunov theory, the finite-time stabilization of the closed-loop system is proved. Simulation studies show the effectiveness of the proposed method.

Bibliography

[Ahanda *et al.* 2017] Joseph Jean-Baptiste Mvogo Ahanda, Jean Bosco Mbede, Achille Melingui, Bernard Essimbi Zobo, Othman Lakhal and Rochdi Merzouki. *Robust control for robot manipulators: support vector regression based command filtered adaptive backstepping approach.* IFAC-PapersOnLine, vol. 50, no. 1, pages 8208–8213, 2017. (Cited on pages v, 12, 17 and 139.)

[Ahanda *et al.* 2018a] Joseph Jean-Baptiste Mvogo Ahanda, Jean Bosco Mbede, Achille Melingui and Bernard Essimbi. *Robust adaptive control for robot manipulators: Support vector regression-based command filtered adaptive backstepping approach.* Robotica, vol. 36, no. 4, pages 516–534, 2018. (Cited on pages v, 12 and 116.)

[Ahanda *et al.* 2018b] Joseph Jean-Baptiste Mvogo Ahanda, Jean Bosco Mbede, Achille Melingui and Bernard Essimbi Zobo. *Robust adaptive command filtered control of a robotic manipulator with uncertain dynamic and joint space constraints.* Robotica, vol. 36, no. 5, pages 767–786, 2018. (Cited on pages v, 12, 15, 17, 120 and 140.)

[Ahanda *et al.* 2018c] Joseph Jean-Baptiste Mvogo Ahanda, Jean Bosco Mbede, Achille Melingui and Bernard Essimbi Zobo. *Robust adaptive command filtered control of a robotic manipulator with uncertain dynamic and joint space constraints.* Robotica, vol. 36, no. 5, page 767, 2018. (Cited on pages 15 and 17.)

[Ananthanarayanan & Ordóñez 2015] Hariharan Ananthanarayanan and Raúl Ordóñez. *Real-time Inverse Kinematics of (2n+ 1) DOF hyper-redundant manipulator arm via a combined numerical and analytical approach.* Mechanism and Machine Theory, vol. 91, pages 209–226, 2015. (Cited on page 94.)

[Anor *et al.* 2011] Tomer Anor, Joseph R Madsen and Pierre Dupont. *Algorithms for design of continuum robots using the concentric tubes approach: a neurosurgical example.* In Robotics and Automation (ICRA), 2011 IEEE International Conference on, pages 667–673. IEEE, 2011. (Cited on page 81.)

[Astudillo *et al.* 2013] Leslie Astudillo, Patricia Melin and Oscar Castillo. *Chemical optimization paradigm applied to a fuzzy tracking controller for an autonomous mobile robot.* International Journal of Innovative Computing, Information and Control, vol. 9, no. 5, pages 2007–2018, 2013. (Cited on page 32.)

[Bakdi *et al.* 2017] Azzeddine Bakdi, Abdelfetah Hentout, Hakim Boutami, Abderraouf Maoudj, Ouarda Hachour and Brahim Bouzouia. *Optimal path planning and execution for mobile robots using genetic algorithm and adaptive*

fuzzy-logic control. Robotics and Autonomous Systems, vol. 89, pages 95–109, 2017. (Cited on page 31.)

[Bemporad 1998] Alberto Bemporad. *Reference governor for constrained nonlinear systems.* IEEE Transactions on Automatic Control, vol. 43, no. 3, pages 415–419, 1998. (Cited on page 121.)

[Bengio *et al.* 2009] Yoshua Bengio*et al. Learning deep architectures for AI.* Foundations and trends® in Machine Learning, vol. 2, no. 1, pages 1–127, 2009. (Cited on page 60.)

[Bensekrane *et al.* 2018] Ismail Bensekrane, Pushpendra Kumar, Achille Melingui, Vincent Coelen, Yacine Amara and Rochdi Merzouki. *Energy Planning for Unmanned Over-Actuated Road Vehicle.* In 2018 IEEE Vehicle Power and Propulsion Conference (VPPC), pages 1–6. IEEE, 2018. (Cited on pages 13, 16 and 18.)

[Bensekrane *et al.* 2020] Ismail Bensekrane, Pushpendra Kumar, Achille Melingui, Vincent Coelen, Yacine Amara, Taha Chettibi and Rochdi Merzouki. *Energy Planning for Autonomous Driving of an Over-Actuated Road Vehicle.* IEEE Transactions on Intelligent Transportation Systems, 2020. (Cited on pages v, 13, 16, 18, 131 and 139.)

[Beom & Cho 1995] Hee Rak Beom and Hyung Suck Cho. *A sensor-based navigation for a mobile robot using fuzzy logic and reinforcement learning.* IEEE transactions on Systems, Man, and Cybernetics, vol. 25, no. 3, pages 464–477, 1995. (Cited on page 31.)

[Bhat & Bernstein 2000] Sanjay P Bhat and Dennis S Bernstein. *Finite-time stability of continuous autonomous systems.* SIAM Journal on Control and Optimization, vol. 38, no. 3, pages 751–766, 2000. (Cited on page 125.)

[Bieze *et al.* 2018] Thor Morales Bieze, Frederick Largilliere, Alexandre Kruszewski, Zhongkai Zhang, Rochdi Merzouki and Christian Duriez. *Finite element method-based kinematics and closed-loop control of soft, continuum manipulators.* Soft robotics, vol. 5, no. 3, pages 348–364, 2018. (Cited on page 51.)

[Bócsi *et al.* 2011] Botond Bócsi, Duy Nguyen-Tuong, Lehel Csató, Bernhard Schoelkopf and Jan Peters. *Learning inverse kinematics with structured prediction.* In 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 698–703. IEEE, 2011. (Cited on page 95.)

[Bonin-Font *et al.* 2008] Francisco Bonin-Font, Alberto Ortiz and Gabriel Oliver. *Visual navigation for mobile robots: A survey.* Journal of intelligent and robotic systems, vol. 53, no. 3, page 263, 2008. (Cited on page 11.)

[bou] *Forward Kinematic Modeling of Conical-Shaped Continuum Manipulators.* (Cited on pages 18, 57, 60 and 132.)

[Bouyom Boutchouang *et al.* 2020] Audrey Hyacinthe Bouyom Boutchouang, Achille Melingui, Joseph Jean-Baptiste Mvogo Ahanda, Othman Lakhal, Frédéric Biya-Motto and Rochdi Merzouki. *Learning-Based approach to Inverse kinematics of Wheeled Mobile Continuum Manipulators.* Submited at IEEE/ASME Transactions on Mechatronics, pages 1–16, 2020. (Cited on pages 12, 17, 18 and 132.)

[Braganza *et al.* 2007] David Braganza, Darren M Dawson, Ian D Walker and Nitendra Nath. *A neural network controller for continuum robots.* Robotics, IEEE Transactions on, vol. 23, no. 6, pages 1270–1277, 2007. (Cited on pages 81 and 85.)

[Bürger & Guay 2010] Mathias Bürger and Martin Guay. *Robust constraint satisfaction for continuous-time nonlinear systems in strict feedback form.* IEEE transactions on automatic control, vol. 15, no. 11, pages 2597–2601, 2010. (Cited on page 120.)

[C. Bing & Chong 2017] L. Xiaoping C. Bing Z. Huaguang and L. Chong. *Neural observer and adaptive neural control design for a class of nonlinear systems.* IEEE transactions on neural networks and learning systems, vol. 29, no. 9, pages 4261–4271, 2017. (Cited on page 125.)

[Camarillo *et al.* 2008] David B Camarillo, Christopher F Milne, Christopher R Carlson, Michael R Zinn and J Kenneth Salisbury. *Mechanics modeling of tendon-driven continuum manipulators.* IEEE transactions on robotics, vol. 24, no. 6, pages 1262–1273, 2008. (Cited on pages 51 and 65.)

[Carpenter *et al.* 1992] Gail A Carpenter, Stephen Grossberg, Natalya Markuzon, John H Reynolds and David B Rosen. *Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps.* IEEE Transactions on neural networks, vol. 3, no. 5, pages 698–713, 1992. (Cited on page 71.)

[Casper & Murphy 2003] Jennifer Casper and Robin R. Murphy. *Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center.* IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 33, no. 3, pages 367–385, 2003. (Cited on page 51.)

[Chakrabarty *et al.* 2017] Ankush Chakrabarty, Vu Dinh, Martin J Corless, Ann E Rundell, Stanisław H Żak and Gregery T Buzzard. *Support vector machine informed explicit nonlinear model predictive control using low-discrepancy sequences.* IEEE Transactions on Automatic Control, vol. 62, no. 1, pages 135–148, 2017. (Cited on page 82.)

[Chang 1987] Pyung Chang. *A closed-form solution for inverse kinematics of robot manipulators with redundancy.* IEEE Journal on Robotics and Automation, vol. 3, no. 5, pages 393–403, 1987. (Cited on page 93.)

[Chen *et al.* 2010] Mou Chen, Shuzhi Sam Ge and Bernard Voon Ee How. *Robust adaptive neural network control for a class of uncertain MIMO nonlinear systems with input nonlinearities.* Neural Networks, IEEE Transactions on, vol. 21, no. 5, pages 796–812, 2010. (Cited on page 85.)

[Chiddarwar & Babu 2010] Shital S Chiddarwar and N Ramesh Babu. *Comparison of RBF and MLP neural networks to solve inverse kinematic problem for 6R serial robot by a fusion approach.* Engineering applications of artificial intelligence, vol. 23, no. 7, pages 1083–1092, 2010. (Cited on page 66.)

[Chien & Huang 2007a] Ming-Chih Chien and An-Chyau Huang. *Adaptive control for flexible-joint electrically driven robot with time-varying uncertainties.* IEEE Transactions on Industrial Electronics, vol. 54, no. 2, pages 1032–1038, 2007. (Cited on page 116.)

[Chien & Huang 2007b] Ming-Chih Chien and An-Chyau Huang. *Continuous control mode transitions for invariance control of constrained nonlinear systems.* Decision and Control, 2007 46th IEEE Conference on, pages 542–547, 2007. (Cited on page 120.)

[Chirikjian & Burdick 1994] Gregory S Chirikjian and Joel W Burdick. *A modal approach to hyper-redundant manipulator kinematics.* Robotics and Automation, IEEE Transactions on, vol. 10, no. 3, pages 343–354, 1994. (Cited on page 65.)

[Cui & Wu 2019] M. Cui and Z. Wu. *Trajectory tracking of flexible joint manipulators actuated by DC-motors under random disturbances.* Journal of the Franklin Institute, vol. 356, no. 16, pages 9330–9343, 2019. (Cited on page 133.)

[Daachi *et al.* 2012] Boubaker Daachi, Tarek Madani and Abdelaziz Benallegue. *Adaptive neural controller for redundant robot manipulators and collision avoidance with mobile obstacles.* Neurocomputing, vol. 79, pages 50–60, 2012. (Cited on page 95.)

[Daszykowski *et al.* 2002] Michael Daszykowski, Beata Walczak and Desire L Massart. *On the optimal partitioning of data with k-means, growing k-means, neural gas, and growing neural gas.* Journal of chemical information and computer sciences, vol. 42, no. 6, pages 1378–1389, 2002. (Cited on page 71.)

[Datouo *et al.* 2017] R Datouo, F Biya Motto, B Essimbi Zobo, Achille Melingui, Ismail Bensekrane and Rochdi Merzouki. *Optimal motion planning for minimizing energy consumption of wheeled mobile robots.* In 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO), pages 2179–2184. IEEE, 2017. (Cited on pages 12, 19 and 131.)

[de Lope *et al.* 2009] Javier de Lope, Matilde Santos*et al.* *A method to learn the inverse kinematics of multi-link robots by evolving neuro-controllers.* Neurocomputing, vol. 72, no. 13-15, pages 2806–2814, 2009. (Cited on page 95.)

[Dehaan & Guay 2005] Darryl Dehaan and Martin Guay. *Extremum-seeking control of state-constrained nonlinear systems.* Automatica, vol. 41, no. 9, pages 1567–1574, 2005. (Cited on page 121.)

[Dekkers & Aarts 1991] Anton Dekkers and Emile Aarts. *Global optimization and simulated annealing.* Mathematical programming, vol. 50, no. 1-3, pages 367–393, 1991. (Cited on page 62.)

[Ding *et al.* 2017] Lian Ding, Kerui Xia, Haibo Gao, Guangjun Liu and Zongquan Deng. *Robust adaptive control for door opening by a mobile rescue manipulator based on unknown-force-related constraints estimation.* Robotica, pages 1–22, 2017. (Cited on page 121.)

[Do 2010] Khac Duc Do. *Control of nonlinear systems with output tracking error constraints and its application to magnetic bearings.* International Journal of Control, vol. 83, no. 6, pages 1199–1216, 2010. (Cited on page 121.)

[Dupont *et al.* 2009] Pierre E Dupont, Jesse Lock, Brandon Itkowitz and Evan Butler. *Design and control of concentric-tube robots.* IEEE Transactions on Robotics, vol. 26, no. 2, pages 209–225, 2009. (Cited on page 51.)

[Eberhart & Shi 1998] Russell C Eberhart and Yuhui Shi. *Comparison between genetic algorithms and particle swarm optimization.* In International conference on evolutionary programming, pages 611–616. Springer, 1998. (Cited on page 63.)

[Escande *et al.* 2015] Coralie Escande, Taha Chettibi, Rochdi Merzouki, Vincent Coelen and Pushparaj Mani Pathak. *Kinematic calibration of a multisection bionic manipulator.* Mechatronics, IEEE/ASME Transactions on, vol. 20, no. 2, pages 663–674, 2015. (Cited on page 52.)

[Ethni *et al.* 2009] SA Ethni, B Zahawi, D Giaouris and PP Acarnley. *Comparison of particle swarm and simulated annealing algorithms for induction motor fault identification.* In 2009 7th IEEE International Conference on Industrial Informatics, pages 470–474. IEEE, 2009. (Cited on page 62.)

[Faisal *et al.* 2013] Mohammed Faisal, Ramdane Hedjar, Mansour Al Sulaiman and Khalid Al-Mutib. *Fuzzy logic navigation and obstacle avoidance by a mobile robot in an unknown dynamic environment.* International Journal of Advanced Robotic Systems, vol. 10, no. 1, page 37, 2013. (Cited on page 31.)

[Farrell *et al.* 2009] Jay A Farrell, Marios Polycarpou, Manu Sharma and Wenjie Dong. *Command filtered backstepping.* IEEE Transactions on Automatic Control, vol. 54, no. 6, pages 1391–1395, 2009. (Cited on pages 110 and 115.)

[Fateh & Khorashadizadeh 2016] Mohammad Mehdi Fateh and Saeed Khorashadizadeh. *Robust control of electrically driven robots by adaptive fuzzy estimation of uncertainty.* Nonlinear Dynamics, vol. 69, no. 3, pages 759–770, 2016. (Cited on page 116.)

[Fatemizadeh *et al.* 2003] Emad Fatemizadeh, Caro Lucas and Hamid Soltanian-Zadeh. *Automatic landmark extraction from image data using modified growing neural gas network.* IEEE Transactions on Information Technology in Biomedicine, vol. 7, no. 2, pages 77–85, 2003. (Cited on page 71.)

[Fletcher 2013] Roger Fletcher. Practical methods of optimization. John Wiley & Sons, 2013. (Cited on page 81.)

[Fritzke 1994] Bernd Fritzke. *Growing cell structures—a self-organizing network for unsupervised and supervised learning.* Neural networks, vol. 7, no. 9, pages 1441–1460, 1994. (Cited on pages 71 and 95.)

[Fritzke 1995] Bernd Fritzke. *A growing neural gas network learns topologies.* In Advances in neural information processing systems, pages 625–632, 1995. (Cited on pages 69 and 70.)

[Furao & Hasegawa 2006] Shen Furao and Osamu Hasegawa. *An incremental network for on-line unsupervised classification and topology learning.* Neural networks, vol. 19, no. 1, pages 90–106, 2006. (Cited on page 71.)

[Galicki 2015] Mirosław Galicki. *Finite-time control of robotic manipulators.* Automatica, vol. 51, pages 49–54, 2015. (Cited on page 125.)

[Ge & Wang 2004] Shuzhi Sam Ge and Cong Wang. *Adaptive neural control of uncertain MIMO nonlinear systems.* Neural Networks, IEEE Transactions on, vol. 15, no. 3, pages 674–692, 2004. (Cited on page 85.)

[Gilbert & Kolmanovsky 2002] Elmer Gilbert and Ilya Kolmanovsky. *Nonlinear tracking control in the presence of state and control constraints: a generalized reference governor.* Automatica, vol. 38, no. 12, pages 2063–2073, 2002. (Cited on page 121.)

[Giorelli *et al.* 2013] M Giorelli, F Renda, G Ferri and C Laschi. *A Feed-Forward Neural Network Learning the Inverse Kinetics of a Soft Cable-Driven Manipulator Moving in Three-Dimensional Space.* in proc. IEEE Int. Conf. on Intelli-gent Robots and Systems, pages 5033–5039, 2013. (Cited on page 80.)

[Giorelli *et al.* 2015] Michele Giorelli, Federico Renda, Marcello Calisti, Andrea Arienti, Gabriele Ferri and Cecilia Laschi. *Neural network and jacobian method for solving the inverse statics of a cable-driven soft arm with nonconstant curvature.* IEEE Transactions on Robotics, vol. 31, no. 4, pages 823–834, 2015. (Cited on pages 51 and 66.)

[Godage *et al.* 2011a] Isuru S Godage, David T Branson, Emanuele Guglielmino, Gustavo A Medrano-Cerda and Darwin G Caldwell. *Shape function-based kinematics and dynamics for variable length continuum robotic arms.* In Robotics and Automation (ICRA), 2011 IEEE International Conference on, pages 452–457. IEEE, 2011. (Cited on pages 53 and 54.)

[Godage *et al.* 2011b] Isuru S Godage, Emanuele Guglielmino, David T Branson, Gustavo A Medrano-Cerda and Darwin G Caldwell. *Novel modal approach for kinematics of multisection continuum arms.* In Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, pages 1093–1098. IEEE, 2011. (Cited on pages 52 and 54.)

[Godage *et al.* 2015] Isuru S Godage, Gustavo A Medrano-Cerda, David T Branson, Emanuele Guglielmino and Darwin G Caldwell. *Modal kinematics for multisection continuum arms.* Bioinspiration & biomimetics, vol. 10, no. 3, page 035002, 2015. (Cited on pages 51, 52, 65 and 74.)

[Gravagne & Walker 2002] Ian A Gravagne and Ian D Walker. *Manipulability, force, and compliance analysis for planar continuum manipulators.* IEEE Transactions on Robotics and Automation, vol. 18, no. 3, pages 263–273, 2002. (Cited on page 51.)

[Grüne & Pannek 2007] Lars Grüne and Jürgen Pannek. *Continuous control mode transitions for invariance control of constrained nonlinear systems.* Nonlinear model predictive control, pages 43–66, 2007. (Cited on page 121.)

[Guilamo *et al.* 2005] Luis Guilamo, James Kuffner, Koichi Nishiwaki and Satoshi Kagami. *Efficient prioritized inverse kinematic solutions for redundant manipulators.* In 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 3921–3926. IEEE, 2005. (Cited on page 65.)

[Hagras *et al.* 2004] Hani Hagras, Victor Callaghan and Martin Colley. *Learning and adaptation of an intelligent mobile robot navigator operating in unstructured environment based on a novel online Fuzzy–Genetic system.* Fuzzy Sets and Systems, vol. 141, no. 1, pages 107–160, 2004. (Cited on page 32.)

[Hagras 2004] Hani A Hagras. *A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots.* IEEE Transactions on Fuzzy systems, vol. 12, no. 4, pages 524–539, 2004. (Cited on pages 31 and 32.)

[Han *et al.* 2018] Yao Han, Jinpeng Yu, Lin Zhao, Haisheng Yu and Chong Lin. *Finite-time adaptive fuzzy control for induction motors with input saturation based on command filtering.* IET Control Theory & Applications, vol. 12, no. 15, pages 2148–2155, 2018. (Cited on page 125.)

[Hartenberg & Denavit 1955] Richard S Hartenberg and Jacques Denavit. *A kinematic notation for lower pair mechanisms based on matrices.* Journal of applied mechanics, vol. 77, no. 2, pages 215–221, 1955. (Cited on page 96.)

[He *et al.* 2016a] Wei He, Amoateng Ofosu David, Zhao Yin and Changyin Sun. *Neural network control of a robotic manipulator with input deadzone and output constraint.* IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 46, no. 6, pages 759–770, 2016. (Cited on page 117.)

[He *et al.* 2016b] Wei He, Yiting Dong and Changyin Sun. *Adaptive neural impedance control of a robotic manipulator with input saturation.* IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 46, no. 3, pages 334–344, 2016. (Cited on page 85.)

[Hollerbach 1985] John M Hollerbach. *Optimum kinematic design for a seven degree of freedom manipulator.* In Robotics research: The second international symposium, pages 215–222. Cambridge, MIT Press, 1985. (Cited on pages 65 and 94.)

[Hornik *et al.* 1989] Kurt Hornik, Maxwell Stinchcombe and Halbert White. *Multilayer feedforward networks are universal approximators.* Neural networks, vol. 2, no. 5, pages 359–366, 1989. (Cited on page 61.)

[Hu & Woo 2006] Hui Hu and Peng-Yung Woo. *Fuzzy supervisory sliding-mode and neural-network control for robotic manipulators.* IEEE Transactions on Industrial Electronics, vol. 53, no. 3, pages 929–940, 2006. (Cited on page 95.)

[Hu *et al.* 2018] Zheng Hu, Yongping Li and Zhiyong Yang. *Improving Convolutional Neural Network Using Pseudo Derivative ReLU.* In 2018 5th International Conference on Systems and Informatics (ICSAI), pages 283–287. IEEE, 2018. (Cited on page 61.)

[Huang & Chen 2004] An-Chyau Huang and Yuan-Chih Chen. *Adaptive sliding control for single-link flexible-joint robot with mismatched uncertainties.* IEEE Transactions on Control Systems Technology, vol. 12, no. 5, pages 770–775, 2004. (Cited on pages 109, 110 and 116.)

[Huang *et al.* 2017] Jian Huang, MyongHyok Ri, Dongrui Wu and Songhyok Ri. *Interval type-2 fuzzy logic modeling and control of a mobile two-wheeled inverted pendulum.* IEEE Transactions on Fuzzy Systems, vol. 26, no. 4, pages 2030–2038, 2017. (Cited on page 32.)

[Hwang & Kim 2006] Jae Pil Hwang and Euntai Kim. *Robust tracking control of an electrically driven robot: adaptive fuzzy logic approach.* IEEE Transactions on Fuzzy systems, vol. 14, no. 2, pages 232–247, 2006. (Cited on page 110.)

[Ivlev & Gräser 1998] Oleg Ivlev and Axel Gräser. *Resolving redundancy of series kinematic chains through imaginary links.* In CESA, volume 98, pages 1–4, 1998. (Cited on page 93.)

[Jin *et al.* 2016] Maolin Jin, Jinoh Lee and Nikolaos G Tsagarakis. *Model-free robust adaptive control of humanoid robots with flexible joints.* IEEE Transactions

on Industrial Electronics, vol. 64, no. 2, pages 1706–1715, 2016. (Cited on page 110.)

[Jin 2000] Yaochu Jin. *Fuzzy modeling of high-dimensional systems: complexity reduction and interpretability improvement.* IEEE Transactions on Fuzzy Systems, vol. 8, no. 2, pages 212–221, 2000. (Cited on page 33.)

[Jiokou K *et al.* 2020] GA Jiokou K, Achille Melingui, Martin Kom and Rochdi Merzouki. *A learning framework to inverse kinematics of redundant manipulators.* accepted to 21th World Congress of the Int. Federation of Aut. Control, berlin, Germany, 11-17 Jully 2014, 2020. (Cited on pages 12, 17 and 18.)

[Jones & Walker 2006] Bryan A Jones and Ian D Walker. *Kinematics for multisection continuum robots.* Robotics, IEEE Transactions on, vol. 22, no. 1, pages 43–55, 2006. (Cited on pages 51 and 65.)

[Jordan & Rumelhart 1992] Michael I Jordan and David E Rumelhart. *Forward models: Supervised learning with a distal teacher.* Cognitive science, vol. 16, no. 3, pages 307–354, 1992. (Cited on pages 66, 67 and 68.)

[Juang & Hsu 2009] Chia-Feng Juang and Chia-Hung Hsu. *Reinforcement ant optimized fuzzy controller for mobile-robot wall-following control.* IEEE Transactions on Industrial Electronics, vol. 56, no. 10, pages 3931–3940, 2009. (Cited on page 32.)

[Juang & Juang 2012] Chia-Feng Juang and Kai-Jie Juang. *Reduced interval type-2 neural fuzzy system using weighted bound-set boundary operation for computation speedup and chip implementation.* IEEE Transactions on Fuzzy Systems, vol. 21, no. 3, pages 477–491, 2012. (Cited on page 33.)

[Junratanasiri *et al.* 2011] Sittichok Junratanasiri, Sansanee Auephanwiriyakul and Nipon Theera-Umpon. *Navigation system of mobile robot in an uncertain environment using type-2 fuzzy modelling.* In 2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011), pages 1171–1178. IEEE, 2011. (Cited on page 32.)

[Kapadia & Walker 2013] Apoorva D Kapadia and Ian D Walker. *Self-motion analysis of extensible continuum manipulators.* In Robotics and Automation (ICRA), 2013 IEEE International Conference on, pages 1988–1994. IEEE, 2013. (Cited on page 66.)

[Karray *et al.* 2016] Amal Karray, Malek Njah, Moez Feki and Mohamed Jallouli. *Intelligent mobile manipulator navigation using hybrid adaptive-fuzzy controller.* Computers & Electrical Engineering, vol. 56, pages 773–783, 2016. (Cited on pages 31 and 32.)

[Kaski & Lagus 1996] Samuel Kaski and Krista Lagus. *Comparing self-organizing maps.* In International Conference on Artificial Neural Networks, pages 809–814. Springer, 1996. (Cited on pages 74 and 99.)

[Khorashadizadeh & Sadeghijaleh 2018] Saeed Khorashadizadeh and Mahdi Sadeghijaleh. *Adaptive fuzzy tracking control of robot manipulators actuated by permanent magnet synchronous motors.* Computers & Electrical Engineering, vol. 72, pages 100–111, 2018. (Cited on page 110.)

[Kim & Chung 2015] Min Jun Kim and Wan Kyun Chung. *Disturbance-observer-based PD control of flexible joint robots for asymptotic convergence.* IEEE Transactions on Robotics, vol. 31, no. 6, pages 1508–1516, 2015. (Cited on page 109.)

[Kim & Chwa 2014] Cheol-Joong Kim and Dongkyoung Chwa. *Obstacle avoidance method for wheeled mobile robots using interval type-2 fuzzy neural network.* IEEE Transactions on Fuzzy Systems, vol. 23, no. 3, pages 677–687, 2014. (Cited on pages 31 and 33.)

[Kim *et al.* 2019] Min Jun Kim, Fabian Beck, Christian Ott and Alin Albu-Schäffer. *Model-Free Friction Observers for Flexible Joint Robots With Torque Measurements.* IEEE Transactions on Robotics, 2019. (Cited on page 110.)

[Kim 2017] Joonyoung Kim. *Two-time scale control of flexible joint robots with an improved slow model.* IEEE Transactions on Industrial Electronics, vol. 65, no. 4, pages 3317–3325, 2017. (Cited on page 109.)

[Klein & Huang 1983] Charles A Klein and Ching-Hsiang Huang. *Review of pseudoinverse control for use with kinematically redundant manipulators.* IEEE Transactions on Systems, Man, and Cybernetics, no. 2, pages 245–250, 1983. (Cited on pages 65 and 94.)

[Klein *et al.* 1995] Charles A Klein, Caroline Chu-Jenq and Shamim Ahmed. *A new formulation of the extended Jacobian method and its use in mapping algorithmic singularities for kinematically redundant manipulators.* IEEE Transactions on Robotics and Automation, vol. 11, no. 1, pages 50–55, 1995. (Cited on pages 65 and 94.)

[Kohonen 1990] Teuvo Kohonen. *The self-organizing map.* Proceedings of the IEEE, vol. 78, no. 9, pages 1464–1480, 1990. (Cited on page 95.)

[Kohonen 1998] Teuvo Kohonen. *The self-organizing map.* Neurocomputing, vol. 21, no. 1-3, pages 1–6, 1998. (Cited on page 69.)

[Kouabon *et al.* 2020a] AG Jiokou Kouabon, A Melingui, JJB Mvogo Ahanda, O Lakhal, V Coelen, M KOM and R Merzouki. *A Learning Framework to inverse kinematics of high DOF redundant manipulators.* Mechanism and

Machine Theory, vol. 153, page 103978, 2020. (Cited on pages v, 12, 17, 18, 69 and 138.)

[Kouabon *et al.* 2020b] AG Jiokou Kouabon, A Melingui, JJB Mvogo Ahanda, O Lakhal, M KOM and R Merzouki. *A Learning Framework to Inverse Kinematics of High DOF Redundant Mobile Manipulators*. submitted to Robotics and Autonomous Systems, 2020. (Cited on pages 12, 17, 18 and 132.)

[Krstic & Bement 2006] Miroslav Krstic and Matt Bement. *Nonovershooting control of strict-feedback nonlinear systems*. IEEE Transactions on Automatic Control, vol. 51, no. 12, pages 1938–1943, 2006. (Cited on page 121.)

[Kumar *et al.* 2010] Swagat Kumar, P Premkumar, Ashish Dutta and Laxmidhar Behera. *Visual motor control of a 7DOF redundant manipulator using redundancy preserving learning network*. Robotica, vol. 28, no. 6, pages 795–810, 2010. (Cited on pages 69, 72, 95 and 97.)

[Lakhal *et al.* 2014a] Othman Lakhal, Achille Melingui, Thor Morales Bieze, Coralie Escande, Blaise Conrard and Rochdi Merzouki. *On the kinematic modeling of a class of continuum manipulators*. In 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014), pages 368–373. IEEE, 2014. (Cited on pages 14 and 18.)

[Lakhal *et al.* 2014b] Othman Lakhal, Achille Melingui, Abdelhakim Chibani, Coralie Escande and Rochdi Merzouki. *Inverse kinematic modeling of a class of continuum bionic handling arm*. In 2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pages 1337–1342. IEEE, 2014. (Cited on page 18.)

[Lakhal *et al.* 2014c] Othman Lakhal, Achille Melingui, Thor Morales, Coralie Escande and Rochdi Merzouki. *Forward Kinematic of a class of Continuum Bionic Handling Arm*. 2014 IEEE Multi-Conference on Systems and Control, 2014. (Cited on page 18.)

[Lakhal *et al.* 2015] Othman Lakhal, Achille Melingui and Rochdi Merzouki. *Hybrid approach for modeling and solving of kinematics of a compact bionic handling assistant manipulator*. IEEE/ASME Transactions on Mechatronics, vol. 21, no. 3, pages 1326–1335, 2015. (Cited on pages v, 12, 13, 14, 18, 57, 60, 66, 80, 81, 85 and 137.)

[Lakhal *et al.* 2019] Othman Lakhal, Achille Melingui, Gérald Dherbomez and Rochdi Merzouki. *Control of a Hyper-Redundant Robot for Quality Inspection in Additive Manufacturing for Construction*. In 2019 2nd IEEE International Conference on Soft Robotics (RoboSoft), pages 448–453. IEEE, 2019. (Cited on pages 13, 14 and 18.)

[Larochelle *et al.* 2009] Hugo Larochelle, Yoshua Bengio, Jérôme Louradour and Pascal Lamblin. *Exploring strategies for training deep neural networks.* Journal of machine learning research, vol. 10, no. Jan, pages 1–40, 2009. (Cited on page 61.)

[Liegeois 1977] Alain Liegeois. *Automatic supervisory control of the configuration and behaviour of multibody mechanisms.* IEEE Transactions on systems, man and cybernetics, vol. 7, no. 12, pages 868–871, 1977. (Cited on page 94.)

[Lin *et al.* 2013] Yang-Yin Lin, Shih-Hui Liao, Jyh-Yeong Chang and Chin-Teng Lin. *Simplified interval type-2 fuzzy neural networks.* IEEE transactions on neural networks and learning systems, vol. 25, no. 5, pages 959–969, 2013. (Cited on pages 33 and 38.)

[Ling *et al.* 2019] Song Ling, Huanqing Wang and Peter X Liu. *Adaptive fuzzy dynamic surface control of flexible-joint robot systems with input saturation.* IEEE/CAA Journal of Automatica Sinica, vol. 6, no. 1, pages 97–107, 2019. (Cited on page 127.)

[Liu & Mendel 2011] Xinwang Liu and Jerry M Mendel. *Connect Karnik-Mendel algorithms to root-finding for computing the centroid of an interval type-2 fuzzy set.* IEEE Transactions on Fuzzy Systems, vol. 19, no. 4, pages 652–665, 2011. (Cited on page 33.)

[Long & Nan 2015] Mai Thang Long and Wang Yao Nan. *Adaptive Position Tracking System and Force Control Strategy for Mobile Robot Manipulators Using Fuzzy Wavelet Neural Networks.* Journal of Intelligent & Robotic Systems, vol. 79, no. 2, page 175, 2015. (Cited on page 85.)

[Lu *et al.* 2017] Zhao Lu, Jing Sun and Kenneth Butts. *Multiscale Support Vector Learning With Projection Operator Wavelet Kernel for Nonlinear Dynamical System Identification.* IEEE transactions on neural networks and learning systems, vol. 28, no. 1, pages 231–243, 2017. (Cited on page 82.)

[M. Li & Guangdeng 2019] Z. Xudong M. Li H. Xin and Z. Guangdeng. *Adaptive fuzzy tracking control for a class of uncertain switched nonlinear systems with multiple constraints: a small-gain approach.* International Journal of Fuzzy Systems, vol. 21, no. 8, pages 2609–2624, 2019. (Cited on page 125.)

[M.-Y. Cui & Xie 2014] Z.-J. Wu M.-Y. Cui and X.-J. Xie. *Output feedback tracking control of stochastic Lagrangian systems and its application.* Automatica, vol. 50, no. 5, pages 1424–1433, 2014. (Cited on page 133.)

[Magni *et al.* 2009] Lalo Magni, Davide Martino Raimondo and Frank Allgöwer. *Nonlinear model predictive control.* Decision and Control, 2007 46th IEEE Conference on, 2009. (Cited on page 121.)

[Mahamat L *et al.* 2020] Imrane Mahamat L, AH Bouyom B, Achille Melingui, Frédéric Biya M and Rochdi Merzouki. *Learning-Based Approaches for Forward Kinematic Modeling of Continuum Manipulators.* accepted to 21th World Congress of the Int. Federation of Aut. Control, berlin, Germany, 11-17 Jully 2014, 2020. (Cited on pages 19 and 54.)

[Mahamat Loufti *et al.* 2020] Imrane Mahamat Loufti, Achille Melingui, Joseph Jean-Baptiste Mvogo Ahanda, Frédéric Biya-Motto and Rochdi Merzouki. *Artificial Potential Field Neurofuzzy Controller for Autonomous Navigation of Mobile Robots.* Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, pages 1–16, 2020. (Cited on pages 11 and 19.)

[Mahl *et al.* 2014] Tobias Mahl, Andreas Hildebrandt and Oliver Sawodny. *A variable curvature continuum kinematics for kinematic control of the bionic handling assistant.* Robotics, IEEE Transactions on, vol. 30, no. 4, pages 935–949, 2014. (Cited on pages 51, 65, 81 and 85.)

[Maldonado *et al.* 2013] Yazmin Maldonado, Oscar Castillo and Patricia Melin. *Particle swarm optimization of interval type-2 fuzzy systems for FPGA applications.* Applied Soft Computing, vol. 13, no. 1, pages 496–508, 2013. (Cited on page 32.)

[Martinetz *et al.* 1990] Thomas M Martinetz, Helge J Ritter and Klaus J Schulten. *Three-dimensional neural net for learning visuomotor coordination of a robot arm.* IEEE Transactions on Neural Networks, vol. 1, no. 1, pages 131–136, 1990. (Cited on pages 69 and 71.)

[Martinetz *et al.* 1991] Thomas Martinetz, Klaus Schulten*et al. A" neural-gas" network learns topologies.* 1991. (Cited on page 95.)

[Masmoudi *et al.* 2016] Mohamed Slim Masmoudi, Najla Krichen, Mohamed Masmoudi and Nabil Derbel. *Fuzzy logic controllers design for omnidirectional mobile robot navigation.* Applied soft computing, vol. 49, pages 901–919, 2016. (Cited on page 31.)

[Matthias *et al.* 2011] Bjoern Matthias, Soenke Kock, Henrik Jerregard, Mats Kallman, Ivan Lundberg and Roger Mellander. *Safety of collaborative industrial robots: Certification possibilities for a collaborative assembly robot concept.* In 2011 IEEE International Symposium on Assembly and Manufacturing (ISAM), pages 1–6. Ieee, 2011. (Cited on page 93.)

[Mbede *et al.* 2012] JB Mbede, A Melingui, B Essimbi Zobo, R Merzouki and BO Bouamama. *zSlices based type-2 fuzzy motion control for autonomous robotino mobile robot.* In Mechatronics and Embedded Systems and Applications (MESA), 2012 IEEE/ASME International Conference on, pages 63–68. IEEE, 2012. (Cited on pages 11 and 31.)

[Medzo Aba *et al.* 2020] Charles Medzo Aba, Joseph Jean-Baptiste Mvogo Ahanda, Achille Melingui and Bernard Essimbi Zobo. *Command filtering approach for robust control of electrically driven flexible joint robot manipulators under random disturbances.* sumbitted to Robotica, pages 1–14, 2020. (Cited on pages 15 and 19.)

[Melin *et al.* 2013] Patricia Melin, Leslie Astudillo, Oscar Castillo, Fevrier Valdez and Mario Garcia. *Optimal design of type-2 and type-1 fuzzy tracking controllers for autonomous mobile robots under perturbed torques using a new chemical optimization paradigm.* Expert Systems with Applications, vol. 40, no. 8, pages 3185–3195, 2013. (Cited on page 32.)

[Melingui *et al.* 2013a] Achille Melingui, Taha Chettibi, Rochdi Merzouki and Jean Bosco Mbede. *Adaptive navigation of an omni-drive autonomous mobile robot in unstructured dynamic environments.* In 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO), pages 1924–1929. IEEE, 2013. (Cited on pages 10, 14, 33, 34 and 35.)

[Melingui *et al.* 2013b] Achille Melingui, Coralie Escande, Benoudjit Nabil, Rochdi Merzouki and JB Mbede. *Qualitative approach for forward kinematic modeling of a Compact Bionic Handling Assistant trunk.* 19th World Congress of the Int. Federation of Aut. Control, Cape Town, South Africa, 24-29 August 2014, 2013. (Cited on page 68.)

[Melingui *et al.* 2014a] A Melingui, R Merzouki and JB Mbede. *Compact bionic handling arm control using neural networks.* Electronics Letters, vol. 50, no. 14, page 979, 2014. (Cited on pages 11, 14, 81, 82 and 85.)

[Melingui *et al.* 2014b] A Melingui, R Merzouki, JB Mbede, C Escande, B Daachi and N Benoudjit. *Qualitative approach for inverse kinematics modeling of a Compact Bionic Handling Assistant trunk.* 2014 IEEE WCCI: International Joint Conference on Neural Networks, Beijing, China, 6-11 July 2014. (Cited on pages 14 and 68.)

[Melingui *et al.* 2014c] Achille Melingui, Coralie Escande, Nabil Benoudjit, Rochdi Merzouki and Jean Bosco Mbede. *Qualitative approach for forward kinematic modeling of a compact bionic handling assistant trunk.* IFAC Proceedings Volumes, vol. 47, no. 3, pages 9353–9358, 2014. (Cited on pages 10, 11, 12, 51 and 88.)

[Melingui *et al.* 2014d] Achille Melingui, Rochdi Merzouki and Jean Bosco Mbede. *Neuro-fuzzy controller for autonomous navigation of mobile robots.* In 2014 IEEE Conference on Control Applications (CCA), pages 1052–1057. IEEE, 2014. (Cited on pages 11 and 14.)

[Melingui *et al.* 2014e] Achille Melingui, Rochdi Merzouki, Jean Bosco Mbede and Taha Chettibi. *A novel approach to integrate artificial potential field and*

fuzzy logic into a common framework for robots autonomous navigation. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, vol. 228, no. 10, pages 787–801, 2014. (Cited on pages 10, 11, 14, 31, 33, 34, 35, 36, 43, 45 and 48.)

[Melingui *et al.* 2014f] Achille Melingui, Rochdi Merzouki, Jean Bosco Mbede, Coralie Escande and Nabil Benoudjit. *Neural networks based approach for inverse kinematic modeling of a compact bionic handling assistant trunk*. In 2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE), pages 1239–1244. IEEE, 2014. (Cited on pages 10, 11 and 14.)

[Melingui *et al.* 2014g] Achille Melingui, Rochdi Merzouki, Jean Bosco Mbede, Coralie Escande, Boubaker Daachi and Nabil Benoudjit. *Qualitative approach for inverse kinematic modeling of a compact bionic handling assistant trunk*. In 2014 International Joint Conference on Neural Networks (IJCNN), pages 754–761. IEEE, 2014. (Cited on pages 10, 11 and 12.)

[Melingui *et al.* 2015] Achille Melingui, Othman Lakhal, Boubaker Daachi, Jean Bosco Mbede and Rochdi Merzouki. *Adaptive neural network control of a compact bionic handling arm*. IEEE/ASME Transactions on Mechatronics, vol. 20, no. 6, pages 2862–2875, 2015. (Cited on pages 11, 12, 14, 51, 66, 77, 80, 81, 82, 85, 88, 89 and 92.)

[Melingui *et al.* 2017a] A Melingui, JJB Mvogo Ahanda, O Lakhal, JB Mbede*et al.* *Performance Improvement of a Class of Continuum Manipulators via Adaptive Algorithms*. IFAC-PapersOnLine, vol. 50, no. 1, pages 4863–4868, 2017. (Cited on pages 12, 14, 15 and 17.)

[Melingui *et al.* 2017b] Achille Melingui, Joseph Jean-Baptiste Mvogo Ahanda, Othman Lakhal, Jean Bosco Mbede and Rochdi Merzouki. *Adaptive algorithms for performance improvement of a class of continuum manipulators*. IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 48, no. 9, pages 1531–1541, 2017. (Cited on pages v, 12, 14, 15, 17, 51, 66, 77, 80 and 137.)

[Melingui *et al.* 2019] Achille Melingui, Audrey Hyacinthe Bouyom Boutchouang, Joseph Jean-Baptiste Mvogo Ahanda, Othman Lakhal, Taha Chettibi and Rochdi Merzouki. *A Learning-Based Approach to Inverse Kinematics of Continuum Manipulators*. submitted to Systems, Man and Cybernetics, IEEE Transactions on, 2019. (Cited on pages 12, 17 and 18.)

[Melingui 2014] Achille Melingui. *Modeling and control of a class of mobile omnidrive: Continuum manipulator robots, case of study Robotino XT*. PhD thesis, Lille 1, 2014. (Cited on pages 11 and 12.)

[Mendel 2017] Jerry M Mendel. *Uncertain rule-based fuzzy systems*. In Introduction and new directions, page 684. Springer, 2017. (Cited on pages 33 and 38.)

[Mendel 2019] Jerry M Mendel. *Type-2 fuzzy sets as well as computing with words.* IEEE Computational Intelligence Magazine, vol. 14, no. 1, pages 82–95, 2019. (Cited on page 32.)

[Méndez & De Los Angeles Hernández 2013] Gerardo M Méndez and Maria De Los Angeles Hernández. *Hybrid learning mechanism for interval A2-C1 type-2 non-singleton type-2 Takagi–Sugeno–Kang fuzzy logic systems.* Information Sciences, vol. 220, pages 149–169, 2013. (Cited on page 32.)

[Mu *et al.* 2018] Zonggao Mu, Han Yuan, Wenfu Xu, Tianliang Liu and Bin Liang. *A segmented geometry method for kinematics and configuration planning of spatial hyper-redundant manipulators.* IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2018. (Cited on page 94.)

[Nagatani & Abe 2007] Takashi Nagatani and Shigeo Abe. *Backward varilable selection of support vector regressors by block deletion.* In 2007 International Joint Conference on Neural Networks, pages 2117–2122. IEEE, 2007. (Cited on page 87.)

[Nagatani *et al.* 2010] Takashi Nagatani, Seiichi Ozawa, Shigeo Abe*et al. Fast variable selection by block addition and block deletion.* Journal of Intelligent Learning Systems and Applications, vol. 2, no. 04, page 200, 2010. (Cited on page 84.)

[Neppalli *et al.* 2009] Srinivas Neppalli, Matthew A Csencsits, Bryan A Jones and Ian D Walker. *Closed-form inverse kinematics for continuum manipulators.* Advanced Robotics, vol. 23, no. 15, pages 2077–2091, 2009. (Cited on page 65.)

[Nguyen *et al.* 2019] Van-Truong Nguyen, Chyi-Yeu Lin, Shun-Feng Su and Wei Sun. *Finite-Time Adaptive Fuzzy Tracking Control Design for Parallel Manipulators with Unbounded Uncertainties.* International Journal of Fuzzy Systems, vol. 21, no. 2, pages 545–555, 2019. (Cited on page 125.)

[Niu & Zhao 2013] Ben Niu and Jun Zhao. *Barrier Lyapunov functions for the output tracking control of constrained nonlinear switched systems.* Systems & Control Letters, vol. 62, no. 10, pages 963–971, 2013. (Cited on page 121.)

[Oh & Lee 1999] Jong H Oh and Jin S Lee. *Control of flexible joint robot system by backstepping design approach.* Intelligent Automation & Soft Computing, vol. 5, no. 4, pages 267–278, 1999. (Cited on page 110.)

[Oltean *et al.* 2010] SE Oltean, M Dulău and R Puskas. *Position control of Robotino mobile robot using fuzzy logic.* In 2010 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR), volume 1, pages 1–6. IEEE, 2010. (Cited on page 34.)

[OptiTrack 2018] OptiTrack. *Motion Capture Systems*, 2018. visited on 06 December 2018. (Cited on page 55.)

[Ott *et al.* 2006] Ch Ott, Oliver Eiberger, Werner Friedl, B Bauml, Ulrich Hillenbrand, Ch Borst, Alin Albu-Schaffer, Bernhard Brunner, H Hirschmuller, S Kielhofer*et al.* *A humanoid two-arm system for dexterous manipulation.* In 2006 6th IEEE-RAS International Conference on Humanoid Robots, pages 276–283. IEEE, 2006. (Cited on page 93.)

[Ott *et al.* 2008] Christian Ott, Alin Albu-Schaffer, Andreas Kugi and Gerd Hirzinger. *On the passivity-based impedance control of flexible joint robots.* IEEE Transactions on Robotics, vol. 24, no. 2, pages 416–429, 2008. (Cited on page 109.)

[Pandey *et al.* 2017] Anish Pandey, Shalini Pandey and Preeti Gupta. *Intelligent navigation and control of a mobile robot in static and dynamic environments using hybrid fuzzy architecture.* International Journal of Autonomic Computing, vol. 2, no. 3, pages 255–281, 2017. (Cited on page 31.)

[Parhi 2005] Dayal R Parhi. *Navigation of mobile robots using a fuzzy logic controller.* Journal of intelligent and robotic systems, vol. 42, no. 3, pages 253–273, 2005. (Cited on page 31.)

[Patle *et al.* 2019] BK Patle, Anish Pandey, DRK Parhi, A Jagadeesh*et al.* *A review: On path planning strategies for navigation of mobile robot.* Defence Technology, 2019. (Cited on page 31.)

[Paul 1981] Richard P Paul. Robot manipulators: mathematics, programming, and control: the computer control of robot manipulators. Richard Paul, 1981. (Cited on page 93.)

[Peiper 1968] Donald L Peiper. *The kinematics of manipulators under computer control.* Technical report, Stanford Univ Ca Dept of computer science, 1968. (Cited on pages 65 and 93.)

[Pérez-Cruz *et al.* 2001] Fernando Pérez-Cruz, Pedro Luis Alarcón-Diana, Angel Navia-Vázquez and Antonio Artés-Rodríguez. *Fast Training of Support Vector Classifiers.* In Advances in Neural Information Processing Systems, pages 734–740, 2001. (Cited on page 84.)

[Poli *et al.* 2007] Riccardo Poli, James Kennedy and Tim Blackwell. *Particle swarm optimization.* Swarm intelligence, vol. 1, no. 1, pages 33–57, 2007. (Cited on pages 62 and 63.)

[Polycarpou *et al.* 1992] Marios M Polycarpou, Petros Ioannou*et al.* *Learning and convergence analysis of neural-type structured networks.* Neural Networks, IEEE Transactions on, vol. 3, no. 1, pages 39–50, 1992. (Cited on page 86.)

[Pratihar *et al.* 1999] Dilip Kumar Pratihar, Kalyanmoy Deb and Amitabha Ghosh. *A genetic-fuzzy approach for mobile robot navigation among moving obstacles.* International Journal of Approximate Reasoning, vol. 20, no. 2, pages 145–172, 1999. (Cited on page 32.)

[Qi *et al.* 2016] Peng Qi, Chuang Liu, Ahmad Ataka, Hak-Keung Lam and Kaspar Althoefer. *Kinematic control of continuum manipulators using a fuzzy-model-based approach.* IEEE Transactions on Industrial Electronics, vol. 63, no. 8, pages 5022–5035, 2016. (Cited on page 51.)

[Qin & Suganthan 2004] A Kai Qin and Ponnuthurai N Suganthan. *Robust growing neural gas algorithm with application in cluster analysis.* Neural networks, vol. 17, no. 8-9, pages 1135–1148, 2004. (Cited on page 71.)

[Raja *et al.* 2019] R Raja, A Dutta and B Dasgupta. *Learning framework for inverse kinematics of a highly redundant mobile manipulator.* Robotics and Autonomous Systems, vol. 120, page 103245, 2019. (Cited on page 95.)

[Remazeilles 2004] Anthony Remazeilles. *Navigation à partir d'une mémoire d'images.* PhD thesis, 2004. (Cited on page 11.)

[Roger *et al.* 2020] Datouo Roger, Mvogo Ahanda Joseph Jean-baptiste, Melingui Achille, Biya-Motto Frédéric and Essimbi Zobo Bernard. *Adaptive fuzzy finite-time command-filtered backstepping control of flexible joint robots.* Robotica, doi:10.1017/S0263574720000910, 2020. (Cited on pages v, 19, 124 and 140.)

[Rolf & Steil 2014] Matthias Rolf and Jochen J Steil. *Efficient exploratory learning of inverse kinematics on a bionic elephant trunk.* IEEE transactions on neural networks and learning systems, vol. 25, no. 6, pages 1147–1160, 2014. (Cited on pages 51, 66, 80, 81 and 85.)

[Sánchez-Fernández *et al.* 2004] Matilde Sánchez-Fernández, Mario de Prado-Cumplido, Jerónimo Arenas-García and Fernando Pérez-Cruz. *SVM multiregression for nonlinear channel estimation in multiple-input multiple-output systems.* Signal Processing, IEEE Transactions on, vol. 52, no. 8, pages 2298–2307, 2004. (Cited on pages 82 and 83.)

[Sanchez *et al.* 2015] Mauricio A Sanchez, Oscar Castillo and Juan R Castro. *Generalized type-2 fuzzy systems for controlling a mobile robot and a performance comparison with interval type-2 and type-1 fuzzy systems.* Expert Systems with Applications, vol. 42, no. 14, pages 5904–5914, 2015. (Cited on pages 31 and 32.)

[Savsani *et al.* 2010] V Savsani, RV Rao and DP Vakharia. *Optimal weight design of a gear train using particle swarm optimization and simulated annealing algorithms.* Mechanism and machine theory, vol. 45, no. 3, pages 531–541, 2010. (Cited on page 62.)

[Shimizu *et al.* 2008] Masayuki Shimizu, Hiromu Kakuya, Woo-Keun Yoon, Kosei Kitagaki and Kazuhiro Kosuge. *Analytical inverse kinematic computation for 7-DOF redundant manipulators with joint limits and its application to redundancy resolution.* IEEE Transactions on Robotics, vol. 24, no. 5, pages 1131–1142, 2008. (Cited on page 93.)

[Shin *et al.* 2009] Jongho Shin, H Jin Kim and Youdan Kim. *Adaptive inverse control using support vector regression.* In Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on, pages 2570–2575. IEEE, 2009. (Cited on page 82.)

[Simaan *et al.* 2004] Nabil Simaan, Russell Taylor and Paul Flint. *A dexterous system for laryngeal surgery.* In Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on, volume 1, pages 351–357. IEEE, 2004. (Cited on pages 51 and 65.)

[Singh & Claassens 2010] Giresh K Singh and Jonathan Claassens. *An analytical solution for the inverse kinematics of a redundant 7dof manipulator with link offsets.* In 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 2976–2982. IEEE, 2010. (Cited on page 94.)

[Singh *et al.* 2018a] Inderjeet Singh, Yacine Amara, Othman Lakhal, Achille Melingui and Rochdi Merzouki. *PH Model-based Shape Reconstruction of Heterogeneous Continuum Closed Loop Kinematic Chain: An Application to skipping Rope.* In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 8682–8688. IEEE, 2018. (Cited on pages 16 and 18.)

[Singh *et al.* 2018b] Inderjeet Singh, Yacine Amara, Achille Melingui, Pushparaj Mani Pathak and Rochdi Merzouki. *Modeling of continuum manipulators using pythagorean hodograph curves.* Soft robotics, vol. 5, no. 4, pages 425–442, 2018. (Cited on pages v, 12, 16, 18, 51, 66 and 138.)

[Smola & Schölkopf 2004] Alex J Smola and Bernhard Schölkopf. *A tutorial on support vector regression.* Statistics and computing, vol. 14, no. 3, pages 199–222, 2004. (Cited on pages 82, 84 and 117.)

[Soltanpour *et al.* 2019] Mohammad Reza Soltanpour, Mazda Moattari*et al. Voltage based sliding mode control of flexible joint robot manipulators in presence of uncertainties.* Robotics and Autonomous Systems, 2019. (Cited on page 109.)

[SU & Stepanenko 1997] CHUN-YI SU and Yury Stepanenko. *Backstepping-based hybrid adaptive control of robot manipulators incorporating actuator dynamics.* International journal of adaptive control and signal processing, vol. 11, no. 2, pages 141–153, 1997. (Cited on page 110.)

[Su *et al.* 1995] Chun-Yi Su, Yury Stepanenko and Tin-Pui Leung. *Combined adaptive and variable structure control for constrained robots.* Automatica, vol. 31, no. 3, pages 483–488, 1995. (Cited on page 121.)

[Subbash & Chong 2019] Panati Subbash and Kil To Chong. *Adaptive network fuzzy inference system based navigation controller for mobile robot.* Frontiers of Information Technology & Electronic Engineering, vol. 20, no. 2, pages 141–151, 2019. (Cited on page 31.)

[Sun *et al.* 2017] Lei Sun, Wei Yin, Meng Wang and Jingtai Liu. *Position control for flexible joint robot based on online gravity compensation with vibration suppression.* IEEE Transactions on Industrial Electronics, vol. 65, no. 6, pages 4840–4848, 2017. (Cited on page 109.)

[Surmann *et al.* 1995] Hartmut Surmann, Joerg Huser and Liliane Peters. *A fuzzy system for indoor mobile robot navigation.* In Proceedings of 1995 IEEE International Conference on Fuzzy Systems., volume 5, pages 71–76. IEEE, 1995. (Cited on page 31.)

[Tanese *et al.* 1989] Reiko Tanese, John H. Holland and Quentin F. Stout. *Distributed Genetic Algorithms for Function Optimization.* PhD thesis, University of Michigan, USA, 1989. (Cited on page 62.)

[Tee & Ge 2011] Keng Peng Tee and Shuzhi Sam Ge. *Control of nonlinear systems with partial state constraints using a barrier Lyapunov function.* International Journal of Control, vol. 84, no. 12, pages 2008–2023, 2011. (Cited on page 121.)

[Toshani & Farrokhi 2014] Hamid Toshani and Mohammad Farrokhi. *Real-time inverse kinematics of redundant manipulators using neural networks and quadratic programming: a Lyapunov-based approach.* Robotics and Autonomous Systems, vol. 62, no. 6, pages 766–781, 2014. (Cited on page 95.)

[Vapnik 2013] Vladimir Vapnik. The nature of statistical learning theory. Springer Science & Business Media, 2013. (Cited on page 82.)

[Walker 2013] Ian D Walker. *Robot strings: long, thin continuum robots.* In 2013 IEEE Aerospace Conference, pages 1–12. Ieee, 2013. (Cited on page 51.)

[Walter & Schulten 1993] Jörg A Walter and KI Schulten. *Implementation of self-organizing neural networks for visuo-motor control of an industrial robot.* IEEE transactions on Neural Networks, vol. 4, no. 1, pages 86–96, 1993. (Cited on page 70.)

[Wang *et al.* 2020] Chengshi Wang, Chase G Frazelle, John R Wagner and Ian Walker. *Dynamic Control of Multi-Section Three-Dimensional Continuum Manipulators Based on Virtual Discrete-Jointed Robot Models.* IEEE/ASME Transactions on Mechatronics, 2020. (Cited on page 51.)

[Webster & Jones 2010] Robert J Webster and Bryan A Jones. *Design and kinematic modeling of constant curvature continuum robots: A review.* The International Journal of Robotics Research, vol. 29, no. 13, pages 1661–1683, 2010. (Cited on pages 51 and 52.)

[Webster *et al.* 2009] Robet J Webster, John P Swensen, Joseph M Romano and Noah J Cowan. *Closed-form differential kinematics for concentric-tube continuum robots with application to visual servoing.* In Experimental Robotics, pages 485–494. Springer, 2009. (Cited on page 65.)

[Wei *et al.* 2014] Yanhui Wei, Shengqi Jian, Shuang He and Zhepeng Wang. *General approach for inverse kinematics of nR robots.* Mechanism and Machine Theory, vol. 75, pages 97–106, 2014. (Cited on page 94.)

[Wu *et al.* 2006] Wei Wu, Hongmei Shao and Zhengxue Li. *Convergence of batch BP algorithm with penalty for FNN training.* In Neural Information Processing, pages 562–569. Springer, 2006. (Cited on page 68.)

[Xia *et al.* 2018] Jianwei Xia, Jing Zhang, Wei Sun, Baoyong Zhang and Zhen Wang. *Finite-time adaptive fuzzy control for nonlinear systems with full state constraints.* IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 49, no. 7, pages 1541–1548, 2018. (Cited on page 125.)

[Xu *et al.* 2011] Wenfu Xu, Bin Liang and Yangsheng Xu. *Survey of modeling, planning, and ground verification of space robotic systems.* Acta Astronautica, vol. 68, no. 11-12, pages 1629–1649, 2011. (Cited on page 93.)

[Xu *et al.* 2018] Wenfu Xu, Tianliang Liu and Yangmin Li. *Kinematics, dynamics, and control of a cable-driven hyper-redundant manipulator.* IEEE/ASME Transactions on Mechatronics, vol. 23, no. 4, pages 1693–1704, 2018. (Cited on page 93.)

[Yahya *et al.* 2014] Samer Yahya, Mahmoud Moghavvemi and Haider AF Mohamed. *Artificial neural networks aided solution to the problem of geometrically bounded singularities and joint limits prevention of a three dimensional planar redundant manipulator.* Neurocomputing, vol. 137, pages 34–46, 2014. (Cited on page 95.)

[Yam *et al.* 1999] Yeung Yam, Péter Baranyi and Chi-Tin Yang. *Reduction of fuzzy rule base via singular value decomposition.* IEEE Transactions on fuzzy Systems, vol. 7, no. 2, pages 120–132, 1999. (Cited on page 33.)

[Yi *et al.* 2019] Zeren Yi, Guojin Li, Shuang Chen, Wei Xie and Bugong Xu. *A navigation method for mobile robots using interval type-2 fuzzy neural network fitting Q-learning in unknown environments.* Journal of Intelligent & Fuzzy Systems, vol. 37, no. 1, pages 1113–1121, 2019. (Cited on page 33.)

[Yin *et al.* 2018] Wei Yin, Lei Sun, Meng Wang and Jingtai Liu. *Nonlinear state feedback position control for flexible joint robot with energy shaping.* Robotics and Autonomous Systems, vol. 99, pages 121–134, 2018. (Cited on page 109.)

[Yip & Camarillo 2014] Michael C Yip and David B Camarillo. *Model-Less Feedback Control of Continuum Manipulators in Constrained Environments.* Robotics, IEEE Transactions on, vol. 30, no. 4, pages 880–889, 2014. (Cited on pages 51 and 81.)

[Yu *et al.* 2005] Shuanghe Yu, Xinghuo Yu, Bijan Shirinzadeh and Zhihong Man. *Continuous finite-time control for robotic manipulators with terminal sliding mode.* Automatica, vol. 41, no. 11, pages 1957–1964, 2005. (Cited on pages 125 and 126.)

[Yu *et al.* 2017] Jinpeng Yu, Lin Zhao, Haisheng Yu, Chong Lin and Wenjie Dong. *Fuzzy finite-time command filtered control of nonlinear systems with input saturation.* IEEE transactions on cybernetics, vol. 48, no. 8, pages 2378–2387, 2017. (Cited on page 125.)

[Z. Wu & Cui 2017] S. Wang Z. Wu and M. Cui. *Tracking controller design for random nonlinear benchmark system.* Journal of the Franklin Institute, vol. 354, no. 1, pages 360–371, 2017. (Cited on page 133.)

[Z. Wu & Shii 2012] M. Cui Z. Wu and P. Shii. *Backstepping control in vector form for stochastic Hamiltonian systems.* SIAM Journal on Control and Optimization, vol. 50, no. 2, pages 925–942, 2012. (Cited on page 133.)

[Zaplana & Basanez 2018] Isiah Zaplana and Luis Basanez. *A novel closed-form solution for the inverse kinematics of redundant manipulators through workspace analysis.* Mechanism and Machine Theory, vol. 121, pages 829–843, 2018. (Cited on page 94.)

[Zhao *et al.* 2010] Dongya Zhao, Shaoyuan Li, Quanmin Zhu and Feng Gao. *Robust finite-time control approach for robotic manipulators.* IET control theory & applications, vol. 4, no. 1, pages 1–15, 2010. (Cited on page 125.)