

POLITECNICO DI MILANO

SCHOOL OF INDUSTRIAL AND INFORMATION  
ENGINEERING

MASTER OF SCIENCE IN  
BIOMEDICAL ENGINEERING



---

**Enhanced Fitting Techniques for  
Model-Based Tracking of Continuum  
Manipulators**

---

*Author:*

Giuseppe BUCCHI

Matr. 852644

*Relator:*

Prof. Elena DE MOMI

*Correlator:*

Prof. Sarthak MISRA

Jakub SIKORSKI

ACCADEMIC YEAR 2017-2018

# Contents

<i>Abstract</i> . . . . .	xi
<i>Sommario</i> . . . . .	xv
<b>1. Introduction</b> . . . . .	<b>1</b>
1.1 Surgical Robotics . . . . .	1
1.2 Cardiac Catheterization . . . . .	4
1.3 Catheter's control . . . . .	6
1.4 State of the Art . . . . .	7
1.4.1 Magnetic Control . . . . .	7
1.4.2 Shape Reconstruction Techniques . . . . .	10
1.4.3 Model-Based Tracking System . . . . .	12
1.4.4 Linear Fitting Technique . . . . .	17
1.4.5 PRBM Technique . . . . .	18
1.5 Aim of the project . . . . .	21
<b>2. Materials and Methods</b> . . . . .	<b>23</b>
2.1 Catheter's Characteristics . . . . .	23
2.2 BigMag . . . . .	24
2.3 Software for Modeling: Solidworks . . . . .	26
2.3.1 Synthetic Model of the Catheter . . . . .	27
2.4 Processing on Matlab . . . . .	29
2.4.1 Linear Fitting Technique Implementation . . . . .	29
2.4.2 Model Fitting with not-fixed number of segments . . . . .	31

2.5	Validation Method A: Curvature Analysis with Linear Fitting Algorithm . . . . .	31
2.5.1	Initial Evaluation . . . . .	31
2.5.2	Experimental Protocol A1: Single Constant Curvature . . . . .	32
2.5.3	Experimental Protocol A2: Synthetic Models with Variable Known Curvature . . . . .	34
2.5.4	Experimental Protocol A3: Evaluation of a 3-D Printed Object with Known Curvatures . . . . .	34
2.6	Validation Method B: Catheter’s Analysis during Animation with Linear Fitting Algorithm . . . . .	37
2.6.1	Comparison Parameters and Animation . . . . .	37
2.6.2	Experimental Protocol B1: Qualitative Evaluation of BigMag dataset . . . . .	39
2.6.3	New Software for Synthetic Models: Maya . . . . .	40
2.6.4	Experimental Protocol B2: Evaluation of Configuration Parameters with Maya’s Dataset . . . . .	42
2.6.5	Experimental Protocol B3: Evaluation of the Radius of Curvature with Maya’s Dataset . . . . .	42
2.7	Flowchart of the Linear Fitting Algorithm . . . . .	44
2.8	Validation Method C: PRBM Algorithm Evaluation and Validation . . . . .	45
2.8.1	PRBM Algorithm: Description and Implementation . . . . .	45
2.8.2	Simulink Implementation of PRBM Algorithm for Validation Protocols . . . . .	57
2.8.3	Experimental Protocol C1: Evaluation of the joints angles with new compliance values . . . . .	59
2.8.4	Experimental Protocol C2: Evaluation of the joints angles with a cubic compliance model implemented on the PRBM . . . . .	60
2.8.5	Luenberger State Observer Implementation on PRBM Algorithm . . . . .	61
2.8.6	Experimental Protocol C3: Evaluation of the joints angles with a Luenberger state observer implemented in the PRBM . . . . .	66
2.8.7	Experimental Protocol C4: Final comparison of all PRBM models in the evaluation of the tip’s position . . . . .	66
2.9	Flowchart of the PRBM Algorithm . . . . .	67

3. <i>Results</i> . . . . .	69
3.1 Initial Evaluation . . . . .	69
3.2 Validation Method A . . . . .	71
3.2.1 Experimental Protocol A1 . . . . .	71
3.2.2 Experimental Protocol A2 . . . . .	73
3.2.3 Experimental Protocol A3 . . . . .	75
3.3 Validation Method B . . . . .	79
3.3.1 Experimental Protocol B1 . . . . .	79
3.3.2 Experimental Protocol B2 . . . . .	84
3.3.3 Experimental Protocol B3 . . . . .	88
3.4 Validation Method C . . . . .	91
3.4.1 Experimental Protocol C1 . . . . .	91
3.4.2 Experimental Protocol C2 . . . . .	96
3.4.3 Experimental Protocol C3 . . . . .	98
3.4.4 Experimental Protocol C4 . . . . .	99
4. <i>Discussion</i> . . . . .	103
4.1 Initial Evaluation . . . . .	103
4.2 Validation Method A . . . . .	104
4.3 Validation Method B . . . . .	106
4.4 Validation Method C . . . . .	109
5. <i>Conclusions and Perspectives</i> . . . . .	114
<i>Appendices</i> . . . . .	118
.1 Moore-Penrose Pseudoinverse . . . . .	119
.2 Main Functions . . . . .	120
.2.1 Linear Least Square Function . . . . .	120
.2.2 Curvature and Radius Evaluation . . . . .	121
.2.3 Validation Error First phase . . . . .	121
.2.4 Iterative Points Selection . . . . .	122
.2.5 Stereo Vision Processing . . . . .	123

.2.6	Tip Position . . . . .	127
.2.7	BigMag Data Extraction . . . . .	128
.2.8	First Joint Evaluation . . . . .	132
.2.9	Angles Evaluation . . . . .	133
.2.10	Angles Evaluation with Cubic Compliance . . . . .	134
.2.11	$\Omega$ Calculation . . . . .	135
.2.12	Get Jacobian . . . . .	135
.2.13	Function Optimized by Fsolve . . . . .	137
.2.14	Function Optimized by Fsolve with Cubic Compliance . . . . .	138
.2.15	Ultrasound Algorithm . . . . .	139
.2.16	Slice Center Estimation . . . . .	142
.2.17	Complex Curvature Validation . . . . .	143
.3	Simulink Additional Code . . . . .	146
.3.1	Simulink: choose the correct Angles and Wrenches . . . . .	146
.3.2	Simulink: Luenberger Observer . . . . .	147

# List of Figures

1.1	Evolution of Minimally Invasive Surgery devices in the last decades (Burgner-Kahrs et al. [2015]). . . . .	2
1.2	Shown the different workspaces of MIS procedures (Burgner-Kahrs et al. [2015]). . . . .	4
1.3	In the picture is shown a common tendon-driven catheter with a coil inserted in the tip. In yellow the external magnetic force applied to steer the catheter ( <a href="http://www.surgicalroboticslab.nl">http://www.surgicalroboticslab.nl</a> ). . . . .	8
1.4	On the top the structure of a Fiber Bragg Grating, in the middle the refractive index profile of the inner part of the fiber and in the bottom the spectral response. . . . .	10
1.5	An example of 3-D Volume Reconstruction with Voxel Carving technique. The resolution of the volume reconstructed is proportional to the number of cameras mounted on the setup. Above a setup with two cameras, below a setup with six cameras. . . . .	12
1.6	(1) Needle (2) Soft-tissue phantom used to simulate a body-like behavior of the material (3) Ultrasound Transducer used to track the needle (4) 2-D slice obtained by the ultrasound probe which provides a cross-sectional view of the needle. During insertion the ultrasound probe is robotically repositioned to track the tip in real-time (Vrooijink et al. [2013]). . . . .	15
1.7	Vision-based 3-D shape sensing system. The manipulator shown is an intracardiac catheter produced by Hansen Medical (Camarillo et al. [2008a]). . . . .	16

1.8	Representation of a PRBM with one single joint. The force $F$ applied to the tip generates a rotation of the joint which characteristics are related to the torsional spring applied to the joint itself ( <a href="http://www.ScienceDirect.com">www.ScienceDirect.com</a> ). . . . .	19
2.1	Examples of catheters with a magnetic component on the tip and different body's material . . . . .	23
2.2	On the left the entire BigMag system. On the right a focus on each detail: (1) Mobile Coils (2) Inserter of continuum manipulators (3) Top Camera (4) Side Camera (Sikorski et al. [2017]). . . . .	25
2.3	In the picture the workspace of BigMag. In the selected region, catheters are inserted in order to be steered magnetically by the system (Sikorski et al. [2017]). . . . .	26
2.4	On the top the initial catheter's model generated on Solidworks. On the bottom the catheter has been bended $60^\circ$ with constant curvature . . . . .	28
2.5	3-D Representation of the catheter generated on Solidworks with the respective coordinate system. . . . .	29
2.6	Representation of the point clouds during one step of the model fitting iteration. . . . .	31
2.7	Representation of the catheter's model to highlight the difference before and after the introduction of noise to the point clouds. Points with noise in red show a more homogeneous spatial distribution than blu initial ones. . . . .	33
2.8	Model of the catheter created on Solidworks. The green part has a curvature of $100m^{-1}$ , the light blue part of $200m^{-1}$ . . . . .	35
2.9	Point clouds from Stereo Vision System representing the catheter. The origins before and after filtering are represented with two different colors: black and red for unfiltered data respectively. . . . .	36
2.10	In the picture are shown the three parameters used to describe the orientation of the catheter in the workspace during the animation. . . . .	38
2.11	In the figure is shown Maya platform. In this case, rotation happened around the x axis in red and the bending is constant during rotation. Below the timeline in which is possible to add key points. . . . .	41

2.12	The image shows the processing of data received from BigMag to simulate the scanning of an ultrasound probe. In this example the slice thickness was 1.5mm. . . . .	43
2.13	Flowchart of the linear fitting algorithm. . . . .	44
2.14	(a) Is the shape of a continuum manipulator: the orange lines are the rigid links connected by rotational joints. Each link represents a constant-curvature arc (ls) that has a particular direction and angle $(\theta_i, \phi_i)$ . (b) The first angle $(q_{\phi,i})$ is related to the bending direction, $(q_{\theta,i})$ is related to the amount of bending. $K_{\theta}$ is a rotational spring assigned to each joint. . . . .	48
2.15	Continuum and pseudo-rigid-body model of the catheter. $\xi_{ix}$ and $\xi_{iy}$ are orthogonal twists and represent the rotation of the $i^{th}$ joint. . . . .	52
2.16	Simulink model: the constant is the wrench applied that is multiplied by a sine function. The result of the product becomes input of the MATLAB function which provides the corresponding angles. . . . .	57
2.17	Simulink model: the second Simulink block is shown. The first function output is wrench composed of a $6 \times 1$ array and the angles also contained in a $6 \times 1$ array. Highlighted in blue BigMag data storage. Display blocks used to show results during the iteration. . . . .	58
2.18	Simulink Model: (1) is a function used to load the angles of the previous iteration (2). It chooses which angles to use depending on whether it is the first iteration or the following ones (appendix .3.1) (3). Is a function that derives the angles exploiting the observer of Luenberger. Corners are saved at the end of each iteration. . . . .	65
2.19	Flowchart of the PRBM algorithm. . . . .	67
3.1	In the vertical axis the relative error between the evaluated distance and the real one. In the horizontal axis the points. . . . .	70
3.2	Catheter's plot, each STL point is represented as a circle. Ideal radius compared to measured ones. . . . .	71
3.3	In blue the real radius of the catheter. Light blue the measured one and in yellow the relative error. . . . .	73



3.4	Above in the vertical axis Solidworks radius values in blue and in orange the values obtained with the average model. Below the relative error, always below the 3%. In the horizontal axis the bending angle. . . . .	74
3.5	In this figure is shown an example of model fitting evaluation with 3-D bending and noise addition. Noise has been added with different amplitudes, 10%, 50% and 90% of the radius length. Black links has been found considering the initial poin clouds, red links after the noise addition.	75
3.6	Three different Tracker evaluations considering different reference joints. First Curvature. . . . .	76
3.7	Three different Tracker evaluations considering different reference joints. Second Curvature. . . . .	77
3.8	Three different Tracker evaluations considering different filtered joints. First Curvature. . . . .	78
3.9	Three different Tracker evaluations considering different filtered joints. Second Curvature. . . . .	78
3.10	In figure the permutation of all frames to compare $x$ . . . . .	80
3.11	In figure the permutation of all frames to compare $d$ . . . . .	80
3.12	In figure the permutation of all frames to compare $\alpha$ . . . . .	81
3.13	X tip position during the helical movement. . . . .	81
3.14	Similar sinusoidal function in the y-axis due to the helical movement generated by BigMag. . . . .	82
3.15	Similar sinusoidal function in the x-axis due to the helical movement generated by BigMag. . . . .	82
3.16	X-axis coordinates after low pass filtering. . . . .	83
3.17	Y-axis coordinates after low pass filtering. . . . .	83
3.18	Z-axis coordinates after low pass filtering. . . . .	84
3.19	In figure the permutation of all frames to evaluate $x$ . Maya's value is constant during all the animation. . . . .	85
3.20	Relative error for the parameter $x$ . . . . .	85
3.21	In figure the permutation of all frames to evaluate $d$ . Maya's value is constant during all the animation. . . . .	86

3.22	Relative error for the parameter $d$ . . . . .	86
3.23	In figure the permutation of all frames to evaluate $\alpha$ 's Angular velocity was not constant during all the animation. . . . .	87
3.24	Relative error and Error for the parameter $\alpha$ . . . . .	87
3.25	In figure the permutation of all frames to evaluate $\alpha$ . Curvature was constant during all the animation and so the radius. . . . .	88
3.26	Relative error for the radius. . . . .	89
3.27	In figure the permutation of all frames to evaluate the radius. Curvature and therefore radius were constant during all the animation. . . . .	90
3.28	Relative error for the radius obtained with the ultrasound simulation. . . . .	90
3.29	In figure the angles given by BigMag for the first test for each joint and direction. . . . .	92
3.30	In blue BigMag's dataset, in red <i>ModelG</i> dataset and in green <i>ModelGQ</i> dataset. . . . .	93
3.31	In blue BigMag's dataset, in red <i>ModelG</i> dataset and in green <i>ModelNC</i> dataset. . . . .	94
3.32	In blue BigMag's dataset, in red <i>ModelS</i> dataset and in green <i>ModelNC</i> dataset. . . . .	96
3.33	In blue BigMag's dataset, in red <i>ModelS</i> dataset and in green <i>ModelSK</i> dataset. . . . .	97
3.34	In blue BigMag's dataset, in red <i>ModelL</i> dataset with 0.9 correction factor and in green <i>ModelSK</i> dataset. . . . .	98
3.35	In blue BigMag's dataset, in red <i>ModelL</i> dataset with 0.5 correction factor and in green <i>ModelSK</i> dataset. . . . .	99
3.36	In blue BigMag's dataset, in black <i>ModelL</i> coordinates, in red <i>ModelSK</i> coordinates and in green the values with <i>ModelG</i> . . . . .	100
3.37	In black <i>ModelL</i> error, in red <i>ModelSK</i> error and in green the error values obtained with <i>ModelG</i> , all compared with BigMag dataset. . . . .	101

# List of Tables

3.1	Model Fitting, Radius Comparison . . . . .	72
3.2	Noise Addition . . . . .	75
3.3	Curvature Evaluation . . . . .	79
3.4	Curvature Error . . . . .	79
3.5	Parameters Evaluation . . . . .	88
3.6	Radius Evaluation . . . . .	91
3.7	Tip Position Coordinate X . . . . .	101
3.8	Tip Position Coordinate Y . . . . .	102
3.9	Tip Position Coordinate Z . . . . .	102

# Abstract

This project, developed at Surgical Robotics Lab, University of Twente, The Netherlands, is part of a vast field of research and development called ‘Minimally Invasive Surgery’ (MIS). MIS deals also with the development of robotic devices for automatic or semi-automatic applications in the surgical field. The introduction of robotics aims to bring innovations especially in terms of reduction of invasivity and recovery time after operation, increase quality and precision of operations and in the performance of complex automated tasks that would be impossible to do for a surgeon without the help of this technology.

Cardiac catheterization is an example of application field of MIS that is undergoing a strong evolution in recent decades. Rigid and passive tools have been replaced over the years by continuous manipulators, a class of elastic instruments able to bend along their length in a continuous manner, in a controlled workspace. This new technology has made possible to improve the performance of some applications and introduce new ones that were previously impossible.

Different types of control of these devices have been developed, including magnetic control. BigMag, a novel platform capable of magnetic steering of endovascular catheters, has been developed in Surgical Robotics Lab. BigMag falls into this field because its purpose is to develop a technology to deflect the tips of continuum manipulators in order to enable automated endovascular surgery, for medical applications like angioplasty, aneurysm repair and electrophysiology. A good estimation of the shape of endovascular catheters is an essential step enabling control of the device.

Since these catheters are part of the continuum manipulator family, it is necessary to derive mathematical models to describe their behavior from the data collected online by

tracking systems: this task is complex in the search for a trade off between accuracy and speed of calculation.

This project aimed at developing an advanced model-based tracking algorithm on MATLAB in order to improve the current techniques used for BigMag. The main purpose of the algorithm was to obtain information on the configuration of the catheters used for BigMag system with high precision, with a particular focus on the evaluation of the catheter's tip, fundamental for the control of the catheter itself.

The catheter exploited in this project has a *5cm* length and *2mm* diameter elastic cylindrical body with a magnetic component mounted on its tip in order to enable the magnetic control. Two algorithms have been developed and validated. The first model-based tracking algorithm has been developed with a linear fitting technique to describe the continuum manipulator as a series of rigid links connected by joints. In order to obtain this model, a discretization of the catheters in cloud-points has been performed to enable the processing on MATLAB.

This first phase of the project has seen the design of catheters models on Solidworks and Maya, in order to create synthetic datasets comparable with the one generated by BigMag's stereo vision system. Synthetic dataset has been useful for improving the linear fitting algorithm, by simulating particular static situations or animations of the model itself, and to validate its performance. The models obtained through these software have been processed differently in order to simulate multiple tracking systems such as stereo vision and ultrasounds, thus comparing their performance. Synthetic point clouds have also been reworked with the addition of a noise in the three directions of the axes in order to make them more similar to the real data collected by clinical tracking systems.

The linear fitting algorithm has been validated in the reconstruction of configuration parameters such as the radius of curvature, the position of the catheter's tip and other parameters to describe its positioning and orientation in the workspace. Two validation methods A and B with three experimental protocols each have been developed in order to validate the algorithm. Both synthetic point clouds and data actually collected from the BigMag's stereo vision system were used and the results were evaluated and reprocessed by filtering to make them more suitable for the perspective of catheter's control.

The second algorithm has been developed exploiting the Pseudo-Rigid-Body-Model tech-

nique. This technique allowed to describe the relationship between the various joints and the segments that linked them through the introduction of torsional springs and local reference systems. Given a force acting on the tip of the model, it became possible to derive the configuration of all the joints belonging to it.

Furthermore, thanks to the PRBM it has been possible to implement the relationship between the magnetic field provided by BigMag and the final configuration of the model and therefore the position and orientation of the catheter. The magnetic field provided by BigMag generated a torque due to interaction with permanent magnet at the tip of the catheter, thus causing a change in the configuration of the whole manipulator; the torque has been processed by the algorithm in order to reconstruct the configuration model through the existing relationships between the joints.

This model has been enhanced to evaluate its performance, constantly improving certain parameters such as the compliance of the springs, which modulates the elastic characteristics of the real catheter, so as to maximize the accuracy of the reconstruction.

The model was also implemented on Simulink in order to automate simulations through the use of real data provided by the stereo vision system. A validation Method C with three experimental protocols has been developed in order to compare results and highlight the weaknesses of the model. A different variable compliance model was also introduced as a function of the bending angle.

Finally, a Luenberger state observer has been implemented in the PRBM algorithm to maximize the accuracy in the reconstruction of the configuration of all the joints through the introduction of a corrective factor that related to the information collected from the previous iteration.

A fourth experimental protocol has been developed in order to validate the PRBM algorithm in the evaluation of the tip position, comparing the results with the validated linear fitting algorithm implemented on BigMag exploiting information from the Optical System. The evaluations of the first linear fitting algorithm showed a behavior comparable with literature with sample mean and standard deviation of the relative error of  $1.2 \pm 0.86\%$  in the reconstruction of the radius of curvature and a maximum variation of  $0.38 \pm 0.1mm$  from the origin positions in the case of noise up to 90% of the radius length. As regards the evaluation of the variable curvature in a 3-D model printed and

analyzed with the linear fitting algorithm, has been reported errors below 10% despite the necessary user intervention in the evaluation of the curvature.

Also the evaluation of the catheter configuration parameters showed a good accuracy and repeatability of the algorithm both with synthetic dataset and with the real one provided by the stereo vision system. Important was the quality of the synthetic data provided by Maya software in the simulation of the catheter movement. The relative maximum errors with regard to the synthetic data were (2.39, 1.64%) in the evaluation of the radius of curvature. The simulation error was much higher (22.81, 5.61%) while exploiting information given by a trial ultrasound scanning processing rather than the stereo vision one, showing a general overestimation of the value and therefore requiring a different processing.

The evaluations of the second algorithm showed a constant improvement of the model's performance during the evaluation of the compliance and a definitive satisfactory result after the implementation of the Luenberger observer with a maximum average sample error in the evaluation of the tip's position of  $(-0.007, 0.11\text{cm})$  compared to BigMag reference values, much better than the maximum result obtained from previous evaluations where the average minimum error in this assessment was  $(0.51, 0.68\text{cm})$ , too high compared to literature.

Taking a cue from what has been achieved in this project it is certainly possible to continue the improvement of the fitting algorithm, always aiming at improving accuracy and adaptability without increasing the computational cost. Improving the quality in the production of synthetic data for model training with the introduction of the environment could make it possible to adapt the algorithm to clinical conditions by testing it directly with realistic virtual animations.

The further study of the materials and the parameters of the models, together with the development of state observers, could lead to real-time, high-precision model-based tracking suitable for clinical applications.

# Sommario

Questo progetto, sviluppato al Surgical Robotics Lab dell'Università di Twente in Olanda, rientra in un campo di ricerca e sviluppo molto ampio denominato 'Chirurgia Robotica Mini-Invasiva' (MIS), il cui scopo principale è lo sviluppo di dispositivi robotici automatici o semiautomatici per applicazioni in campo chirurgico. L'introduzione della robotica e l'utilizzo di innovazioni tecnologiche d'avanguardia si pongono come obiettivo la riduzione dell'invasività e del tempo di ricovero ospedaliero, l'aumento della qualità e della precisione delle stesse operazioni e la possibilità di svolgere operazioni complesse altresì impossibili per un chirurgo con i metodi utilizzati tradizionalmente.

La cateterizzazione cardiaca è un esempio di campo applicativo della MIS che si sta evolvendo rapidamente negli ultimi decenni. Strumentazioni rigide e passive sono state sostituite negli anni da manipolatori continui, cioè quella classe di strumenti che grazie alle loro caratteristiche elastiche possono essere flessi in modo continuo in uno spazio di lavoro. Questa nuova tecnologia ha permesso di migliorare la performance di alcune applicazioni di cateterizzazione cardiaca ed introdurne di nuove, prima impossibili.

BigMag, un nuovo dispositivo che permette di controllare magneticamente il movimento di cateteri intravascolari, è stato sviluppato al Surgical Robotics Lab. BigMag rientra all'interno di questo campo di ricerca poiché mira a sviluppare una tecnologia per la movimentazione ed il controllo di manipolatori continui al fine di automatizzare la chirurgia intravascolare per applicazioni come l'angioplastica, la riparazione di aneurismi e l'elettrofisiologia. Proprio perché questa tipologia di cateteri intravascolari rientra nella famiglia dei manipolatori continui, è necessario implementare modelli matematici, per descriverne il comportamento a partire dai dati raccolti online da sistemi di tracking. Lo sviluppo di questo campo è complesso poiché in una continua ricerca di equilibrio fra



accuratezza del modello e velocità dell'algoritmo.

Questo progetto si è proposto di sviluppare un avanzato algoritmo di model-based tracking su MATLAB per migliorare le tecniche di visualizzazione utilizzate su BigMag. Lo scopo principale dell'algoritmo era quello di ottenere informazioni sulla configurazione dei cateteri utilizzati dal sistema BigMag con elevata precisione, con particolare attenzione alla valutazione della punta del catetere, fondamentale per il controllo di BigMag sul catetere stesso. Il catetere sfruttato per questo progetto era composto da un corpo cilindrico lungo  $5\text{cm}$  con diametro di  $2\text{mm}$  dotato di una componente magnetica fissata sulla punta per permetterne il controllo magnetico. Due algoritmi sono stati sviluppati e validati.

Il primo algoritmo di model-based tracking è stato sviluppato grazie ad una tecnica di regressione lineare al fine di descrivere il manipolatore continuo come una serie di segmenti rigidi connessi da giunti. Per ottenere questo modello è stato necessario discretizzare il catetere per ottenere una nuvola di punti analizzabile su MATLAB. Questa prima fase ha visto la progettazione di modelli di catetere comparabili con quelli utilizzati da BigMag, attraverso l'utilizzo di software CAD quali Solidworks e Maya. Questi modelli hanno permesso la generazione di dati sintetici, utili per lo studio ed il miglioramento dell'algoritmo. I dati sintetici hanno infatti permesso di ricostruire simulazioni del modello in particolari situazioni statiche e animazioni, grazie alle quali è stata eseguita una parte della validazione dell'algoritmo di fitting lineare.

I modelli ottenuti grazie a questi software sono stati elaborati diversamente su MATLAB per riprodurre virtualmente il funzionamento di due sistemi di tracking ampiamente utilizzati in letteratura: il sistema ottico ed il sistema ad ultrasuoni. Le nuvole di punti ottenute dagli algoritmi sono state inoltre rielaborate per introdurre un rumore nelle tre direzioni degli assi di riferimento con lo scopo di renderle il più simili possibile a quanto ottenuto dal sistema di tracking ottico.

L'algoritmo di fitting lineare è stato valutato nella ricostruzione di parametri fondamentali per la valutazione della configurazione quali il raggio di curvatura del catetere, la posizione della sua componente magnetica ed altri parametri utili per la ricostruzione della posizione ed orientamento nello spazio di lavoro. Due metodi di valutazione A e B con tre protocolli sperimentali ciascuno sono stati sviluppati per la validazione dell'algoritmo.

Per questa validazione sono stati utilizzati sia i dati sintetici, sia dati reali ottenuti dal sistema ottico installato su BigMag; entrambi i dataset sono stati poi elaborati affinché risultassero piú efficaci ai fini del controllo del catetere eseguito da BigMag.

Il secondo algoritmo é stato sviluppato sfruttando la tecnica del modello di corpo pseudo rigido (PRBM). Questo modello ha permesso di descrivere la relazione fra i vari giunti rotazionali del modello ed i segmenti di collegamento attraverso l'introduzione di molle torsionali e sistemi di riferimento solidali ai giunti grazie ai quali, applicata una torsione agente sulla punta del catetere, é stato possibile ottenere la configurazione di ciascun giunto del modello stesso. Sfruttando il PRBM é stata inoltre implementata la relazione fra il campo magnetico prodotto da BigMag e la configurazione finale del modello, cioé la posizione e l'orientamento del catetere stesso. Il campo magnetico fornito da BigMag permetteva infatti di generare una torsione sulla punta del catetere, dovuta all'interazione fra il campo stesso ed il magnete permanente del catetere, provocando in tal modo una variazione della configurazione di tutto il manipolatore continuo. Il momento torcente é stato quindi elaborato dall'algoritmo per poter ricostruire la configurazione del modello attraverso la relazione fra i giunti.

L'algoritmo é stato progressivamente migliorato per valutarne le performance, intervenendo su parametri come la rigidezza delle molle, che permetteva di modellizzare le caratteristiche elastiche del catetere reale, massimizzando cosí l'accuratezza nella ricostruzione.

Questo secondo algoritmo é stato implementato il modello su Simulink per automatizzare la simulazione grazie all'uso di dati forniti dal sistema di ottico di BigMag. Un metodo di valutazione C con tre protocolli sperimentali sono stati sviluppati per comparare i risultati ed evidenziare i punti di forza e debolezza del modello. E' stato inoltre introdotto un modello di compliance variabile in relazione all'angolo di bending.

Nella fase finale del progetto é stato invece sviluppato un osservatore di stato di Lu-enberger nell'algoritmo PRBM per massimizzare l'accuratezza nella ricostruzione della configurazione del catetere attraverso l'introduzione di un fattore correttivo che tenesse conto delle informazioni ottenute nell'iterazione precedente.

Un quarto protocollo sperimentale é stato sviluppato per per la validazione dell'algoritmo PRBM nella valutazione della posizione della punta del catetere, comparando i risul-

tati con i valori di riferimento forniti dall'algorithm di fitting lineare validato ed implementato nel sistema ottico di BigMag. La valutazione delle performance del modello di fitting lineare ha mostrato un comportamento dell'algorithm paragonabile alla letteratura, con una media campionaria e relativa deviazione standard dell'errore relativo pari a  $(1.2 \pm 0.86\%)$  per il raggio di curvatura ed una variazione massima della posizione dei giunti di  $(0.38 \pm 0.1mm)$  nel caso di rumore sui punti fino al 90% della lunghezza del raggio. Anche la valutazione di due curvature note sullo stesso modello 3-D stampato e successivamente analizzato da BigMag ha mostrato valori d'errore sempre inferiori al 10%, evidenziando però la necessità di un operatore nella scelta dei giunti da utilizzare per la valutazione della curvatura.

Anche la valutazione dei parametri sulla configurazione del catetere ha mostrato una buona accuratezza e ripetibilità dell'algorithm sia con dati sintetici sia con dati reali dal sistema ottico. Importante è stato inoltre il contributo fornito dal software Maya nel generare set di dati sintetici di elevata qualità permettendo anche di simulare il movimento del catetere. L'errore relativo massimo è risultato basso  $(2.39, 1.64\%)$  per quanto riguarda il raggio di curvatura. Simulando invece il processing di una sonda ad ultrasuoni sui dati sintetici gli errori si sono rivelati più alti  $(22.81, 5.61\%)$ , mostrando una generale sovrastima del parametro che richiederebbe quindi una variazione dell'algorithm di processing. La valutazione del secondo algorithm ha mostrato un costante miglioramento della performance del modello durante la correzione dei parametri dei giunti ed un definitivo incremento dell'accuratezza nella ricostruzione dopo l'implementazione dell'osservatore di stato di Luenberger. L'errore medio campionario e relativa deviazione standard nella valutazione della posizione della punta del catetere dopo l'implementazione dell'osservatore sono risultati pari a  $(-0.007, 0.11cm)$  rispetto ai valori di riferimento forniti da BigMag, nettamente più performante rispetto al miglior risultato ottenuto nelle valutazioni precedenti  $(0.51, 0.68cm)$ , troppo alto se comparato alla letteratura. Dall'analisi dei risultati di questo progetto è possibile affermare che vi sia ancora la possibilità di migliorare ulteriormente l'algorithm di fitting, con l'obiettivo di incrementarne l'accuratezza e l'adattabilità senza incrementarne eccessivamente il costo computazionale. Il miglioramento della qualità nella produzione di dati sintetici per i test sui modelli con l'introduzione dell'environment potrebbe inoltre permettere test di adattabilità dell'algorithm in condizioni

cliniche testandolo direttamente con animazioni virtuali realistiche.

Lo studio ulteriore sui materiali e sui parametri dei modelli, insieme allo sviluppo di osservatori di stato, potrebbe infine portare ad un'accuratezza nella ricostruzione e quindi predizione della configurazione del catetere tale da permettere future applicazioni nel campo chirurgico.

# 1

## Introduction

### 1.1 Surgical Robotics

Robotics in general and surgical robotics in particular are examples of translational research: originally, research funding was provided primarily by the defense and development departments of the countries: for example the prototype of Surgical's da Vinci System, a robotic system that nowadays is commonly used in the prostate removal procedure, cardiac valve replacement, gynecological surgery and any abdominal or thoracic surgical procedures, was developed in the late 1980s under contract to the US Army for performing remote surgery in the battlefield (Gomes [2011]).

Robotic surgery is a field of research in strong development which aim is to explore new approaches to improve outcomes for patients by making procedures safer and more effective through the application of cutting-edge technologies.

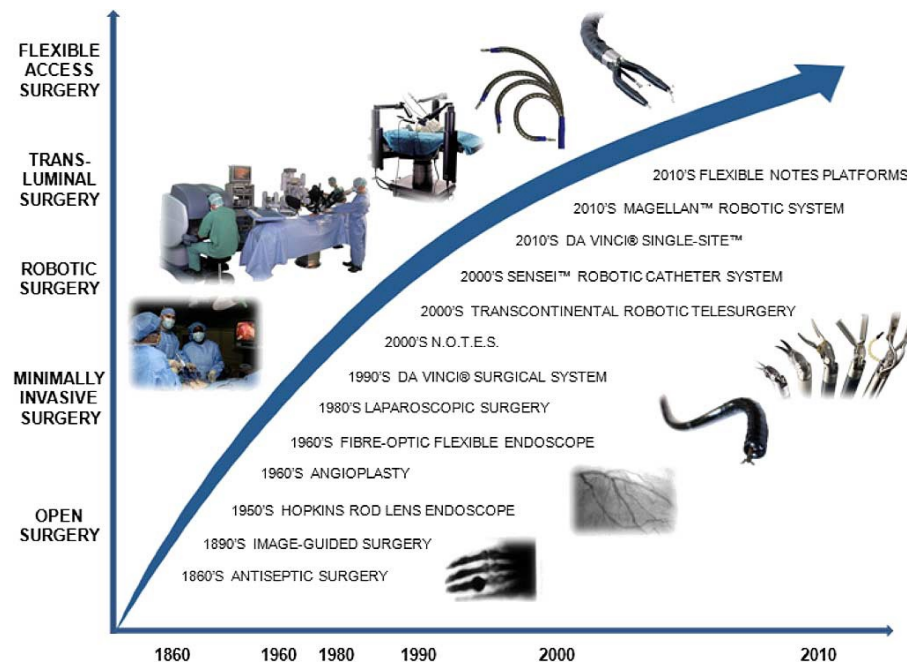


Fig. 1.1: Evolution of Minimally Invasive Surgery devices in the last decades (Burgner-Kahrs et al. [2015]).

A strongly growing branch within this field is certainly minimally invasive surgery (MIS): ‘minimally invasive’ means that surgical instruments could be inserted into little incision instead of operating the patients through large incisions.

The agenda for minimally invasive surgery continues to be driven towards the minimization of the number and size of visible skin incisions increasing surgical precision, thus reducing postoperative pain to the patient, shortening recovery time, improving cosmesis and ensuring overall cost-effectiveness. The main object of such smart instruments is to reduce drastically or eliminate patient trauma caused to the tissue due to open surgery, in order to improve the diagnostic accuracy and therapeutic outcome requiring often no more than ‘a few minutes’ training for the surgeons (Burgner-Kahrs et al. [2015]).

The use of robot assistance has helped to realize the full potential of MIS with improved consistency, safety and accuracy.

In addition, compared to open surgery, robotics surgery allows surgeons to perform complex surgical tasks like operations in very narrow spaces in the body that would otherwise only be accessible through open surgery with long incisions. Specialized instrumenta-

tion with enhanced, integrated flexibility, stability and dexterity to reach the operative target sites through complex anatomical pathways is then required. The development of articulated precision tools to enhance the surgeon's dexterity has evolved in parallel with advances in imaging and human-robot interaction, improving the hand-eye coordination and manual precision down to micro scales (Burgner-Kahrs et al. [2015]), (Vitiello et al. [2013]).

Despite all these benefits, new technologies require surgeons to be trained with the latest tools or platforms, available to modify the conventional surgical workflow and sometimes they have to compensate for counter intuitive ergonomic principles. Furthermore, surgeons rely mostly on sight and touch feedback that came through considerable changes: the latest robotics surgical platforms have recently been introduced to the clinical arena and offer solutions to increase dexterity, manipulability as well as stability and motion accuracy (Burgner-Kahrs et al. [2015]).

In addition, surgical robots are often very expensive compared to traditional approaches and do not replace human intelligence, skill and experience. There are still significant drawbacks associated with conventional MIS including inaccurate control of the instrument caused by rigid instrumentation (Center [2017]).

Generally, minimally invasive access can be splitted in four different categories:

1. Extraluminal: procedures where a cavity is accessed directly through skin incisions like Laparoscopy and thoracoscopy.
2. Intraluminal: procedures performed through tubular anatomical structures like urethra, esophagus and arteries without breaching their luminal boundaries; an example of application is the use of the endoscope.
3. Transluminal Access: expansion of intraluminal operations thanks to new MIS technologies.
4. Hybrid Approaches: these procedures are a combination of the ones described before and allow a strategy to mitigate the invasiveness of particular procedures that otherwise might require large incisions.

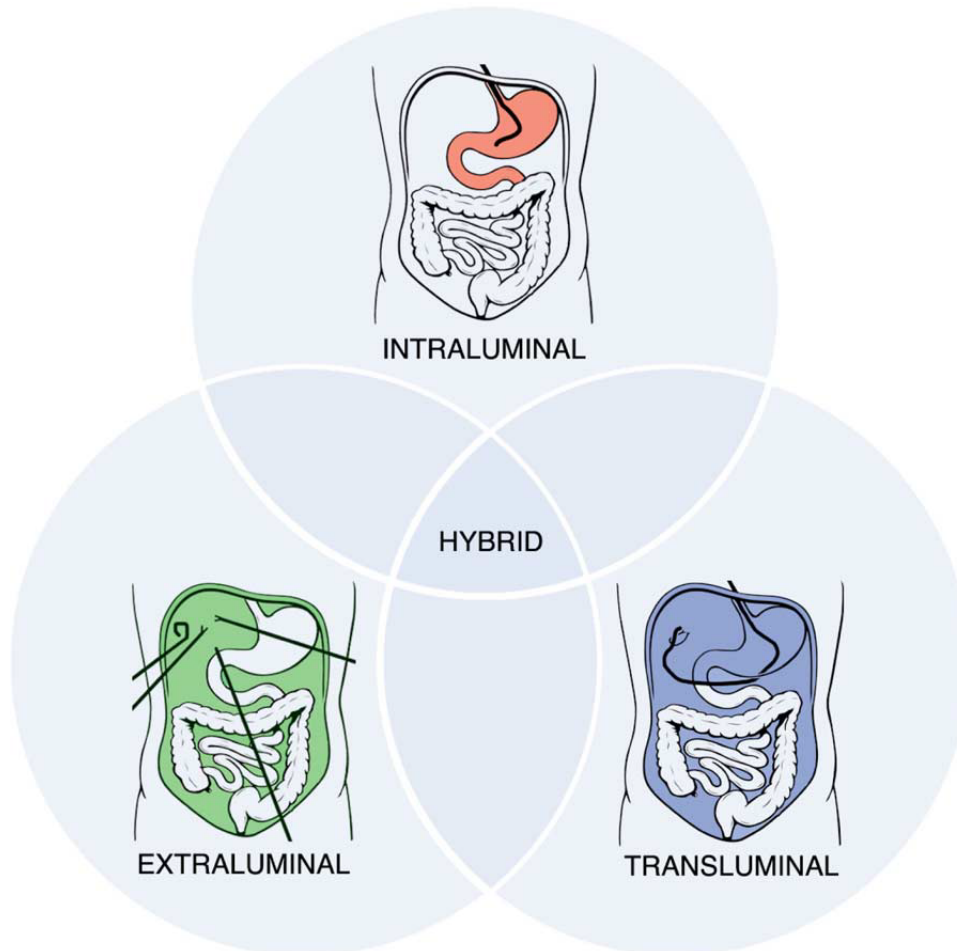


Fig. 1.2: Shown the different workspaces of MIS procedures (Burgner-Kahrs et al. [2015]).

## 1.2 Cardiac Catheterization

Cardiac catheterization is a common example of Hybrid approach of robotic surgery. A thin hollow tube called catheter is usually inserted into the iliac artery that leads to the heart.

A cardiac catheterization provides information on how well the organ works, identifies problems and allows for procedures to open blocked arteries. For example, during cardiac catheterization (Association [2017]) is possible to:

- Take X-rays using contrast dye injected through the catheter to look for narrowed or blocked coronary arteries. This is called coronary angiography or coronary arte-



riography.

- Perform a percutaneous coronary intervention (PCI) such as coronary angioplasty with stenting to open up narrowed or blocked segments of a coronary artery.
- Take samples of blood to measure the oxygen content in the four chambers of the heart.
- Perform Intracardiac electrogram
- Look for defects in the valves or chambers of heart with cameras.
- Remove and examine under microscope a small piece of heart tissue.

In order to perform these operations it is necessary to use tools that allow to reach the target enabling the transportation or insertion of miniaturized support devices such as electrodes, cameras, surgical instruments and other types of sensors.

The class of instruments used for such applications are called continuum manipulators and consist of an elastic structure which allows to bend along their length in a continuous manner (Roesthuis and Misra [2016]), (Khoshnam et al. [2012]). This class of instruments exhibits several useful properties in medical applications, due to the elastic characteristics and a non rigid structure, that allow the instruments to bend through narrow vessels and soft materials easily without producing dangerous mechanical loads during contacts with human tissues, as instead it could be done using traditional methods (Piorkowski et al. [2011]). The insertion into an unknown environment must be safe and accurate and therefore, during exploration through tortuous pathways inside the body, reliable methods to minimize the force interaction between tissues and continuum robots are required (Goldman et al. [2014]).

The elongated shape of these catheters enables their use in endovascular surgery in which the motion environment is made of narrow and winding cavities of the human body: the main requirement for such endeavours is the ability to access different target anatomy from access sites that are not aligned in the most direct and ergonomic position. Flexible access surgery characterizes these novel techniques for the next evolutionary phase of

MIS.

Catheters can be definitely considered essential instruments used in cardiology. Whereas in the past they were designed as flexible tubes, used for drug dealing or monitoring, nowadays catheters have evolved into more dedicated instruments with additional tip functionality (Ali et al. [2016]). A very large variety of commercially available catheters exists, being adopted in different treatments like heart valve repair or replacement, ablation of atrial fibrillation (AF) or repair of congenital heart defects.

However, the position of the catheter's tip inside the heart must be as precise as possible and remains one of the most important challenge in interventional cardiology due to the heart's continuous motion generated by breathing and the heart-beat. Additionally, the shape of the vessels, the complex three-dimensional structure and the lack of vessel wall support in the heart further limit the ability for accurate steering.

### 1.3 Catheter's control

In an effort to overcome these complications in catheter-based therapies, both manually and robotically steerable systems have emerged in the past years. Several studies have shown improvements in operation time and patient safety by using steerable catheters in cardiology instead of conventional unsteerable ones. Furthermore, in some cases, the use of steerable catheters even allowed successful treatments in cardiac patients who couldn't be treated otherwise. These new devices can be called continuum robots and do not contain rigid links and rotational joints; instead they bend continuously along their length and produce motion performing smooth curves (Robinson and Davies [1999]). Continuum robots can be classified in three different groups:

- Single backbone: Central elastic structure that supports the passage of actuation/transmission elements.
- Multi backbone: Multiple elements in parallel
- Concentric tubes: Multiple elements arranged one inside the other

However, with the various degrees of steerability being available and numerous techniques being developed to achieve this steerability, different steering options have been developed.

Based on the actuation method in the distal tip, steerable catheters were labeled under the term ‘Actuation-Dependent Steering’ to differentiate them from the conventional ones. Subsequently, two categories of steering were identified (Ali et al. [2016]):

- Force transmission to tip, (Camarillo et al. [2008b])
  1. Hydraulic chamber actuation
  2. Micro-motors
  3. Pneumatic actuation
  4. Tendon and cable actuation (Vrooijink et al. [2013])
  
- Force generation in tip
  1. Electric actuation: Based on the direct or indirect response of materials to an electric current by bending
  2. Thermal actuation: Generation of a bending force and motion when thermally active elements are actuated by an electric current. When placed in the tip, these materials respond to a sudden change of temperature generating a deflection of the catheter
  3. Magnetic actuation

## 1.4 State of the Art

### 1.4.1 Magnetic Control

Magnetic actuation is based on alterations of a magnetic field, either internal or external, which is applied to the tip of the catheter having electro-magnetic elements inside its body. In the state of art are present different types of catheters with magnetic components

inside them and also the number of components is variable according to the different applications or researches (Liu et al. [2016]).

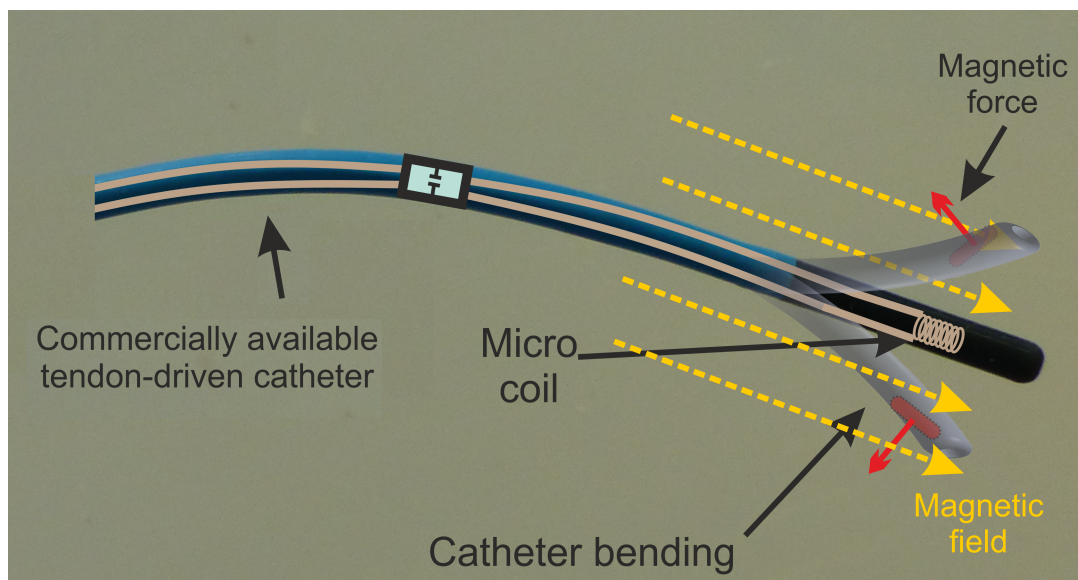


Fig. 1.3: In the picture is shown a common tendon-driven catheter with a coil inserted in the tip. In yellow the external magnetic force applied to steer the catheter (<http://www.surgicalroboticslab.nl>).

Magnetically-actuated continuum manipulators are steered by introducing them an external magnetic field, thus exerting a torque on their magnetic components. The magnitude and the direction of the resulting magnetic torque are determined by the properties of the magnetic element as well as the magnetic field. In (Le et al. [2016]) for example, the external magnetic field is generated by driving the electromagnets with a set of electric currents, called ‘driving current’: once the workspace is calibrated with respect to the external magnetic field, by controlling its orientation and magnitude, the magnetic component of the catheter’s tip can be steered inside the workspace. Therefore the magnetic field generated by the eight electromagnet systems has been accurately modeled.

Magnetic control of the catheter allows to minimize interactions with the body in which it is inserted keeping invasiveness very low. For devices controlled by cable, for example, there is a higher risk of friction which increases with the length of the device itself. With no need of transmission elements, magnetic devices can have a simple structure ensuring

high accuracy as the tip is directly actuated.

The only risk of these equipment is the high temperatures that are generated on the coils crossed by current to generate the external magnetic field. Therefore, distance between the patient and the magnetic system is a factor not to be overlooked in the design of these devices.

These technologies allowed an effective control of this type of devices making them usable also in tough tasks of surgical field.

There are three different types of devices to exert magnetic actuation:

- Mobile permanent magnets
- Static coils
- Mobile coils

In the state of art there are many examples of devices that use this kind of technology (Sikorski et al. [2017]),(Le et al. [2016]). With the introduction of these new systems in the clinical field, for example, the surgeon no longer has to insert the catheter into the patient's vessel manually during the surgical operation. Manual operation is usually very imprecise and requires skilled personnel in order to perform the task correctly without endangering patient's safety.

In addition, with respect to conventional surgical operation, techniques using flexible instruments presents a higher degree of difficulty. The relationship between hand gestures and the actual instruments trajectories is not that linear and visual or tactile feedback systems often do not provide an efficient support to the surgeon during complex steps of the operation.

The introduction of partially and fully automated steering mechanisms has shown potential facilitations for many endovascular surgery procedures, improving the outcomes in terms of safety, precision and invasiveness (Ali et al. [2016]).

However, these new automatic steering systems require real-time information during the surgical operation in order to perform the task in an appropriate manner.

## 1.4.2 Shape Reconstruction Techniques

One of the main parameters to be monitored in real-time is the shape of the manipulator. Indeed for rigid robotic manipulators the end-effector position and the joint angles would be enough for a good control; however, in continuum manipulators the flexibility introduces motions that are much more difficult to measure and predict. For a flexible device, there are not well-defined joints and the continuum manipulators articulate by means of material compliance (Camarillo et al. [2008a]). To overcome this problem, sensing for flexible devices had been developed.

In the state of art can be found examples of strain measurement to quantify the articulation of the catheter applying piezoelectric actuators and sensors to the device's body. Applying these types of sensors can be really difficult considering the thin and reduced size of the catheters.

Fiber optic sensors have also been developed for measuring strain of flexible objects: as an example Fiber Bragg grating (FBG) is a type of distributed Bragg reflector constructed in a segment of an optical fiber.

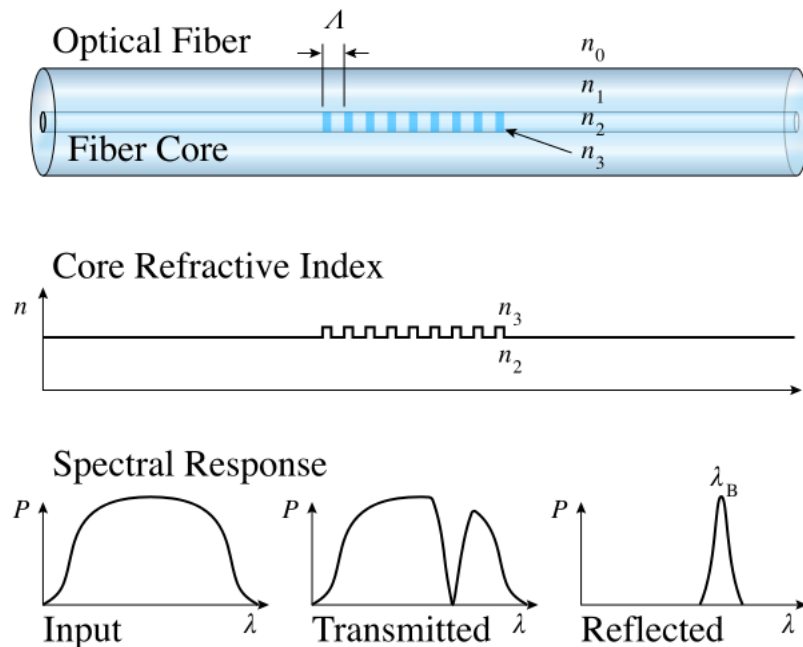


Fig. 1.4: On the top the structure of a Fiber Bragg Grating, in the middle the refractive index profile of the inner part of the fiber and in the bottom the spectral response.

This sensor has a different reflection behavior with specific wavelengths varying the shape of the optic fiber. In (Roesthuis and Misra [2016]) FBG sensors are used both for 3-D shape reconstruction of a continuum manipulator and as a feedback for closed-loop control of the manipulator demonstrating also the possibility of FBG sensors to measure interaction forces while using them for shape sensing. In order to have a good reconstruction of the 3-D shape, arrays of FBG must be placed in several locations among the manipulator shaft.

Many applications use FBG technology for shape sensing as a technology that guarantees good accuracy as demonstrated by the state of the art (Roesthuis and Misra [2016]) in which the average error in the estimation of the tip position is below  $1mm$ . It also provides information at a sufficiently high speed for real-time applications without increasing the invasiveness of the object on which it is applied.

However, shape sensing based on strain measurement can be inaccurate due to incomplete and over-simplified models. This is one of the reasons why other shape-sensing technologies evolved in the last decades exploiting stereo vision and ultrasound systems, as it is possible to obtain a reconstruction of the studied device from the diagnostic images they can generate. Stereo vision systems usually use silhouettes-based or Voxel carving technologies to reconstruct the 3-D object (Camarillo et al. [2008a]).

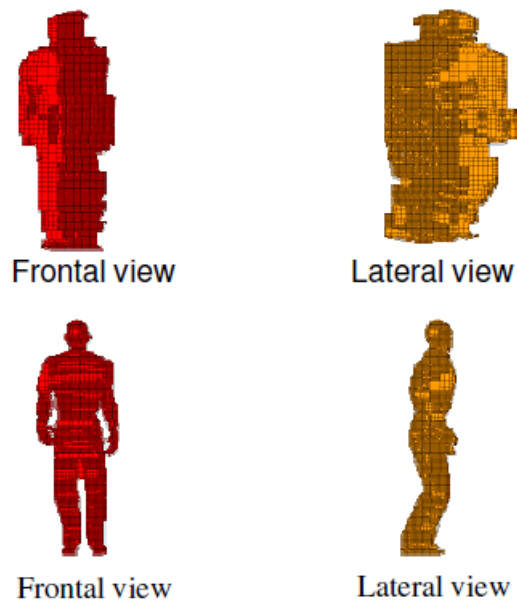


Fig. 1.5: An example of 3-D Volume Reconstruction with Voxel Carving technique. The resolution of the volume reconstructed is proportional to the number of cameras mounted on the setup. Above a setup with two cameras, below a setup with six cameras.

Voxel carving is a particular volume reconstruction technique that exploits images from calibrated cameras and their orientation to reconstruct the volume of an object. The object is then described as point cloud situated with a precise position and orientation in the workspace.

Voxel carving technique is commonly used in many tracking systems in the state of art (Sikorski et al. [2017]).

### 1.4.3 Model-Based Tracking System

Automated detection and tracking of robotic instruments is an important problem in minimally invasive surgery. Due to the indirect visualization of the workspace during MIS intervention, image guidance is necessary to improve the accuracy of the procedures: with intraoperative imaging, and the use of a patient specific model, it becomes possible to guide the robot along a planned path, minimizing risk for the patient.

Indeed, in order to perform this type of intervention effectively, it is necessary to know



and monitor the position of the catheter throughout the operation, paying particular attention to the position of the tip in the workspace. Monitoring the tip is crucial as the drug is supplied right from the end of the catheter as well as surgical instruments.

By automatically detecting and tracking instruments, it is possible to provide enhanced navigational cues to the clinician in the form of overlays of virtual fixtures and is also possible to employ image-based servoing to perform certain tasks of a procedure using medical robotics.

For this purpose, the surgical rooms are equipped with systems for real-time monitoring of instruments used by the surgeons during these types of intervention. In order to determine the position of the catheter tip in the surgical room, it is important to map the local coordinate system of the catheter into the global coordinate system of the magnetic system or the surgical room and retrieve information about its shape and orientation.

Unlike a rigid robotic manipulator which employs encoders and potentiometers to measure joints angle and determine the end effector position, continuum manipulators are flexible introducing motions much difficult to measure.

Multiple sensing modalities are used to track these new surgical instruments, such as electromagnetic tracking (EMT), image-based tracking, ultrasonic sensor tracking (UST), optical tracking (OPT), mechanical tracking, as well as some hybrid systems integrating more than one of these modalities (Vrooijink et al. [2013]), (Reiter et al. [2011]) and (Chmarra et al. [2007]).

In addition Ultrasound imaging can be used to aid with guidance and soft tissue identification in MIS, X-rays together with contrast material can be useful for endovascular application and MRI technology is exploited in cardiac procedures and research (Vrooijink et al. [2013]). This technology is currently used in medical imaging such as echography, but can also be used to monitor the position and the behavior of a surgical object during an intra-body operation.

However, the ultrasound system has an important disadvantage compared to the optical system: the ultrasound technology does not allow a volume scan in real-time with the use of simple 2-D probe that scan only one 2-D slice at a time. With this technique is not possible to have information about the whole workspace in real-time as it would be necessary to move the probe very quickly in order to scan all of it. Moreover, the probe

would not provide enough frames per second to perform the reconstruction of the entire volume in real-time with a high resolution.

Therefore there is no perfect methodology and for each type of application it is necessary to evaluate the optimal tracking or monitoring system.

### *Ultrasound System*

In the state of the art there are tracking systems that use ultrasound technology to monitor and model fixed or moving objects inside a body. Sounds or sound waves are radiations of a mechanical nature and, for this, they need a material to propagate.

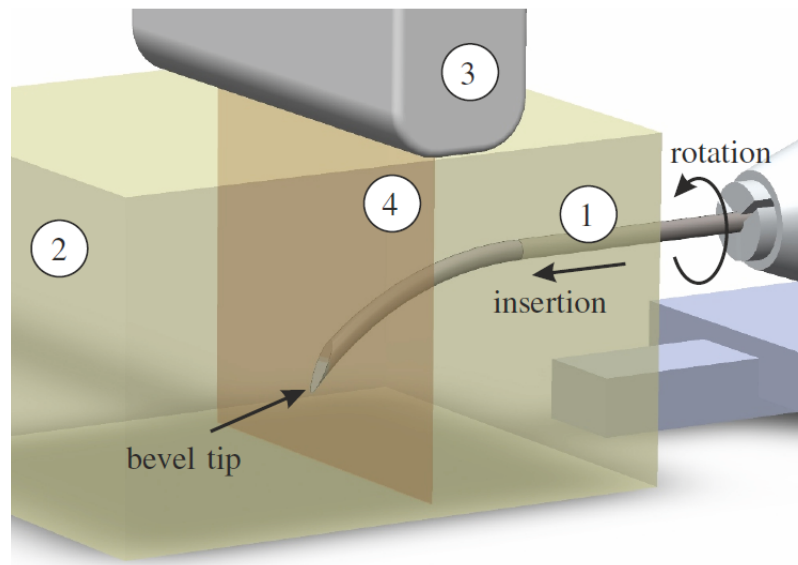
Ultrasounds are very high frequency sound waves and are so called because they are not audible to the human ear, which is able to perceive sounds in the range of 20-20'000 Hz, which is the unit of measurement of the frequency of cyclical phenomena and is equal to one cycle per second; ultrasounds are located above 20'000 Hz.

Ultrasound systems usually include an instrument called probe or transducer which is responsible for the production and reception of ultrasounds.

The ultrasounds emitted by the probe pass through the tissues with a speed and, above all, with an impedance that will be characteristic of each tissue. During the crossing of the various tissue structures, the energy possessed by the ultrasound is progressively attenuated mainly for reflection, transmission and refraction.

The change of impedance between different materials generates echoes which are received by the probe and processed, so the diagnostic images that exploit this technology usually present a good contrast between different materials highlighting the boundary that separates them. In contrast to other types of waves such as X-rays, ultrasound technology is not harmful to body tissues, but it can not be used to study tissues very deeply due to the rapid attenuation along the body path that characterizes the ultrasounds.

Therefore, accurate ultrasound tracking systems allow to recognize different materials that make up an object inside a workspace (Vrooijink et al. [2013]),(Abayazid et al. [2015]).



*Fig. 1.6:* (1) Needle (2) Soft-tissue phantom used to simulate a body-like behavior of the material (3) Ultrasound Transducer used to track the needle (4) 2-D slice obtained by the ultrasound probe which provides a cross-sectional view of the needle. During insertion the ultrasound probe is robotically repositioned to track the tip in real-time (Vrooijink et al. [2013]).

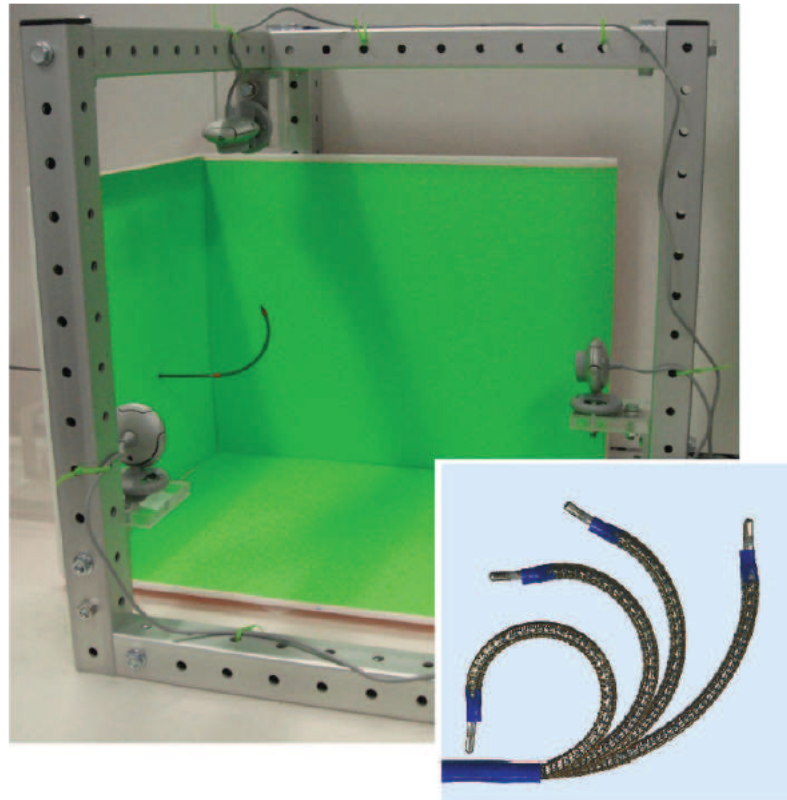
### *Stereo Vision System*

Computer stereo vision is the extraction of 3-D information from digital images. By comparing images of a scene from two cameras in two different positions, 3-D information can be extracted by examination of the relative positions of objects in the two panels.

One of the most important information for surgical purposes is the position of the objects in the workspace. Through a stereo visual system, it is indeed possible to obtain the position of all the devices inside the workspace with respect to a pre-established common reference system (Reiter et al. [2011]).

For example in (Camarillo et al. [2008a]) is described a new vision based system for estimating and registering the full 3-D shape of a continuum manipulator in real-time using a technique called shape-from-silhouette, which is a tracking algorithm usually applied for human body pose estimation. This algorithm doesn't require markers, but multiple cameras are placed at different viewing angles around the workspace, and each camera

extracts a silhouette of the moving target. By projecting these silhouettes and finding their intersections is possible to obtain the final 3-D shape of the object. The mean Relative Error in the estimation of the tip position reported in the article is 4,9%.



*Fig. 1.7:* Vision-based 3-D shape sensing system. The manipulator shown is an intracardiac catheter produced by Hansen Medical (Camarillo et al. [2008a]).

However, this system has limits in the surgical field: indeed, as it is possible to see in the figure ( 1.7), the workspace is well defined and the catheter is easily recognizable with respect to the background and it is in direct contact with the cameras arranged around it. Therefore, this method can be used only in the absence of materials or objects around the catheter that could limit or prevent visibility.

For example, by using stereo vision systems exploiting only the light of the visible, it is not possible to track a moving object inside the patient's body without open-surgery due

to the presence of the body structures around the region of interest.

### 1.4.4 Linear Fitting Technique

As in the example above, tracking systems do not have the only function of recognizing an object with respect to a workspace, but they can rework the information collected to obtain a model of the observed object. Subsequently, the model derives useful information on the object itself. For example a point cloud without processing provides few information about the shape of the device. It is therefore necessary to derive a model that describes the dataset as accurately as possible in order to retrieve such information. It may involve either interpolation, where an exact fit to the data is required, or smoothing, in which a ‘smooth’ function approximately fits the data.

Linear fitting technique is a mathematical procedure for finding the best-fitting curve to a given set of points by minimizing the sum of the squares of the offsets (‘the residuals’) of the points from the curve. A commonly used technique in literature to obtain useful information from the point cloud of a thin flexible catheter is to approximate it to a series of links connected by rotational joints. Linear least square is the method of solving mathematics/statistical problems that can be used for this purpose: indeed, linearity allows to approximate a dataset of points to the segment that best approximates them (Camarillo et al. [2008a]).

Therefore, from the initial point cloud obtained with the previously described tracking systems, it is reached a simplified model of the continuous manipulator. The modeling of a flexible object is indeed very complex because it can be considered as an object with infinite degrees of freedom and therefore requires simplifications especially if the purpose of modeling is real-time monitoring and control of the device itself. Due to the flexible nature of continuum manipulators, steering them is challenging and requires models that describe the manipulator shape during steering.

Therefore it is not sufficient to approximate the point cloud as a single segment because it would simplify the model too much and would not adequately describe the elastic characteristics and behavior of the continuum manipulator. Multiple links with rotational joints

allow to approximate better the real shape of the catheter with the relationship between each couple of links described by the bending angle of the rotational joint between them. A model with these characteristics that is widely used in the literature is called Pseudo-Rigid-Body Model (PRBM).

### 1.4.5 PRBM Technique

The Pseudo-Rigid-Body Model was introduced by Larry L. Howell in 1994. It is an approximation technique for modeling a compliant/flexure mechanism, especially for those where the compliant joints experience large deflection and lead to non-linear behavior. It can be considered a bridge between classical rigid-body mechanism theory and compliant mechanism analysis and offers a good trade-off between computational complexity and accuracy by reducing the dimensionality of the system while retaining its compliant characteristic. It also consists in a serial chain of rigid links connected by revolute joints. The elasticity of a continuum manipulator can be modeled by torsional springs linked to the joints in order to reconstruct its behavior. In the state of art are present multiple studies about compliant mechanisms modeling via the PRBM (Goldman et al. [2014]), (Greigarn and Çavuşoğlu [2014]), (Gravagne and Walker [2002]).

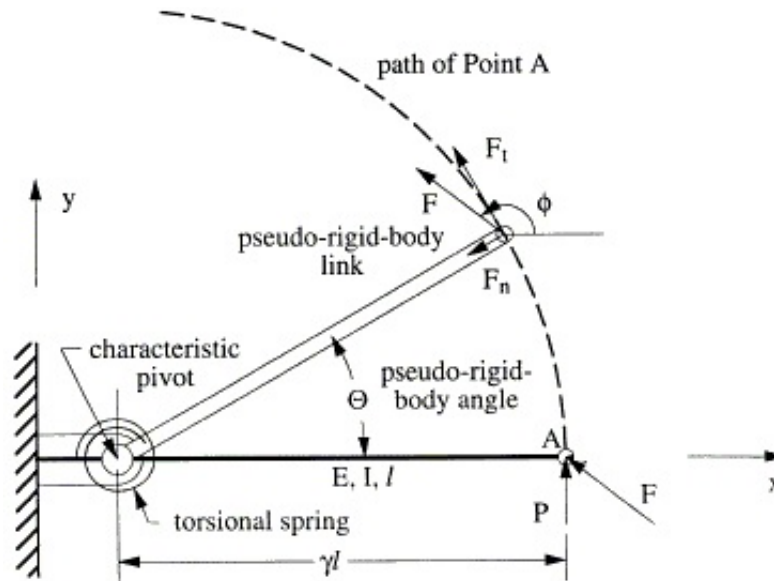


Fig. 1.8: Representation of a PRBM with one single joint. The force  $F$  applied to the tip generates a rotation of the joint which characteristics are related to the torsional spring applied to the joint itself (www.ScienceDirect.com).

This model is then reconstructed with the information collected from the object during the movement.

The great potential of the PRBM compared to the simple curve fitting is that it is possible to implement a relationship between the configuration of the model and an external force acting on it, thus not depending on the quality of the tracking. Using this technique, the model reconstruction is not based on the point clouds obtained from optical or ultrasound systems, but from the forces acting, for example, on the end-effector of the model. In case of catheters with external actuation, such as those with magnetic control, it is necessary to derive the relationship between the configuration of the continuum manipulator and the external magnetic field applied to the magnetic component of the catheter (Greigarn and Çavuşoğlu [2014]).

Thanks to this technique, a model-based tracking system based on PRBM can combine information provided either by an optical system or ultrasound, or by the control system used for the catheter's actuation. Indeed the Optical System can be exploited to obtain the position of the catheter's base in the absolute reference system, while from the magnetic

field applied to the magnetic component, known its magnetic characteristics, it is possible to derive the force or the moments acting on the model itself. Subsequently, it is possible to obtain the angles of each single joint of the PRBM and therefore the final configuration in the workspace.

This type of reconstruction does not always adequately describe the current behavior of the device as a modeling often includes simplifications: sometimes it can happen that a real behavior can not be completely defined by the model that does not adequately describe its evolution over time. The model is an approximation of reality and the simulation often generates errors that add up over time, worsening the quality of the simulation itself. To overcome this problem is necessary to retrieve information regarding the real behavior of the device in order to correct possible model's errors during activity.

### *PRBM Technique with State Observer*

A possible solution widely used in the state of art is the implementation of a state observer (Tripathi et al. [2016]), (Taylor et al. [2016]). In control theory, a state observer is a system that provides an estimate of the internal state of a given real system, from measurements of the input and output of the real system.

Given the model's state equations, it is possible to reconstruct the behavior of the model itself and perform an estimate of the future behavior if manipulator properties are known. The Kalman filter is an efficient recursive filter that evaluates the state of a dynamic system starting from a series of measures subjected to noise and can be used as a state observer. Due to its intrinsic characteristics, it is an excellent filter for noise and noise acting on non-zero Gaussian system, it is usually computer-implemented and can provide the basis for multiple practical applications.

The Extended Kalman Filter (EKF) (Yang and Baum [2017]) is a non-linear version of the Kalman filter used when the evolution or observation of the system state is non-linear. In (Hajimolahoseini et al. [2014]), for example, an Extended Kalman Filter is used as an observer to estimate and tracks the movement parameters of an object in space using the model's equations and the error in the angle evaluation is less than 0.2 degrees.



A particular type of state observer is called Luenberger observer, which introduces an additional corrective term in order to ensure the convergence of the state to a condition of stability. In particular the Luenberger observer exploit information about the outputs of the system. In literature there are not Luenberger state observers implementations in PRBM models.

## 1.5 Aim of the project

The project, in collaboration with Surgical Robotics Lab, University of Twente, The Netherlands, aimed at developing an advanced model-based tracking algorithm in order to improve the current techniques used for BigMag, a device that allows to control endovascular catheters with magnetic components through the use of six mobile coils.

The main purpose of the algorithm was to obtain information on the configuration of the catheters used for BigMag system with high precision, with a particular focus on the evaluation of the catheter's tip, fundamental for the control of the catheter itself.

A first model-based tracking algorithm has been developed with a linear fitting technique to describe the continuum manipulator as a series of rigid links connected by joints. In order to obtain this model, a discretization of the catheters in cloud-points has been performed.

Two validation methods with different protocols have been developed to evaluate its performance:

### 1. Catheter's Curvature Analysis

- Evaluation of the curvature by analyzing a synthetic catheter model with known constant curvature (with subsequent introduction of random noise)
- Evaluation of 90 models of synthetic catheters with known variable curvature
- Evaluation of a 3-D printed object with known curvatures, with data provided by the BigMag stereo vision system

### 2. Configuration Parameters Evaluation during Catheter's Animation

- Qualitative evaluation of the configuration parameters obtained from a real dataset provided by BigMag (comparison of results before and after filtering with evaluation of the position of the catheter)
- Evaluation of configuration parameters with synthetic data created on Maya
- Evaluation of the radius of curvature with synthetic data created on Maya

Subsequently, a second algorithm was implemented with a PRBM model known in the literature that allowed to obtain the catheter configuration starting from the data on the torque provided by BigMag to the magnetic component of the catheter. Therefore this algorithm did not exploit the point cloud to obtain the catheter configuration but rather the wrench applied to the catheter's magnetic tip. To evaluate and improve the quality of this model compared to the state of the art, a validation divided into four protocols was performed:

1. Evaluation of the joints angles after improving the initial compliance values of the PRBM (recalculation of the initial compliance values exploiting the new datasets)
2. Evaluation of the joints angles by introducing a cubic compliance model to the PRBM
3. Evaluation of the joints angles by introducing a Luenberger state observer to the PRBM
4. Final comparison of all PRBM models in the evaluation of the tip's position

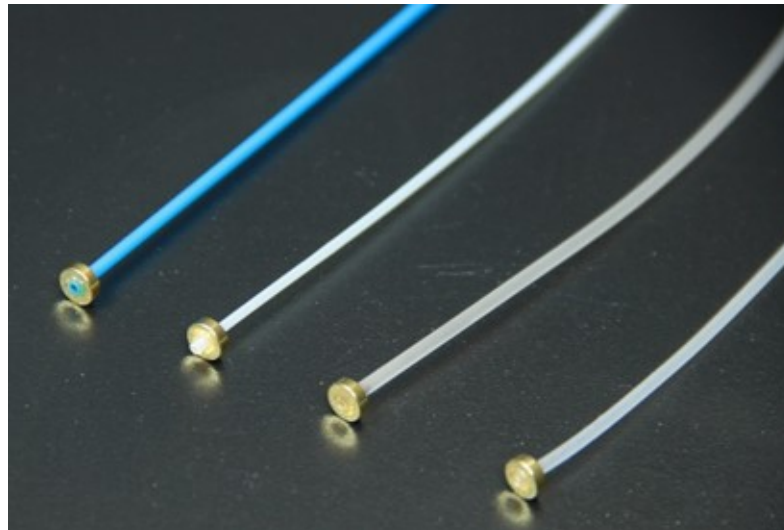
All four evaluations of the PRBM models were performed comparing the results with the information provided by the linear fitting algorithm implemented on BigMag stereo vision system and used as a reference value.

## 2

# Materials and Methods

### 2.1 Catheter's Characteristics

The catheter used for this project was designed as a  $5\text{cm}$  length and  $2\text{mm}$  diameter cylinder with a single, axially-magnetized, ring shaped magnet mounted at its tip. The presence of the magnetic component increased the diameter of the distal part of the catheter up to  $6\text{mm}$ .



*Fig. 2.1:* Examples of catheters with a magnetic component on the tip and different body's material

As shown in figure (2.1), the material of which they were composed was elastic to

allow them to bend. The only forces acting on these catheters were the force of gravity and the torque generated on the magnetic component by an external magnetic field.

### 2.2 BigMag

BigMag is a novel platform developed in the Surgical Robotics Lab of Twente University. This device uses an array of mobile coils for 3-D magnetic actuation of continuum manipulators mounted on a 1.55x1.35m aluminium structure. It is composed by six coils, held in two symmetrical mobile frames.

Three coils are suspended above the horizontal centre plane of the workspace, the other three are located below it. Such an arrangement is proposed in order to make the workspace accessible through its centre plane with 360° clearance (Sikorski et al. [2017]).

It can't provide enough workspace for trials on human-like models but is sufficient for trials on ex vivo specimen and on models of human heart. The system generates magnetic fields of at least 40mT in any direction within its workspace, and provides gradients of up  $1T/m$ . The field of each coil of Big Mag can be controlled independently. This structure is capable to apply known magnetic wrenches to a continuum manipulator in the workspace. The small size of the coils allows to reduce the inertia and inductance, increasing the bandwidth of the system.

Each mobile frame is moved by a motor and rotates independently along a common vertical axis and the current in each coil can reach 8A maximum.

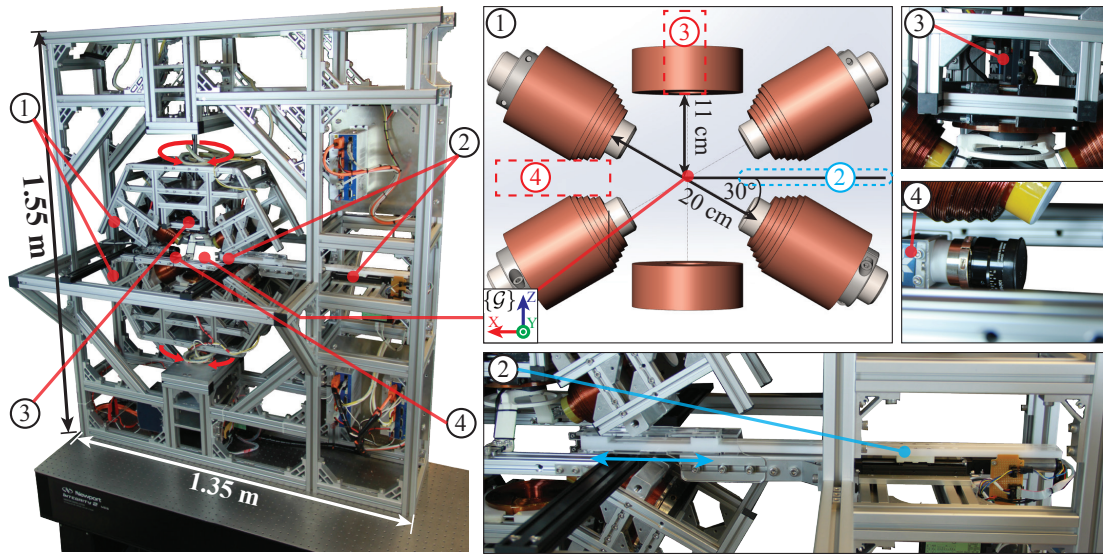


Fig. 2.2: On the left the entire BigMag system. On the right a focus on each detail: (1) Mobile Coils (2) Inserter of continuum manipulators (3) Top Camera (4) Side Camera (Sikorski et al. [2017]).

On BigMag has been developed a fully automated closed-loop steering method for continuum manipulator, under the guidance of widely used clinical tracking modalities. The two cameras used are Dalsa Genie Nano C1940 Red-Green-Blue (RGB). Different technologies can be used to monitor the catheter's movement and collect information in real time about its behavior, geometrical characteristics and position in the workspace. BigMag is equipped with an Optical Tracking System made of two cameras orthogonal to each other and arranged above and to the side of the workspace. This arrangement is made to visualize the entire workspace and allowing for 3-D reconstruction of the position of objects in it. Jelly phantoms could be inserted into the workspace to simulate the behavior of continuum manipulators in body-like environments.

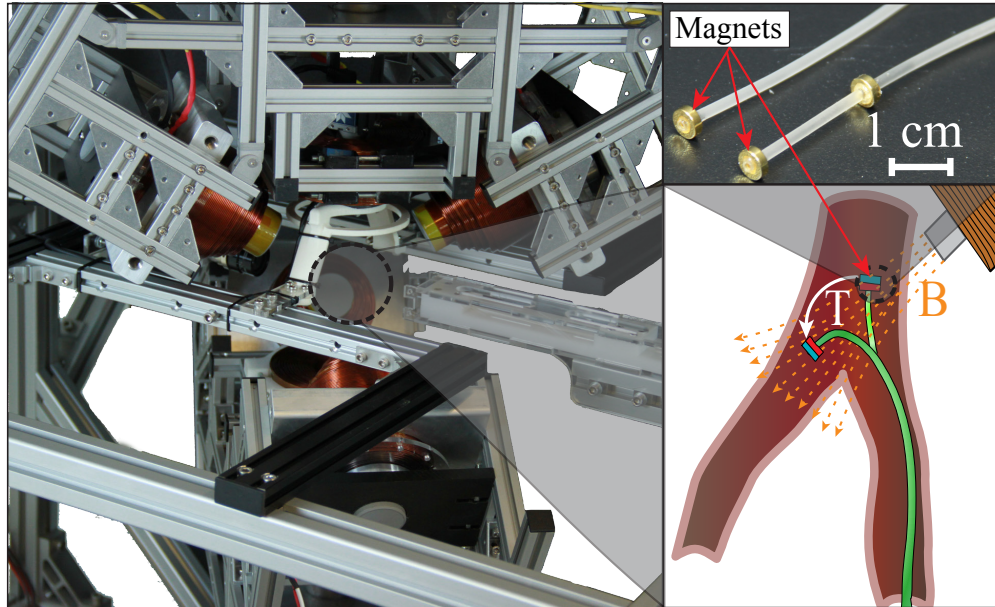


Fig. 2.3: In the picture the workspace of BigMag. In the selected region, catheters are inserted in order to be steered magnetically by the system (Sikorski et al. [2017]).

The system is also equipped with an automated advancer that allows the insertion of continuum manipulators with diameters between 1 and 5 *mm* in the workspace. The advancer is steered by a linear actuator. The field generated by the six coils is used to steer the tip of the catheter in any direction in the workspace.

### 2.3 Software for Modeling: Solidworks

For the validation methods, it was necessary to generate synthetic data for the evaluation of the first fitting algorithm. In order to generate synthetic models and extract datasets similar to those provided by the stereo vision system present on BigMag, Solidworks has been used.

SolidWorks is a three-dimensional parametric design software, produced and marketed by Dassault Systèmes. Born as software specifically dedicated for mechanical engineering, is therefore particularly useful for the design of complex mechanical devices.

The software is useful for the creation of 2-D and 3-D drawings of solids and surfaces, through a parametric and fully customizable geometric system. Solidworks allows to bring drawings from two to three dimensions and vice versa, with simple operations, im-

porting and exporting AutoCAD files (DWG, DXF).

The software is also able to import and export geometries in a large variety of three-dimensional formats and the parametric design allows you to set up many types of relationships (parallelism, concentricity, perpendicularity, collinearity, equality, coaxiality, symmetry and many others). The equations tool also allows you to set relationships between geometric dimensions.

### 2.3.1 Synthetic Model of the Catheter

The model of the catheter to obtain synthetic data has been developed using Solidworks with known geometrical information: a cylindrical body with  $5\text{cm}$  length and  $2\text{mm}$  diameter. The tip of the catheter had a bigger diameter up to  $6\text{mm}$  due to the magnetic component's dimension. The curvature has been parameterized to allow the generation of known curvature models in order to compare Solidworks real information with what has been obtained by the developed algorithm. Each catheter's model has been then exported as a STL file: STL (Stereo Lithography interface format) format represents a solid whose surface has been discretized into triangles. The STL format has advantages such as simplicity, since it is very easy to generate and to process, while to its disadvantage it has an approximate geometry and its data structure, which is simple, can present the same vertex multiple times.

The first model generated on Solidworks had a single curvature constant on the  $XZ$  plane with a bending angle between the base and the tip of the catheter equal to  $60^\circ$  as shown in figure ( 2.4).

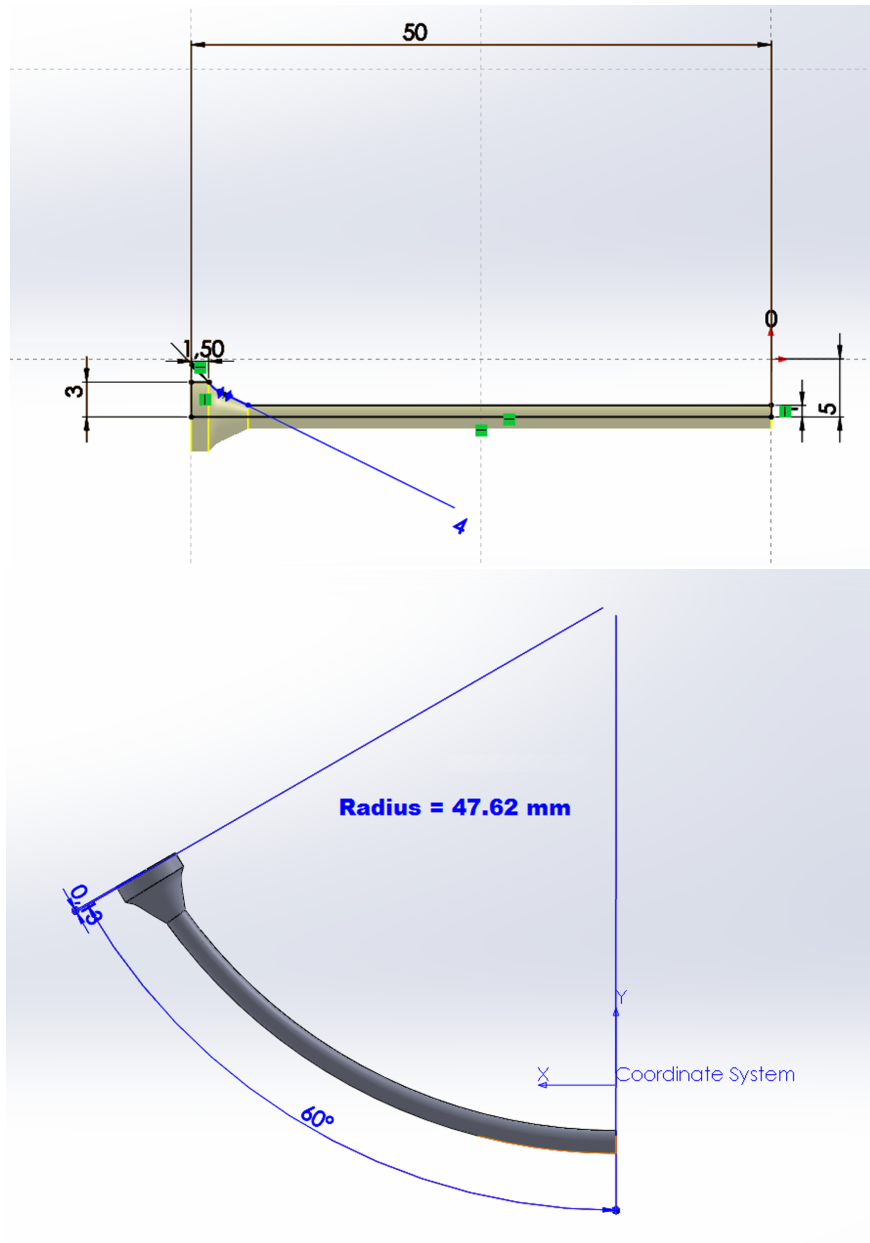


Fig. 2.4: On the top the initial catheter's model generated on Solidworks. On the bottom the catheter has been bended  $60^\circ$  with constant curvature

The catheter has been rendered in a 3-D space so as to be able to export a synthetic volume of the object to be analyzed on Matlab.

The reference frame on Solidworks has been set up to be similar to the catheter insertion position in BigMag workspace reference frame. The positioning of the reference system was necessary for the correct initialization of the fitting algorithm.



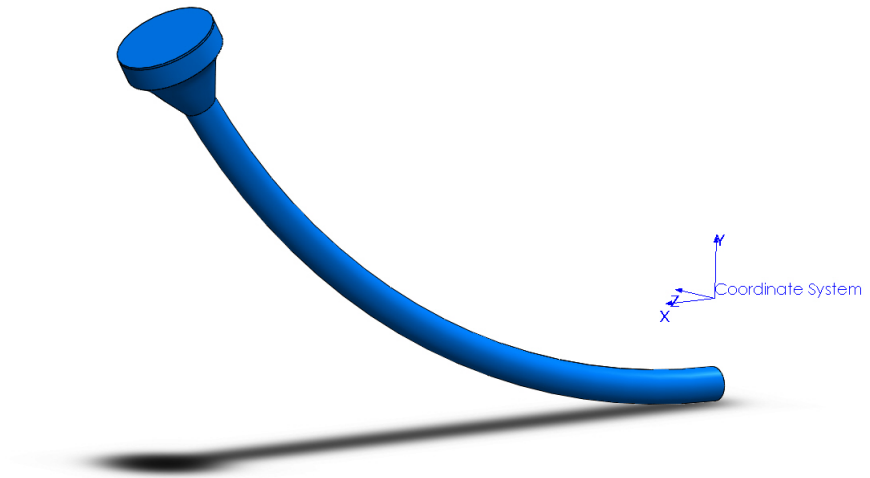


Fig. 2.5: 3-D Representation of the catheter generated on Solidworks with the respective coordinate system.

## 2.4 Processing on Matlab

### 2.4.1 Linear Fitting Technique Implementation

This algorithm has been developed and tested on the synthetic catheter point clouds simulating a stereo vision system dataset and processing. Once the points were sorted and filtered, a function that allowed to approximate the position of the center of the catheter base in the workspace has been developed, as reported in appendix (.2.8).

Subsequently, a linear regression model with the least linear squares method was used to derive a good approximation of the point clouds.

$$\beta = (X^T * X^{-1})X^T y \quad (2.1)$$

where  $X$  contains the information on the data points,  $\beta$  is the coefficient vector which fit the equations “best”, solving the quadratic minimization problem,  $X^T$  is called Gramian matrix of  $X$  and  $X^T y$  is known as the moment matrix of regressand by regressors.

The least linear squared method in equation (2.1) has been adapted to the fitting problem to find the coefficient of the line which better approximates the selected point cloud, as reported in appendix (.2.1).

The continuum manipulator has been modeled as a series of rigid links with known length connected by rotational joints. In order to obtain the model, the regression algorithm has been applied iteratively. Known the position of the initial center and the number of links of the final model, the points were selected with a distance to the initial center less than or equal to the length of a single segment, as showed in appendix (.2.4).

The position of the successive center was then obtained thanks to the linear least square method as shown in (.2.1): this method allowed to retrieve the line that best approximates the selected points; the point on the line with a distance equal to the length of the segment is set as the subsequent center that is used as the initial origin of the successive iteration. During the final step of each iteration, the points selected were removed from the point clouds so that each point could be selected just once.

## 2.4.2 Model Fitting with not-fixed number of segments

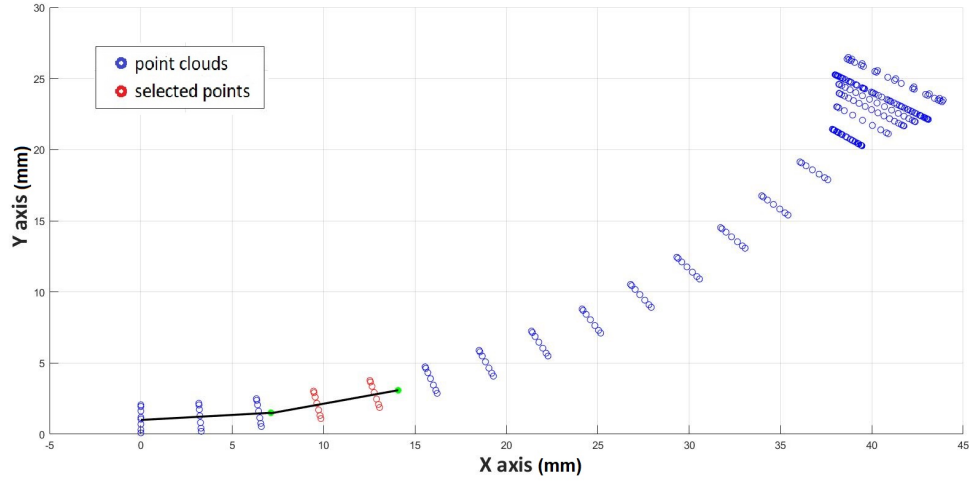


Fig. 2.6: Representation of the point clouds during one step of the model fitting iteration.

At the end of the model fitting algorithm a model was obtained with  $n$  joints connected by  $n - 1$  segments of length  $\frac{5cm}{(n-1)}$ .

Curvature of a continuum manipulator could be not so easy to be modeled with a series of links, so the number of segments of the fitting algorithm could be changed to find the best ratio between computational cost and accuracy.

## 2.5 Validation Method A: Curvature Analysis with Linear Fitting Algorithm

### 2.5.1 Initial Evaluation

Initially a geometric consideration was made: the file format STL indeed generates triangles and therefore vertices only on the surface of the 3-D object. The cloud of points obtained was then made of a series of points organized in circles whose centres lied on the hypothetical line that passed through the center of the catheter's body. The distance of the

points from the segments of the fitting model was then evaluated considering as real value  $1mm$  which was equivalent to the radius of the cylinder, as reported in appendix (.2.3).

This simple method of geometric validation was however limited to the catheter body as the diameter of the tip progressively increased up to  $6mm$  due to the permanent magnet. Furthermore, the catheter's tip was an important component to be studied because the catheter control was exerted by BigMag through the magnetic component.

The ultimate goal of the model fitting was to gather information on the geometry of the catheter in order to accurately trace the position of the magnetic component. For this purpose, to evaluate the quality of the fitting algorithm two validation methods has been used. Curvature refers to a series of geometric concepts linked to each other, which intuitively refer to the measure of how much a given object deviates from being flat. The measurement of the curvature is defined in different ways according to the geometric entity to which it is applied. The curvature of a continuum manipulator bended in a 3-D volume could be very hard to be described.

Therefore, the validation of the model fitting algorithm has been achieved considering a constant curvature of the catheter's body on a single plane: in this way the model could be considered as a part of a circumference.

In appendix (.2.2) has been implemented the function to reconstruct the circumference from three known points in order to evaluate the curvature.

The circumference offers the simple model of extrinsic curvature: since in a plane the curvature  $k$  ad a point  $P$  indicates the sensitivity of the tangent in  $P$ , circumferences with greater diameter have a smaller radius  $R$  and vice versa. The curvature is defined as the reciprocal of the radius of the osculating circle at a point.

$$k = \frac{1}{R} \quad (2.2)$$

### 2.5.2 *Experimental Protocol A1: Single Constant Curvature*

In the first validation phase, the evaluation of the curvature by analyzing a synthetic model with known constant curvature has been performed. Values obtained with the linear fitting

algorithm has been compared to the reference value given by Solidworks model. After this first comparison, we focused on making more realistic synthetic point clouds. As shown in figure ( 2.7), the distribution of the points is not homogeneous in the body of the catheter, but well organized in circles equally spaced from each other and in which the points are equidistant. Datasets from BigMag’s stereo vision system were very different because the points obtained with a voxel carving technique were much more spread in the entire volume of the catheter and not just on the surface like in the STL format.

Therefore, we introduced a random noise to the coordinates of each point in all three directions of the reference system, as reported for example in appendix (.2.5).

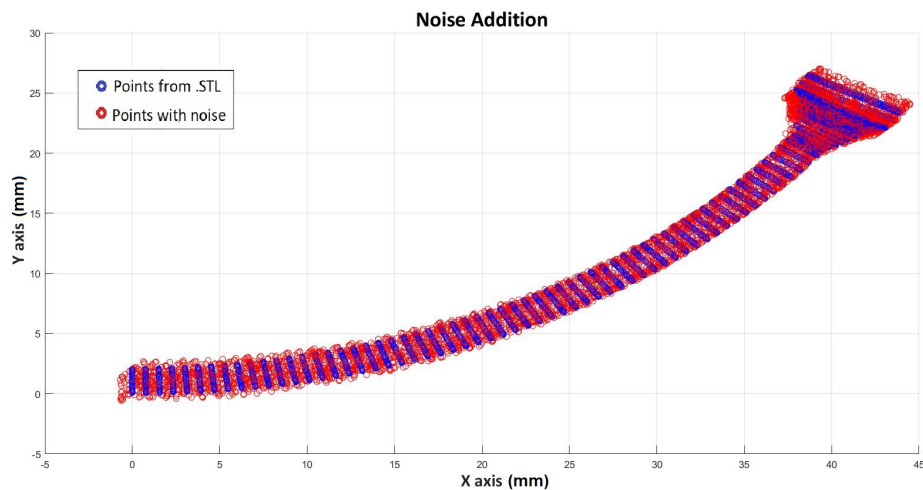


Fig. 2.7: Representation of the catheter’s model to highlight the difference before and after the introduction of noise to the point clouds. Points with noise in red show a more homogeneous spatial distribution than blu initial ones.

A multiplicative factor has also been added to the noise in order to modify its maximum amplitude to verify the different response of the fitting algorithm. With the introduction of noise in synthetic data, the point clouds was more similar to the one obtained by the system installed on BigMag.

The performance of the algorithm with and without the noise to the point clouds has been evaluated.

### 2.5.3 *Experimental Protocol A2: Synthetic Models with Variable Known Curvature*

The second validation protocol saw the generation of multiple models on SolidWorks in which the bending was not constant, as in the case of the first validation protocol, to test the robustness of the fitting algorithm. For this purpose, the function to derive the initial origin of the catheter has been modified in order to avoid errors caused by excessive bending of the catheter: a control on the distance of the recruited points was introduced to avoid points not belonging to the base of the catheter body.

On Solidworks, one hundred and eighty models of the catheter have been generated with a progressively increasing bending angle from  $1^\circ$  to  $180^\circ$  degrees. This progressive increase in the angle was designed to simulate a hypothetical movement over time of the catheter. STL files have been loaded iteratively on MATLAB in order to save the data of each model on the same array.

In this way each file could be theoretically compared to a frame obtained from the stereo vision system during a test; frame-by-frame it has been possible to derive the curvature parameter through the model fitting algorithm and compare it with the known information provided by the Solidworks models.

### 2.5.4 *Experimental Protocol A3: Evaluation of a 3-D Printed Object with Known Curvatures*

The final validation protocol of the linear fitting algorithm and the evaluation of its behavior was obtained with the following procedure:

### *Development of a catheter model in Solidworks*

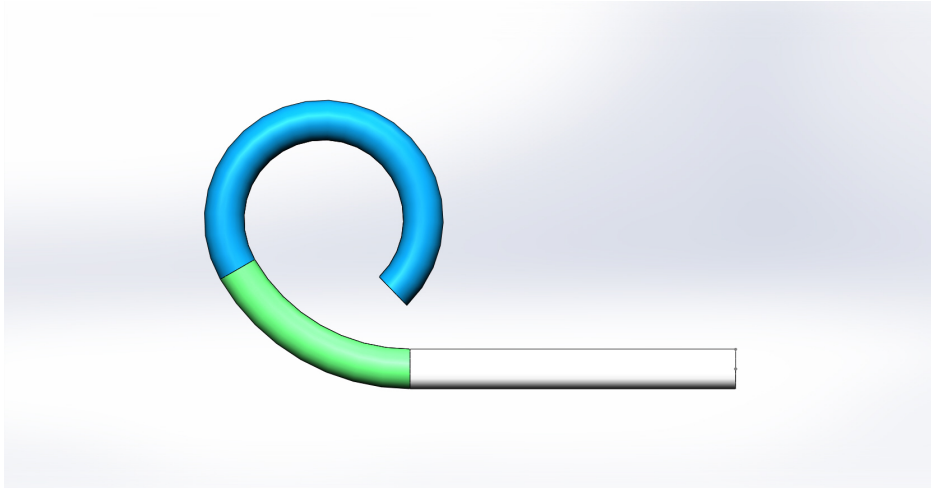


Fig. 2.8: Model of the catheter created on Solidworks. The green part has a curvature of  $100m^{-1}$ , the light blue part of  $200m^{-1}$ .

The catheter has been approximated as a cylinder with  $5cm$  length and  $2mm$  diameter. Subsequently were applied two different bendings to the model with a curvature of  $100m^{-1}$  and  $200m^{-1}$  respectively. The dimensions were therefore comparable with the real ones of the catheters used for steering.

### *Object's Fabrication*

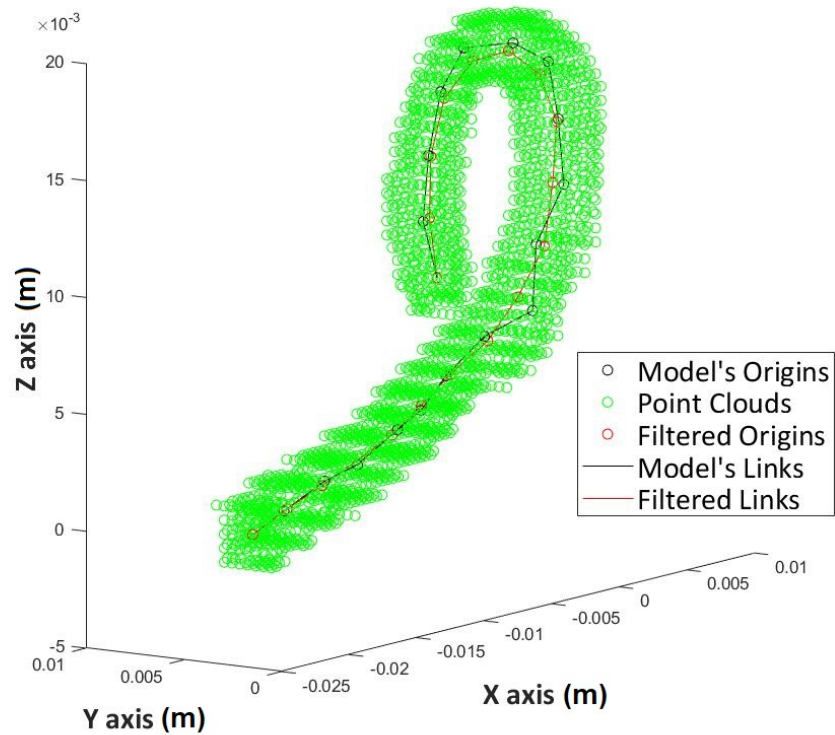
The catheter has been fabricated with a 3-D printer (resolution of  $0.2mm$ ) using the model generated on Solidworks. The material used for the fabrication was a rigid plastic material that did not allow the object to bend due to gravity during the validation operations.

### *Setup: Image Recording*

The 3-D printed object was then positioned in BigMag's workspace and eighty-five images were recorded through the Optical System mounted and calibrated on BigMag System. The object was not moved during the entire acquisition.

*Image Processing*

A Voxel Carving technique was applied by the Optical System to each couple of images, one for each camera, in order to reconstruct the volume of the catheter as a 3-D point clouds. Later on has been used the linear fitting algorithm to describe the point clouds as a series of nineteen rigid links connected by joints. Subsequently, the position of each joint has been filtered through a three-points window moving average filter.



*Fig. 2.9:* Point clouds from Stereo Vision System representing the catheter. The origins before and after filtering are represented with two different colors: black and red for unfiltered data respectively.

Curvature has been evaluated both before and after filtering.



## 2.6 Validation Method B: Catheter's Analysis during Animation with Linear Fitting Algorithm

### 2.6.1 Comparison Parameters and Animation

A first study of the movement of a magnetically controlled real catheter by BigMag was performed by making the tip of the catheter accomplish a specific circular trajectory within the workspace. This apparently simple movement allowed to verify the quality of the catheter tip control in all directions.

All the information obtained by the system during this test were saved in CSV files organized in arrays in which each row gathered all the information regarding one frame: in each row the coordinates of each voxel obtained from the stereo vision system were reported.

On these images information on the position and orientation of the catheter during the movement has been evaluated rather than the quality of the model fitting algorithm.

Being able to derive high precision key parameters frame-by-frame of catheter's configuration was an important result to be achieved.

For this purpose, three parameters were introduced, making it possible to have a complete reconstruction of the catheter configuration at every instant:

- Tip X axis Distance  $x$
- Tip Distance  $d$
- Tip Bending angle  $\alpha$

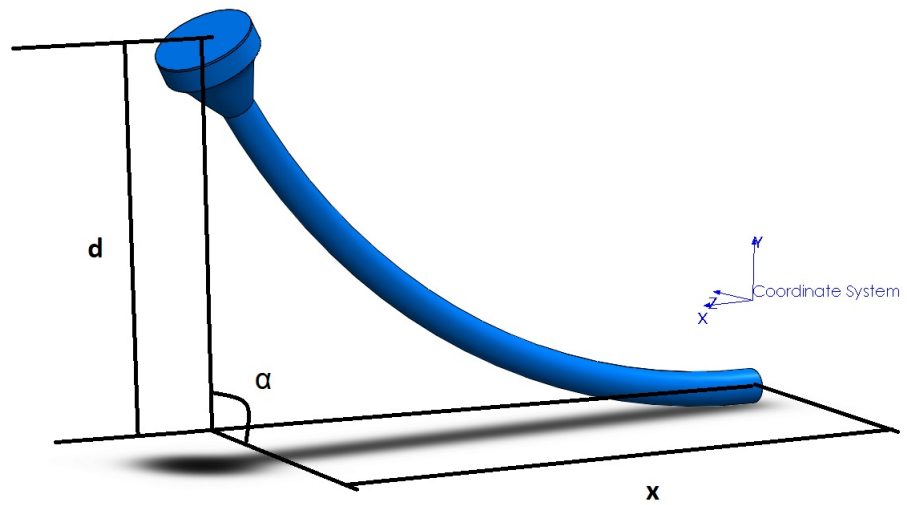


Fig. 2.10: In the picture are shown the three parameters used to describe the orientation of the catheter in the workspace during the animation.

The distance  $x$  was measured considering the distance in the X axis between the first origin of the catheter and the projection of the catheter tip P on the line parallel to the X axis passing through the origin itself.

The distance  $d$  has been instead evaluated measuring the distance between the tip position and the point P.

Finally, the bending angle  $\alpha$  was the angle in the YZ plane formed by the intersection between the line on which lay the distance  $d$  and the axis parallel to the y-axis passing through the origin at the base of the catheter. These three parameters allowed the description of the catheter configuration in the case of a constant bending of the catheter as described previously.

2.6.2 *Experimental Protocol B1: Qualitative Evaluation of BigMag dataset*

Through the linear fitting algorithm, performed iteratively for all the frames collected in the file obtained from the stereo vision system mounted on BigMag, it has been possible to obtain the three parameters throughout the catheter movement, thus being able to study its behavior. MATLAB function is reported in (.2.5).

In particular, it was decided to represent them by performing a permutation with respect to the frames they belonged to.

*Low Pass Filter*

On this data, a qualitative evaluation was carried out initially to evaluate the efficiency of BigMag in the catheter control, subsequently a low pass filter was applied to the obtained parameters.

$$y(n) = \frac{1}{M} \sum_{k=0}^{M-1} x(n-k) \quad (2.3)$$

In (2.3) a moving average filter where  $x(n)$  is the filter input,  $y(n)$  the output and  $M$  the number of points in the window.

A moving average low pass filter has been implemented on MATLAB and has been applied to the coordinates of the tip's position in each frame. The main function of the filter was to limit the oscillations at high frequency of the tip position and therefore of the three parameters, generated mainly by a not yet perfect control of the magnetic component and the accuracy limits of the model fitting algorithm.

Through filtering, we tried to make the variation of the tip's position smoother, being able to derive filtered parameters accordingly. Any sudden changes in these parameters and in the coordinates of the catheter tip would negatively affect the catheter's control, which is based precisely on the position of the magnetic component in the tip to establish the magnetic field to be generated for a given task.

### 2.6.3 *New Software for Synthetic Models: Maya*

After studying the catheter's movement in BigMag's workspace, Solidworks limits for device animation were evident: indeed, the generation of a specific movement of the catheter was anything but intuitive and easy to do, so much so that in the generation of synthetic data described previously, where the bending angle grew progressively, it was necessary to modify the curvature value before exporting each frame.

To avoid this limit, a search was carried out to identify a software that could make the catheter animation easier.

Two software with similar features to Solidworks but with a more intuitive and simple possibility to develop an animation were Blender and Maya. Maya is 3-D computer animation software with powerful modeling, rendering, simulation, texturing and animation tools.

The particularity of Maya compared to Solidworks is the presence of a very simple timeline on which is possible insert key-points. These key-points are nothing but values attributed to the parameters of interest. For example, once a value has been assigned to a parameter and is different from the initial one, in all the frames contained in the interval between the beginning frame and the key-point, the parameter progressively changes to reach exactly the value at the key-point and then remains constant until the end of the animation.

The number of key-points that can be added is unlimited, so it was very easy to make the catheter moving with a precise task.

An animation of the catheter model similar to the one seen previously was developed in order to obtain the same  $x$ ,  $d$  and  $\alpha$  parameters, thus comparing the results obtained by the algorithm by analyzing the data on the movement of a real and a synthetic catheter.

Although Maya offered a wide choice of functions to simulate the real behavior of the model, such as force and gravity fields and material's characteristics, to simulate the specific task of the catheter with circular motion already performed on BigMag it was sufficient to use the bending functions and rotation around the axes of the reference system based on the model itself.

## 2. MATERIALS AND METHODS

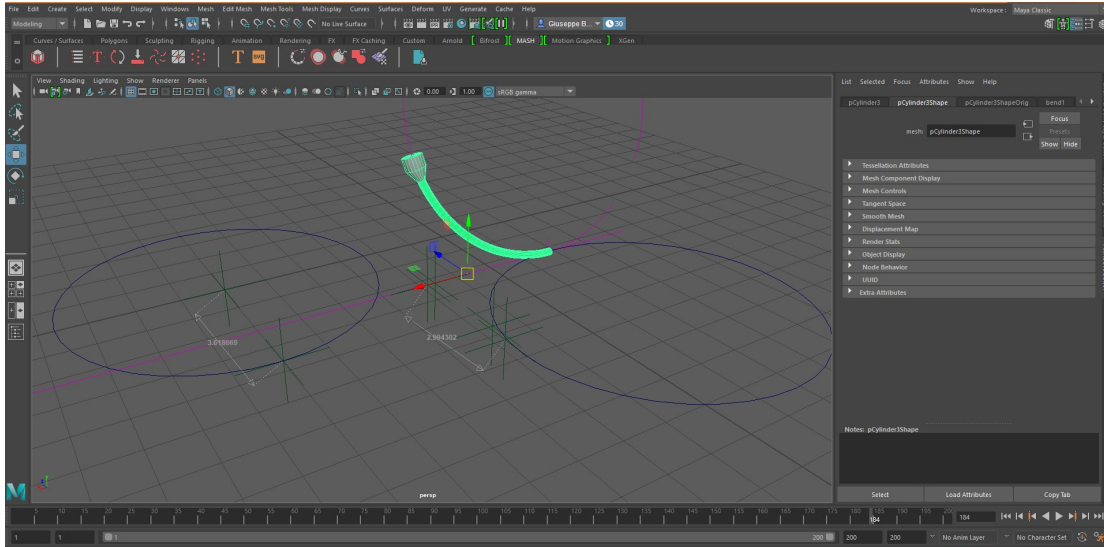


Fig. 2.11: In the figure is shown Maya platform. In this case, rotation happened around the x axis in red and the bending is constant during rotation. Below the timeline in which is possible to add key points.

In figure (2.11) the catheter is shown in green with the relative reference system. The dimensions of the catheter are similar to those of the Solidworks model and the circumferences have been traced in order to measure the curvature geometrically. Therefore, the subsequent synthetic datasets to simulate the catheter movement with the stereo vision system have been generated through this new software.

However, it was necessary to partially modify the data loading function on MATLAB because the default Maya software did not have the possibility to export the files in STL format and on the updated version there was no working patch that would allow it.

The OBJ format has been chosen as an alternative: OBJ is a file format for defining 3-D geometries. It has a data-format that represents only the 3-D geometry, that is the position of each vertex, the normals and the faces that build up the model. The vertices are stored by default in a counter clockwise order and do not have units of measure.

The movement of the catheter generated by BigMag was then reconstructed setting key-points on Maya with a constant 2-D bending and a rotation with constant angular velocity around the longitudinal axis of the catheter before bending. Naturally, at each iteration the constant curvature and the bending angle  $\alpha$  with respect to the rotation axis integral

with the base of the model were known and stored.

### *2.6.4 Experimental Protocol B2: Evaluation of Configuration Parameters with Maya's Dataset*

Known the real values of the configuration parameters provided by the Maya software, it was possible to validate the quality of the linear fitting algorithm in the reconstruction of the three parameters  $x$ ,  $d$  and  $\alpha$ .

A comparison between the results and the Maya's reference values has been performed exploiting 200 frames stored in the dataset.

### *2.6.5 Experimental Protocol B3: Evaluation of the Radius of Curvature with Maya's Dataset*

Subsequently, a further validation of the algorithm in the evaluation of the radius of curvature of the catheter was performed, exploiting the formula reported in (2.2). Two different methodologies has been compared. In the first methodology, used in the first validation phase, MATLAB processing was performed on the entire point cloud, thus simulating a stereo vision system. In the second methodology instead, the data processing was modified to simulate the use of an ultrasound system, obtaining the whole point clouds as the sum of many slices, each obtained in one iteration. The validation has been performed on the same dataset as the Protocol B2.

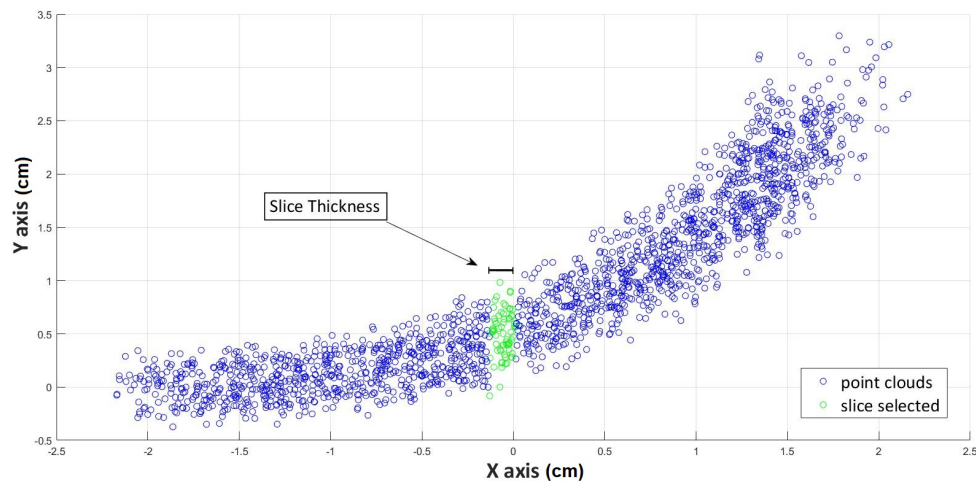
#### *Ultrasound Probe Scanning Simulation*

Since the stereo vision system was not an optimal tracking mode for objects within a volume, a possible reliable alternative system was an ultrasound system. The production of synthetic data to simulate the activity of an ultrasound scanner was then developed, in order to test new solutions. The characteristics we have considered in creating synthetic data similar to ultrasound diagnostic images, implemented and reported in appendices (.2.15) and (.2.16), are mainly three:

1. The point cloud is not read all at the same time but is generated one slice for each

iteration, since the ultrasound technology that does not allow to obtain information on the whole volume with a simple probe, but only about the single slice in which the ultrasounds are emitted. Therefore, the previous data loading algorithm has been changed and adapted to the new technology.

2. The 2-D slice generated by the ultrasound probe doesn't have a thickness that is considered only for a 3-D reconstruction of the tissue from a series of 2-D images, so we have used different values of this parameter to recruit a different number of points to describe each slice.
3. Simulation of the US scanning (Linear Actuator): the scanning direction was also designed to simulate the presence of a probe that moves at a certain velocity and direction guided by a linear motor. Due to the size of the workspace of BigMag it would have been necessary to insert a linear motor to move the probe along the entire workspace in order to scan the catheter inside the phantom; the analysis of the volume and therefore the data acquisition in the MATLAB simulation had to respect the movement due to the motor.



*Fig. 2.12:* The image shows the processing of data received from BigMag to simulate the scanning of an ultrasound probe. In this example the slice thickness was 1.5mm.

The main difference between the synthetic data and the data from diagnostic images of an ultrasound machine is the absence of the background that it is necessary to remove. Usually the tracking with ultrasound systems is equipped with tracking and filtering algorithms that allow to isolate the object studied from the background. The synthetic data that were generated on Maya already refers to the dataset already without the environment.

### 2.7 Flowchart of the Linear Fitting Algorithm

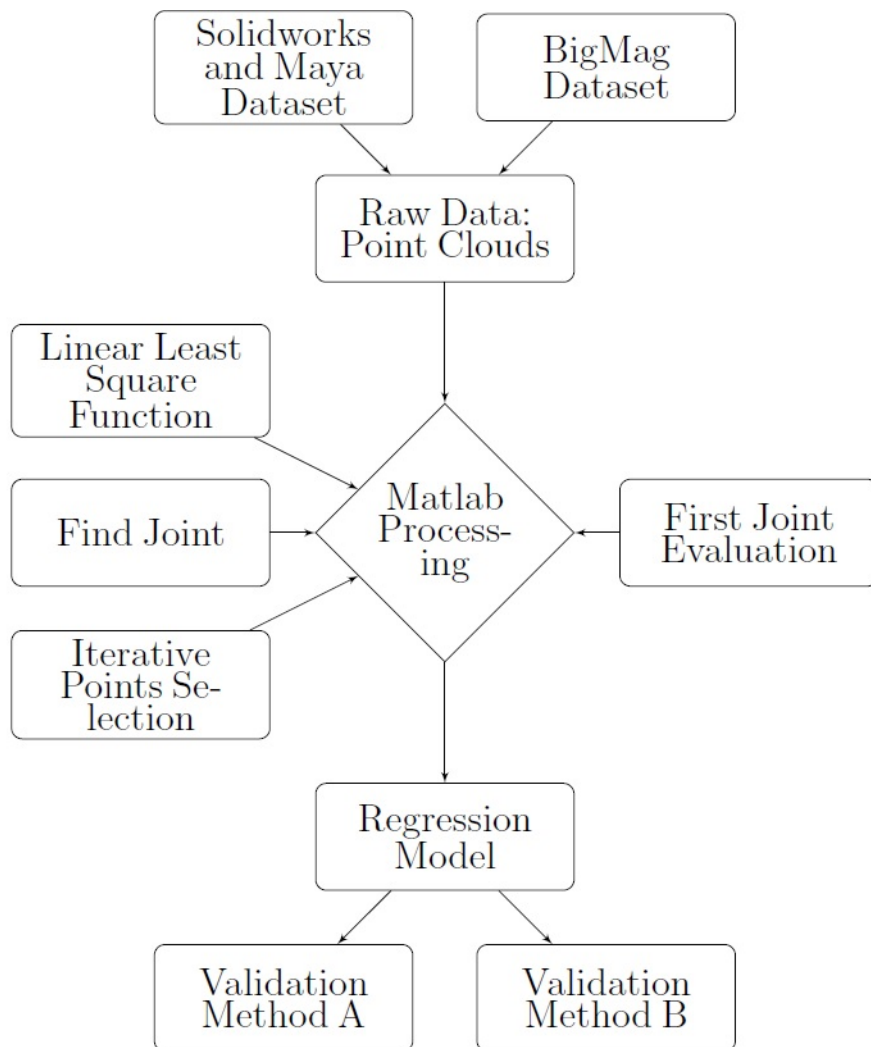


Fig. 2.13: Flowchart of the linear fitting algorithm.



At the top of 2.13 the sources from which the datasets were derived, CAD software and BigMag respectively.

Raw data, made of point-clouds coordinates, were subsequently processed on MATLAB. On the right the function in order to obtain the first joint of the regression model. On the left three MATLAB functions that were used iteratively to build the final model.

Once the model was obtained, validation methods were exploited to evaluate the performance of the algorithm. All the validation methods have been implemented on MATLAB in the main function of the linear fitting algorithm.

### **2.8 Validation Method C: PRBM Algorithm Evaluation and Validation**

After a first evaluation analysis and validation of the linear fitting algorithm, a second algorithm has been implemented on MATLAB with a PRBM model known in the literature that allowed to obtain the catheter configuration starting from the data on the torque provided by BigMag to the magnetic component of the catheter. Therefore this algorithm did not exploit the point clouds to obtain the catheter configuration. Two models have been evaluated before MATLAB implementation: Roesthuis's and Greigarn's PRBM (Roesthuis and Misra [2016]), (Greigarn and Çavuşoğlu [2014]).

#### *2.8.1 PRBM Algorithm: Description and Implementation*

A good approximation of continuum manipulators like the catheters used with BigMag could be obtained with Pseudo-Rigid-Body Models: a serial chain of rigid links connected by elastic joints with variable characteristics of rigidity could be considered sufficiently performing for the description of the catheter as we saw in the state of art.

The angles between each joint were not derive directly from the images or from Solidworks models, as the reconstruction had to be done starting from information about the magnetic field provided by BigMag.

Indeed, the goal was to derive a good and accurate relationship between joints angles and the magnetic field applied to the tip of the catheter. In addition to gravity, the only bending

component of the catheter was generated by the activity of the coils installed on BigMag. This magnetic force could be defined as wrench  $W$  and is made of forces  $f_t$  and moments  $\tau_t$  generated on the three axes. In our case, being the permanent magnet installed in the tip of the catheter, the wrench has been applied only in the tip.

$$W = [f_t \quad \tau_t] \quad (2.4)$$

It was therefore necessary to introduce a parameter  $K_r$  describing the stiffness of each joint along the axes of the reference system solidal with the catheter: this parameter has been necessary to allow a more realistic description of the continuum manipulator's stiffness. Rotational springs were assigned at the origin and the end of each link.

In the simplification of the continuum manipulator to a PRBM, constant curvature arcs have been approximated to rigid links and the energy  $U_s$  stored in the springs could be described in terms of joints bending angles (Roesthuis and Misra [2016]).

$$U_s = \frac{1}{4}K_r\theta^2 \quad (2.5)$$

Introducing the Young's modulus  $E$ , the link's length  $l_s$  and the area moment of inertia  $I$  of the manipulator, the following expression for  $K_r$  can be found:

$$K_r = \frac{2EI}{l_s} \quad (2.6)$$

Given this stiffness  $K_r$ , the torque at  $i^{th}$  joint can be written as:

$$\tau_i = K_r q_{\theta,i} \quad (2.7)$$

where  $q_{\theta,i}$  is the joint angle related to the amount of bending.

Starting from the equation (2.7), it was therefore possible to develop a relationship between torque applied to the tip and configuration of the PRBM: the equilibrium between magnetic wrench and internal bending moment of each joint was the hypothesis necessary

to find the joints angles.

### *Roesthuis's PRB Model*

The first PRBM that has been studied consider a three-dimensional shape of a continuum catheter as a serial chain of rigid links connected by rotational joints. Each links describes a constant-curvature arc with fixed length  $l_s$  and each arc is defined by a bending angle ( $\theta_i$ ) and bending direction  $\phi_i$ . In addition, the orientation of each joint is defined by two angles  $q_{\phi,i}$  and  $q_{\theta,i}$  that are related to the bending direction and the amount of bending respectively. A rotational spring ( $K_{\theta}$ ) is assigned to each joint in order to describe the manipulator stiffness  $K_r$  (Roesthuis and Misra [2016]). Since the length of the catheter is considered much higher than the diameter, shear strains are small compared with bending ones and therefore the deflection is generated by bending only. In addition, the extension of the manipulator is neglected. Each links describes a constant-curvature arc defined by the three parameters  $l_s$ ,  $\theta_i$ , and  $\phi_i$ .

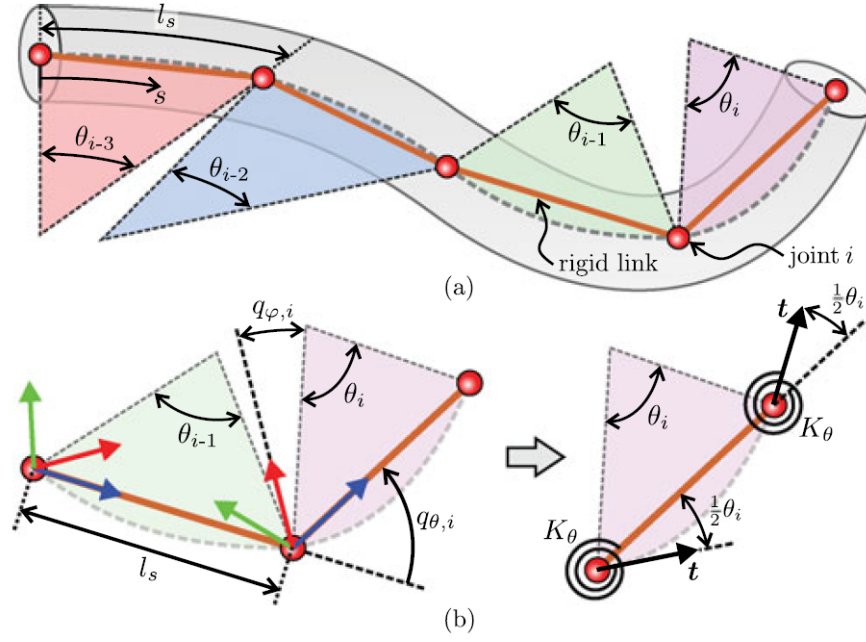


Fig. 2.14: (a) Is the shape of a continuum manipulator: the orange lines are the rigid links connected by rotational joints. Each link represents a constant-curvature arc ( $l_s$ ) that has a particular direction and angle ( $\theta_i, \phi_i$ ). (b) The first angle ( $q_{\phi,i}$ ) is related to the bending direction, ( $q_{\theta,i}$ ) is related to the amount of bending.  $K_{\theta}$  is a rotational spring assigned to each joint.

The transformation between two joints can be described by three consecutive rotation:

$$R_{i+1}^i = R_z(q_{\phi,i})R_y(q_{\theta,i})R_z(-q_{\phi,i}) \in SO3 \quad (2.8)$$

where  $R_z, R_y \in SO3$  are rotations about the z-axis and y-axis respectively.

The final transformation  $H$  between the joints  $i$  and  $i+1$  can be written as follow:

$$H_{i+1}^i = \begin{bmatrix} R_{i+1}^i & p_{i+1}^i \\ 0^3 & 1 \end{bmatrix} \in SO3 \quad (2.9)$$

where  $p_{i+1}^i$  is the position of the current joint with respect to the previous one.

Once evaluated the stiffness  $K_r$  of each joint, the relation with the torque  $\tau_i$  at each joint

can be written as:

$$\tau_i = K_r * q_{\theta,i} \quad (2.10)$$

In order to determine the manipulator shape is defined the internal bending moment  $m_i$  that is related directly to the joints angles of the rigid link model and can be defined as follow:

$$m_i^* = \sum M_j + p_F^i \times F \quad (2.11)$$

where  $q_{\theta,i}$  and  $q_{\varphi,i}$  can be related to the magnitude and direction of the bending moment.

$$K_r * q_{\theta,i} = \|m_i^*\| \quad (2.12)$$

$$q_{\varphi,i} = \angle m_i^* \quad (2.13)$$

In (2.11)  $M_j$  is the moment resulting from actuation,  $F$  denotes an external force applied to the joint. Since the position of the vector  $p_F^i$  is the result of the configuration of the PRBM, in order to calculate the manipulator shape is necessary to solve a set of non linear equation to find the most suitable angles.

From the rigid-link model is then easy to derive the manipulator Jacobian  $J(q)$  that is very useful for the control of the tip because it relates joint velocities  $\dot{q}$  to end-effector velocity  $v_e$  (Murray et al. [1994]).

The Jacobian (2.14) is the matrix of which the elements are the partial derivatives of the kinematics equations. The relationship between the end-effector velocity and the joint velocities is thus fully described by the Jacobian.

$$J = \begin{pmatrix} \frac{\partial x_1}{\partial \theta_1} & \dots & \frac{\partial x_1}{\partial \theta_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial x_m}{\partial \theta_1} & \dots & \frac{\partial x_m}{\partial \theta_n} \end{pmatrix} \quad (2.14)$$

The end-effector velocity is a linear function of the joint velocities and therefore can be evaluated using the Jacobian.

$$v_e = \begin{bmatrix} \omega_e \\ \dot{p}_e \end{bmatrix} = J(q)\dot{q} \quad (2.15)$$

where  $\omega_e$  and  $\dot{p}_e$  are the angular velocity and the linear velocity of the end-effector respectively. In Roesthuis's model, the Jacobian  $J$  can be related to the twist coordinates  $\xi_\varphi$  and  $\xi_\theta$  of the joints as expressed in the following equation:

$$J_i(q) = \begin{bmatrix} \xi'_{\varphi} & \xi'_{\theta} \end{bmatrix} \quad (2.16)$$

$\xi'_{\varphi}$  and  $\xi'_{\theta}$  are related to the direction and the bending angle respectively, that can be found directly by transforming the unit joint twist  $\xi_j$  to the current manipulator configuration.

$$\xi'_j = Ad_{H_{j-1}^o} \xi_j \quad (2.17)$$

in (2.17)  $Ad_{H_{j-1}^o}$  is the adjoint matrix, which depends on manipulator configuration (Roesthuis and Misra [2016]).

*"The most general rigid body displacement can be produced by a translation along a line followed (or preceded) by a rotation about that line."* (Kumar [2000]). Starting from the screw theory, described in Chasles's Theorem, we could say that a twist describes the velocity of a rigid body as an angular velocity around an axis and a linear velocity along the same axis, as described in equation (2.18). Therefore all points of a rigid body have the same angular velocity along the axis, but greater is the distance from the axis, greater is the velocity in the plane perpendicular to the same axis.

For a revolute joint, the axis of rotation passes through the point  $r$  and is directed along the vector  $\hat{\omega}$ . Since all joints are revolute, the unit twist of each joint with respect to the previous joint is given as:

$$\xi_j = \begin{pmatrix} \hat{\omega} \\ r \times \hat{\omega} \end{pmatrix} \quad (2.18)$$

where  $\hat{\omega}$  is the angular velocity and  $r$  the position vector from the origin of the previous joint and the origin of the current one. This PRBM has been studied from literature in order to understand the model's characteristics and behavior, but not implemented in our project.

### *Greigarn PRB Model*

In this section is described the mathematical derivation and implementation of the PRBM obtained by Greigarn, (Greigarn and Çavuşoğlu [2014]), and achieved through three steps: forward kinematics, angular velocity evaluation and the actuation of the catheter. The part of the article concerning the analysis of the movement has been neglected.

As for direct kinematics, like the previous model, the PRBM consists of a series of  $n$  rigid links connected by rotational joints and the characteristics of the material are modeled as torsional springs attached to the joints. Shear strains are neglected as described the previous model.

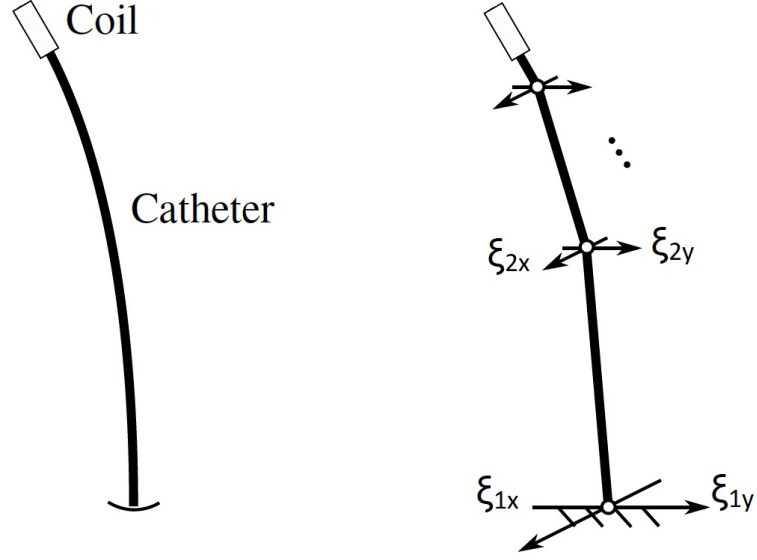


Fig. 2.15: Continuum and pseudo-rigid-body model of the catheter.  $\xi_{ix}$  and  $\xi_{iy}$  are orthogonal twists and represent the rotation of the  $i^{th}$  joint.

Each joint can be modeled as a 2-DOF spherical joint with only bending and the rotation of each joint is expressed in terms of two orthogonal twists, denoted by  $\xi_{ix}$  and  $\xi_{iy}$  among the two orthogonal axes  $\omega_{ix}$  and  $\omega_{iy}$  of the  $i^{th}$  joint. The linear combination between the two can be written as follow:

$$\omega_i \theta_i = \omega_{ix} \theta_{ix} + \omega_{iy} \theta_{iy} \quad (2.19)$$

where  $\theta_{ix}$  and  $\theta_{iy}$  are the rotation angles around each axis respectively.  $\omega_{ix}$  and  $\omega_{iy}$  are chosen to be orthogonal to the initial direction of the next link.

Therefore, the configuration of the catheter's tip can be found using the product of exponentials formula as follow,

$$g_{st}(\theta) = e^{\hat{\xi}_1 \theta_1} e^{\hat{\xi}_2 \theta_2} \dots g_{st}(0) \quad (2.20)$$

In (2.20)  $g_{st}(\theta) \in SE(3)$  is the configuration of the catheter's tip for a given set of angles  $\theta$  with  $\theta = [\theta_{1x} \ \theta_{1y} \ \dots \ \theta_{nx} \ \theta_{ny}]$ ,  $g_{st}(0)$  is the initial configuration and as



reported in (Greigarn and Çavuşoğlu [2014]),  $\hat{\xi}_i \in se(3)$  is the twists given by:

$$\hat{\xi}_i = \begin{bmatrix} \hat{\omega}_i & -\omega_i \times q_i \\ 0 & 1 \end{bmatrix} \quad (2.21)$$

where  $q_i$  is the location of the  $i^{th}$ .

Together with the forward kinematics, another parameter that is calculated is the angular velocity, which is necessary to construct the Jacobian manipulator and study the singularities of the model. The angular velocity can be expressed as the sum between the two components on the  $x$  – axis and  $y$  – axis. From the equation (2.19) could be considered  $\omega$  as a parameter that contain both magnitude and direction of the rotation.

$$\omega(t) = \omega_x \theta_x(t) + \omega_y \theta_y(t) \quad (2.22)$$

The rotation can be written as an element in the Lie algebra  $so(3)$  as follows,

$$\hat{\omega} = \begin{bmatrix} 0 & 0 & \theta_y(t) \\ 0 & 0 & -\theta_x(t) \\ -\theta_y(t) & \theta_x(t) & 0 \end{bmatrix} \quad (2.23)$$

Knowing that the magnitude of  $\theta$  can be easily obtained from  $\theta_x$  and  $\theta_y$  from the equation

$$\theta = (\theta_x^2 + \theta_y^2)^{1/2}, \quad (2.24)$$

the angular velocity can be expressed as a linear combination of the velocity axes like in (2.22):

$$\omega^s = \begin{bmatrix} 1 - \frac{\theta_y^2}{\theta^2} (1 - \frac{\sin\theta}{\theta}) \\ \frac{\theta_x \theta_y}{\theta^2} (1 - \frac{\sin\theta}{\theta}) \\ \frac{\theta_y}{\theta^2} (1 - \cos\theta) \end{bmatrix} \dot{\theta}_x + \begin{bmatrix} \frac{\theta_x \theta_y}{\theta^2} (1 - \frac{\sin\theta}{\theta}) \\ 1 - \frac{\theta_x^2}{\theta^2} (1 - \frac{\sin\theta}{\theta}) \\ \frac{\theta_x}{\theta^2} (1 - \cos\theta) \end{bmatrix} \dot{\theta}_y \quad (2.25)$$

From the equations (2.22) and (2.25), it is possible to introduce the parameter  $\Omega = [\omega_x \quad \omega_y]$ , which has been implemented on MATLAB to retrieve the angles (.2.11) as follow:

$$\Omega = \begin{bmatrix} 1 - \frac{\theta_y^2}{\theta^2} \left(1 - \frac{\sin\theta}{\theta}\right) & \frac{\theta_x \theta_y}{\theta^2} \left(1 - \frac{\sin\theta}{\theta}\right) \\ \frac{\theta_x \theta_y}{\theta^2} \left(1 - \frac{\sin\theta}{\theta}\right) & 1 - \frac{\theta_x^2}{\theta^2} \left(1 - \frac{\sin\theta}{\theta}\right) \\ \frac{\theta_y}{\theta^2} (1 - \cos\theta) & \frac{\theta_x}{\theta^2} (1 - \cos\theta) \end{bmatrix} \quad (2.26)$$

The Jacobian  $J_u$  can be calculated starting from  $J_{su}^b$  in the body frame.  $J_{su}^b$  can be expressed as follow,

$$J_{su}^b = \begin{bmatrix} \xi_{1x}^\dagger & \xi_{1y}^\dagger & \dots & \xi_{kx}^\dagger & \xi_{ky}^\dagger & 0 & \dots \end{bmatrix} \quad (2.27)$$

where for each  $i = 1, \dots, k$ ,

$$\xi_{1x}^\dagger = Ad_{g_i(\theta)}^{-1} \xi_{ix} \quad \text{and} \quad \xi_{1y}^\dagger = Ad_{g_i(\theta)}^{-1} \xi_{iy} \quad (2.28)$$

$Ad^{-1}$  is the inverse of the adjoint described before and  $g_i(\theta)$  is defined as

$$g_i(\theta) = e^{(\hat{\xi}_{ix}\theta_{ix} + \hat{\xi}_{iy}\theta_{iy})} \dots e^{(\hat{\xi}_{kx}\theta_{kx} + \hat{\xi}_{ky}\theta_{ky})} g_{su}(0) \quad (2.29)$$

Considering  $g$  as a matrix containing a rotation matrix, the inverse adjoint transformation can be calculated as below,

$$g = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \rightarrow Ad_g^{-1} = \begin{bmatrix} R^T & -R^T \hat{p} \\ 0 & R^T \end{bmatrix} \quad (2.30)$$

the final equations of the Jacobian's columns can be written as:

$$\xi_{ix}^\dagger = \begin{bmatrix} R_i^T & -R_i^T \hat{p}_i \\ 0 & R_i^T \end{bmatrix} \begin{bmatrix} -\omega_{ix} \times q_{ix} \\ \omega_{ix} \end{bmatrix} \quad (2.31)$$

$$\xi_{iy}^\dagger = \begin{bmatrix} R_i^T & -R_i^T \hat{p}_i \\ 0 & R_i^T \end{bmatrix} \begin{bmatrix} -\omega_{iy} \times q_{iy} \\ \omega_{iy} \end{bmatrix}$$

Greigarn PRBM has been implemented on MATLAB, (.2.12), to have a performing modeling of the continuum manipulator and has been used some features from the Robotics Toolbox, (P.I.Corke [2017]). As described before, the evaluation of the angles in each joint has been made not graphically from the images, but as the solution of a system of non-linear equations.

Indeed, considering the stereo vision data obtained from BigMag, initially only the wrench applied to the tip of the catheter was known, as shown in the given function reported in appendix (.2.7), and not each single moment applied to the joints of the PRBM. Therefore the last component implemented in the modeling of the continuum manipulator has been the actuation of the catheter, which allowed to relate the moments applied to the tip of the catheter to the angles of the joint through the Jacobian obtained previously. All the stresses and the angles were set to zero by default and the Jacobian was initialized with those parameters too.

Knowing from (Greigarn and Çavuşoğlu [2014]) that:

$$\tau = (J_{su}^b)^T(\theta) \begin{bmatrix} 0_{3 \times 1} \\ m \times b \end{bmatrix} \quad (2.32)$$

where  $\tau$  is the joint torque vector,  $m$  is the magnetic moment of the actuator,  $b$  is the magnetic field applied and  $J_{su}^b$  the body manipulator Jacobian, through the minimization of the difference between the current stresses ( $\tau_{total}$ ) and the product between the transposed of the Jacobian and the wrench ( $\tau_{res}$ ) it has been possible to find the optimal stress in each single joint and therefore the angles, as reported in appendix (.2.9).

$$\tau_{total} - \tau_{res} \rightarrow \approx 0 \quad (2.33)$$

$$\tau_{res} = J^T W \quad (2.34)$$

where  $W$  is the wrench described in (2.32).

Once found all the joints angles the final configuration of the PRBM would have been reconstructed.

In order to minimize this difference and find the angles, we decided to use *fsolve*.

### *Fsolve Function*

*Fsolve* is a MATLAB function that solves systems of non-linear equations of several variables. It requires an optimization function, a parameter to be optimized and general operating options such as the maximum number of iterations, the tolerance value that allows to exit the iteration.

The function in our case was the minimization of the difference described by the equation (2.33), which MATLAB implementation is reported in appendix (.2.13).

The second unknown factor in the reconstruction of the angles was the stiffness applied to each joint of the continuum manipulator. The stiffness indeed had to be certainly different between the joints given the different elasticity of the catheter base with respect to the tip for example. Starting from stiffness values inherited from previous studies, we analyzed real data obtained with the BigMag system in order to obtain the best approximation of the stiffness possible.

From the obtained PRBM, together with the applied wrench and the stiffness values, the angles of each joint has been reconstructed and compared to the values provided by the stereo vision system. This comparison has been made made in all subsequent steps to evaluate the improvement of model performances.

To simplify all the analysis, four-joint PRBM models were built and the comparison has been made between the angles in the two directions  $x$  and  $y$ , in the reference system of each joint, for each subsequent step.

## 2.8.2 Simulink Implementation of PRBM Algorithm for Validation Protocols

In order to perform the validation protocols, the PRBM algorithm has been implemented on Simulink. Simulink is a software for modeling, simulation and analysis of dynamic systems, developed by the American company MathWorks. This software is tightly integrated with MATLAB.

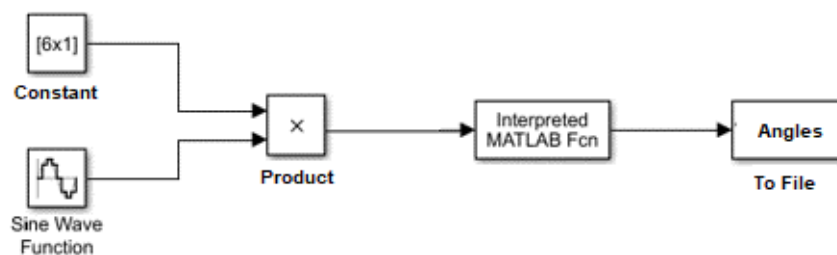


Fig. 2.16: Simulink model: the constant is the wrench applied that is multiplied by a sine function. The result of the product becomes input of the MATLAB function which provides the corresponding angles.

The first Simulink implementation has been performed to verify graphically the behavior of the model. As shown in the picture ( 2.16), it has been introduced a sinusoidal input which, multiplied to a constant wrench, provided a periodic input to the function. Through this implementation we wanted to obtain a first graphic feedback of the algorithm simulating a hypothetical wrench generated by BigMag coils, evaluating the quality of the functioning.

The algorithm that allowed to obtain the angles starting from a wrench applied to the tip was inserted inside a functional block of Simulink, called *Interpreted MATLAB Function*, in which was possible to set both the input and the output ports of the block. All the angles obtained during the simulation have been saved in order to be analyzed on MATLAB. Subsequently, the Simulink model was extended to allow the comparison between the angles obtained directly from BigMag and those reconstructed through the PRBM algo-

rithm.

Saving all information during the simulation would have allowed parallel processing of both datasets simplifying the identification of possible improvements to the previously implemented algorithm.

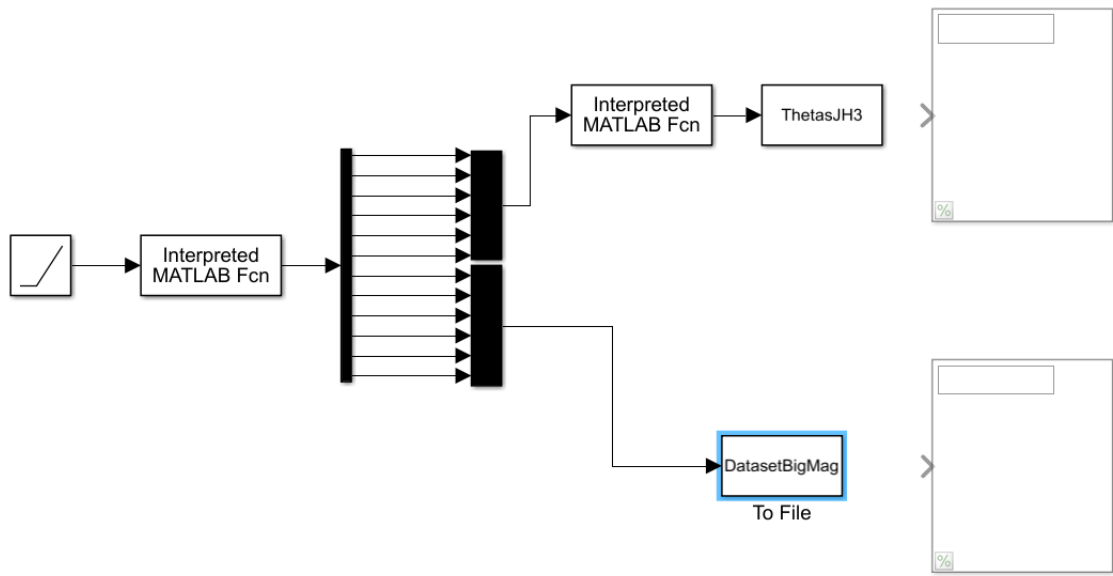


Fig. 2.17: Simulink model: the second Simulink block is shown. The first function output is wrench composed of a  $6 \times 1$  array and the angles also contained in a  $6 \times 1$  array. Highlighted in blue BigMag data storage. Display blocks used to show results during the iteration.

As it is possible to see in the image ( 2.17), the first functional block allowed to execute the algorithm of loading and processing of BigMag’s dataset thus obtaining frame-by-frame the wrench applied to the tip and the configuration of the PRB model.

The output of the first functional block was then splitted and redirected: the angles obtained by BigMag’s stereo vision system were saved directly while the applied wrench was used as input for the previously implemented functional block. The angles obtained from the second block were also saved and subsequently compared with BigMag data.

2.8.3 *Experimental Protocol C1: Evaluation of the joints angles with new compliance values*

For all four experimental protocols, the same dataset provided by the linear fitting tracking system implemented on BigMag was used and compared. This dataset has been generated by moving the catheter with a helical path and consists of 1640 frames.

*Compliance Evaluation*

Compliance is an intrinsic characteristic of a material and can be defined as the reciprocal of the stiffness (Mahvash and Dupont [2011]). BigMag stereo vision files has been processed to find the compliance of each individual test. The linear fitting algorithm implemented on BigMag allowed in fact to obtain the angles of the joints directly from the optical system; initially the angles and the wrench applied to each frame were saved and loaded on MATLAB.

Subsequently, compliance was obtained by modifying the equation previously used to obtain the angles as follow:

$$C = pinv(\theta) * J^T * W \quad (2.35)$$

where  $C$  is the compliance,  $\theta$  is a matrix  $\begin{bmatrix} 2 * n * m, & 2 * n \end{bmatrix}$ , in which  $n$  and  $m$  are the number of joints and frames respectively, composed of  $m$  diagonal matrices containing  $\theta_x$  and  $\theta_y$  angles of each joint,  $J^T$  is the transpose of the Jacobians and  $W$  is a matrix  $\begin{bmatrix} 2 * n * m, & 1 \end{bmatrix}$  containing all the wrenches applied to the tip for each frame.

The function  $pinv$ , described in appendix (.1), is the Moore-Penrose pseudoinverse and has been used the MATLAB function  $cond$  to evaluate the reliability of the results. The condition number, that is the output of the function  $cond$ , of a matrix measures the sensitivity of the solution of a system of linear equations to errors in the data. It gives an indication of the accuracy of the results from matrix inversion and the linear equation solution. Values of  $cond(X)$  or  $cond(X, p)$  near 1 indicate a well conditioned matrix.

Since the outputs from the *cond* function were less than 3, we considered the compliance's values obtained reliable.

It has been also necessary to perform a temporal shift to overcome a delay between the reconstructions of the angles and the real ones. This delay has been also assessed in obtaining the compliance the accuracy in the reconstruction of the PRBM configuration.

In the International System, flexural stiffness is measured in Newton-meters on radians, as explained in this relation  $k = \frac{M}{\theta}$ , therefore compliance  $C$  is measured in radians on Newton-meters.

Compliance values were obtained from the new BigMag datasets. With the new PRBM algorithm (called *ModelNC*), joints angles were obtained from the wrench applied at the tip. Results were compared with reference values detected by the stereo vision system (*BigMagDataset*), with the PRBM algorithm with initial compliance (*ModelQ*) and with a different PRBM algorithm including the initial bending of the catheter generated by gravity force (*ModelGQ*).

### 2.8.4 Experimental Protocol C2: Evaluation of the joints angles with a cubic compliance model implemented on the PRBM

#### *Cubic Compliance Model*

From previous studies on BigMag in the control of the catheter in the workspace, it was evident a lack of homogeneity of compliance both according to the position of the joint with respect to the catheter base, and to the variation of curvature: in the first case, the base of the catheter was less elastic than the central part of the catheter and furthermore an increase or a decrease of the bending caused a variation of the elastic characteristics of the material.

Therefore, a constant compliance might not be the best solution to have an optimal performance of the algorithm. Later on a compliance which also introduced a corrective factor to compensate for the angle's variation has been studied developed. In order to add a corrective factor, a cubic rather than linear compliance was also evaluated, such as previously calculated.

In equation (2.35),  $\theta$  was changed as follow:



$$\theta = [\theta^3 \quad \theta] \quad (2.36)$$

The cubic compliance provided two terms for each joint and for each direction  $x$  and  $y$  of which a first-degree factor and a corrective third-degree factor. It has been necessary to modify both the algorithm to obtain the angles and the function used by *fsolve* to derive the optimal configuration of the PRBM. The algorithm was then able to exploit both the terms of the cubic compliance, as implemented in appendices (.2.10) and (.2.14).

The time necessary for the *fsolve* function to derive a solution starting from the two different optimization functions was also compared to verify the possibility of implementing them in a real-time control algorithm. To evaluate the convergence speed of *fsolve* the stopwatch function of MATLAB was used, in which the command *tic* starts a stopwatch timer to measure performance.

The function records the internal time at execution of the *tic* command and display the elapsed time after the *toc* command. By inserting in the algorithm, the commands *tic* and *toc* before and after the *fsolve* line respectively it has been possible to measure its performance.

The PRBM algorithm with the cubic compliance model (called *ModelSK*) has been evaluated in comparison with reference values of the stereo vision system. Joints angles have been compared both with reference values and the best result of the previous validation protocol.

### 2.8.5 Luenberger State Observer Implementation on PRBM Algorithm

The final step of the project involved the implementation of a state observer on PRBM algorithm to limit sudden oscillations in the catheter control as shown in the tests performed on BigMag. A state observer could be very useful for this purpose because knowing the system state is necessary to stabilize a system using for example state feedback; in this way the algorithm did not obtain information about angles only from the wrench applied at the current iteration, but also taking advantage of the values obtained previously.

Therefore a state observer was designed to perform mainly a filtering that would make the reconstruction of the corners more smoothed and precise.

Analyzing the state observer to be used, the extended Kalman filter and Luenberger state observer were evaluated.

However, the extended Kalman filter has been discarded as it was not possible to adapt the merger between the algorithm and BigMag stereo vision system to the filter itself.

### *Luenberger State Observer*

Luenberger observer is a dynamic system that can be used to obtain one or some of the state of a system and it is called full-state observer when used to estimate all the states. In our project it has been adapted for a discrete-time system because we were working with images recorded by the stereo vision system.

In general, the main purpose of the observer is to obtain an estimate of a state  $x(t)$  based on the measured output  $y(t)$  and the input to the system  $u(t)$ , which are assumed to be exactly measurable.

The number of state's variable  $n$  determine the order of the dynamic system and if the observer is used as a part of a feedback control system, the estimate  $\hat{x}(t)$  will be used by the controller as if it were the true state  $x(t)$ .

There are several ways to derive the state equations for a state observer and can be added a correction term based on the output and the estimate of what that output is expected to be.

For example, for an actual system described by

$$\begin{cases} x'(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases} \quad (2.37)$$

the observer is modeled as follows,

$$\hat{x}'(t) = A\hat{x}(t) + Bu(t) + L(y(t) - \hat{y}(t)) \quad (2.38)$$

where  $\hat{y}(t) = Cx(t) + Du(t)$  and  $L$  is called Luenberger correction term and can be considered as a gain matrix for the observer. Higher is the gain, higher is the correction velocity.  $L$  is multiplied by the difference between the true output measured and its estimation.

*Model's Equations for Implementation in the PRBM algorithm*

To derive control's equations to implement the state observer it has been necessary to start from the relation between the Jacobian and the configuration of the joints described in (2.32), introducing also the force generated by gravity  $F_g$ , explaining the compliance  $C$  and the configuration of the joints  $q$ :

$$J^T(q)W_t = J^T(q) \begin{bmatrix} 0_{2 \times 1} \\ -F_g \\ \tau_\mu^x \\ \tau_\mu^y \\ 0 \end{bmatrix} = Cq \quad (2.39)$$

the differentiation of (2.39) allows to rewrite it as follows,

$$\dot{J}(q)^T W_t + J(q)^T \dot{W}_t = C\dot{q} \quad (2.40)$$

where  $\dot{q}$  is the PRBM joint rate. Using the chain rule which says that:

$$D[f(g(x))] = f'(g(x)) \cdot g'(x) \quad (2.41)$$

the differentiation of becomes:

$$\dot{J}(q)^T W_t = \begin{bmatrix} -\frac{\delta J^{3.1}}{\delta q} F_g + \frac{\delta J^{4.1}}{\delta q} \tau_x^\mu + \frac{\delta J^{5.1}}{\delta q} \tau_y^\mu \\ \dots \\ -\frac{\delta J^{3.2n}}{\delta q} F_g + \frac{\delta J^{4.2n}}{\delta q} \tau_x^\mu + \frac{\delta J^{5.2n}}{\delta q} \tau_y^\mu \end{bmatrix} \dot{q} \quad (2.42)$$

where the matrix can be defined as  $\Gamma(q, \tau_\mu)$ .

The final equation for PRB joint rate used for the Luenberger observer becomes,

$$\dot{q} = (C - \Gamma(q, \tau_\mu))^{-1} J^T(q) [0_{1 \times 3} \quad \dot{\tau}_\mu^T]^T \quad (2.43)$$

Since the data on the BigMag model were discrete data, to simplify the differentiation of the parameters to obtain  $\Gamma$ , we chose to perform a numerical differentiation, which in numerical analysis describes the algorithms for estimating the derivative of a mathematical function using the values of the same function.

The easiest way to perform the numerical differentiation is to use finite difference approximations. A two-point estimation is to compute the slope of a nearby secant line through the points  $(x, f(x))$  and  $(x+h, f(x+h))$ . Chosen a small number  $h$ , which represents a small change in  $x$  and can be either positive or negative.

The slope of the secant line differs from the slope of the tangent line by a value that is proportional to  $h$ . The true derivative of  $f$  at  $x$  is the limit of the value, for  $h$  that approaches zero, of the difference quotient as the secant lines get closer and closer to being the tangent line.

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} \quad (2.44)$$

For our purpose we used a different two-point formula as follows,

$$\frac{f(x+h) - f(x-h)}{2h} \quad (2.45)$$

This formula is known as the symmetric difference quotient and has been implemented and is reported in appendix (.3.2), where  $f$  is the Jacobian and  $x$  the configuration angles.

#### *Correction Parameter*

The gain matrix  $L$  in equation (2.38) that is used as a correction parameter and its values should be chosen so that the eigenvalues of  $A - LC$  are strictly left half of the complex plane to guarantee the stability of the system, and the estimation error should decay to

zero over time. As a first attempt  $L$  has been considered as a diagonal matrix multiplied with three constants minor than one to guarantee the stability of the observer. Has been chosen the constant 0.3, 0.5 and 0.9.

The final equation becomes:

$$\dot{q}_i = \dot{q}_{i-1} + L(C - \Gamma(q_i, \tau_\mu))^{-1} J^T(q_i) [0_{1 \times 3} \quad \dot{\tau}_\mu^T]^T \quad (2.46)$$

where  $i$  is the current iteration.

### Simulink Model of PRBM algorithm with Luenberger Implementation

The Luenberger observer has been implemented on Simulink by expanding the previous model. A data saving and reading system was generated in order to obtain information on the previous state during the iteration, thus simulating a hypothetical real-time test on BigMag.

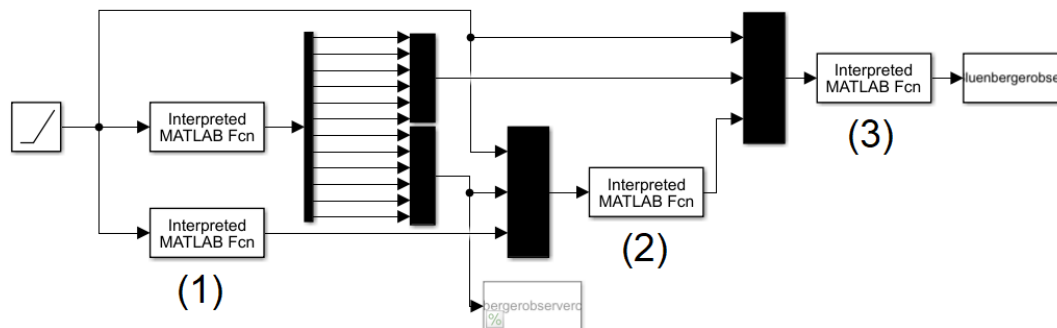


Fig. 2.18: Simulink Model: (1) is a function used to load the angles of the previous iteration (2). It chooses which angles to use depending on whether it is the first iteration or the following ones (appendix .3.1) (3). Is a function that derives the angles exploiting the observer of Luenberger. Corners are saved at the end of each iteration.

### 2.8.6 *Experimental Protocol C3: Evaluation of the joints angles with a Luenberger state observer implemented in the PRBM*

The results of the Luenberger PRBM algorithm (called *ModelL*) with different  $L$  values were then compared with the results obtained from the stereo vision system present on BigMag, always in relation with the best performances of the previous validation protocols.

### 2.8.7 *Experimental Protocol C4: Final comparison of all PRBM models in the evaluation of the tip's position*

The position of the catheter tip in the workspace was calculated as the last validation phase. For each model used for the previous validation protocols, the coordinates of the end-effector, corresponding to the catheter tip, were calculated. The data obtained from the first model (*ModelG*), the cubic compliance model (*ModelSK*) and the latest PRBM model with Luenberger (*modelL*) were compared to the BigMag reference values.

## 2.9 Flowchart of the PRBM Algorithm

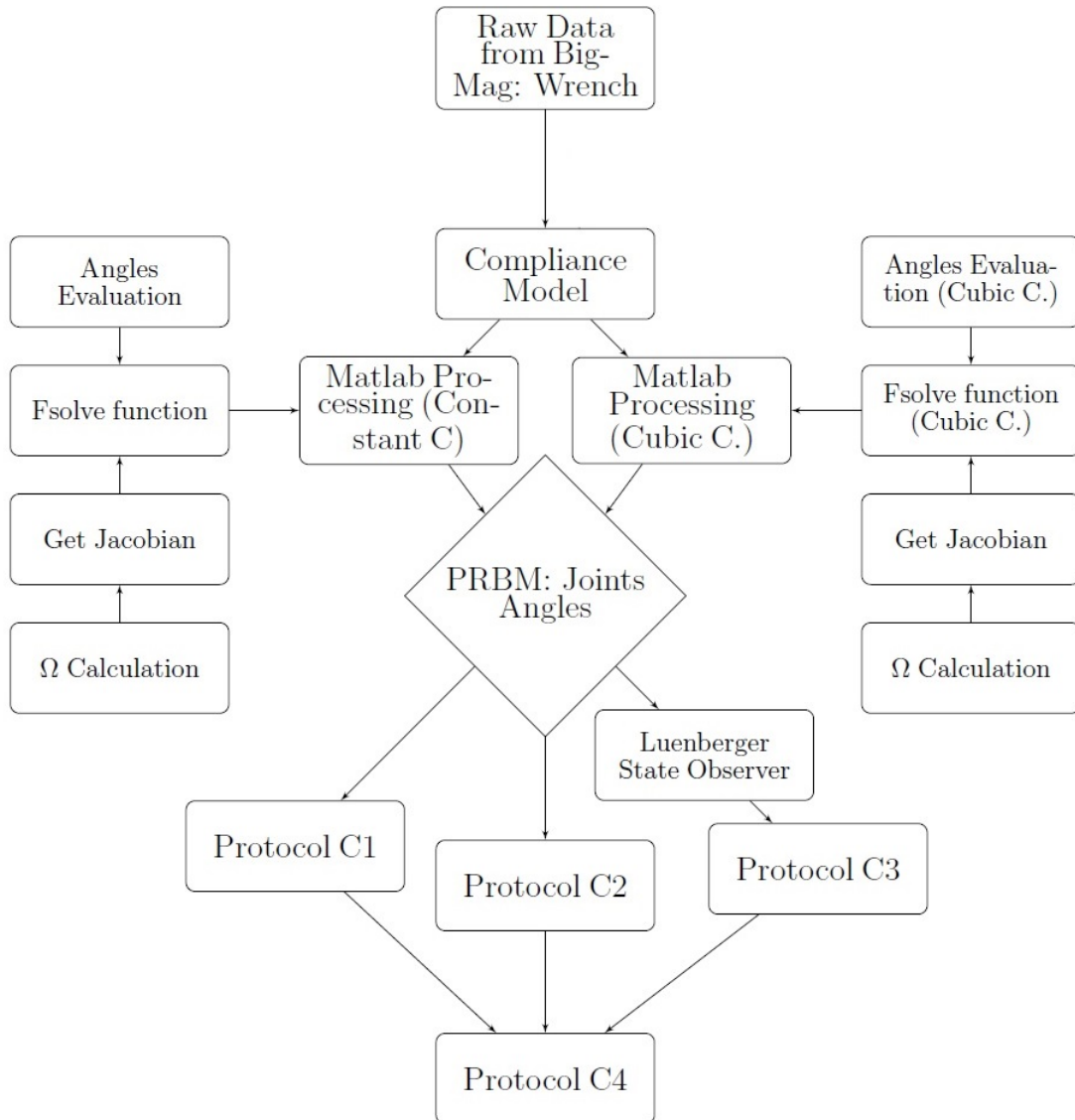


Fig. 2.19: Flowchart of the PRBM algorithm.

In flowchart (2.19) is shown the organization of the PRBM algorithm and the evaluation protocols. Raw dataset at the top of the chart is given by BigMag and is composed by the wrench applied by BigMag to the magnetic component in the catheter's tip. MATLAB processing was differentiated into two blocks depending on the compliance model used, due to differences in minor functions used to find a solution to the *fsolve* function.

## 2. MATERIALS AND METHODS

---

The output of both MATLAB Processing blocks is the PRBM. Joints angles have been evaluated in Protocol C1, C2 and C3 and finally in Protocol C4 the evaluation of the tip position.



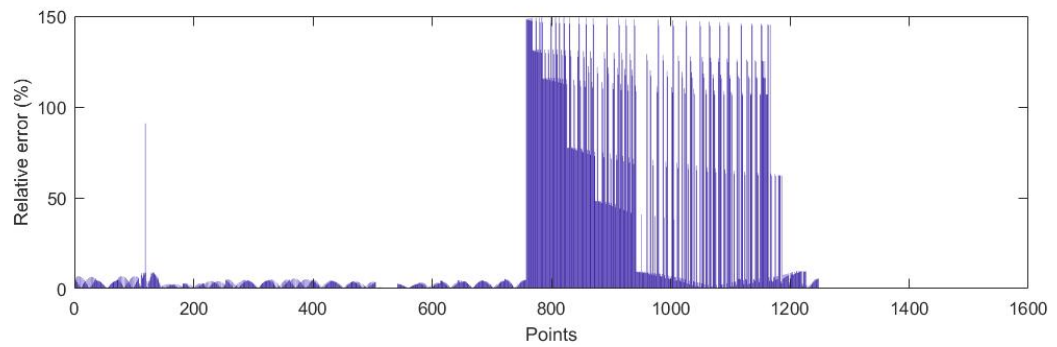
# 3

## Results

### 3.1 Initial Evaluation

The first phase of the project involved the generation of synthetic data in order to develop a model fitting algorithm. Once the point clouds were extracted from Solidworks models with the same characteristic curvature but different resolution, the quality of the reconstruction of the catheter model was evaluated.

As points generated by the STL file were arranged in circles whose centers all lay along the hypothetical line passing through the center of the catheter, the minimum distance between each circle and the fitting model has been evaluated. This distance should correspond to the radius of the catheter body that was known.



*Fig. 3.1:* In the vertical axis the relative error between the evaluated distance and the real one. In the horizontal axis the points.

#### *Variable number of segments*

By varying the number of segments the relative error did not show changes worthy of note, nevertheless a variation of the effectiveness of the model was reported, as evident in figure ( 3.2). Indeed, considering the description of the tip's position the most important detail, the model fitting algorithm behaved differently changing the number of segments.

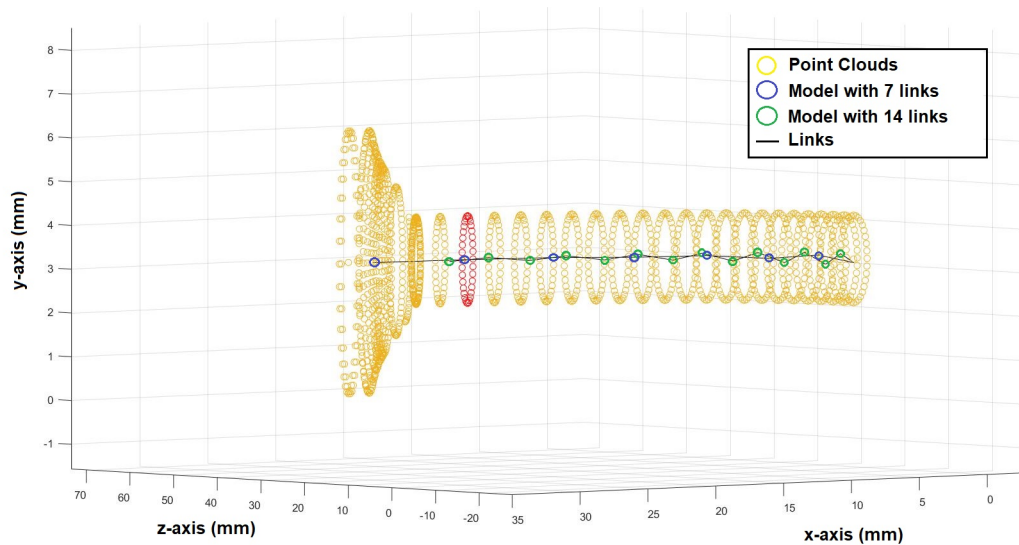


Fig. 3.2: Catheter's plot, each STL point is represented as a circle. Ideal radius compared to measured ones.

Clearly, as shown in ( 3.1), this parameter could be considered valid only for the catheter body because the tip of the catheter, equipped with the magnetic component, had a different body size, not homogeneous and not spherical so that it was not possible to accurately evaluate the catheter distance from the hypothetical line mentioned above.

## 3.2 Validation Method A

### 3.2.1 Experimental Protocol A1

Exploiting the relation (2.2) between the curvature of the circle and the radius of the circumference, curvature radius reconstruction has been evaluated starting from the joints obtained from the model fitting, considering iteratively the circle passing through three successive origins.

## Curvature Evaluation

Tab. 3.1: Model Fitting. Radius Comparison

Origins	Real Radius (mm)	Model Radius (mm)	R.Error (%)
<b>1-2-3</b>	47.62	42.86	9.98
<b>2-3-4</b>	47.62	46.01	3.38
<b>3-4-5</b>	47.62	52.4	10.04
<b>4-5-6</b>	47.62	44.27	7.04
<b>5-6-7</b>	47.62	47.26	0.77
<b>5-7-8</b>	47.62	46.24	2.89
<b>1-4-8 (Av.)</b>	47.62	47.57	0.1

The first value of the third column expressed in the table ( 3.1) is therefore equivalent to the radius of curvature obtained from the circle inscribed respectively in the 1<sup>st</sup>, 2<sup>nd</sup> and 3<sup>rd</sup> origin of the fitting model, the same value in the second row from the 2<sup>nd</sup>, 3<sup>rd</sup> and 4<sup>th</sup> origin respectively and so on.

The graph shows the results for a model with eight origins. The last value represents instead the radius obtained considering the 1<sup>st</sup>, the 4<sup>th</sup> and the 8<sup>th</sup> origins respectively. The last column represents the relative error of the measurement compared to the real one in the second column taken from Solidworks model and has been obtained by the following equation:

$$E_{Rel}^{\%} = \left| \frac{V_{mis} - V_{real}}{V_{real}} \right| \cdot 100 \quad (3.1)$$

In ( 3.3) are shown grafically table's results.

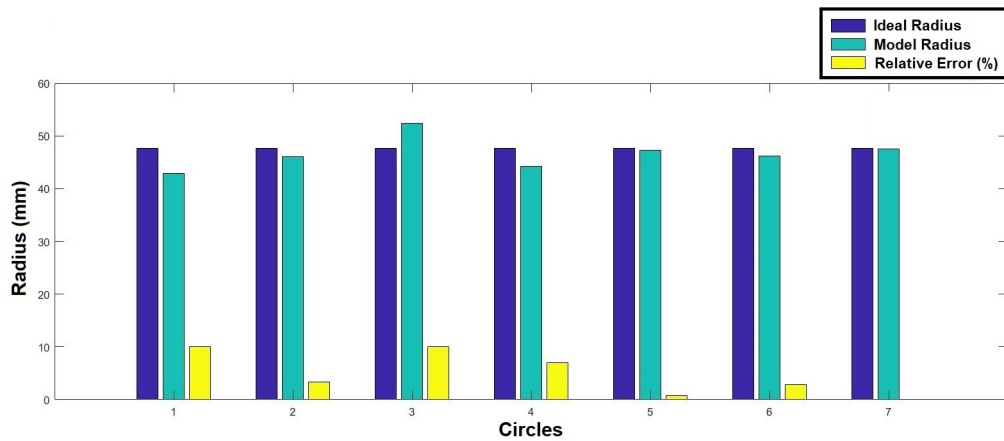


Fig. 3.3: In blue the real radius of the catheter. Light blue the measured one and in yellow the relative error.

### 3.2.2 Experimental Protocol A2

#### 2-D Bending

The fitting algorithm was also tested with a catheter model with variable curvature over time. The algorithm's results were evaluated with respect to the real values obtained from Solidworks.

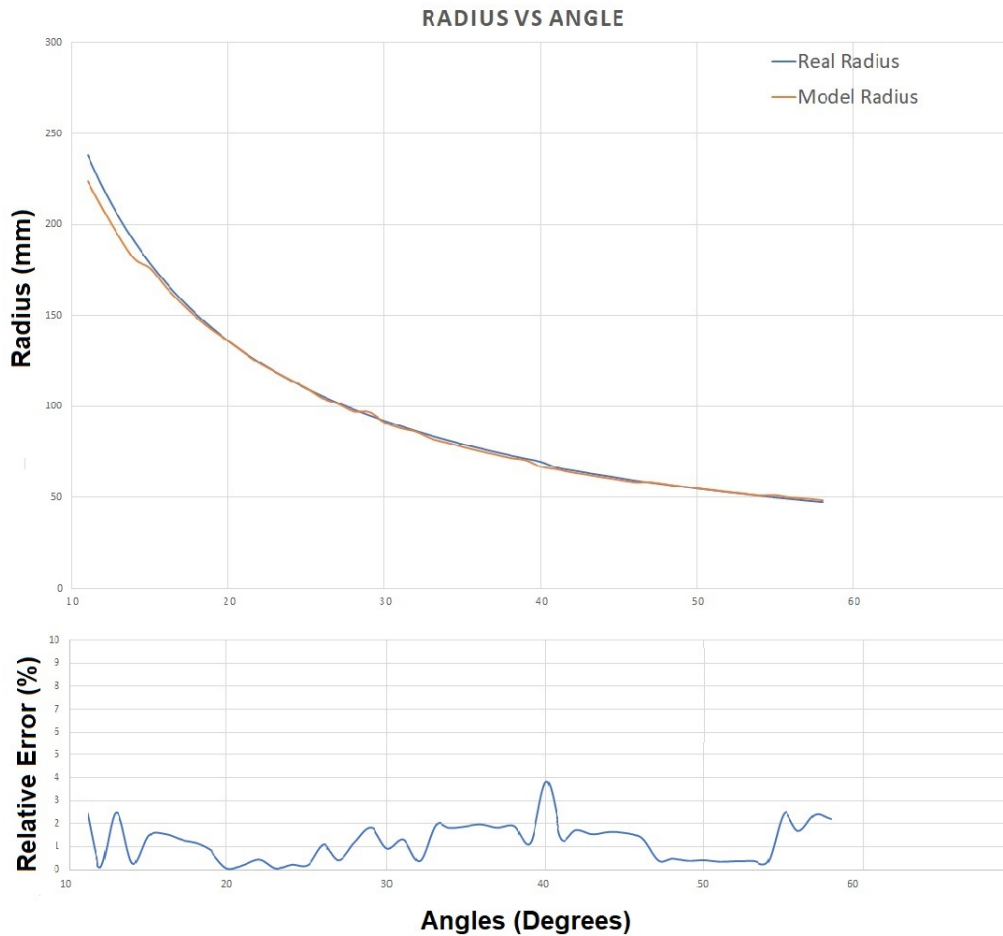


Fig. 3.4: Above in the vertical axis Solidworks radius values in blue and in orange the values obtained with the average model. Below the relative error, always below the 3%. In the horizontal axis the bending angle.

Test performed with Solidworks models with increasing bending angle from  $10^\circ$  to  $60^\circ$  degrees. Sample mean and standard deviation has been equal to  $1.2 \pm 0.86\%$ .

#### *Noise Addition*

After adding the noise in the coordinates of the points to make the point cloud more realistic, the comparison between the models generated before and after the addition of noise has been evaluated as shown in the figure ( 3.5):

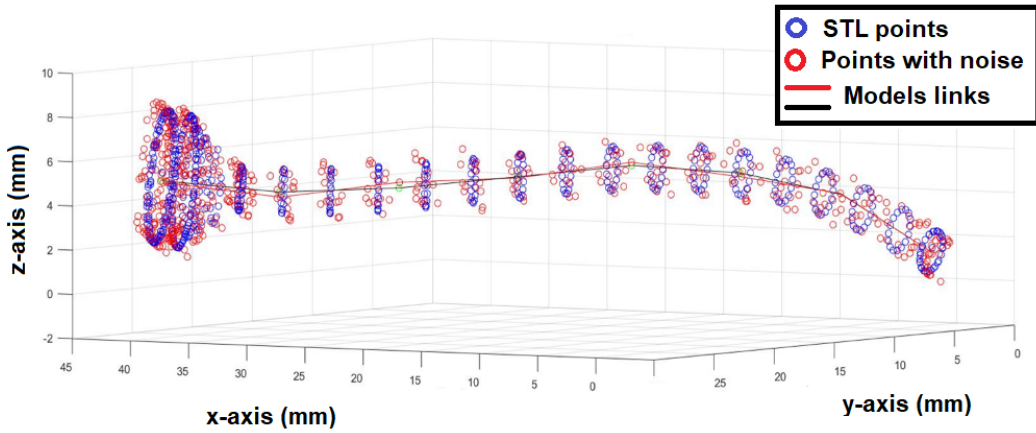


Fig. 3.5: In this figure is shown an example of model fitting evaluation with 3-D bending and noise addition. Noise has been added with different amplitudes, 10%, 50% and 90% of the radius length. Black links has been found considering the initial poin clouds, red links after the noise addition.

The differences compared to the values before adding the noise are shown in the table below.

Tab. 3.2: Noise Addition

<b>Max Noise Amplitude (% radius)</b>	<b>10% r.</b>	<b>50% r.</b>	<b>90% r.</b>
Mean (mm)	0.07	0.2	0.38
Std (mm)	0.046	0.176	0.103

### 3.2.3 Experimental Protocol A3

All the data collected by the stereo vision system have been processed for the extraction of point clouds and the evaluation of the two curvatures present in the printed object for validation. After a first part of the catheter with zero bending angle there was a curvature of  $100m^{-1}$  and then  $200m^{-1}$ .

The data were obtained considering three different reference joints, since the exact curvature starting points were not known.

*Curvature before filtering*

Below the two graphs with the values obtained for the two curvatures.

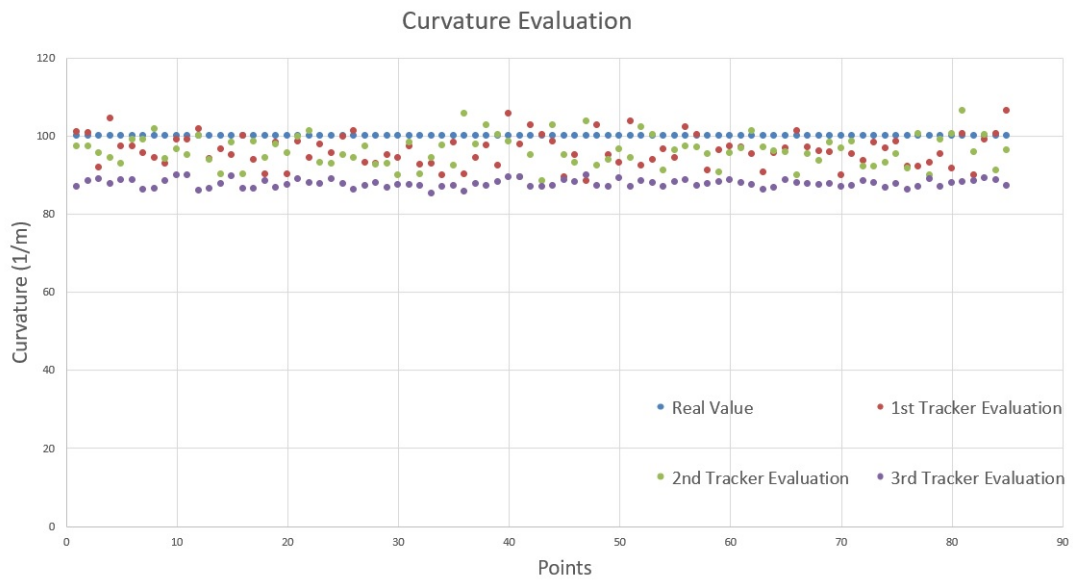


Fig. 3.6: Three different Tracker evaluations considering different reference joints. First Curvature.



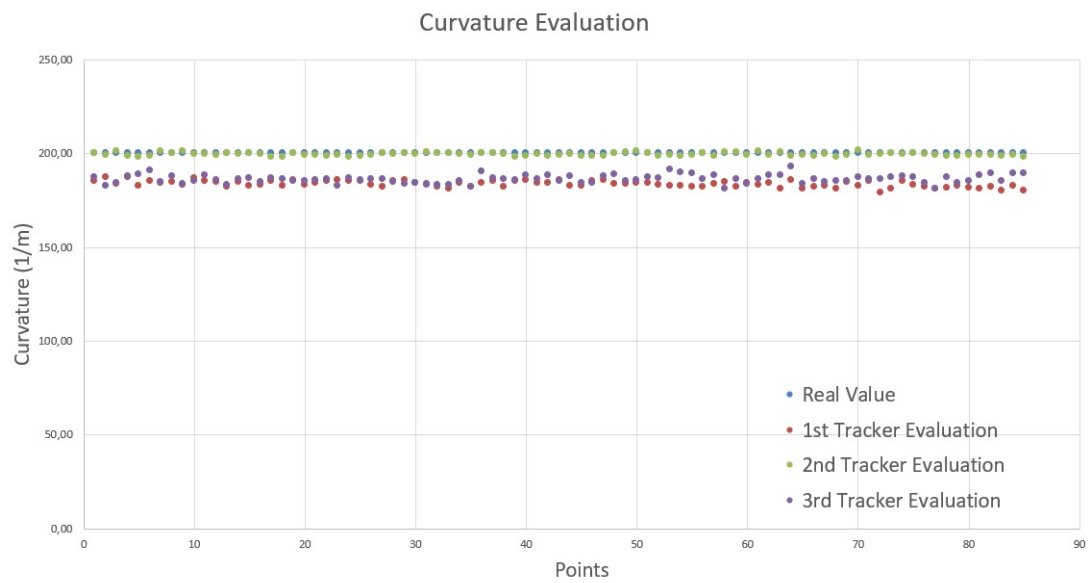


Fig. 3.7: Three different Tracker evaluations considering different reference joints. Second Curvature.

#### *Curvature after filtering*

Before calculating the curvature again, the coordinates of the joints obtained from the fitting model were filtered.

For the filtering a 3-sample moving average was implemented per window with the MATLAB *movmean* function.

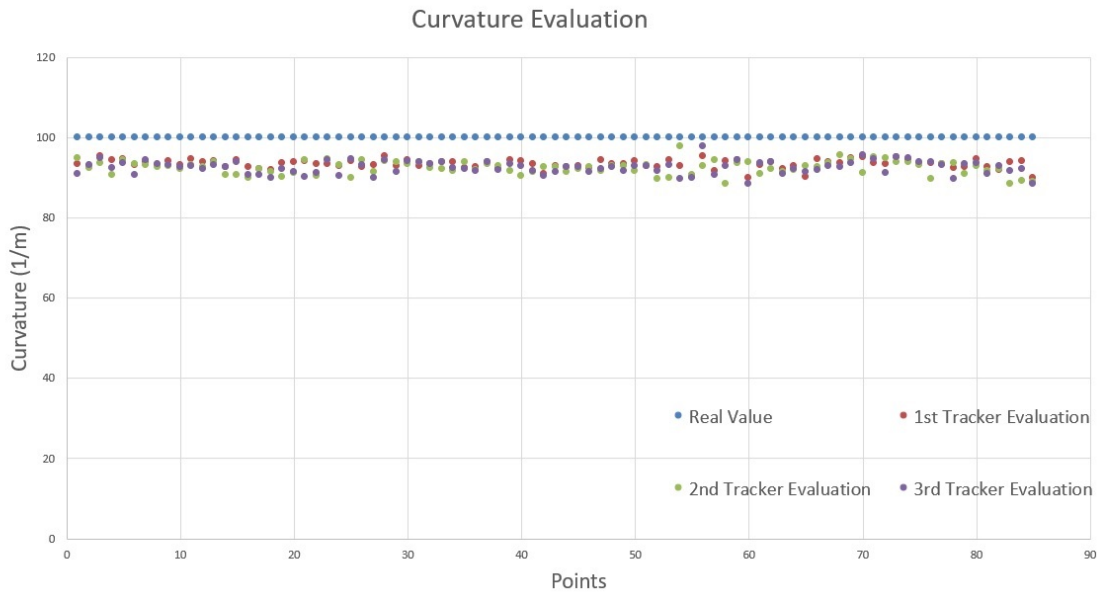


Fig. 3.8: Three different Tracker evaluations considering different filtered joints. First Curvature.

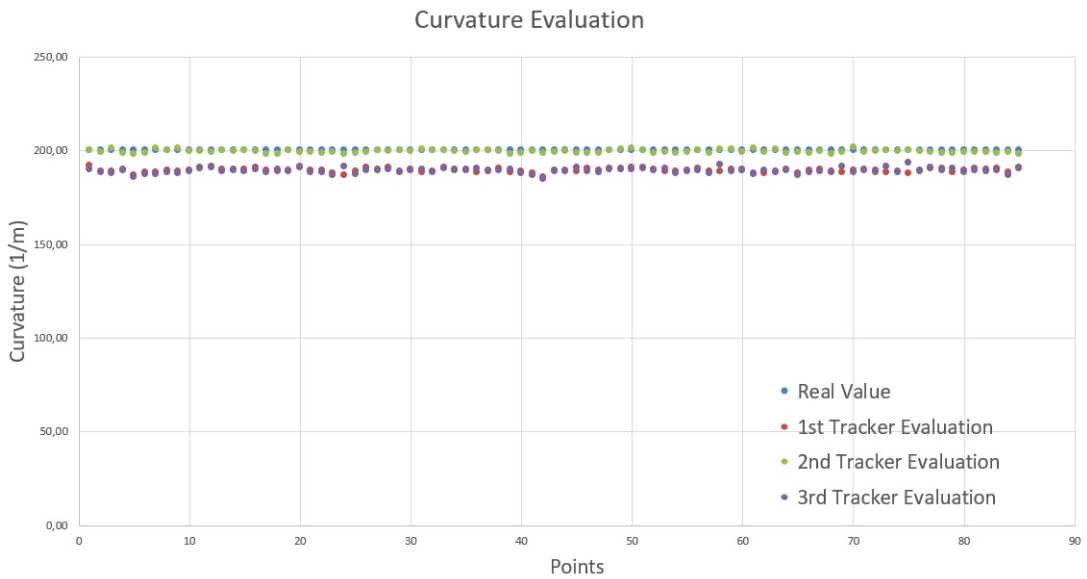


Fig. 3.9: Three different Tracker evaluations considering different filtered joints. Second Curvature.

In tables below the results for the two different curvatures both before and after filtering and the simple difference between both cases and the real value.

Tab. 3.3: Curvature Evaluation

<b>Curvature (<math>m^{-1}</math>)</b>	<b>Mean</b>	<b>Std</b>
1 <sup>st</sup> <b>Curv.</b> ( $100m^{-1}$ )	93.32	5.13
1 <sup>st</sup> <b>Curv.</b> ( $100m^{-1}$ ) <b>Filt.</b>	92.73	1.57
2 <sup>nd</sup> <b>Curv.</b> ( $200m^{-1}$ )	189.79	7.14
2 <sup>nd</sup> <b>Curv.</b> ( $200m^{-1}$ ) <b>Filt.</b>	192.66	4.95

Tab. 3.4: Curvature Error

<b>Evaluation Error (<math>m^{-1}</math>)</b>	<b>Mean</b>	<b>Std</b>
1 <sup>st</sup> <b>Curv.</b> ( $100m^{-1}$ )	-6.68	5.13
1 <sup>st</sup> <b>Curv.</b> ( $100m^{-1}$ ) <b>Filt.</b>	-7.27	1.57
2 <sup>nd</sup> <b>Curv.</b> ( $200m^{-1}$ )	-10.21	7.14
2 <sup>nd</sup> <b>Curv.</b> ( $200m^{-1}$ ) <b>Filt.</b>	-7.34	4.95

### 3.3 Validation Method B

#### 3.3.1 Experimental Protocol B1

Then we moved on to the data set provided by BigMag's stereo vision system, made of 600 frames describing a helical trajectory. On this data collected on the catheter, being supplied by an optical system and not directly from a synthetic software design, a qualitative rather than quantitative evaluation has been performed.

Therefore, the accuracy of the parameters obtained was not evaluated, rather more how much they were realistic. The three evaluated parameters were represented and the tip position has been evaluated before and after filtering.

### Evaluation before Low Pass Filter

Given the particular helical motion generated by BigMag, the parameters  $x$  and  $d$  would have had to settle at a constant value after a first settling moment. All the frames generated by BigMag were processed to compare their parameters.

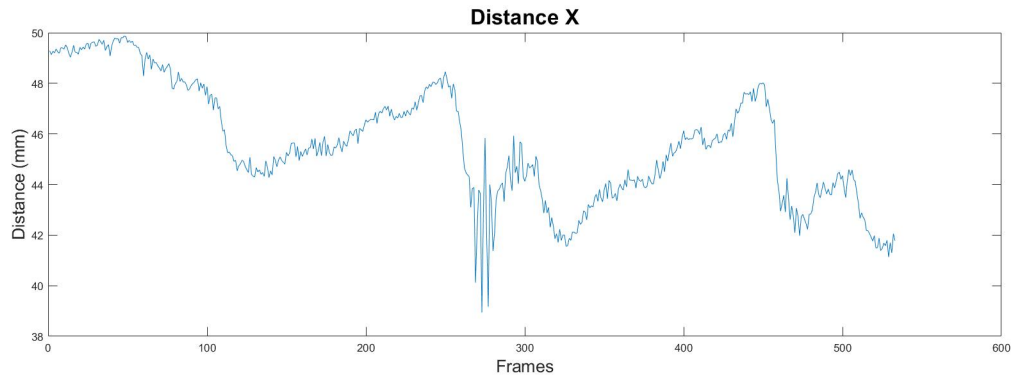


Fig. 3.10: In figure the permutation of all frames to compare  $x$ .

The figure ( 3.10) shows the trend of  $x$  that represents the distance along the x axis between the catheter tip and catheter's insertion point.

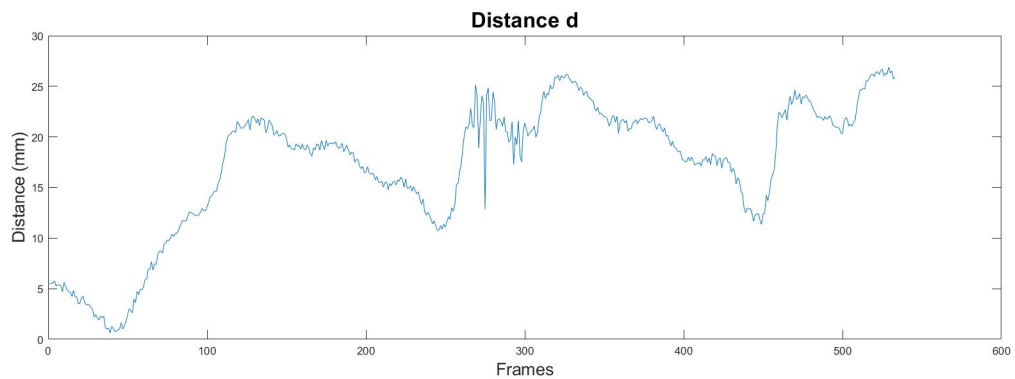


Fig. 3.11: In figure the permutation of all frames to compare  $d$ .

The figure ( 3.11) shows the trend of  $d$  which represents the distance between the

catheter tip and line parallel to the x-axis passing through catheter's insertion point.

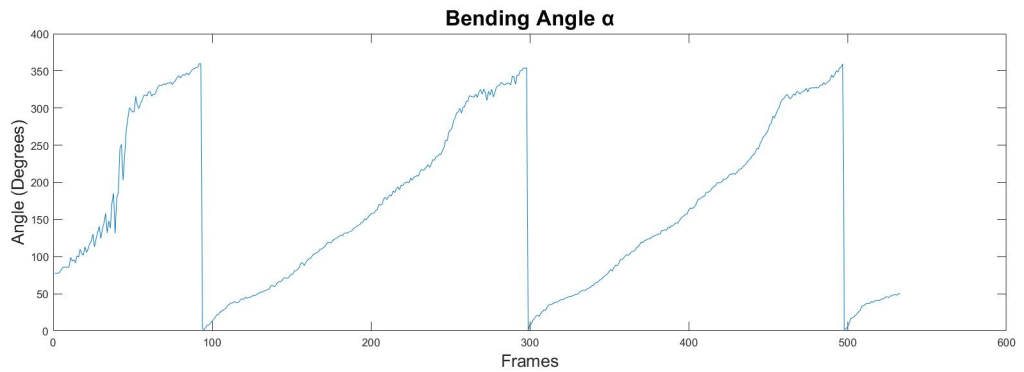


Fig. 3.12: In figure the permutation of all frames to compare  $\alpha$ .

The figure ( 3.12) shows the trend of  $\alpha$  that represents the angle between the line passing through  $d$  and the line parallel to the z-axis passing through intersection between the previous line and line parallel to the x-axis passing through catheter's insertion point.

### Tip Coordinates

In the figures below the coordinates of the catheter tip before filtering.

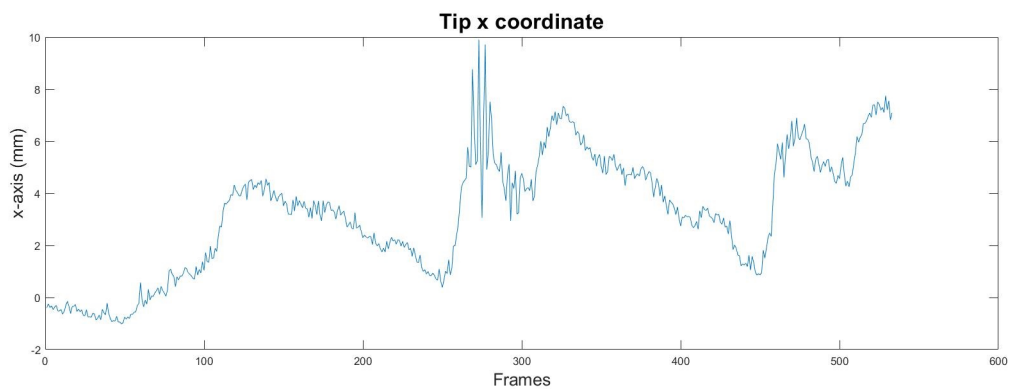


Fig. 3.13: X tip position during the helical movement.

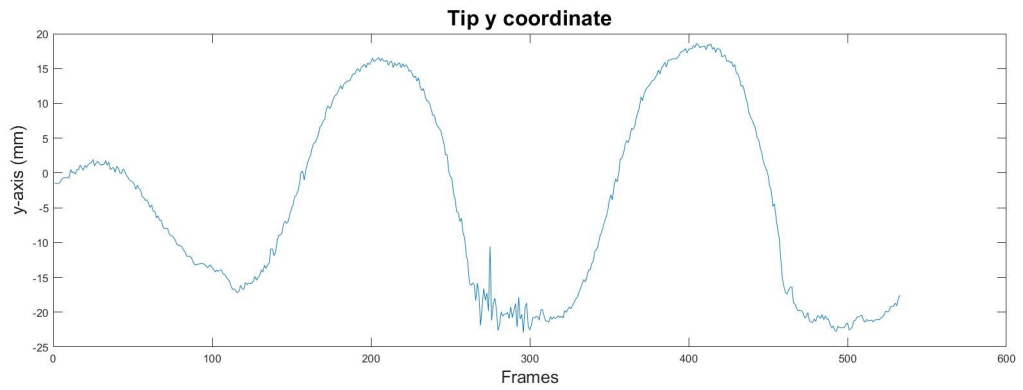


Fig. 3.14: Similar sinusoidal function in the y-axis due to the helical movement generated by Big-Mag.

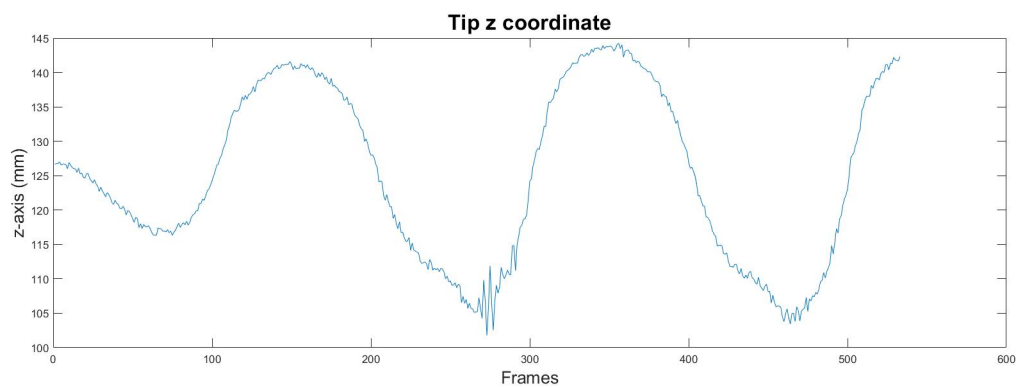


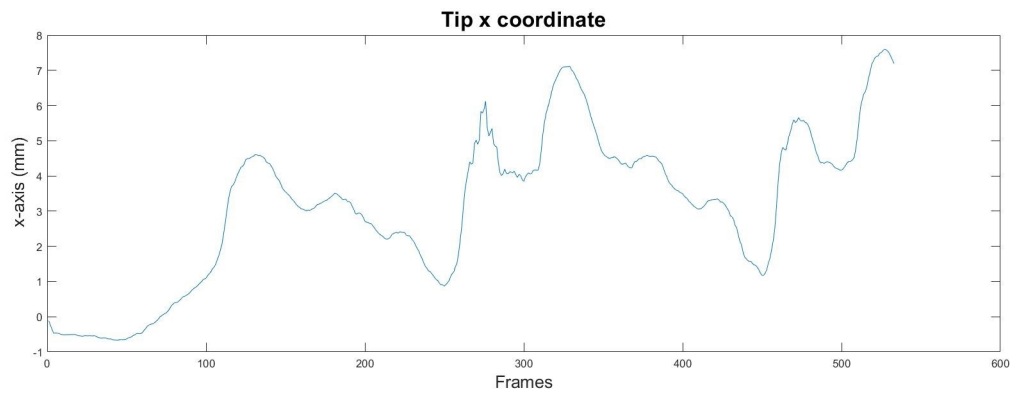
Fig. 3.15: Similar sinusoidal function in the x-axis due to the helical movement generated by Big-Mag.

#### *Evaluation after Low Pass Filter*

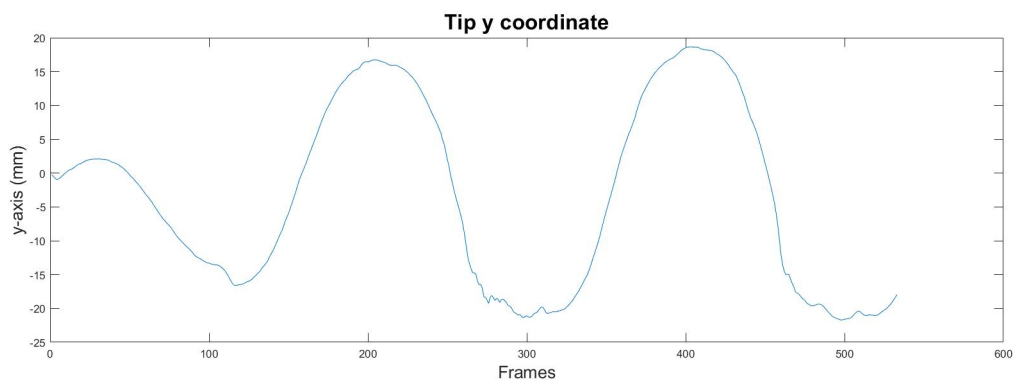
A moving average has been performed using MATLAB function *movmean* with four samples in each window. The filtering was performed not on the three parameters but on the position of the catheter tip, the main element for the control of the catheter itself.

*Tip Coordinates*

In the figures below the coordinates of the catheter tip, in the same test as ( 3.13), ( 3.14) and ( 3.15), after the moving average.



*Fig. 3.16: X-axis coordinates after low pass filtering.*



*Fig. 3.17: Y-axis coordinates after low pass filtering.*

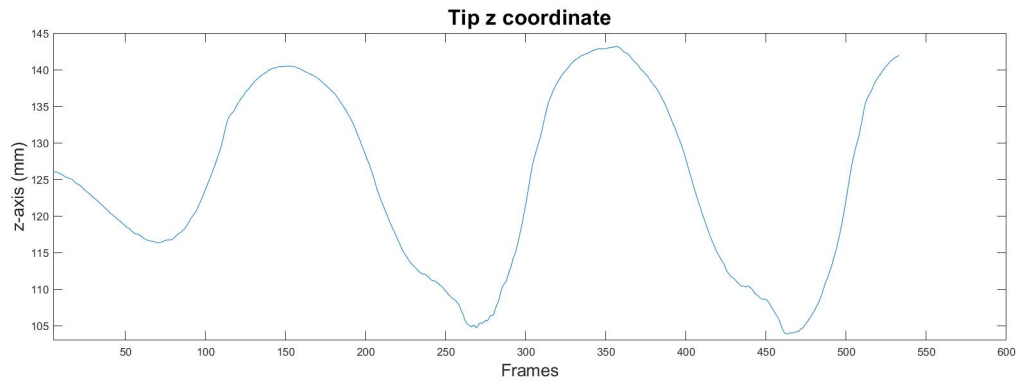


Fig. 3.18: Z-axis coordinates after low pass filtering.

### 3.3.2 Experimental Protocol B2

#### *Maya's Synthetic Dataset Evaluation*

Maya software allowed to obtain a point clouds of the synthetic catheter that was very similar to the one obtained from the data provided by BigMag.

An animation of the catheter model was also simulated performing a movement very similar to the one performed previously. The animation has generated a total of 200 frames saved and processed with the OBJ format.

The results for the three configuration parameters  $x$ ,  $d$  and  $\alpha$ , obtained by the linear fitting algorithm during the simulation, are shown below.

#### *Parameter $x$*

In ( 3.19), ( 3.20) show the values obtained for the parameter  $x$  compared with the data provided by the Maya model with a respective graph of the relative error with respect to the real value.



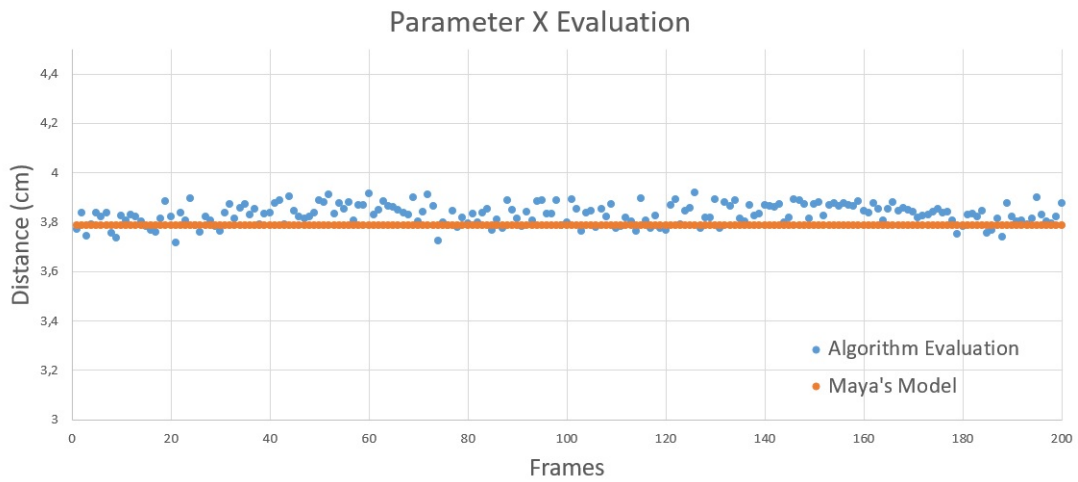


Fig. 3.19: In figure the permutation of all frames to evaluate  $x$ . Maya's value is constant during all the animation.

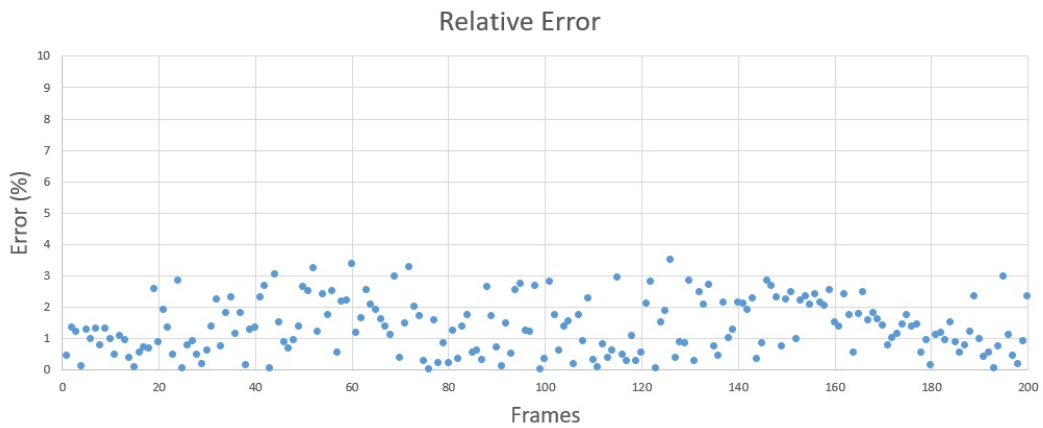


Fig. 3.20: Relative error for the parameter  $x$ .

### Parameter $d$

In ( 3.21), ( 3.22) are show the values obtained for the parameter  $d$  compared with the data provided by the Maya model with a respective graph of the relative error with respect to the real value.

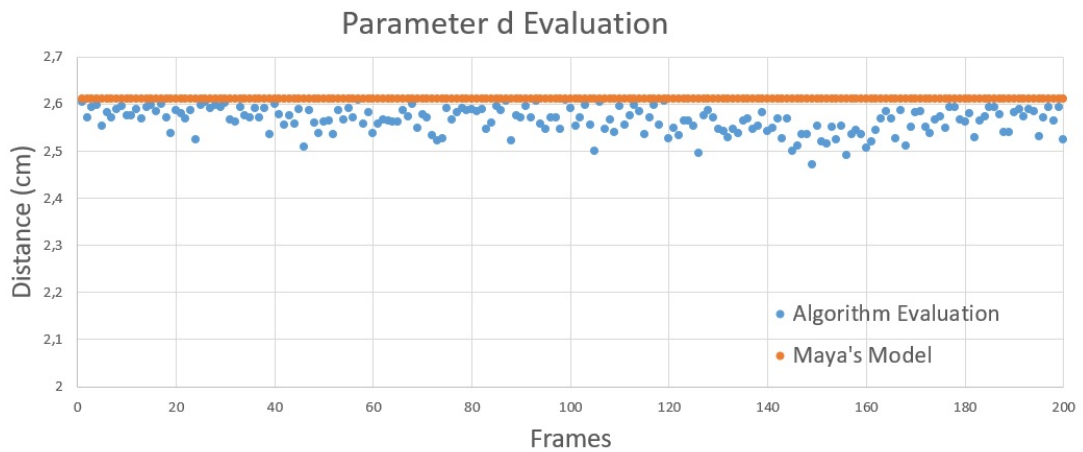


Fig. 3.21: In figure the permutation of all frames to evaluate  $d$ . Maya's value is constant during all the animation.

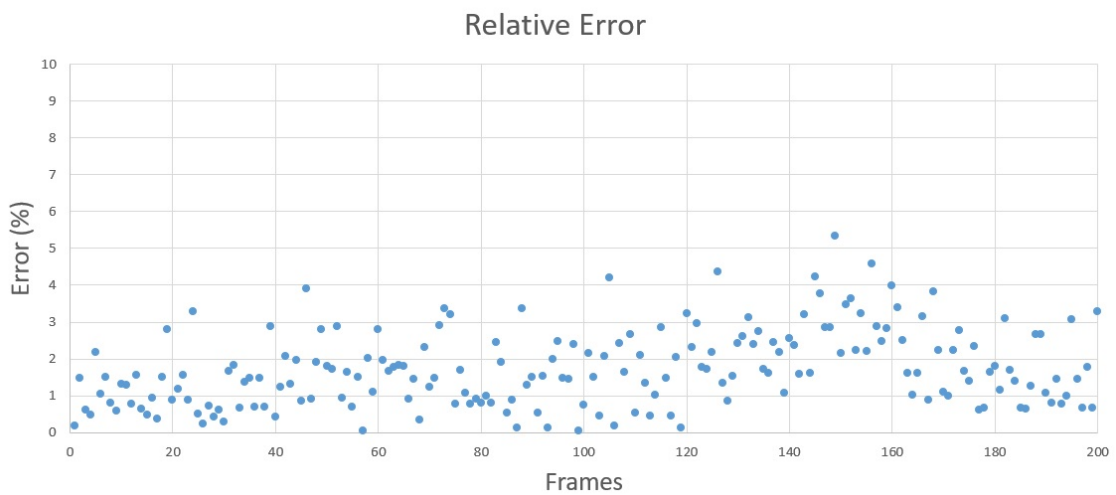


Fig. 3.22: Relative error for the parameter  $d$ .

### *Bending Angle Evaluation*

Figures ( 3.23), ( 3.24) show the values obtained by the linear fitting algorithm for the parameter  $\alpha$  compared with the data provided by the Maya model with a respective graph of the relative error with respect to the real value.

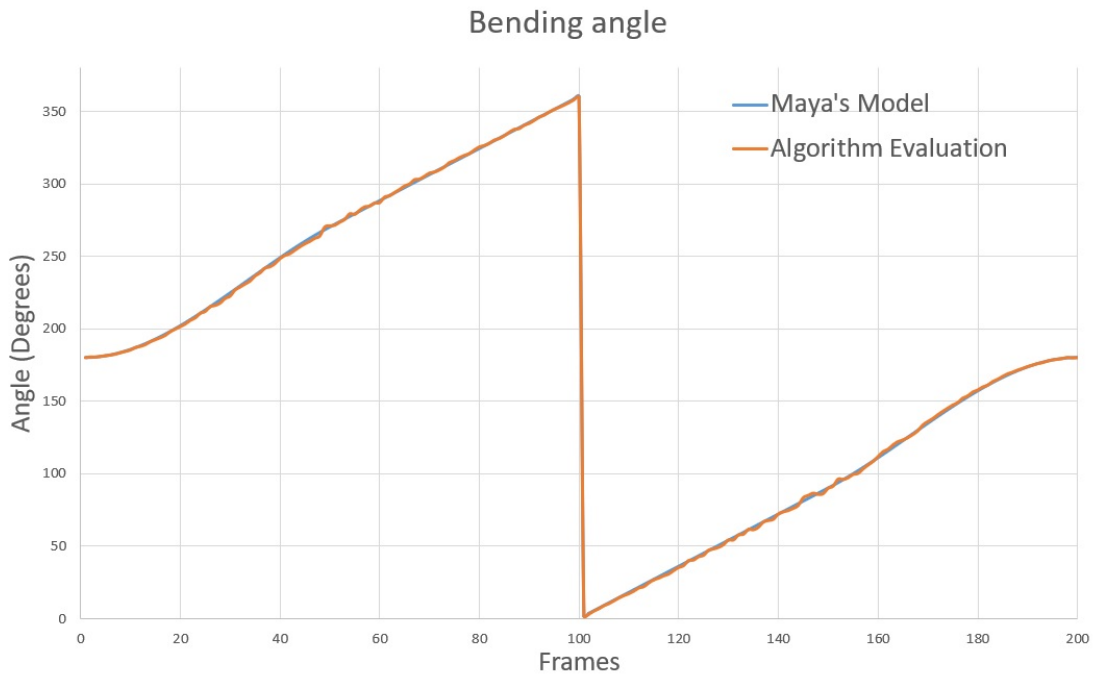


Fig. 3.23: In figure the permutation of all frames to evaluate  $\alpha$ 's Angular velocity was not constant during all the animation.

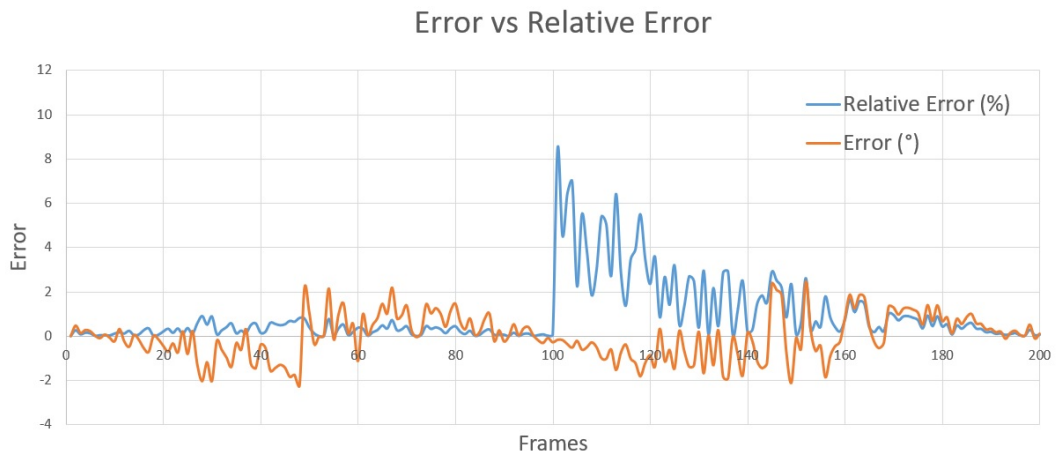


Fig. 3.24: Relative error and Error for the parameter  $\alpha$ .

All the results for parameters  $x$ ,  $d$  and  $\alpha$  have been reported in the table below.

Tab. 3.5: Parameters Evaluation

Relative Error (%)	Parameter $x$	Parameter $d$	Parameter $\alpha$
Mean	1.35	1.72	0.95
Std.	0.86	1.02	1.44

The simple difference between the  $\alpha_{mis}$  and  $\alpha_{real}$  reported with the orange line in figure ( 3.24), showed a very low sample average error and a good standard deviation  $(-0.02, 0.97^\circ)$ .

### 3.3.3 Experimental Protocol B3

#### Radius Evaluation

Figures ( 3.25), ( 3.26) show the values obtained by the linear fitting algorithm for the radius compared with the data provided by the Maya model with a respective graph of the relative error with respect to the real value.

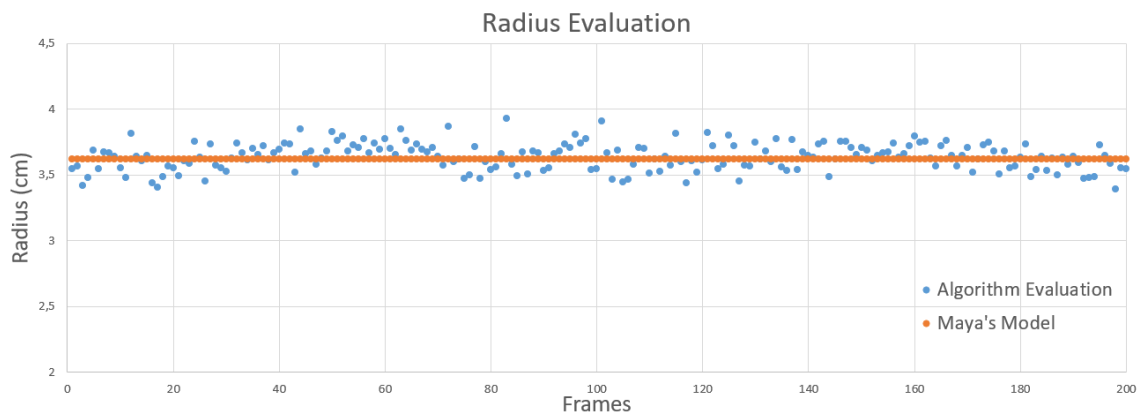


Fig. 3.25: In figure the permutation of all frames to evaluate  $\alpha$ . Curvature was constant during all the animation and so the radius.

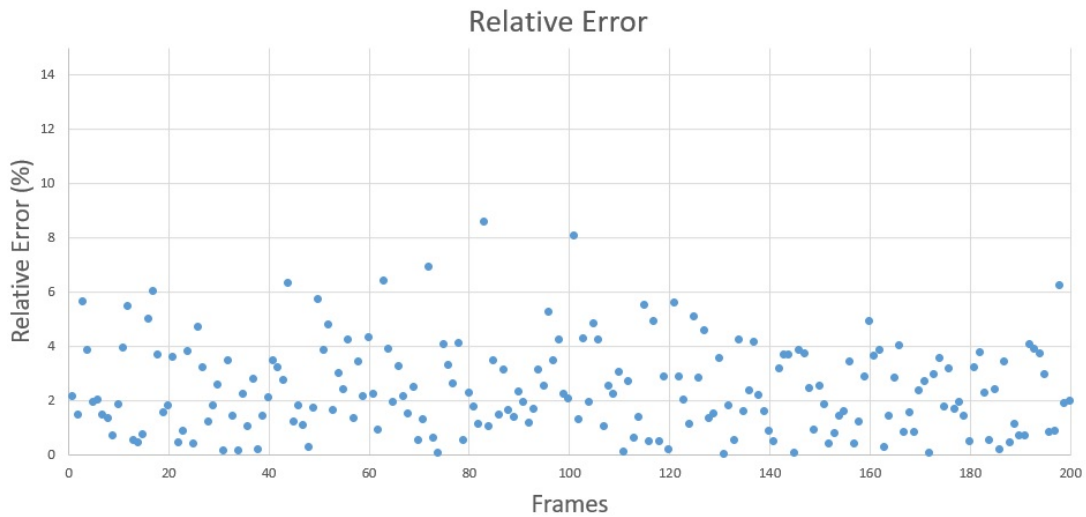


Fig. 3.26: Relative error for the radius.

#### *Ultrasound Scanning Simulation Evaluation*

After developing the data processing algorithm to simulate a scan with an ultrasound probe, the compatibility of the stereo vision algorithm in the reconstruction of the radius of curvature has been evaluated starting from the centers of the slices obtained with the simulation. Maya's model has been analyzed simulating the ultrasound probe.

#### *Radius evaluation*

Figures ( 3.27), ( 3.28) show the values of the radius obtained in relation to the information provided by Maya's model and the graph of the relative error with respect to the real value.

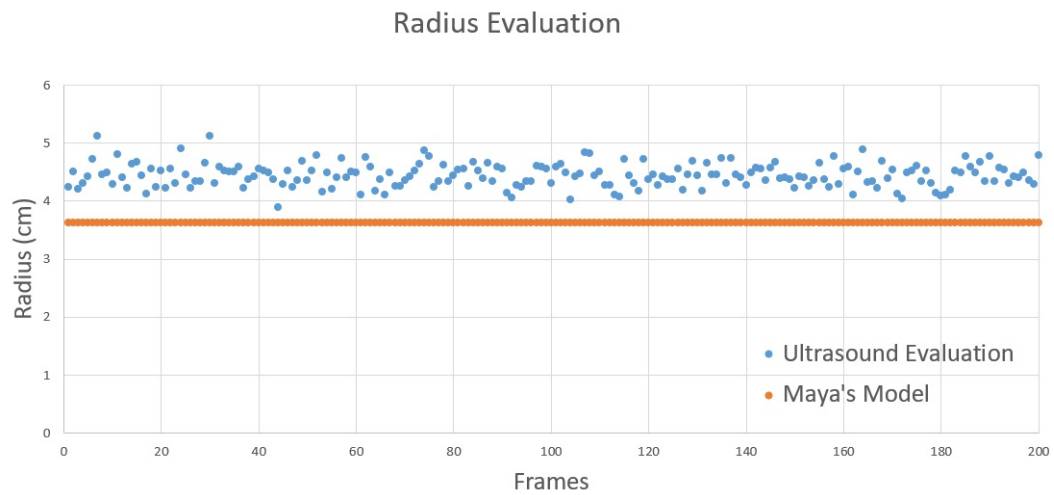


Fig. 3.27: In figure the permutation of all frames to evaluate the radius. Curvature and therefore radius were constant during all the animation.

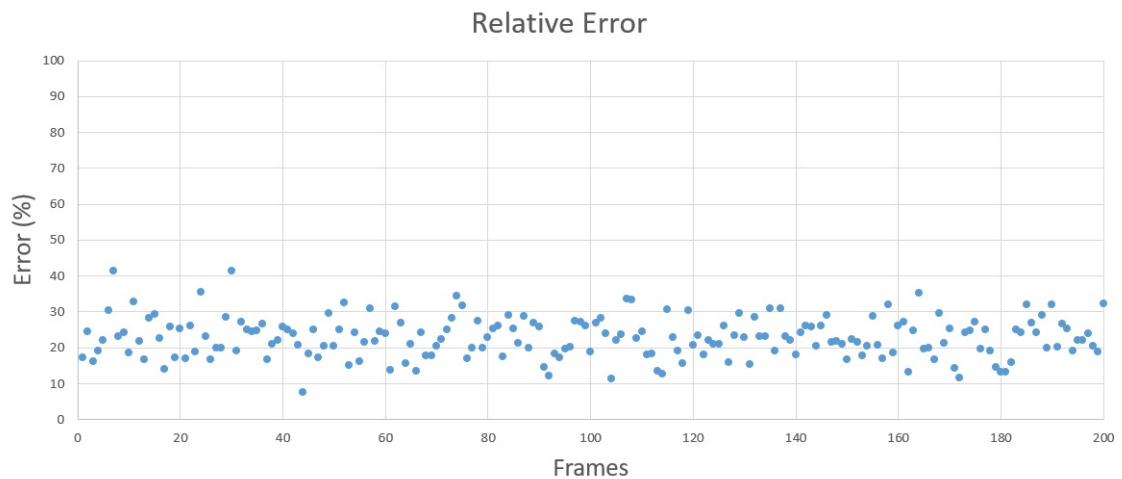


Fig. 3.28: Relative error for the radius obtained with the ultrasound simulation.

The comparison between results obtained with stereo vision and ultrasound processing evaluating the radius have been reported in the following table:

Tab. 3.6: Radius Evaluation

<b>Relative Error (%)</b>	<b>Stereo Vision</b>	<b>Ultrasound</b>
Mean	2.39	22.81
Std.	1.64	5.61

For stereo vision processing, sample mean and standard deviation of the simple difference between the measured and the real radius resulted  $0.017cm$  and  $0.1cm$  respectively.

### 3.4 Validation Method C

#### 3.4.1 Experimental Protocol C1

Once the development PRBM algorithm on MATLAB and Simulink was completed and the model's behavior was verified with data supplied artificially through functional blocks on Simulink, the joints angles obtained by the linear fitting algorithm, implemented in BigMag's stereo vision system, and those obtained by the PRBM model were compared. To simplify the analysis of the PRBM model and the comparison between the angles in the two directions  $x$  and  $y$ , in the reference system of each joint, four-joints models, considering the tip too, were built.

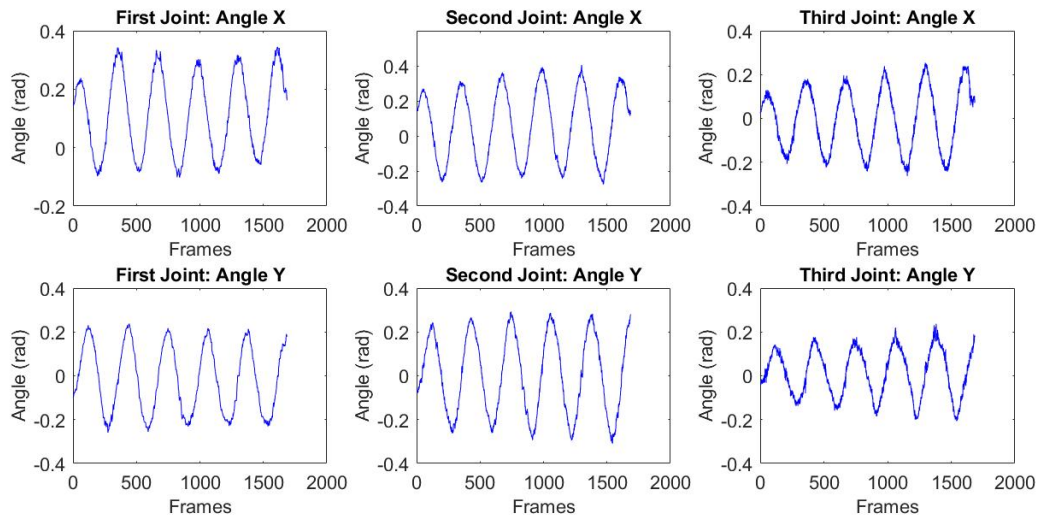


Fig. 3.29: In figure the angles given by BigMag for the first test for each joint and direction.

The simulations of the PRBM were performed with a 1640 frames dataset in which was stored and helical trajectory of the catheter. In the first two simulations the contribution due to the gravity force in the *ModelG* was considered and the comparison was made with a model that included also the initial deflection due to the gravity itself (*ModelGQ*).



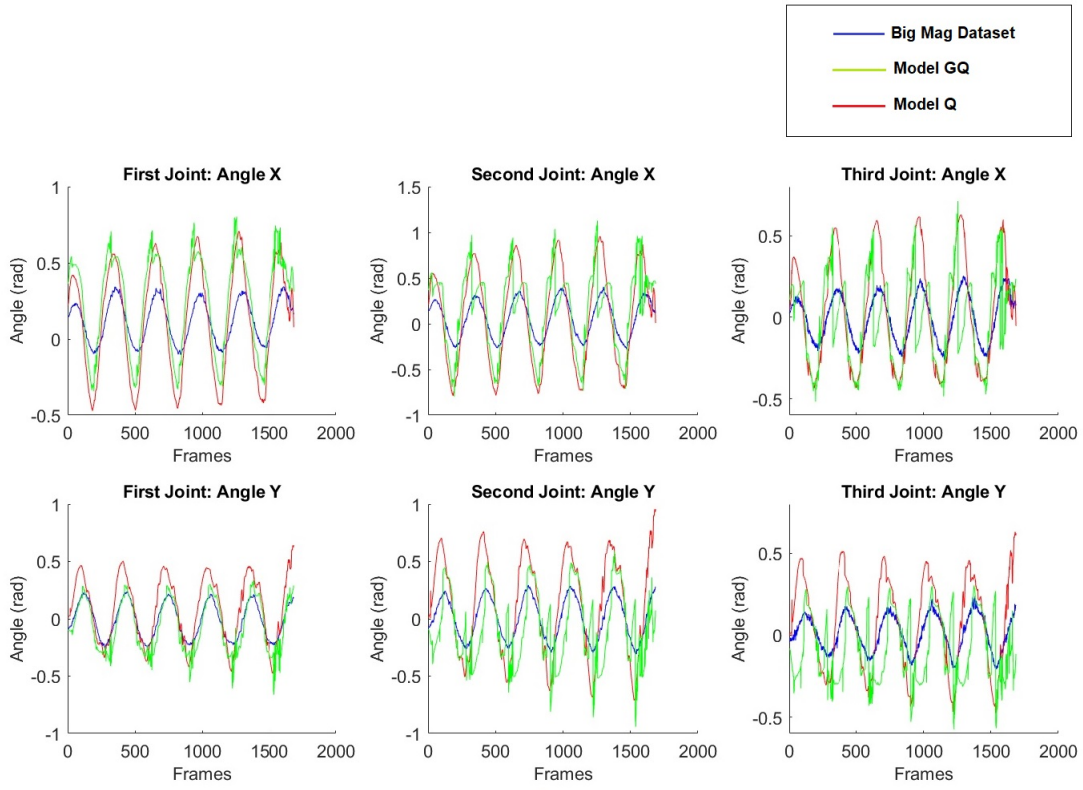


Fig. 3.30: In blue BigMag’s dataset, in red *ModelG* dataset and in green *ModelGQ* dataset.

### Compliance evaluation

Since the first results showed an excessive amplitude of the output evaluating the model, it has been decided to modify the elastic constant of the springs in the joint so that the error in the reconstruction of the corners could be reduced.

The initial guess for compliance values for each joint and direction was:

$$C = \begin{bmatrix} 0.0076 & 0.008 & 0.0011 & 0.0011 & 0.0022 & 0.0022 \end{bmatrix} \quad \frac{rad}{Nm} \quad (3.2)$$

in which:

$$C = \begin{bmatrix} C_1^x & C_1^y & C_2^x & C_2^y & C_3^x & C_3^y \end{bmatrix} \quad (3.3)$$

From the supplied dataset the compliance has been obtained again through the formula described in (2.35) to verify if the quality of the reconstruction of the angles depends on

this parameter.

The new compliance values obtained (3.4) were used to perform the new simulations.

$$C = \begin{bmatrix} 0.0021 & 0.0019 & 0.0018 & 0.0017 & 0.0027 & 0.0028 \end{bmatrix} \frac{\text{rad}}{\text{Nm}} \quad (3.4)$$

The model which included the new compliance has been called *ModelNC*.

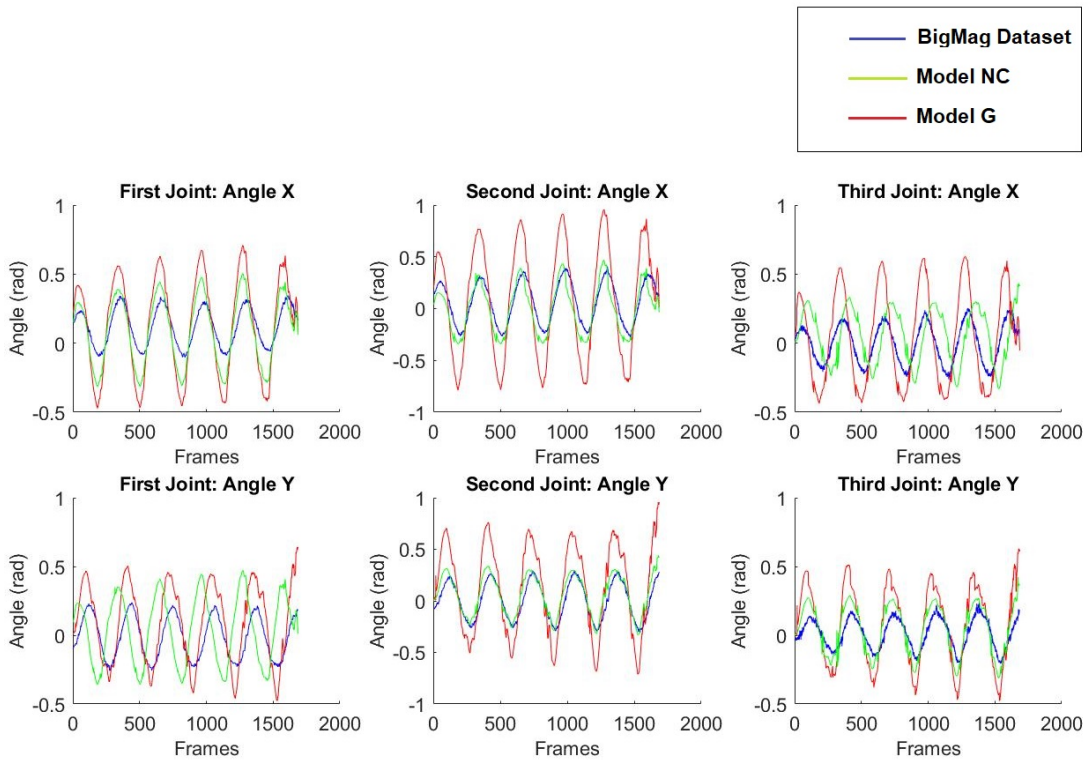


Fig. 3.31: In blue BigMag's dataset, in red *ModelG* dataset and in green *ModelNC* dataset.

### Temporal Shift

Since the first graphs we noticed a temporal shift between the corners of the dataset and the reconstructed angles from the applied wrench, probably due to some delay in the BigMag control operations and saving. The delay has been evaluated and the compliance has been recalculated by shifting the wrenches values to make them matching with the correct

angles of the catheter.

The delay was evaluated with the *finddelay* function on MATLAB and quantified in about 19 frames as the average between the tests performed.

#### *Shifted Compliance Evaluation*

Applying the same procedure used previously and taking into account the calculated delay, new compliance values were obtained as follows:

$$C = \left[ 0.0023 \quad 0.0019 \quad 0.0020 \quad 0.0018 \quad 0.0029 \quad 0.0029 \right] \quad \frac{rad}{Nm} \quad (3.5)$$

The model with the compliance obtained considering the delay was named *ModelS*, which has been compared with *ModelNC*.

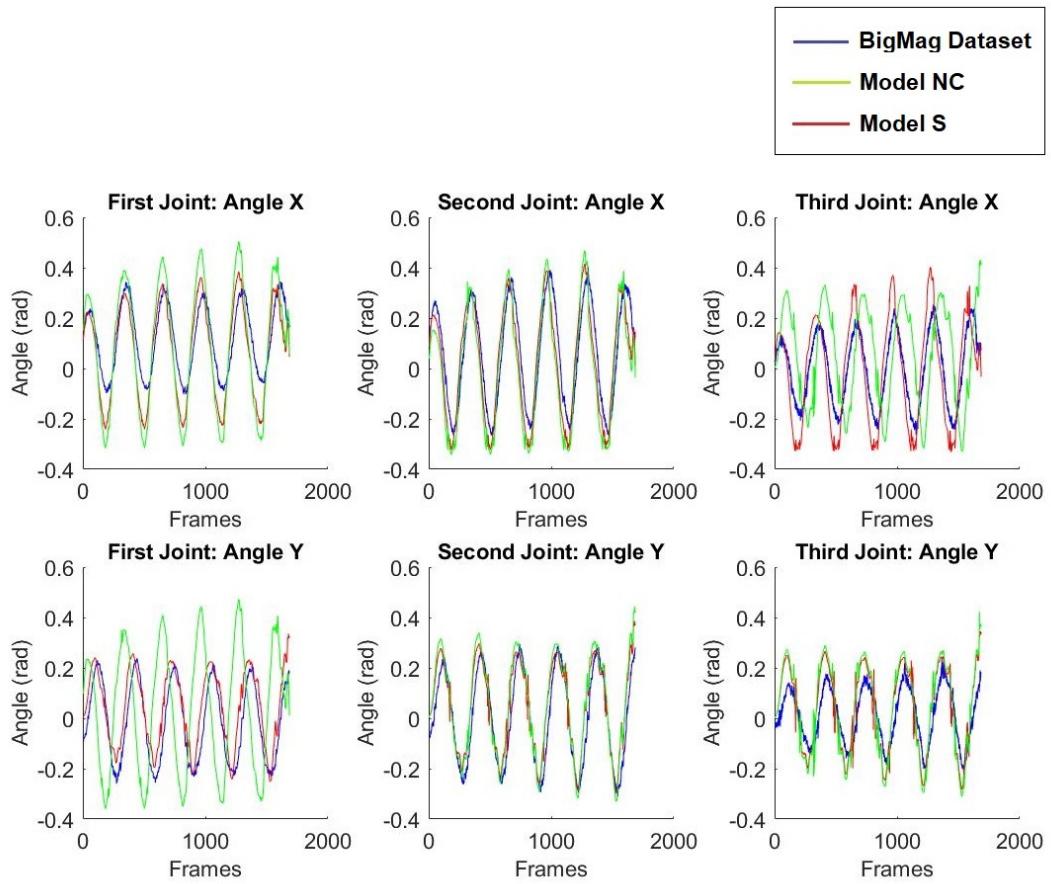


Fig. 3.32: In blue BigMag's dataset, in red *ModelS* dataset and in green *ModelNC* dataset.

### 3.4.2 Experimental Protocol C2

#### *Cubic Compliance Evaluation*

Subsequently, obtained the variable cubic compliance through (2.36), according to the amount of bending, a new model has been designed and called *ModelSK*. The model was modified because the new compliance was not made of six values only, one for each joint and management, but twelve. Both the optimization function contained in *fsolve* and the angle calculation function have been updated and adapted to the new compliance while maintaining the same convergence speed of the optimization function, around 0.082s and 0.075s respectively.

The new compliance has been set as follow:

$$C = \begin{bmatrix} C_1^{3x} & C_1^{3y} & C_2^{3x} & C_2^{3y} & C_3^{3x} & C_3^{3y} \\ C_1^x & C_1^y & C_2^x & C_2^y & C_3^x & C_3^y \end{bmatrix} \quad (3.6)$$

$$C = \begin{bmatrix} -0.0073 & -0.0029 & -0.0033 & -0.0005 & -0.0010 & -0.0188 \\ 0.0027 & 0.0020 & 0.0023 & 0.0017 & 0.0033 & 0.0034 \end{bmatrix} \quad \frac{rad}{Nm} \quad (3.7)$$

This new *ModelSK* has been compared with *ModelS*.

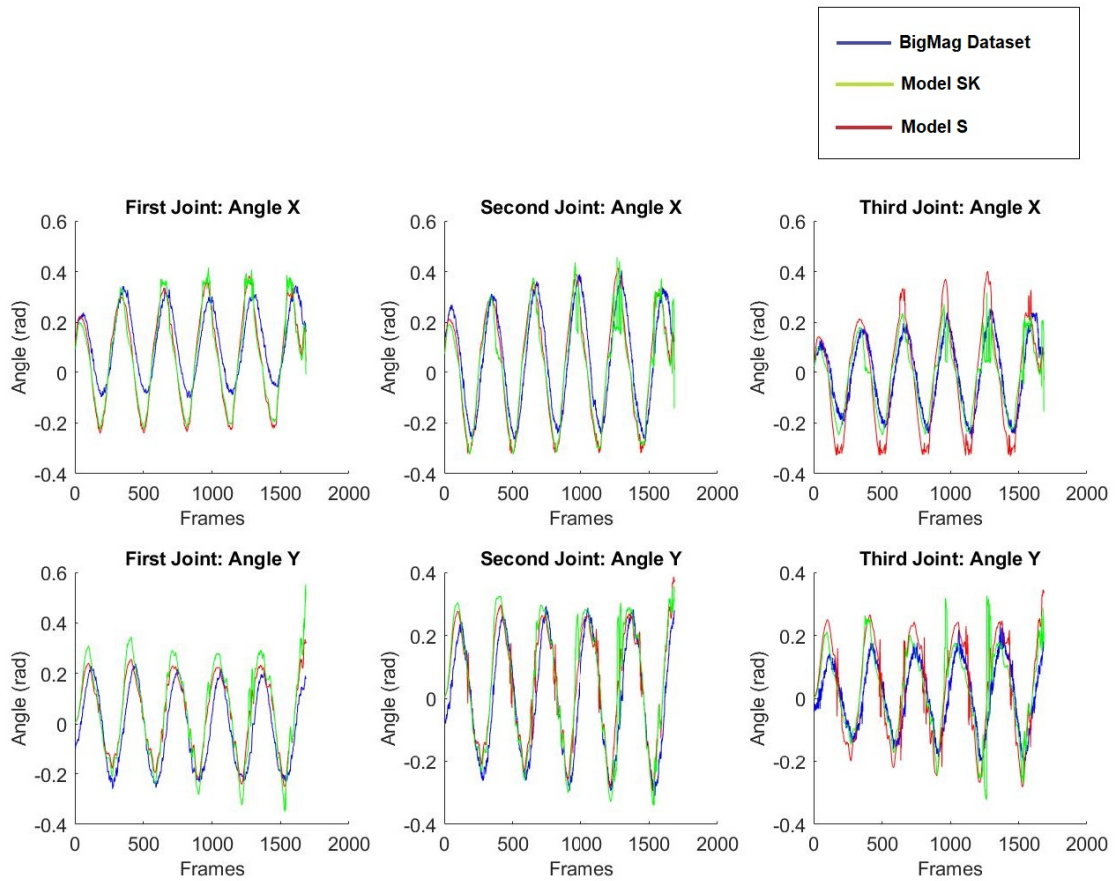


Fig. 3.33: In blue BigMag’s dataset, in red *ModelS* dataset and in green *ModelSK* dataset.

## 3.4.3 Experimental Protocol C3

## Luenberger Observer

Finally, the Luenberger observer was implemented in the PRBM algorithm with different correction coefficients. The new model, called *ModelL* has been compared with the previous models. In the figures below is reported the comparison with *ModelSK* with corrective factors of 0.9 and 0.5 respectively.

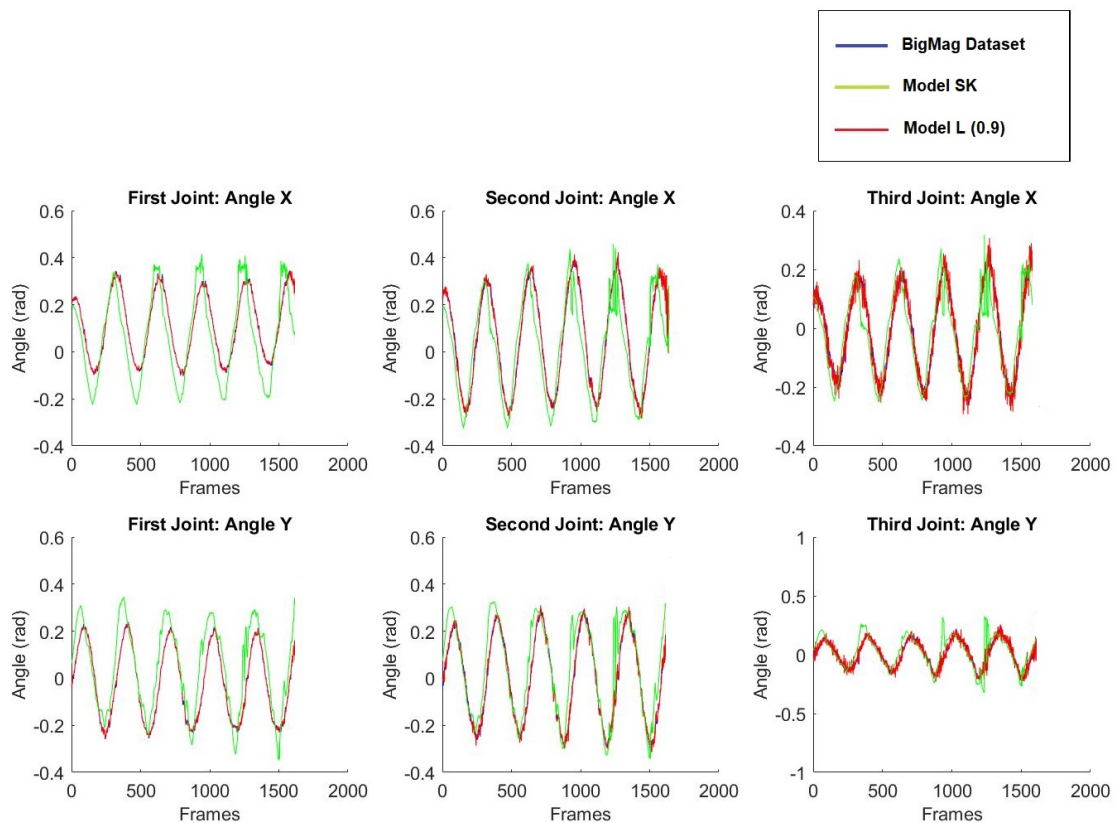


Fig. 3.34: In blue BigMag's dataset, in red *ModelL* dataset with 0.9 correction factor and in green *ModelSK* dataset.

Three different corrective factors were tested to verify the different behavior of the model in the prediction of angles.

The value of this fitter was directly proportional to the actual amount of correction performed on the reconstruction of the angle, thus varying the speed of convergence to the

true value of the previous output.

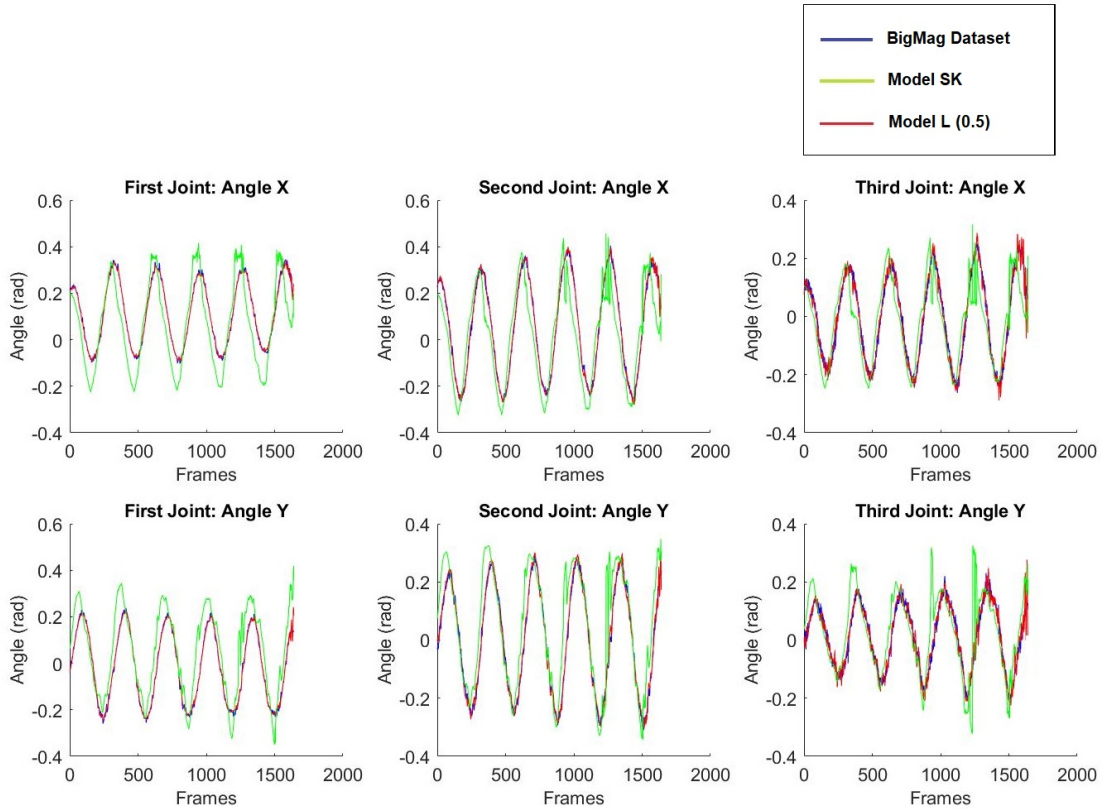


Fig. 3.35: In blue BigMag’s dataset, in red *ModelL* dataset with 0.5 correction factor and in green *ModelSK* dataset.

#### 3.4.4 Experimental Protocol C4

To verify the quality of the models, the position of the catheter tip has been evaluated during the test. Fundamental for steering control purposes was the accuracy in the evaluation of the position of the catheter, because the magnetic component in the tip should have been located accurately to evaluate the correct wrench for a certain task in real time.

The graph below shows the comparison between the coordinates of the tip obtained with the first *ModelG* model compared to *ModelSK* and *ModelL*.

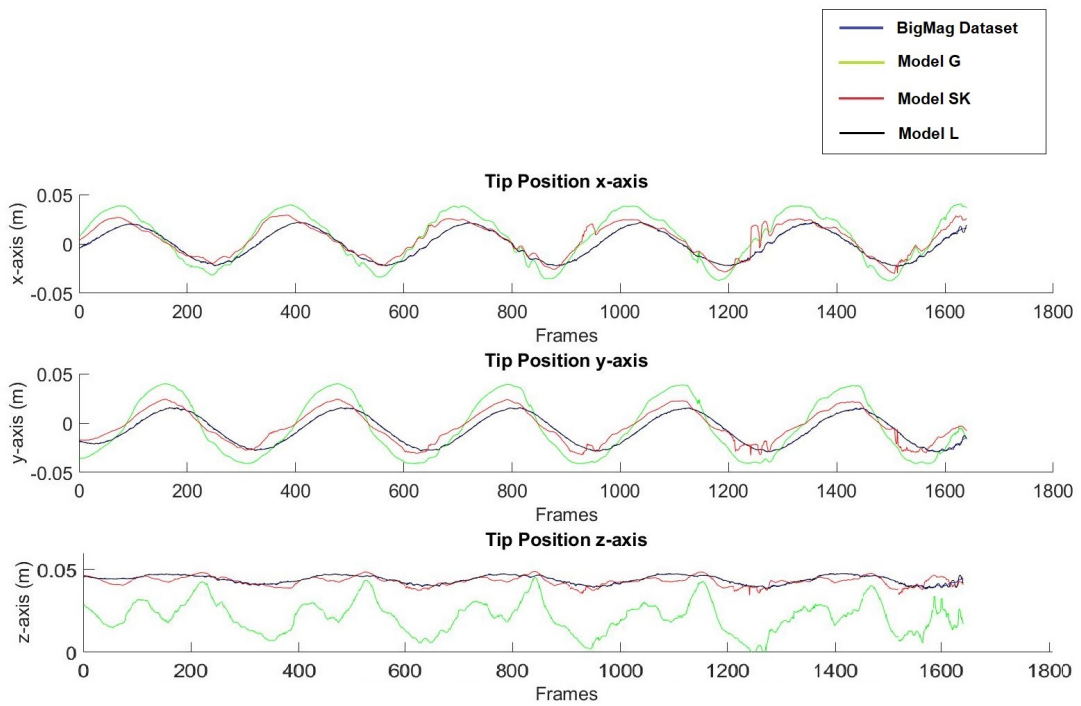


Fig. 3.36: In blue BigMag's dataset, in black *ModelL* coordinates, in red *ModelSK* coordinates and in green the values with *ModelG*.

From the data obtained in, the errors in the reconstruction of the angles were calculated to evaluate the reliability of the developed models.



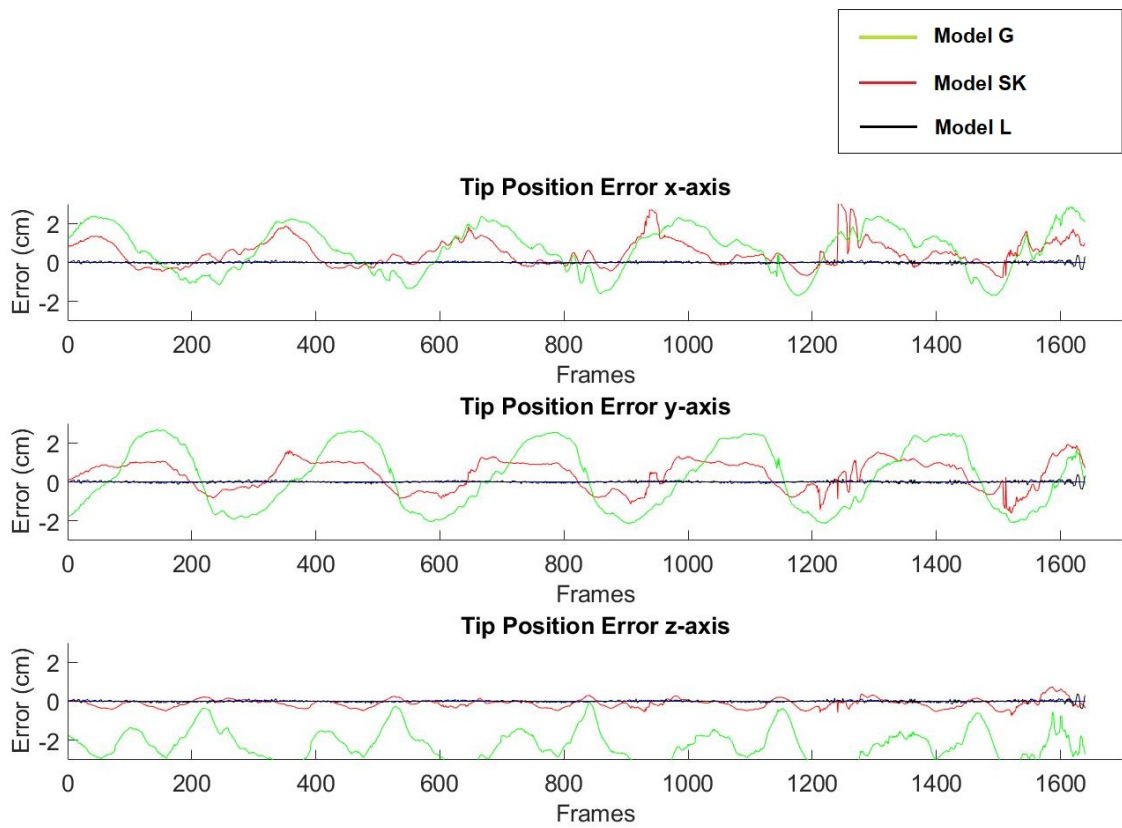


Fig. 3.37: In black *ModelL* error, in red *ModelSK* error and in green the error values obtained with *ModelG*, all compared with BigMag dataset.

The average sample errors and relative standard deviations are shown in the tables below, grouped according to the considered axis.

Tab. 3.7: Tip Position Coordinate X

<b>Error (cm)</b>	<i>ModelG</i>	<i>ModelSK</i>	<i>ModelL</i>
Mean	0.69	0.51	-0.007
Std.	1.16	0.68	0.11

Tab. 3.8: Tip Position Coordinate Y

---

---

<b>Error (cm)</b>	<i>ModelG</i>	<i>ModelSK</i>	<i>ModelL</i>
Mean	0.24	0.38	-0.004
Std.	1.61	0.76	0.08

---

---

Tab. 3.9: Tip Position Coordinate Z

---

---

<b>Error (cm)</b>	<i>ModelG</i>	<i>ModelSK</i>	<i>ModelL</i>
Mean	-2.26	-0.12	0.001
Std.	0.86	0.25	0.05

---

---

# 4

## Discussion

Once finished the writing of numerical results, it has been fundamental to reflect on them in order to describe the performance of the developed algorithm. In this section I have tried to give an objective interpretation of the outcome focusing on its strengths and limits. I have divided the review of all the results in three macro areas: in the first part I will focus on the modeling of the catheter and the initial development and evaluation of the algorithm, in the second part I will discuss the evaluation of the parameters, obtained with the linear fitting algorithm, useful for shape reconstruction and movement evaluation both for tracking and control purposes using different types of datasets. Finally I will analyze the development of the PRBM focusing on the possible limits and future developments.

### 4.1 Initial Evaluation

The first phase of the project focused on the generation of a catheter model with Solidworks in order to obtain good synthetic data to develop the linear fitting algorithm. The first limit showed by this method is the particular generation of the point clouds, in which the points were distributed in ordered circles due to the approximation method of the surface used by the STL format.

This problem increased further in cases where the catheter flexion angle was zero or a few degrees. In this case the simplified modeling of the STL reduced the point clouds to very few points, insufficient in order to use them properly in the linear fitting algorithm.

During the development phase of the model it we have decided to increase the resolution of the STL format or introduce alterations on the surface to increase the generation of triangles on it.

The initial evaluation the linear fitting quality, described in figure ( 3.1), showed a good performance of the algorithm in evaluating the body of the catheter; the only outlier was a point in the initial part of the catheter body, generated by Solidworks. The high error values in the right part of the graph are justified by the fact that the distance of the points in the catheter's tip was much greater than the radius of the body.

The algorithm has been designed to be easily adapted to different cases, for example by setting a variable number of segments. A model with only a few segments would be sufficient for low bidimensional flexions. Increasing the complexity, for example with a 3-D flexion, or the amount of the flexion would instead increase the number of segments to better describe the evolution of the catheter. However, the graph ( 3.2) has shown that the increase in the number of segments does not always correspond to an improvement in the description of the catheter for two reasons:

- A fixed segment length combined with the inevitable errors of the fitting algorithm have produced a quality loss in the description of the tip. The model with 7 links showed a better modeling of the tip compared to the 14 links model. Indeed, due to the errors, the model 'takes a longer path' than the one that theoretically should, reaching in this case the catheter's length of *5cm* before reaching the tip of the catheter.
- Increase of computational cost.

### 4.2 Validation Method A

The first experimental Protocol A1, developed for the linear fitting algorithm, showed a poor reliability in evaluating the curvature radius considering three consecutive points with relative errors up to 10%: probably this is due to the fitting error that does not exactly follow the line passing through the center of the catheter body but deviates laterally generating errors in the calculation of the curvature.

However, as it is possible to see in table (3.1), for a single measurement the radius evaluated with the average model has given a relative error of 0.1% which can be considered a very good result compared to the literature.

The goodness in the evaluation of the radius of curvature has been demonstrated in the Protocol A2 in the subsequent test, showed in (3.4), in which models with decreasing curvature radius were implemented. These tests showed a good behavior with sample mean and standard deviation of the relative error of  $1.2 \pm 0.86\%$ .

Figure ( 2.9) showed how a stereo vision point clouds looked like. To increase the similarity between the real stereo vision point clouds and the synthetic one, the noise has been added with different amplitude.

With a high resolution of the STL model and the addition of random noise in the three dimensions it has been possible to generate populations of points similar to those generated by the system mounted on BigMag. The addition of noise clearly affected the effectiveness of the fitting algorithm together with 3-D flexion attempts. The table (3.2) shows however a very low average sample variation, equal to  $0.07 \pm 0.046mm$ , with a maximum noise equal to 10% of the radius length; increasing the variations proportionally to the noise up to a maximum of  $0.38 \pm 0.1mm$  in the case of noise up to 90% of the radius length, which is still comparable with literature. Figure ( 3.5) has shown a good behavior of the model fitting with both point clouds.

The evaluation of the curvature tested in the Protocol A3, showed a further limitation in the evaluation of the fitting algorithm using only the simple positions of the virtual joints. The variation of curvature, clearly visible in the 3-D printed object, could not be related to the model if not a priori by the user because on the catheter there was no reference mark that allowed to recognize the exact starting point or variation of the bending.

The intervention of the user has led to more variable results compared to previous tests, demonstrating however in each test a good quality of evaluation of the angle and repeatability of the linear fitting algorithm.

In general the results are similar for the both curvatures, with a comparable underestimation of the parameter both before and after filtering. As reported in the table (3.3), in the evaluation of the first curvature of  $100m^{-1}$ , the unfiltered and filtered models behaved similar with an estimation of  $(93.32, 5.13m^{-1})$  and  $(92.73, 1.57m^{-1})$  respectively, with

the only exception of the sample standard deviation which has been reduced after filtering operations.

Of the same trend the results obtained for the curvature equal to  $200^{-1}$ , in which for unfiltered and filtered model the results has been  $(189.79, 7.14m^{-1})$  and  $(192.66, 4.95m^{-1})$  respectively.

Therefore it has been possible to state that the filtered model was slightly better performing from the point of view of the mean error, with a maximum of  $-7.3m^{-1}$  against  $-10.2m^{-1}$  of the unfiltered one, with a lower variability of the measurement confirmed by the range of sample standard deviation of the error of  $(1.57 - 4.95m^{-1})$  against  $(5.13 - 7.14m^{-1})$  of the unfiltered one, as reported in table (3.4).

The evaluation of the curvature has once again proved to be good compared to the state of art and considering the necessary intervention of the user in the selection of the joints to be used for the calculation of the parameters.

### 4.3 Validation Method B

The Protocol B1 of this validation method was not aimed at assessing the quality of the linear fitting algorithm, but rather to test a first approach with real data provided by BigMag and verify the quality of the magnetic control on the catheter in the performance of a helical trajectory.

In this task, after an initial settling phase, the parameters  $x$  and  $d$  should have set to a constant value and the variation of the bending angle  $\alpha$  due to the constant angular velocity should be linear as required by the task.

Graphs ( 3.10) and ( 3.11) showed a lack of control of BigMag system that was still not perfect because the values obtained by the linear fitting were not constant during the whole test. However, it has been possible to see a periodic behavior of both parameters, that could mean a different behavior of the catheter depending on the direction of bending due to an inhomogeneity of the material and not only an imperfection in the control, which is much more likely to be the reason for high-frequency noise visible for example just before frame 300.

Also the evaluation of  $\alpha$  parameter has shown an angular velocity not completely constant, with a less important noise at high frequencies.

The evaluation of the quality of the catheter control has moved to the position of the catheter tip, which is essential for the control. Graphs ( 3.13), ( 3.14) and ( 3.15) has shown a good estimation of the tip's coordinates with the exception of the frames before the 300<sup>th</sup> where the large high frequency noise generates abrupt variations of the tip position.

This effect should be clearly avoided from the perspective of a future biomedical application, so it has been decided to perform a low pass filtering in order to eliminate these sudden variations and provide a smoother movement to the control algorithm. Indeed, a rapid variation generates a different response of correction of the movement, thus engaging a chain reaction that causes the large variations at high frequency shown in the graphs. The moving average filter applied to the tip coordinates has given good results for control purposes showing a much smoother behavior.

After a first qualitative analysis of the three parameters, a validation of the linear fitting algorithm's quality was performed through the comparison with known values provided by the new Maya software used for modeling and catheter animation, as described in the Protocol B2.

As regards parameter  $x$ , from table (3.5), an overestimation of the parameter was generally obtained with a relatively small relative error (1.35, 0.86%) which has been confirmed opposite but similar also for parameter  $d$  (1.72, 1.02%). The two results showed a correlation because a greater distance  $x$  from the origin of the catheter assumes a lower bending angle and therefore a smaller parameter  $d$ . The error could therefore be due to the lack of accuracy in the estimation of the tip's position by the fitting algorithm.

From table (3.5) is also described the behavior of the fitting algorithm in bending angle  $\alpha$  evaluation. Results for relative error showed a high sample standard deviation compared to the mean sample error (0.96, 1.44%): this particular dynamic is explained by the graph ( 3.24) in which we can see a relative error peak at the 100<sup>th</sup> frame, which correspond to the 0° angle from picture ( 3.23).

Clearly the angle's relative error is strongly influenced by the amplitude of the angle value in case of equal angular variation. The measured difference between the two angles has

been reported too, showing a more constant behavior over time with a good performance of the algorithm in the estimate of  $\alpha$  ( $-0.02, 0.97^\circ$ ).

It also showed that the algorithm did not undergo any noticeable changes in the performance between the first and last phases of the simulation with variable angular velocity and the central phase with almost constant angular velocity.

The simulation also highlighted a high performance of the linear fitting algorithm comparable to the literature. Furthermore was clear the possibility of further development of the Maya software for movement and animation simulations with the possible introduction of gravity force and force fields for the simulation not only of the catheter but of the entire environment.

In Protocol B3, the evaluation of the curvature radius with the model fitting algorithm for stereo vision system showed a good performance regarding the difference between measured and real value ( $0.017, 0.1\text{cm}$ ) with a relative average error of ( $2.39, 1.64\%$ ) as reported in table (3.6). On the contrary, the sample mean of the relative error found by the same algorithm exploiting not the model's joints but the centers of the slices obtained with the ultrasound processing showed a much higher error ( $22.81, 5.61\%$ ).

Comparing it to what has been obtained with the processing of stereo vision data, it has been noticed that the average error is ten times higher, therefore an excessive relative error in order to be considered as a good result; however the sample standard deviation showed a smaller value in proportion to the magnitude of the error, reduced to three times the error with stereo vision processing, meaning a general overestimation of the parameter but a good robustness and a good repeatability of the algorithm.

Moreover, for real-time tracking purposes, it would not be possible to exploit the centers of all the slices given the limited speed of movement of the probe. A hypothetical future algorithm should use only a small part of the slices corresponding to a small part of the catheter in order to provide the parameters in real-time.



#### 4.4 Validation Method C

With the introduction of the PRBM algorithm, the validation was concentrated in the evaluation of the algorithm comparing it to an optical tracking system according to the literature and based on the linear fitting algorithm developed previously. In the various validation protocols, a progressive improvement of the performance has occurred thanks to the changes made on the initial PRBM algorithm.

The model used for the implementation of the PRB has been intuitive in conceiving the bending angle of each joint as a simple rotation around a vector lying on a plan orthogonal to the initial direction of the link, as inspired by Chasles's screw theory. Moreover, the creation of a relationship between the joints and the local reference systems allowed to generate a single global response of the model to an external input provided in the tip only.

After the first phase of implementation of the model on MATLAB, we have passed directly to the simulation on Simulink in order to verify the behavior of the model based on the artificial inputs supplied.

For the first time during the whole project, a relationship between the applied magnetic field and the model configuration has been created through a model implemented on MATLAB.

Thanks to the comparison with the data provided by BigMag, shown in figure ( 3.29), it was possible to immediately verify the quality of the configuration reconstruction. All the models described below have been implemented on MATLAB and loaded on the Simulink model in order to simulate a hypothetical test in real time.

In Protocol C1, results in figure ( 3.30) showed immediately that the PRBM algorithm was correctly implemented because the reconstruction of the angles followed the periodic trend of BigMag's dataset.

In *ModelG*, only the gravity component has been inserted in addition to the magnetic component provided by BigMag. The trend was smoother than the *ModelGQ* in which the initial bended configuration of the catheter due to gravity was also considered, calculated with the stereo vision system.

Precisely because in the graph was evident an excessive amplitude both positive and neg-

ative in the measurement of the angles for all joints and directions, we have decided to evaluate the compliance of the joints, the parameter that directly connect the wrench applied to the magnetic tip and the configuration of the whole model.

Starting from BigMag's dataset, the compliance was then analytically calculated and replaced with the one supplied at the beginning and the subsequent *ModelNC* implemented with the new compliance proved to be much more performing than the initial model, as shown in the figure ( 3.31).

The new compliance showed comparable but not equal values in both directions  $x$  and  $y$  for the first two joint and a higher value for the third joint, justified by the fact that in the last part of the catheter was installed the permanent magnet, much more rigid of the flexible catheter body. In these early phases the amount of the error was not evaluated quantitatively but only qualitatively because the error in the prediction was much higher than 100%.

However, analyzing the images, we found a temporal shift between the angles saved on the BigMag CSV file and the wrench applied in the same frame. The shift, explained both by the fact that the introduction of a magnetic moment in the tip would inevitably generate a delay in the generation of the respective angle, and a possible delay due to the control algorithm, was quantified by *finddelay* MATLAB function in 19 frames in average, corresponding to about 1.9s. However, an in-depth study of the delay has not been performed.

After the new evaluation of the compliance taking into account the temporal shift, the angles obtained by the new *ModelS*, shown in figure ( 3.32), were revalued, showing a further improvement of the performance both in amplitude and in phase shift with respect to the previous model.

At this point of the development, the error was qualitatively still too high compared to the literature and the reference value of the linear fitting algorithm. Therefore it was not possible to allow a validation of these initial models of PRBM algorithm, with a too high difference with the dataset in the positive and negative peaks rather than in the ascending or descending phases.

Analyzing the real catheter, a different elasticity of the catheter has been found when the bending tip was changed. Compliance could therefore not be constant throughout the test

and in part this information could explain the greater reconstructed angle than the real one.

For this reason, we decided to introduce non-constant compliance with a cubic component linked to the bending angle and a first-degree component, therefore introducing a corrective factor to compliance directly proportional to the bending angle. The new compliance has required the modification of the whole optimization algorithm for the reconstruction of the angles, but has nevertheless led to a further improvement of the model, impossible to achieve with the previous constant compliance.

In (3.7) are reported the final values evaluated for the compliance, showing a negative corrective cubic factor and a linear component similar to the previous one.

From figure (3.33) of the Experimental Protocol C2, it has been possible to say that the *ModelSK* with cubic compliance behaved better than the constant one especially in the maximum and minimum peaks during the task, areas where the previous models made the bigger errors in the reconstruction of the joints angles.

The analysis of the resolution speed of the optimization algorithm produced comparable values in terms of the resolution time of the *fsolve* function, both below the tenth of a second (0.082,0.075s); the time can be considered valid for real-time implementation of the algorithm.

However, even this model showed still high errors especially in the two directions of the first joints and sudden variations concentrated in the maximum and minimum points mainly. Analyzing the results we have concluded that the complexity of the elastic behavior of the whole catheter in different directions most likely could not be described sufficiently adequate by a single parameter in order to obtain an accurate reconstruction of the angles throughout the entire trajectory required in the task.

Therefore, notwithstanding the initial model, a clear improvement in the quality of reconstruction had been achieved and it was decided that, from a perspective of development and further improvement, it was necessary to exploit further information for the reconstruction of the parameters.

A further development of the PRBM algorithm performance considering only the parameter compliance would have required an analysis and improvement of the real catheter itself, in the search of similar elastic characteristics for all directions of movement, anal-

ysis that was not carried out in this project.

Since BigMag system was operating at a speed sufficient to allow real-time control, it was decided to introduce a state observer that would have exploited both the model equations seen previously in (2.46) and the information obtained in output in the previous frame for the evaluation of the current angles.

Luenberger state observer has been implemented in a necessarily simplified version, using finite difference approximations to obtain numerical differentiation and thus obtain the solutions of the model equation and by simplifying the Luenberger  $L$  factor to a diagonal matrix multiplied by a preset corrective factor. The observer's gain  $L$  had to be chosen to make the continuous-time error dynamics converge to zero asymptotically.

Luenberger state observer has been chosen instead of a common Kalman Filter or an Extended Kalman filter because in our case the only information available were the equations of the model and the output of the previous iteration. Kalman filters are called estimators because they are mainly used for stochastic cases, in which measurements and measurement errors are known or introduced. In our case the estimation of the measurement error was not easily quantifiable since it was a reconstruction from a model and not a true measurement for which we chose to use a Luenberger observer and consider the test as deterministic.

In Protocol C3, Luenberger observer has been implemented in the PRBM algorithm (called *ModelL*) and evaluated. The results obtained in the reconstruction of the joints angles, shown in the figures ( 3.34) and ( 3.35), described a significant improvement in the quality of the output, comparing what obtained with the best result of the previous steps. In each direction and for all the joints, the model adequately described the values of the parameters, highlighting a wider oscillation at another frequency in the case of a higher correction factor.

This is determined by the fact that when the observer gain  $L$  is high, the linear Luenberger observer converges to the system states very quickly. However, high observer gain leads to a peaking phenomenon in which initial estimator error can be prohibitively large. Reducing the gain factor of the model slowed down the speed of convergence, achieving a smoother output that could be more effective from the point of view of control over the catheter.

Once obtained these results and considered them qualitatively consistent, the actual position of the catheter tip was evaluated with respect to the value provided by the BigMag stereo vision system, evaluating the error in the estimate of the coordinates.

After the evaluation of the angles of the joints, fundamental for the improvement of the PRBM algorithm, the validation of the main models obtained was performed. In Protocol C4, the initial model *ModelG*, the *ModelSK* and *ModelL* have been compared to the linear fitting algorithm implemented on BigMag, which could be considered a comparison with literature. Tip's position has been evaluated for each model and compared in figure ( 3.36).

The graph showed a marked improvement in the quality of the reconstruction between the first and last model, especially in the  $z$  axis. The error in tip's position has been evaluated in figure ( 3.37) and reported in tables (3.7), (3.8) and (3.9).

From the results it could be seen that for the  $x$  coordinate the average sample error with the *ModelL* has been almost zero with a small deviation  $(-0.007, 0.11cm)$ , much higher the results for the other two models *ModelSK* and *ModelG*,  $(0.51, 0.68cm)$  and  $(0.69, 1.16cm)$  respectively, with a sample standard deviation at least six times *ModelL*. The evaluation on the other two axes can be considered analogous with a particularly high error of the *ModelG* with respect to the mean error in the evaluation of the  $z$  coordinate equal to  $-2.26cm$ .

The results thus confirm the goodness of *ModelL* which performance is totally comparable to the system mounted on BigMag that exploits visual information widely used in the state of the art.

# 5

## Conclusions and Perspectives

This thesis, in collaboration with Surgical Robotics Lab of the University of Twente, has supported the development of the complex BigMag system by approaching and deepening different issues.

The whole project has been developed on three main themes: the first one included catheter modeling, conceived as a continuum object in a workspace, data processing and the development of a fitting algorithm to model the catheter itself. As a result it has been possible to describe its position and orientation with respect to a pre-established reference system. The second theme has been the validation of this linear fitting algorithm, focusing mostly on the evaluation of important parameters, especially in a perspective of control of the same catheter by BigMag. The last one consisted in the development and validation of a PRMB that widened the relationship between the various joints, calculated with the fitting model, in order to relate the catheter model with the forces acting on it.

The first phase of the project showed the fundamental contribution of synthetic data for the development and testing of the linear fitting algorithm, through the use of software such as Solidworks and Maya. The former, a software widely used in the designing field, it is characterized by high precision, but lacks of the ability to generate a large volume of data in an automated way.

In this sense, Maya proved to be more suitable for this purpose, allowing to animate the same object and obtain the automatic saving of each frame, speeding up the generation of synthetic data and allowing to simulate with a good design quality a real movement of a

catheter, as shown in this study.

In the deepening of Maya software, I also discovered the possibility of inserting gravity, force fields and disturbing agents such as wind, and countless other components that could be present in an environment in the workspace. From this point of view, there could be also a strong evolution and improvement of synthetic data in order to increase the development of technology in this field.

The initial development of the linear fitting algorithm showed a good overall quality in describing a point clouds in comparison with literature, highlighting however the limits related to the relationship between quantity of flexion and quality in the description. A model with many segments could adequately provide high flexion while the quality in the description of the final part of the catheter is lost due to geometric limits and errors and vice versa. Therefore this type of modeling needs to be able to adapt to different situations, and in this sense a dynamic model could be implemented.

To overcome the quality problem a different kind of model could be generated. It should either be made by curves or functions of a higher degree than the first, or be a cylindrical one that already incorporates the volumetric characteristics of the object it describes. Clearly the evolution in this sense would rationally lead to an increase in the computational cost.

The second step showed how an algorithm with an overall reduced computational cost could still provide valid information for use in real time.

The evaluation of the parameters addressed in this study provided good results also in the description of the catheter tip position, fundamental both for the control and for a biomedical application, where the tip becomes the application point of an end-effector instrument, of medicine release or a control point.

Low-pass filtering of data has proved to be a concrete help, especially in terms of external control of the device.

The limit of the linear fitting algorithm found in the evaluation of complex shapes, with the presence of two different curves in the model, has also introduced the problem of user intervention in the evaluation of the parameter. It could be overcome in different ways: a possibility is to introduce a control on the positions of adjacent joints in order to recognize a geometric variation of the configuration and identify the joints where the change occurs

or from which it originates. Another option is to use other types of sensors such as FBGs, which due to their small size could be installed inside the catheter, thus more effectively monitoring a configuration of the catheter even more complex.

The comparison in Validation method B showed that processing of stereo vision data seemed more accurate than ultrasound data processing in the analysis of motion parameters exploiting the same linear fitting algorithm. A further study on the ultrasound system could lead to an improvement of the performances, considering that with a simple 2-D ultrasound probe it would not be possible to analyze the entire workspace. Therefore it would be necessary to develop an algorithm able to provide information on the whole model starting from only a part of it.

However, this system would have greater potential than the stereo vision in the perspective on an application of the catheter within a body volume, in which the latter would not be able to track it correctly.

In the end, a PRBM was implemented that provided further information on the model, starting from its basic elastic characteristics, and a relationship between each joint was generated. In this way the effect of an external event applied to the catheter tip would affect the configuration of the whole model.

Thanks to the PRBM algorithm, it has been possible to relate the external magnetic field acting on the magnetic component to the configuration of each joint and therefore to the catheter's.

All the study of the parameters indicated how wide the margins for improvement of a model are in updating and developing the parameters used: an example of this is the introduction of a more complex compliance. In order to help describing the not always homogeneous behavior of the catheter due to imperfections of the material or the setup, further studies could focus on other parameters, which also allow the model to adapt more easily to different conditions of use.

Despite the marked improvement in the early stages of the development of the PRBM, it was not possible to obtain qualitatively comparable results with the tracking technique mounted on BigMag and used as a reference. Furthermore, a thorough study of the material and a high quality realization of the catheter could lead to an improvement of the algorithm's performance.



## 5. CONCLUSIONS AND PERSPECTIVES

---

Finally, an initial approach to the theory of state observers and a first implementation of a Luenberger state observer in the PRBM algorithm showed a remarkable improvement of the model through the control on the previous output in order to correct the current evaluation. The results in the reconstruction of the position of the catheter's tip are comparable to the reference technology and therefore to the tracking systems used in literature.

Such a small error in the evaluation of the position of the catheter tip could allow the development of a clinical application of the instrument at least in the performance of simple tasks. The great potential of the PRBM algorithm is to obtain the configuration directly from the relation between the known applied magnetic field and the magnetic component of the catheter, an additional information compared to a simple visual reconstruction. These two technologies could be merged together in order to maximize the collection of real-time information on catheter behavior.

The development direction of the last phase of the project seems to lead to very satisfactory results from the point of view of quality, reconstruction reliability and therefore the control by optimizing the corrective factors and reducing the oscillations at high frequency. However, it would be necessary to verify the possibility of implementing it on BigMag without preventing the system from working in real time.

# **Appendices**

---

## .1 Moore-Penrose Pseudoinverse

Moore-Penrose pseudo-inverse provides a least squares solution to a system of linear equations. There are two main modalities to compute the pseudo-inverse:

- Singular Value Decomposition (SVD): Decomposition of a matrix  $C$  ( $m \times n$ ) is made by factorization of  $C$  into the product of three different matrices:

$$C = U\Sigma V^T \quad (.1)$$

where  $U$  is a  $m \times m$  unitary matrix,  $\Sigma$  a rectangular diagonal matrix with non-negative real numbers of the same dimension on  $U$  and  $V$  is a  $n \times n$  unitary matrix.

It can be proved that:

$$C^\dagger = U\Sigma^\dagger V^T \quad (.2)$$

$\Sigma$  is a diagonal matrix so it is simple to compute its pseudo-inverse: the elements can be considered smaller than a tolerance  $\varepsilon$  to be null and we substitute every non-null elements on the diagonal with their reciprocal.

- $C$  has linearly independent columns, therefore  $(C^T C)$  is invertible and:

$$C^\dagger = (C^T C)^{-1} C^T \quad (.3)$$

---

## .2 Main Functions

### .2.1 Linear Least Square Function

```
1 %Linear least squares function
2 function [m,R,Q2]=lsfit(pos,vect_lsdata,Q,sum_vect,seg_len)
3 %vect_lsdata contains the coordinates of the selected
   points
4 %Q is the current joint
5 %seg_len is the length of each link
6 i=0;
7 C=zeros(pos*3,1);
8 T=zeros(pos*3,3);
9 %y=mx+q looking for m with least squares fitting
10 for j= 1:pos
11     for k = 1:3
12         i = i+1;
13         %C= difference (y-q)
14         C(i)= vect_lsdata(j,k)-Q(k);
15         %T is the diagonal distance vector
16         T(i,k)=sum_vect(j);
17     end
18 end
19 %Transpose of the diagonal distance
20 R=transpose(T);
21 %Finding the coefficient m
22 m=inv(R*T)*R*C;
23 m=m/norm(m);
24 Q2=seg_len*transpose(m)+Q;
25 %New joint calculation with new fitting line
26 end
```

---

## .2.2 Curvature and Radius Evaluation

```
1 %Function for radius and curvature evaluation
2
3 function [c, r, k] = calc_circle3D(pt1, pt2, pt3)
4 %Vectors generation
5 t = pt2-pt1;
6 u = pt3-pt1;
7 v = pt3-pt2;
8 w = cross(t,u);
9 t2 = sum(t.^2);
10 u2 = sum(u.^2);
11 w2 = sum(w.^2);
12 %Parameters evaluation
13 c = pt1+(t2*sum(u.*v)*u-u2*sum(t.*v)*t)/(2*w2); %Centre
14 r = 1/2*sqrt(t2*u2*sum(v.^2)/w2); %Radius
15 k= 1./r; %Curvature
16 end
```

## .2.3 Validation Error First phase

```
1 %Function used to evaluate the error in the first
  validation phase
2
3 function [error, count]= checkError(vect_lsdata, origin, k,
  error, count)
4
5 Q1= origin(k,:);
6 Q2= origin(k+1,:);
7 for i = 1:length(vect_lsdata)
8     %Find distance of each point to the line
```

```

9         error(count,1) = norm(cross(Q2-Q1,vect_lsdata(i,:)-
10             -Q1))./norm(Q2-Q1);
11         %Relative error with ideal radius equal to lmm
12         error(count,2) = abs(1 - error(count,1)).*100;
13         count= count+1;
14     end
end

```

#### .2.4 Iterative Points Selection

```

1 %Function used to check which points have a distance from
   the current origin lower than the segment's length
2
3 function [vect_data ,data_out ,j ,sum_vect]=checkPoints (data ,Q
   ,seg_len)
4 j = 0;
5 pos=1;
6 to_remove=zeros (size (data ,1) ,1);
7 data_out=data ;
8 for k = 1:length (data)
9     sum_current = norm (data (k,:) -Q);
10    if sum_current < seg_len
11        j = j+1;
12        sum_vect (j)=sum_current ;
13        to_remove (k)=1;
14    end
15    k = k+1;
16 end
17 vect_data= zeros (j ,3);
18 sum_vect=zeros (j ,1);
19 for z = 1:length (data)

```

```

20     if to_remove((length(data)-z+1),:)== 1
21         %Saving each point distance
22         sum_vect(pos) = norm(data((length(data)-z+1),:)-Q);
23         %Coping each point value
24         vect_data(pos,:) = data((length(data)-z+1),:);
25         pos= pos+1;
26         %Cleaning data (each point can be considered once)
27         data_out((length(data)-z+1),:)=[];
28     end
29 end
30 end

```

## .2.5 Stereo Vision Processing

```

1 seg_num=7;
2 seg_len = 5./seg_num;
3 folder_name='curvaturefourty /';
4 %Noise amplitude
5 noise_amplitude = 0.3;
6 %Data storage
7 load('curvaturefourty/voxels_frame.mat');
8 origins_global=zeros(seg_num+1,3,size(voxels_frame,2));
9 stability_total=zeros(seg_num,3,size(voxels_frame,2));
10 stability = zeros(seg_num,3);
11 %Frame-by-frame curvature and radius
12 curvature_total=zeros(seg_num,1,size(voxels_frame,2));
13 radius_total=zeros(seg_num,1,size(voxels_frame,2));
14
15 for z= 1:size(voxels_frame,2)
16     %Array of curvature ,radius and center info below
17     curvature = zeros(seg_num,1);

```

```

18 radius = zeros(seg_num,1);
19 circle = zeros(seg_num,3);
20 noise_signal = zeros(size(voxels_frame{z},1),3);
21
22 %Noise addition to point clouds
23 for u= 1: size(voxels_frame{z},1)
24     disturb = noise_amplitude*rand(1);
25     if mod(disturb,0.002) > 0.001
26         noise_signal(u,1) = voxels_frame{z}(u,1) +
27             disturb;
28     else
29         noise_signal(u,1) = voxels_frame{z}(u,1) -
30             disturb;
31     end
32 end
33 for u= 1: size(voxels_frame{z},1)
34     disturb = noise_amplitude*rand(1);
35     if mod(disturb,0.002) > 0.001
36         noise_signal(u,2) = voxels_frame{z}(u,2) +
37             disturb;
38     else
39         noise_signal(u,2) = voxels_frame{z}(u,2) -
40             disturb;
41     end
42 end
43 for u= 1: size(voxels_frame{z},1)
44     noise_signal(u,3) = voxels_frame{z}(u,3);
45 end
46 %Sortening array
47 B = sortrows(noise_signal,1);

```



```

44     noise_signal = B;
45     Q2 = findOrigintorsionMaya(noise_signal); %First vector
         origin
46     Q = Q2;
47     origins(1,:) = Q; %Array of origins from lssquare
48     lsdata_n = noise_signal; %Create a copy of noisy data
49
50 %% Processing
51 %New points depending on distance
52     for k=1:seg_num
53         [vect_lsdata ,lsdata_n ,pos ,sum_vect] = checkPoints(
         lsdata_n ,Q, seg_len);
54         dataX= vect_lsdata(:,1);
55         dataY= vect_lsdata(:,2);
56         dataZ= vect_lsdata(:,3);
57         %Least square error
58         [m,R,Q] = lsfit(pos ,vect_lsdata ,Q, sum_vect , seg_len
         );
59         %Update origins datastore
60         origins(k+1,:) = Q;
61         %Computing x parameter
62         stability(k,1) = abs(origins(k+1,1)-origins(1,1));
63         %Computing d parameter
64         stability(k,2) = norm(origins(k+1,:)-[origins(k
         +1,1),origins(1,2:3)]);
65         %Computing alpha parameter
66         stability(k,3) = (((asin(norm(cross([0,1,0],[
         origins(k+1,:)-[origins(k+1,1),origins(1,2:3)
         ]])))/norm([origins(k+1,:)-[origins(k+1,1),
         origins(1,2:3)]])))*360)/2)/pi;

```

```

67      %Bending angle , 4 quadrants
68      if origins(k+1,3) < origins(1,3)
69          if origins(k+1,2) < origins(1,2)
70              stability(k,3) = 270 + (90-stability(k,3));
71          else
72              stability(k,3) = 180 + stability(k,3);
73          end
74      else
75          if origins(k+1,2) > origins(1,2)
76              stability(k,3) = 90 + (90-stability(k,3));
77          end
78      end
79      if (k >= 2 && k < seg_num)
80          [ circle(k-1,:), radius(k-1,:), curvature(k-1,:)
            ] = calc_circle3D(origins(k-1,:),origins(k
            ,:),origins(k+1,:)); %curvature evaluation
81      end
82      if k == seg_num
83          [ circle(seg_num,:), radius(seg_num,:),
            curvature(seg_num,:) ] = calc_circle3D(
            origins(1,:),origins(4,:),origins(8,:)); %
            curvature evaluation
84      end
85  end
86  %Data Storage
87  origins_global(:,:,z)=origins;
88  stability_total(:,:,z)=stability;
89  curvature_total(:,:,z)=curvature;
90  radius_total(:,:,z)=radius;
91  for q= 1:200

```

```

92         angles(q,1)=stability_total(7,3,q);
93         curvature(q,1)=curvature_total(7,1,q);
94         distanced(q,1)=stability_total(7,2,q);
95         distancex(q,1)=stability_total(7,1,q);
96     end
97 end
98 %% Example of Parameter Plot
99 figure
100 plot(permute(stability_total(7,3,:),[3,2,1]));

```

## .2.6 Tip Position

```

1 %Function to evaluate the position of the catheter's tip
  from joints angles.
2 function [EEpos] = EndEffectorPosition (thetas)
3 seg_num = size(thetas,1)/2;
4 theta=zeros(2,seg_num);
5 tot_len= 0.05;          %Catheter length
6 Hframestack_r=zeros(4*(seg_num+1),4);
7 H0=eye(4);
8 Hframestack_r(1:4,:)=H0;
9 pos=0;
10 %Angles organization
11 for i=1:seg_num
12     pos=pos+1;
13     theta(1,i)=thetas(pos);
14     pos=pos+1;
15     theta(2,i)=thetas(pos);
16 end
17 %Initial trasformation matrix
18 for i=1:seg_num

```

```

19     Hframestack_r((i+1)*4-3:(i+1)*4,:)=H0;
20     Hframestack_r((i+1)*4-3:(i+1)*4-1,4)=H0(1:3,1:3)*[0;0;i
        *tot_len/seg_num];
21 end
22 Hframestack=Hframestack_r;
23 twistTot=eye(4);
24 Twiststack=zeros(4*size(theta,2),4);
25 %Trasformation matrix with current angles
26 for i=1:size(theta,2)
27     Hprev=Hframestack_r((1:4)+4*(i-1),:);
28     Hcurr=Hframestack_r((1:4)+4*(i),:);
29     thetacurr=[1;0;0]*theta(1,i)+[0;1;0]*theta(2,i);
30     Twiststack((1:4)+4*(i-1),:)=[[skew(thetacurr),-cross(
        thetacurr,Hprev(1:3,4))];zeros(1,4)];
31     twistTot=twistTot*expm(Twiststack((1:4)+4*(i-1),:));
32     Hframestack((1:4)+4*(i),:)=twistTot*Hcurr;
33 end
34 %End Effector position
35 EEpos= Hframestack(size(Hframestack,1)-3:end-1,4);
36 end

```

## .2.7 BigMag Data Extraction

```

1 %BigMag's algorithm to extract data (given for the project)
2 data=csvread('PIDExperimentJH6.csv');
3 cut=1;
4 %Global reference system
5 Hglobal=[[ -1,0,0,2.8];
6          [0,1,0,3.7];
7          [0,0,-1,119]
8          [0,0,0,1]];

```

---

```

9 %Initial compliance
10 compliance=[0.0012;0.0008;0.0013;0.0011;0.0023;0.0021];
11 %%
12 voxbeg=64; %Beginning Index of Voxel Vector
13 manipulatorLength=50;
14 NofPoints=11;
15 tip_dipole=0.04;
16 %%
17 i=50;
18 dataline=data(i,:);
19 magneticField_deb=dataline(52:54)';
20 nofVoxels=dataline(voxbeg);
21 currentVoxelMat=zeros(3,nofVoxels);
22
23 Hmanipulator=eye(4);
24 Hmanipulator(1,1:3)=[0;0;1]';
25 Hmanipulator(2,1:3)=[1;0;0]';
26 Hmanipulator(3,1:3)=[0;1;0]';
27 Hmanipulator(:,4)=[dataline((11:13))';1];
28 Hmanipulator=inv(Hmanipulator);
29
30 for iI=1:nofVoxels
31     currentVoxel=Hglobal*[dataline((voxbeg+(iI-1)*3+1):(
32         voxbeg+(iI-1)*3+3))';1]/1000;
33     currentVoxel(4)=1;
34     currentVoxel=Hmanipulator*currentVoxel;
35     currentVoxelMat(:,iI)=currentVoxel(1:3);
36 end
37 magneticField=Hmanipulator(1:3,1:3)*magneticField_deb;
38 %%

```

```

38 segment_positions=zeros(3,NofPoints);
39 for iI=1:NofPoints
40     current_seg_position=[dataline((1:3)+10+3*(iI-1)')';1];
41     current_seg_position=Hmanipulator*current_seg_position;
42     segment_positions(:,iI)=current_seg_position(1:3);
43 end
44 %% Reducing number of segments
45 Pmat=zeros(3*size(segment_positions,2)-3,1);
46 regmat=zeros(3*size(segment_positions,2)-3,9);
47 for iI=2:size(segment_positions,2)
48     tt=(iI-1)*50/(NofPoints-1);
49     Pmat((3*iI-5):(3*iI-3))=segment_positions(:,iI)-
        segment_positions(:,1);
50     regmat((3*iI-5):(3*iI-3),:)= [tt*eye(3), tt^2*eye(3), tt
        ^3*eye(3)];
51 end
52 cvec=pinv(regmat)*Pmat;
53 NofSegmentsReduced=4;
54 segment_positions_reduced=zeros(3,NofSegmentsReduced);
55     for iI=1:NofSegmentsReduced
56         tt=(iI-1)*manipulatorLength/(NofSegmentsReduced-1);
57         segment_positions_reduced(:,iI)=segment_positions
            (:,1)+tt*cvec(1:3)+tt^2*cvec(4:6)+tt^3*cvec(7:9);
58     end
59 %% Reference Frame Construction
60 thetas=zeros(2,NofSegmentsReduced-1);
61 Hframes_ref=zeros(NofSegmentsReduced*4,4);
62 Hframes=zeros(NofSegmentsReduced*4,4);
63 twists=zeros(4*(NofSegmentsReduced-1),4);
64 fieldBody=zeros(size(magneticField));

```

---

```

65 forceBody=zeros ( size ( magneticField ) );
66 torqueBody=zeros ( size ( magneticField ) );
67 %%
68 H0=eye ( 4 ) ;
69 Hframes ( 1 : 4 , : ) = H0 ;
70 Hframes_ref ( : , : ) = repmat ( H0 , NofSegmentsReduced , 1 ) ;
71 segments_temp = segment_positions_reduced ( : , 2 : end ) ;
72 qvec = zeros ( 3 , ( NofSegmentsReduced - 1 ) ) ;
73 qvec ( 3 , 2 ) = norm ( segment_positions_reduced ( : , 2 ) -
    segment_positions_reduced ( : , 1 ) ) ;
74 qvec ( 3 , 3 ) = norm ( segment_positions_reduced ( : , 3 ) -
    segment_positions_reduced ( : , 2 ) ) + qvec ( 3 , 2 ) ;
75 qvec ( 3 , 4 ) = norm ( segment_positions_reduced ( : , 4 ) -
    segment_positions_reduced ( : , 3 ) ) + qvec ( 3 , 3 ) ;
76 tmag = zeros ( 2 , ( NofSegmentsReduced - 1 ) ) ;
77 %%
78 for iF = 2 : NofSegmentsReduced
79 %%
80 H0 = eye ( 4 ) ;
81 qloc = [ qvec ( : , iF ) ; 1 ] ;
82 H0 ( : , 4 ) = [ qloc ( 1 : 3 ) ; 1 ] ;
83 id = ( 4 * iF - 3 ) : ( 4 * iF ) ;
84 Hframes ( id , : ) = H0 ;
85 Hframes_ref ( id , : ) = H0 ;
86 segments_temp = segment_positions_reduced ( : , iF : end ) ;
87 %%
88 for iI = 1 : size ( segments_temp , 2 )
89     s_t = inv ( Hframes ( id - 4 , : ) ) * [ segments_temp ( : , iI ) ; 1 ] ;
90     segments_temp ( : , iI ) = s_t ( 1 : 3 ) ;
91 end

```

```

92 thetas (1 ,iF -1)=atan2(-segments_temp (2 ,1) ,segments_temp (3 ,1)
    );
93 thetas (2 ,iF -1)=atan2 (segments_temp (1 ,1) ,segments_temp (3 ,1))
    ;
94 theta =[1;0;0]* thetas (1 ,iF -1)+[0;1;0]* thetas (2 ,iF -1);
95 tmag (1 ,iF -1)=norm ( theta );
96 tmag (2 ,iF -1)=sqrt ( thetas (1 ,iF -1)^2+ thetas (2 ,iF -1)^2);
97 qloc2 =[qvec (: ,iF -1);1];
98 twists ((1:3)+4*(iF -2) ,:)= [skew ( theta ) , -skew ( theta ) *
    Hframes_ref (id (1:3) -4,4)];
99 total_twist =eye (4);
100 for iT=1:iF -1
101     total_twist=total_twist*expm (twists ((1:4)+4*(iT -1) ,:));
102 end
103 Hframes (id ,:)=total_twist*H0;
104 end
105 %%
106 fieldBody=inv (Hframes (id (1:3) ,1:3))*magneticField;
107 forceBody=inv (Hframes (id (1:3) ,1:3))*([0; -0.00055*9.81;0]);
108 torqueBody=cross ([0;0;1]* tip_dipole ,fieldBody);

```

## .2.8 First Joint Evaluation

```

1 %Function to estimate the first origin position
2
3 function [Q]=findOrigin (vect_points)
4 %Q is the first origin position
5 %vect_points is the array of 3D points
6 min = vect_points (1,1);
7 check = 's';
8 i = 0;

```



```

9 y = 0;
10 x = 0;
11 z = 0;
12 %While loop end condition depends on distance
13 while check == 's'
14     i = i + 1;
15     if vect_points(i,1) < (min+1) %Control on the distance
16         x = x+vect_points(i,1);
17         y = y+vect_points(i,2);
18         z = z+vect_points(i,3);
19     else
20         check = 'n';
21     end
22 end
23 Q = [x./i,y./i,z./i]; %Evaluating the average position
24 end

```

## .2.9 Angles Evaluation

```

1 %Function to estimate the angles from the moments applied
  to each joint
2
3 function [theta] = getTheta(tau_tot , k_valuesinverse ,
  seg_num , q_0)
4 tau_x=zeros(1,seg_num);
5 tau_y=zeros(1,seg_num);
6 pos= 0;
7 %Dividing tau_tot in 2 components x and y
8 for p=1:seg_num
9     pos= pos+1;
10    tau_x(p)=tau_tot(pos);

```

```

11     pos= pos+1;
12     tau_y(p)=tau_tot(pos);
13 end
14 theta=zeros(2, size(tau_x,2));
15 for i=1:length(theta)
16     theta(1,i)=(k_valuesinverse(i)*tau_x(i)+q_0(1,i));
17     theta(2,i)=(k_valuesinverse(i)*tau_y(i)+q_0(2,i));
18 end

```

### *.2.10 Angles Evaluation with Cubic Compliance*

```

1 %Function to estimate the angles from the moments applied
  to each joint
2 %with cubic compliance
3
4 function [theta] = getTheta_cubic(tau_tot, k_valuesinverse,
  k_valuescubic, seg_num, q_0)
5 tau_x=zeros(1, seg_num);
6 tau_y=zeros(1, seg_num);
7 pos= 0;
8 %Dividing tau_tot in 2 components x and y
9 for p=1:seg_num
10     pos= pos+1;
11     tau_x(p)=tau_tot(pos);
12     pos= pos+1;
13     tau_y(p)=tau_tot(pos);
14 end
15 theta=zeros(2, size(tau_x,2));
16 for i=1:length(theta)
17     theta(1,i)=(k_valuesinverse(i)*tau_x(i)+q_0(1,i));

```

```

18     theta(1,i)=(k_valuescubic(i)*(theta(1,i)).^3 +
        k_valuescubic(i+6)*theta(1,i));
19     theta(2,i)=(k_valuesinverse(i)*tau_y(i)+q_0(2,i));
20     theta(2,i)=k_valuescubic(i)*(theta(2,i)).^3 +
        k_valuescubic(i+6)*theta(2,i);
21 end

```

### .2.11 $\Omega$ Calculation

```

1 function [w_x,w_y] = calc_wx_wy(theta_x ,theta_y)
2 %%
3 theta=sqrt(theta_x ^2+theta_y ^2);
4 w_x=zeros(3,1);
5 w_y=w_x;
6 if((theta_x ~=0) && (theta_y ~=0))
7 w_x(1)=1-(theta_y / theta)^2*(1-sinc(theta/pi));
8 w_x(2)=theta_x *theta_y / (theta ^2)*(1-sinc(theta/pi));
9 w_x(3)=-theta_y / (theta ^2)*(1-cos(theta));
10 w_y(1)=w_x(2);
11 w_y(2)=1-(theta_x / theta)^2*(1-sinc(theta/pi));
12 w_y(3)=theta_x / (theta ^2)*(1-cos(theta));
13 else
14 w_x=[1;0;0];
15 w_y=[0;1;0];
16 end
17 end

```

### .2.12 Get Jacobian

```

1 function [Jacobian] = get_jacobian(thetas ,Twiststack ,
    Hframestack_init)

```

---

```

2 %%
3 Jacobian=zeros(6,length(thetas));
4 Hsaving= zeros(size(thetas,2)*4,4);
5 totalTwist=eye(4);
6 Hcurrent= Hframestack_init(1:4,1:4);
7 Hsaving(1:4,:)=Hcurrent;
8 for i=1:size(thetas,2)
9     %%
10    [w_x,w_y]=calc_wx_wy(thetas(1,i),thetas(2,i));
11    %Total twist from current to last joint
12    for iI=i:size(thetas,2)
13        totalTwist=totalTwist*expm(Twiststack((1:4)+4*(iI
14        -1),:));
15    end
16    %Total Transformation H
17    Hcurrent=totalTwist*Hframestack_init((end-3):(end),:);
18    Hsaving(i*4+1:i*4+4,:)=Hcurrent;
19    %Initial position of the ith joint
20    q_init= Hframestack_init((1:3)+4*(i-1),4);
21    %Rotation after total twist ith joint
22    Rot_ith=Hcurrent(1:3,1:3);
23    %Final position of the ith joint
24    pos_ith=Hcurrent(1:3,4);
25    %Inverse Adjoint evaluation
26    Adjinv=[[ Rot_ith',-Rot_ith'*skew(pos_ith) ] ;[ zeros(3,3),
27    Rot_ith' ]];
28    %%
29    twist_x=[cross(-w_x,q_init);w_x];
30    twist_y=[cross(-w_y,q_init);w_y];
31    Jacobian(:,(2*i-1))=Adjinv*twist_x;

```

```

30     Jacobian(:,(2*i))=Adjinv*twist_y;
31 end
32 end

```

### .2.13 Function Optimized by Fsolve

```

1 %Function optimized by fsolve. Evaluation of the moments in
   each joint to
2 %generate the applied wrench. tau_tot is the optimized
   parameter.
3
4 function [tau_diff] = solve_model_greigarn(tau_tot ,wrench ,
   seg_num ,k_values ,q_0 ,Hframestack_init)
5 %%
6 %Obtaining theta from the current tau_tot
7 theta = getTheta(tau_tot , k_values , seg_num ,q_0);
8 Twiststack=zeros(4* size (theta ,2) ,4);
9
10 %Update of joints Trasformation matrix with new thetas
11 for i=1:size (theta ,2)
12     Hprev=Hframestack_init ((1:4)+4*(i-1) ,:);
13     thetacurr=[1;0;0]*theta (1 ,i)+[0;1;0]*theta (2 ,i);
14     Twiststack ((1:4)+4*(i-1) ,:)=[[skew (thetacurr) ,-cross (
   thetacurr ,Hprev (1:3 ,4)) ]; zeros (1 ,4) ];
15 end
16
17 %Jacobian construction
18 [Jacobian] = get_jacobian(theta ,Twiststack ,Hframestack_init
   );
19 %Moment from the new jacobian and the applied wrench
20 tau_res=Jacobian '*wrench;

```

```
21 %Difference optimized by fsolve
```

```
22 tau_diff=tau_res-tau_tot';
```

```
23
```

```
24 end
```

#### .2.14 Function Optimized by Fsolve with Cubic Compliance

```
1 %Function optimized by fsolve with cubic compliance.
```

```
    Evaluation of the moments in each joint to
```

```
2 %generate the applied wrench. Theta guess is the changing  
    parameter
```

```
3
```

```
4 function [tau_diff] = solve_model_greigarn_cubic(  
    theta_guess , wrench , seg_num , k_values , k_valuesscubick , q_0 ,  
    Hframestack_init)
```

```
5 %%
```

```
6 theta_vec=theta_guess(:);
```

```
7
```

```
8 %New equation for cubic k. I don't need to find theta  
    because theta is the changing parameter
```

```
9 moments_from_compliance=diag(k_valuesscubick)*diag(theta_vec  
    ).^2*theta_vec + diag(k_values)*theta_vec;
```

```
10 Twiststack=zeros(4*size(theta_guess,2),4);
```

```
11
```

```
12 for i=1:size(theta_guess,2)
```

```
13     Hprev=Hframestack_init((1:4)+4*(i-1),:);
```

```
14     thetacurr=[1;0;0]*theta_guess(1,i)+[0;1;0]*theta_guess  
        (2,i);
```

```
15     Twiststack((1:4)+4*(i-1),:)=[[skew(thetacurr),-cross(  
        thetacurr ,Hprev(1:3,4)]]; zeros(1,4)];
```

```
16 end
```

---

```

17 %Find the manipulator Jacobian with the angle guess
18 [Jacobian] = get_jacobian(theta_guess , Twiststack ,
    Hframestack_init);
19 %Moments from the new jacobian and the applied wrench
20 moments_from_jacobian=Jacobian'*wrench;
21 %Difference that must be optimized by fsolve
22 tau_diff=moments_from_compliance-moments_from_jacobian;
23 end

```

### .2.15 Ultrasound Algorithm

```

1 seg_num=7;
2 seg_len = 5./seg_num;
3 folder_name='curvaturefourty/';
4 noise_amplitude = 0.5;
5 load('curvaturefourty/voxels_frame.mat');
6 origins_global=zeros(seg_num+1,3,size(voxels_frame,2));
7 stability_total=zeros(seg_num,3,size(voxels_frame,2));
8 stability = zeros(seg_num,3);
9 curvature_total=zeros(seg_num,1,size(voxels_frame,2));
10 radius_total=zeros(seg_num,1,size(voxels_frame,2));
11 orvoxels_frame = voxels_frame;
12 num_slice =30;
13 voxels_slice = cell(size(orvoxels_frame,2),num_slice);
14 center_slice = cell(size(orvoxels_frame,2),num_slice);
15 %Adding noise to point clouds
16 for p = 1:size(voxels_frame,2)
17     noise_signal = zeros(size(voxels_frame{p},1),3);
18     for u= 1: size(voxels_frame{p},1)
19         disturb = noise_amplitude*rand(1);
20         if mod(disturb,0.002) > 0.001

```

---

```

21         noise_signal(u,1) = voxels_frame{p}(u,1) +
           disturb;
22     else
23         noise_signal(u,1) = voxels_frame{p}(u,1) -
           disturb;
24     end
25 end
26 for u= 1: size(voxels_frame{p},1)
27     disturb = noise_amplitude*rand(1);
28     if mod(disturb ,0.002) > 0.001
29         noise_signal(u,2) = voxels_frame{p}(u,2) +
           disturb;
30     else
31         noise_signal(u,2) = voxels_frame{p}(u,2) -
           disturb;
32     end
33 end
34 for u= 1: size(voxels_frame{p},1)
35     noise_signal(u,3) = voxels_frame{p}(u,3);
36 end
37 %Ordering data
38 orvoxels_frame{p} = sortrows(noise_signal ,1);
39 end
40 for z= 1:size(orvoxels_frame ,2)
41     noise_signal = orvoxels_frame{z};
42     uscan_points = size(noise_signal ,1);
43     %Probe position. Set at first origin position
44     uscan_comparison= noise_signal(1,1);
45     %Slice thickness

```



```

46 slice_th = abs(noise_signal(size(noise_signal,1),1)-
noise_signal(1,1))./ num_slice;
47 voxels_th = zeros(size(noise_signal,1),3);
48 jj = 1;
49 j = 0;
50 s = 0;
51 %Probe linear movement simulation
52 while jj < uscan_points
53     jj = jj+1;
54     %Check points in the slice
55     if abs(noise_signal(jj,1)- uscan_comparison) <
slice_th
56         s = s+1;
57         voxels_th(s,:) = noise_signal(jj,:);
58     else
59         uscan_comparison = noise_signal(jj,1);
60         j = j+1;
61         voxels_th(~any(voxels_th,2), :) = [];
62         voxels_slice{z,j} = voxels_th;
63         %Evaluation of initial position
64         center_slice{z,j} = findCenterslice(
voxels_th, slice_th);
65         s = 1;
66         voxels_th = zeros(size(noise_signal,1),3);
67         voxels_th(s,:) = noise_signal(jj,:);
68     end
69 end
70 %Arrays for storing curvature, radius and center
71 curvature = zeros(seg_num,1);
72 radius = zeros(seg_num,1);

```

```

73     circle = zeros(seg_num,3);
74     %First vector origin
75     Q2 = findOrigintorsionMaya(noise_signal);
76     Q = Q2;
77     origins(1,:) = Q;
78     %Create a copy of noisy data
79     lsdata_n = noise_signal;
80     clf;
81     %Plot for Video Reconstruction
82     scatter3(noise_signal(:,1),noise_signal(:,2),
83             noise_signal(:,3),'blue');
84     hold on;
85     scatter3(Q2(1,1),Q2(1,2),Q2(1,3),'red');
86     axis([-10 10 -10 10 -10 10]);
87     hold off;
88     F(z)=getframe(gcf);
89     clf;
90
91
92 %% Video Maker
93 video = VideoWriter('Ultrasound','MPEG-4')
94 video.FrameRate = 30;
95 open(video);
96 writeVideo(video,F);
97 close(video);

```

## .2.16 Slice Center Estimation

---

```

2 %Function to evaluate the average position of the center of
   each slice
3
4 function [C]=findCenterslice (data , thickness)
5
6 %C is the average center of the slice
7 %Data is the array of 3D points
8 check = 1;
9 i = 0;
10 y = 0;
11 %x axis position is fixed by the moving probe simulation
12 x = (data(1,1)+data(size(data,1),1))./2;
13 z = 0;
14 %Loop to collects points. Slice's thickness is an important
   parameter
15
16 while check < size(data,1)
17     if abs(data(check,1) - data(1,1)) <= thickness
18         i = i + 1;
19         y = y+data(i,2);
20         z = z+data(i,3);
21     end
22     check = check+1;
23 end
24 %Average position
25 C = [x,y./(i),z./(i)];
26 end

```

### .2.17 Complex Curvature Validation

```

1 data=csvread('TrackingValidation.csv');

```

---

```

2 Hglobal=[[ -1,0,0,2.8];
3         [0,1,0,3.7];
4         [0,0,-1,119]
5         [0,0,0,1]];
6 voxbeg=data(1,1)*3+2;
7 %Number of segments parameter
8 seg_num=19;
9 seg_len = 0.05./seg_num;
10 origins=zeros(seg_num+1,3);
11 curvaturefilt = zeros(size(data,1),seg_num,1);
12 curvaturedata = zeros(size(data,1),seg_num,1);
13 radius = zeros(size(data,1),seg_num,1);
14 circle = zeros(size(data,1),seg_num,3);
15 %%
16 for i=2:size(data,1)
17 %%
18     close
19     nofVoxels=data(i,voxbeg);
20     NofPoints=data(1,1);
21     currentVoxelMat=zeros(3,nofVoxels);
22     for iI=1:nofVoxels
23         segment_positions=zeros(3,NofPoints);
24         %%
25         for iII=1:NofPoints
26             current_seg_position=data(i,(1:3)+1+3*(iII-1));
27             segment_positions(:,iII)=current_seg_position
28                 (1:3);
29         end

```

```

30         currentVoxel=Hglobal*[ data(i,((iI-1)*3+1+voxbeg):((
           iI-1)*3+3+voxbeg))';1]/1000;
31         currentVoxelMat(:,iI)=currentVoxel(1:3);
32     end
33     %% Movin Average to the joints' positions and
           evaluation of the results
34     origins_filt(1,:)= movmean(segment_positions(1,:),3);
35     origins_filt(2,:)= movmean(segment_positions(2,:),3);
36     origins_filt(3,:)= movmean(segment_positions(3,:),3);
37     origins_filt(:,1)=segment_positions(:,1);
38     origins_filt(:,19)=segment_positions(:,19);
39     scatter3(origins_filt(1,:),origins_filt(2,:),
           origins_filt(3,:), 'red');
40     plot3(segment_positions(1,:),segment_positions(2,:),
           segment_positions(3,:), 'black');
41     plot3(origins_filt(1,:),origins_filt(2,:),origins_filt
           (3,:), 'red');
42
43     %% Curvature is evaluated here, variable joints to
           check quality
44     stability= segment_positions';
45     stabilityf=origins_filt';
46     [c, d, curvaturefilt(i,seg_num,:)] = calc_circle3D(
           origins_filt(:,11),origins_filt(:,14),origins_filt
           (:,19));
47     cur1f(i,1) = (((asin(norm(cross([0,1,0],[stabilityf
           (10,:)-[stabilityf(10,1),stabilityf(6,2:3)])))/norm
           ([stabilityf(10,:)-[stabilityf(10,1),stabilityf
           (6,2:3)])))*360)/2)/pi;

```

```

48 [c, d, curvaturefilt(i,seg_num-1,:)] = calc_circle3D(
    origins_filt(:,6),origins_filt(:,8),origins_filt
    (:,12));
49 [c, d2, curvaturedata(i,seg_num,:)] = calc_circle3D(
    segment_positions(:,11),segment_positions(:,14),
    segment_positions(:,19));
50 [c, d, curvaturedata(i,seg_num-1,:)] = calc_circle3D(
    segment_positions(:,6),segment_positions(:,9),
    segment_positions(:,11));
51 curl(i,1) = (((asin(norm(cross([0,1,0],[stability(10,:)
    -[stability(10,1),stability(6,2:3)])))/norm([
    stability(10,:)-[stability(10,1),stability(6,2:3)]])
    ))*360)/2)/pi;
52 end

```

### .3 Simulink Additional Code

#### .3.1 Simulink: choose the correct Angles and Wrenches

```

1 %Function to choose the correct angle
2 function [theta]= Chosethetas(k,thetam ,thetak)
3 k=round(k);
4 if k<2
5     theta=thetam;
6 else
7     theta=thetak;
8 end
9 end
10 %Function to choose the correct wrench
11 function [wrench]= Chosewrenches(k,wrenchm ,wrenchk)
12 k=round(k);

```

```

13 if k<2
14     wrench=wrenchm;
15 else
16     wrench=wrenchk;
17 end
18 end
19 %Function to extract the correct iteration angles
20 function [theta]= Displayhome(k)
21 k=round(k);
22 if k<2
23     theta=[0;0;0;0;0;0];
24 else
25     data=load('thetasJH3luenbergerobserver0_2k2.mat');
26     theta=data.ans(2:7,k-1);
27 end
28 end

```

### .3.2 Simulink: Luenberger Observer

```

1 %%
2     fieldBody=inv(Hframes(id(1:3),1:3))*magneticField;
3     forceBody=inv(Hframes(id(1:3),1:3))
4         *([0;-0.00055*9.81;0]);%+magForce(:,i));
5     torqueBody=cross([0;0;1]*-tip_dipole,fieldBody);
6     %Correct output for Simulink
7     wrench = [0;0;0;torqueBody];
8     u=zeros(6,1);
9     u(4:5)=(wrenchplus(4:5)-wrench(4:5))./0.1;
10
11     %New loop for Hframestack with changing segment number
12     H0=eye(4);

```

```

12 Hframestack_r(1:4,:) = H0;
13 for i = 1:seg_num
14     Hframestack_r((i+1)*4-3:(i+1)*4,:) = H0;
15     Hframestack_r((i+1)*4-3:(i+1)*4-1,4) = H0(1:3,1:3)
        * [0;0;i*tot_len/seg_num];
16 end
17 Hframestack = Hframestack_r;
18 %%
19 %Initial configuration
20 twistTot = eye(4);
21 Twiststack = zeros(4*size(thetas,2),4);
22
23 for i = 1:size(theta,2)
24     Hprev = Hframestack_r((1:4)+4*(i-1),:);
25     Hcurr = Hframestack_r((1:4)+4*(i),:);
26     thetacurr = [1;0;0]*theta(1,i) + [0;1;0]*theta(2,i);
27     Twiststack((1:4)+4*(i-1),:) = [[skew(thetacurr),-
        cross(thetacurr,Hprev(1:3,4))]; zeros(1,4)];
28     twistTot = twistTot*expm(Twiststack((1:4)+4*(i-1),:))
        ;
29     Hframestack((1:4)+4*(i),:) = twistTot*Hcurr;
30 end
31
32 pos = 0;
33 Jacobian = get_jacobian(thetas, Twiststack, Hframestack_r)
        ;
34 for j = 1:3
35     for k = 1:2
36         pos = pos + 1;
37         thetaplus = thetas;

```



```

38     thetaminus=thetas ;
39     deltaj=zeros (6,6) ;
40     % Numerical Differentiation 1st Parameter
41     thetaplus (k,j)=thetaplus (k,j)+constant ;
42     %Update reference frame
43     H0=eye (4) ;
44     Hframestack_r (1:4 ,:)=H0;
45     for i=1:seg_num
46         Hframestack_r ((i+1)*4-3:(i+1)*4 ,:)=H0;
47         Hframestack_r ((i+1)*4-3:(i+1)*4-1,4)=H0
48             (1:3 ,1:3) * [0;0;i*tot_len/seg_num] ;
49     end
50     Hframestack=Hframestack_r ;
51     %%
52     %Initial configuration 1st phase
53     twistTot=eye (4) ;
54     Twiststack=zeros (4* size (thetaplus ,2) ,4) ;
55     for i=1: size (theta ,2)
56         Hprev=Hframestack_r ((1:4)+4*(i-1) ,:);
57         Hcurr=Hframestack_r ((1:4)+4*(i) ,:);
58         thetacurr=[1;0;0]* thetaplus (1,i) +[0;1;0]*
59             thetaplus (2,i) ;
60         Twiststack ((1:4)+4*(i-1) ,:)= [[ skew (
61             thetacurr) ,- cross (thetacurr ,Hprev (1:3 ,4)
62             ) ]; zeros (1,4) ];
63         twistTot=twistTot*expm ( Twiststack ((1:4)+4*(
64             i-1) ,:));
65         Hframestack ((1:4)+4*(i) ,:)=twistTot*Hcurr ;
66     end
67     %End configuration 1st phase

```

```

63     Jacobianplus=get_jacobian( thetaplus , Twiststack ,
64         Hframestack_r );
65     %Numerical Differentiation 2nd Parameter
66     thetaminus(k,j)=thetaminus(k,j)-constant;
67
68     %Update reference frame
69     H0=eye(4);
70     Hframestack_r(1:4,:)=H0;
71     for i=1:seg_num
72         Hframestack_r((i+1)*4-3:(i+1)*4,:)=H0;
73         Hframestack_r((i+1)*4-3:(i+1)*4-1,4)=H0
74         (1:3,1:3)*[0;0;i*tot_len/seg_num];
75     end
76     Hframestack=Hframestack_r;
77     %%
78     %Initial configuration 2nd phase
79     twistTot=eye(4);
80     Twiststack=zeros(4*size(thetaminus,2),4);
81
82     for i=1:size(theta,2)
83         Hprev=Hframestack_r((1:4)+4*(i-1),:);
84         Hcurr=Hframestack_r((1:4)+4*(i),:);
85         thetacurr=[1;0;0]*thetaminus(1,i)+[0;1;0]*
86             thetaminus(2,i);
87         Twiststack((1:4)+4*(i-1),:)=[[skew(
88             thetacurr),-cross(thetacurr,Hprev(1:3,4)
89             )];zeros(1,4)];
90         twistTot=twistTot*expm(Twiststack((1:4)+4*(
91             i-1),:));
92         Hframestack((1:4)+4*(i),:)=twistTot*Hcurr;

```

```

87         end
88     %End configuration 2nd phase
89
90     Jacobianminus=get_jacobian(thetaminus ,
91         Twiststack ,Hframestack_r);
92     %Numerical Differentiation Formula
93     deltaj=(Jacobianplus-Jacobianminus)/(2*
94         constant);
95     %Gamma matrix evaluation
96     for z=1:6
97         gamma(z, pos)=gamma(z, pos)-deltaj(3,z)*9.81+
98             deltaj(4,z)*u(4)+deltaj(5,z)*u(5);
99     end
100     end
101     %Output
102     thetaKplus= thetaK + 0.5.*inv(diag_c-gamma)*(Jacobian'*
103         u);
104 end
105 end

```

# Bibliography

Momen Abayazid, Pedro Moreira, Navid Shahriari, Sachin Patil, Ron Alterovitz, and Sarthak Misra. Ultrasound-guided three-dimensional needle steering in biological tissue with curved surfaces. *Medical Engineering and Physics*, 37(1):145–150, 2015.

Awaz Ali, Dick H Plettenburg, and Paul Breedveld. Steerable catheters in cardiology: Classifying steerability and assessing future challenges. *IEEE Transactions on Biomedical Engineering*, 63(4):679–693, 2016.

American Heart Association. Cardiac catheterization. <http://www.heart.org>, 2017.

Jessica Burgner-Kahrs, D Caleb Rucker, and Howie Choset. Continuum robots for medical applications: A survey. *IEEE Transactions on Robotics*, 31(6):1261–1280, 2015.

David B Camarillo, Kevin E Loewke, Christopher R Carlson, and J Kenneth Salisbury. Vision based 3-d shape sensing of flexible manipulators. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 2940–2947. IEEE, 2008a.

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*, 24(6):1262–1273, 2008b.

Robotic Surgery Center. What is robotic surgery. <https://med.nyu.edu/robotic-surgery/physicians/what-robotic-surgery>, 2017.

M. K. Chmarra, C. A. Grimbergen, and J. Dankelman. Systems for tracking minimally invasive surgical instruments. *Minimally Invasive Therapy & Allied Technologies*, 16(6):328–340, 2007. doi: 10.1080/13645700701702135. URL <https://doi.org/10.1080/13645700701702135>.

- Roger E Goldman, Andrea Bajo, and Nabil Simaan. Compliant motion control for multisegment continuum robots with actuation force sensing. *IEEE Transactions on Robotics*, 30(4):890–902, 2014.
- Paula Gomes. Surgical robotics: Reviewing the past, analysing the present, imagining the future. *Robotics and Computer-Integrated Manufacturing*, 27(2):261–266, 2011.
- Ian A Gravagne and Ian D Walker. Manipulability, force, and compliance analysis for planar continuum manipulators. *IEEE Transactions on Robotics and Automation*, 18(3):263–273, 2002.
- Tipakorn Greigarn and M Cenk Çavuşoğlu. Task-space motion planning of mri-actuated catheters for catheter ablation of atrial fibrillation. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 3476–3482. IEEE, 2014.
- H Hajimolahoseini, R Amirfattahi, and S Khorshidi. Real-time pose estimation and tracking of rigid objects in 3d space using extended kalman filter. In *Electrical Engineering (ICEE), 2014 22nd Iranian Conference on*, pages 1545–1549. IEEE, 2014.
- Mahta Khoshnam, Mahdi Azizian, and Rajni V Patel. Modeling of a steerable catheter based on beam theory. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 4681–4686. IEEE, 2012.
- V Kumar. *The theorems of euler and chasles*, 2000.
- Vi NT Le, Nghia H Nguyen, Kamal Alameh, Rukshen Weerasooriya, and Peter Pratten. Accurate modeling and positioning of a magnetically controlled catheter tip. *Medical physics*, 43(2):650–663, 2016.
- Taoming Liu, Nate Lombard Poirot, Dominique Franson, Nicole Seiberlich, Mark A Griswold, and M Cenk Çavuşoğlu. Modeling and validation of the three-dimensional deflection of an mri-compatible magnetically actuated steerable catheter. *IEEE Transactions on Biomedical Engineering*, 63(10):2142–2154, 2016.

- Mohsen Mahvash and Pierre E Dupont. Stiffness control of surgical continuum manipulators. *IEEE Transactions on Robotics*, 27(2):334–345, 2011.
- Richard M Murray, Zexiang Li, S Shankar Sastry, and S Shankara Sastry. *A mathematical introduction to robotic manipulation*. CRC press, 1994.
- P.I.Corke. *Robotics, Vision and Control*. ISBN 978-3-319-54413-7, 2017.
- Christopher Piorkowski, Charlotte Eitel, Sascha Rolf, Kerstin Bode, Philipp Sommer, Thomas Gaspar, Simon Kircher, Ulrike Wetzel, Abdul Shokor Parwani, Leif-Hendrik Boldt, et al. Steerable versus nonsteerable sheath technology in atrial fibrillation ablationclinical perspective: A prospective, randomized study. *Circulation: Arrhythmia and Electrophysiology*, 4(2):157–165, 2011.
- Austin Reiter, Roger E Goldman, Andrea Bajo, Konstantinos Iliopoulos, Nabil Simaan, and Peter K Allen. A learning algorithm for visual pose estimation of continuum robots. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 2390–2396. IEEE, 2011.
- Graham Robinson and J Bruce C Davies. Continuum robots-a state of the art. In *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, volume 4, pages 2849–2854. IEEE, 1999.
- Roy J Roesthuis and Sarthak Misra. Steering of multisegment continuum manipulators using rigid-link modeling and fbg-based shape sensing. *IEEE transactions on robotics*, 32(2):372–382, 2016.
- Jakub Sikorski, Imro Dawson, Alper Denasi, Edsko EG Hekman, and Sarthak Misra. Introducing bigmag, a novel system for 3d magnetic actuation of flexible surgical manipulators. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 3594–3599. IEEE, 2017.
- Liana Ellen Taylor, Midriem Mirdanies, and Roni Permana Saputra. Optimized object tracking technique using kalman filter. *Mechatronics, Electrical Power & Vehicular Technology*, 7(1), 2016.

- Ravi Pratap Tripathi, Soumyabrata Ghosh, and JO Chandle. Tracking of object using optimal adaptive kalman filter. In *Engineering and Technology (ICETECH), 2016 IEEE International Conference on*, pages 1128–1131. IEEE, 2016.
- Valentina Vitiello, Su-Lin Lee, Thomas P Cundy, and Guang-Zhong Yang. Emerging robotic platforms for minimally invasive surgery. *IEEE reviews in biomedical engineering*, 6:111–126, 2013.
- Gustaaf J Vrooijink, Momen Abayazid, and Sarthak Misra. Real-time three-dimensional flexible needle tracking using two-dimensional ultrasound. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 1688–1693. IEEE, 2013.
- Shishan Yang and Marcus Baum. Extended kalman filter for extended object tracking. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pages 4386–4390. IEEE, 2017.

# Acknowledgments

I would like to start by thanking Professor Elena De Momi and Professor Sarthak Misra, who gave me the possibility to live this experience. With them all the guys of the Surgical Robotics Lab, starting from Jakub Sikorski for his fundamental help and patient support during the development of my project. A sincere gratitude to my family, my girlfriend and all my friends from Bologna, Milan and Enschede who shared with me these years. Thanks, finally, to the Polytechnic of Milan and to all the Professors for bringing me up to here.