POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Free-Floating Robot Inertial Parameter Identification Towards in Orbit Servicing

Tesi di Laurea Magistrale in
Space Engineering - Ingegneria Spaziale

Author: **Michele Spirolazzi**

Student ID: 964707
Advisor: Prof. Michèle Roberta Lavagna
Co-advisor: Dr. Roberto Lampariello (DLR)
Academic Year: 2021-2022

Abstract

This master thesis work focuses on the inertial parameter identification of a space robot by exploiting an object of known inertial properties placed at the end-effector of the robotic arm and angular momentum conservation.

On-orbit servicing tasks are becoming every day more crucial due to the exponential growth experienced by the space sector in the recent years. The accurate knowledge of the inertial parameters of a servicing platform is fundamental to accomplish complex missions which require high precision.

In this context the method developed in this research work, which was carried out at the DLR's Institute of Robotics and Mechatronics in Oberpfaffenhofen (Germany), will extend the already well-covered topic of space manipulators in-orbit identification with algorithms tailored for platforms that do not have reaction wheels on board (e.g. ISS Astrobees).

Besides validating the method with offline simulations, tests were performed for a free-floating robot with a 7 degrees of freedom arm on DLR's OOS-SIM experimental facility, providing an onground validation in a close to representative environment. The identification results show that the full dynamic model of the free-floating robot can be identified with the known load at its end-effector, giving comparable results to those in the literature, ready to be used in a model-based control framework.

Keywords: On-Orbit Servicing, Space Robot, Free-Floating Dynamics, Identification, Angular Momentum Conservation

POLITECNICO
MILANO 1863

DLR
Deutsches Zentrum
für Luft- und Raumfahrt

Abstract in lingua italiana

Questo lavoro di tesi magistrale si occupa dell'identificazione dei parametri inerziali di un satellite equipaggiato con un braccio robotico, sfruttando un oggetto di proprietà inerziali note posto all'end-effector e la conservazione del momento angolare.
Le operazioni da portare a termine in orbita stanno diventando ogni giorno più cruciali nella crescita esponenziale subita dal mercato spaziale negli ultimi anni. La conoscenza accurata dei parametri inerziali di una piattaforma adibita a queste operazioni è fondamentale per completare compiti difficili che richiedono alta precisione.
In questo contesto il metodo sviluppato per questo lavoro di tesi, svolto all'Istituto di Robotica e Meccatronica del DLR di Oberpfaffenhofen (Germania), vuole estendere l'argomento già ben trattato dell'identificazione in orbita di satelliti equipaggiati con bracci robotici con algoritmi appositi per piattaforme sprovviste di ruote di reazione a bordo (come ad esempio gli Astrobee a bordo della ISS).
Oltre a validare il nuovo metodo con simulazioni offline, sono stati fatti dei test su un robot flottante a sette gradi di libertà sulla piattaforma OOS-SIM del DLR, fornendo in questo modo una validazione a terra in un ambiente molto vicino a quello reale. I risultati mostrano che l'intero modello dinamico del robot flottante può essere identificato usando l'oggetto noto all'end-effector, dando risultati comparabili a quelli in letteratura, e pronti per essere usati in uno schema di controllo basato sul modello dinamico.

Parole chiave: Servizio in Orbita, Robot Spaziale, Dinamica Flottante, Identificazione, Conservazione del Momento Angolare

Acknowledgements

Proverò a rendere questi ringraziamenti poco scontati, forse compito più arduo della scrittura di tutta la mia tesi di laurea magistrale.

Alla mia famiglia (partiamo bene), a mia madre, a mio padre, punti di riferimento della mia vita e supporto costante. E' grazie a loro se alla fine ho deciso di mollare la cucina italiana per nove mesi per passare alla raffinata cucina della mensa DLR ("Ancora patate come contorno?").

A mia madre, così simile a me nel carattere da litigare, a volte, anche solo per salutarsi al telefono, ma sempre disponibile ad ascoltare i miei problemi, quando a quel santo telefono risponde.

A mio padre, l'AirBoss (grazie Alitalia per i viaggi al 5% del costo del biglietto), per la passione per l'aeronautica e l'ingegneria, non sono diventato un pilota solo per il mio stomaco debole.

Alle mie cugine (tutte e quattro), agli zii, ai nonni, per essere stati sempre presenti.

A Benedetta, migliore amica e psicologa nei momenti più bui (ma studi medicina, forse hai sbagliato), nonostante i tuoi consigli siano essenzialmente fatti per non essere da me seguiti ("Tu cosa ne pensi Giòrgio?").

A Guglielmo, che mi ha seguito a Milano per studiare ed è sempre stato lì quando ne avevo bisogno ("Non è che mi accompagneresti a Malpensa ho il volo tra due ore").

Agli inquilini di via Murat, a Giacomo, a Giuseppe, a Jacopo, a Davide, per avermi accolto e adottato in casa soprattutto nei primi anni, ai giochi di carte che sono poi stati esportati nelle pause tra le lezioni (l'annale diatriba tra giochissimo/gatto).

Alle persone che sono arrivato a chiamare amici nei cinque anni di università, grazie per aver reso lo studio quasi sopportabile. Alle serate per Milano, alle feste in via Alcuino, ai quindici posti occupati ogni mattina in mezzo all'aula.

Ai lord di Skyward e alle loro perle, ad Albi, Tommi, Fede, Paul, Matteo, Giamma, Restu, Fabio, Staffo.

Al gruppo CARBO, ormai famiglia estesa, a Giorgia, Silvia, Pietro, Greta, Restu, Fabio, Staffo, Berto.

A Giorgia, praticamente mia sorella acquisita, non sceglierei nessun altro al mio fianco per cucinare e per farmi un volo sopra Milano, tanto pilota lei ("Acchittiamo una carbonara per dieci?").

A Fabio, amico vero dal profondo sud (bellissima città Matera), sei sempre stato un punto di riferimento nel mondo accademico per rigorosità e dedizione allo studio. Scegliere una frase che ti rappresenti è stata tosta, quindi ne ho selezionate due ("Sapete quante sono le probabilità che in un gruppo di trenta persone almeno due siano nate lo stesso giorno?")- ("Ma v********o Fabio", "Come scusi?").

A Silvia, essendo entrambi abbastanza diffidenti del prossimo ci è voluto del tempo per far ingranare il rapporto, ma adesso per capirsi basta un'occhiata ("Potevi dirmelo che eri da solo in cucina, avrei evitato di intrattenere la conversazione con loro").

A Staffo, fotocopiati nel carattere non solo da essere diventati grandi amici, ma soprattutto da essere stati scambiati il più delle volte per fratelli. Alle nostre chiamate la sera a programmare il futuro ("Finisco questo e andiamo in lab").

A Restu, agli interessi in comune, dall'NBA alle serate su COD durante il lockdown. La tua dedica è breve, come si confà al personaggio ("Me lo sento questa ce la portiamo a casa").

A Berto, poco presente fisicamente ma sempre nei nostri cuori (sembra un epitaffio ma mi faceva ridere). Anche per te la dedica è breve, tu e Restu avete molte cose in comune ("Mi manderesti gli appunti di (*inserire materia a piacere*)?").

A Greta, per essere diventata rapidamente e inaspettatamente una delle persone che mi conoscono meglio di chiunque altro, come se parlassimo la stessa madre lingua ("Ehi si vede la Luna da qui").

Alla Professoressa Lavagna, che mi ha permesso di cogliere questa opportunità e che mi ha fatto appassionare alla materia.

Al Dottor Lampariello, per la sua disponibilità nei mesi al DLR e la sua guida durante il lavoro di tesi.

Grazie di cuore a tutti, non sarei diventato la persona che sono senza l'apporto dato da ognuno di voi ("Come concludere dei ringraziamenti spero non banali con la più comune delle frasi di circostanza" - Corso da due pagine).

Contents

Abstract **i**

Abstract in lingua italiana **ii**

Acknowledgements **iii**

Contents **v**

List of Figures **viii**

List of Tables **x**

Nomenclature **xi**

1 Introduction **1**

1.1 Motivation . . . 1

1.2 Thesis Contribution . . . 3

1.3 Thesis Outline . . . 4

2 State Of The Art **6**

2.1 Inertial Parameters Identification Methods . . . 6

2.1.1 Inverse Dynamic Identification With Least-Squares . . . 6

2.1.2 Instrumental Variable And Maximum Likelihood . . . 8

2.1.3 Output Error And Input Error . . . 9

2.1.4 Kalman Filters and Neural Networks . . . 9

2.2 Parametric Identification Of Free-Floating Robots . . . 11

3 Mathematical Formulation Of The Problem **19**

3.1 Kinematics Of A Space Manipulator . . . 19

3.1.1 Geometrical Description Of The Manipulator . . . 19

3.1.2 Positions And Velocities . . . 20

3.2 Free Floating Dynamics . . . 21
3.3 Linear And Angular Momenta . . . 22
3.4 Regressor Form Of The Angular Momentum Equations . . . 23
3.4.1 Formulation Independent From Base Body Position and Velocity . . 26
3.4.2 Simplification For 2D Systems . . . 29
3.4.3 Symbolic Implementation Of The Angular Momentum Equations In Regressor Form . . . 30
3.5 Exciting Trajectories Optimization . . . 32
3.5.1 Collision Avoidance . . . 33

4 Preliminary Research: Baseline Simulations **35**
4.1 Simulink Model Description . . . 35
4.1.1 Noise Models . . . 36
4.1.2 Joints Angles & Angular Rates . . . 37
4.1.3 Gyroscope . . . 37
4.1.4 Random Numbers Generation in Simulink . . . 38
4.2 Baseline Simulations Results: Case Study 1 . . . 38
4.3 Reaction Wheels Accumulated Momentum Case . . . 41
4.4 Baseline Simulations Results: Case Study 2 . . . 42

5 Novel Identification Framework And Simulation Results **46**
5.1 Methodology Description . . . 46
5.1.1 Considerations On The Symbolic Implementation . . . 49
5.2 Methodology Validation . . . 50
5.3 Case Study N1: Free-Floating Seven Degrees Of Freedom Robotic Arm . . 52
5.3.1 Identification Results . . . 57
5.3.2 Validation Tests . . . 60

6 Simulations On DLR Experimental Facility **63**
6.1 OOS-SIM Description . . . 63
6.2 OOS-SIM: IMU Calibration . . . 64
6.3 Required Measurements For The Identification Experiments . . . 66
6.3.1 Signals Processing . . . 67
6.4 Trajectory Optimization For OOS-SIM Experiments . . . 68
6.5 Experimental Results . . . 68
6.5.1 Optimized Trajectory For OOS-SIM . . . 70
6.5.2 Sensors Gathered Data . . . 73
6.5.3 Identification Results . . . 76

7 Conclusions And Future Developments 79

7.1 Conclusions 79

7.2 Future Developments 80

7.2.1 Optimization of the Computational Times 80

7.2.2 Sensitivity Analyses 81

7.2.3 Experimental Tests 81

7.2.4 Model Complexity 82

Bibliography 83

A Appendix A 87

A.1 Free-Floating 7 DOF Robotic Arm Inertial Parameter Clusters 87

List of Figures

1.1 Artist Render of a MEV Unit Attached to a Client Satellite, (*Credits*: Northrop Grumman) . . . 1
1.2 Compliant Assistance and Exploration SpAce Robot (CAESAR), (*Credits*: DLR) . . . 2

3.1 Modified DH Parameters . . . 20
3.2 Space Robot Representation . . . 21
3.3 Collision/No Collision Visualization . . . 34

4.1 Simulink Model . . . 35
4.2 Case Study 1: Optimized Joints Trajectories . . . 38
4.3 Case Study 1: Maximum Relative Error . . . 39
4.4 Case Study 1: π_1 Convergence Analysis . . . 40
4.5 Case Study 2: Optimized Joints Trajectories . . . 43
4.6 Case Study 2: π_4 Convergence Analysis . . . 43
4.7 Case Study 2: Maximum Relative Error . . . 44
4.8 Case Study 2: Identification with Moving Window . . . 44

5.1 Render of the Astrobee Free-Flying Robot (*Credits:* NASA) . . . 46
5.2 Case Study N1: Angular Momentum Norm . . . 54
5.3 Case Study N1: Joints Angles . . . 55
5.4 Case Study N1: Joints Angular Rates . . . 56
5.5 Case Study N1: Base Body Angular Velocity . . . 56
5.6 Case Study N1: Base Body Attitude . . . 57
5.7 Case Study N1: π_1 Convergence Analysis . . . 58
5.8 Case Study N1: π_i Components Identified With The Highest Relative Errors In Each Run . . . 58
5.9 Case Study N1: Base Body Angular Velocity Error . . . 60
5.10 Validation: Joints Angles & Angular Rates . . . 61
5.11 Validation: Base Body Attitude & Angular Velocity . . . 61

5.12 Validation: Distribution of the Absolute Value of the Maximum Error for the Base Body Angular Velocity 62

6.1 OOS-SIM in DLR Institute of Robotics and Mechatronics, (*Credits*: DLR) 63

6.2 Uncalibrated vs Calibrated Accelerometers Measurements 65

6.3 Uncalibrated vs Calibrated Gyroscopes Measurements 65

6.4 Joints Angles For Experimental Test 71

6.5 Joints Angular Rates For Experimental Test 71

6.6 Base Body Angular Velocity For Experimental Test 72

6.7 Base Body RPY Angles For Experimental Test 72

6.8 Joints Angles Registered On Board (Filtered) 73

6.9 Joints Angular Rates Computed (Filtered) 73

6.10 Base Body Angular Velocity Registered On Board (Filtered) 74

6.11 Base Body Body Angular Velocity In The Inertial Reference Frame (Filtered) 74

6.12 Base Body Attitude Computed (Filtered) 75

6.13 Experimental total angular momentum showing error from expected zero value (left y-axis), compared with the base body angular momentum (right y-axis) to show the magnitude of the error. The two y-axis have different scales to make the chart readable. 76

6.14 OOS-SIM: Comparison between the Base Body Experimental Angular Velocity and the Reconstructed Angular Velocity 78

6.15 OOS-SIM: Absolute Error Between Experimental Angular Velocity And Reconstructed Angular Velocity 78

List of Tables

4.1 Case Study 1: Optimized Fourier Coefficients 39
4.2 Case Study 1: Identification Results 40
4.3 Case Study 2: Optimized Fourier Coefficients 42
4.4 Case Study 2: Identification Results 45

5.1 Case Study N1: 7 DoF Simulation Model Parameters 52
5.2 Case Study N1 Trajectory: Optimized Fourier Coefficients 54
5.3 Case Study N1: Identification Results 59

6.1 Calibration Parameters 64
6.2 Regressor Matrix Required Measurements 66
6.3 Experimental Model Parameters 69
6.4 Experimental Trajectory: Optimized Fourier Coefficients 70
6.5 OOS-SIM: Identification Results 76

Nomenclature

Symbols

α	Roll Angle
β	Pitch Angle
$\boldsymbol{\omega}$	Angular Velocity
$\boldsymbol{\theta}$	Angular rotation
$\bar{\boldsymbol{\pi}}$	Mean Identified Param. Vec.
$\boldsymbol{\pi}$	Inertial Parameters Vector
$\boldsymbol{\Sigma}$	Singular Values Matrix
γ	Yaw Angle
$\mathcal{L}$	Total Momentum
σ	Standard Deviation
$\dot{\mathbf{q}}$	Joint Angular Rate
$\mathbf{a}_i$	Link i center of mass position in Joint i reference frame
$\mathbf{b}_i$	Joint $i+1$ position in Joint i reference frame from Link i center of mass
$\mathbf{d}$	Quaternion
$\mathbf{H}$	Angular Momentum
$\mathbf{M}$	Composite Inertia Matrix
$\mathbf{P}$	Linear Momentum
$\mathbf{q}$	Joint Angle
$\mathbf{R}$	Rotation Matrix
$\mathbf{r}$	Joint Position in Inertial reference frame
$\mathbf{U}$	Left-Singular Vector
$\mathbf{V}$	Right-Singular Vector
$\mathbf{v}$	Velocity
$\mathbf{Y}$	Regressor Matrix
$\mathbf{z}$	Rotation Axis
$\mathbf{E}_{3\times3}$	Identity Matrix
$\mathbf{f}$	Frequency
$\mathbf{I}$	Inertia
M	Mass
${}^{i-1}\mathbf{T}_i$	Homogeneous Tranformation Matrix
$\mathbf{c}$	Center of Mass Position in Inertial reference frame

1 | Introduction

1.1. Motivation

The document by NASA in [1] presents a great overview of the operations to be performed in orbit in the upcoming years: the possibilities range from refuelling the satellites orbiting the Earth in Low Earth Orbits (LEO), Medium Earth Orbits (MEO) or Geostationary Earth Orbits (GEO), the mitigation of the rising problem of space debris by removing them and thus lowering the collision probabilities, and the assembly of large structures in space (e.g. a space station at the lagrangian point between the Earth and the Moon).

A great example is embodied by the Northrop Grumman mission MEV (Mission Extension Vehicle, Figure 1.1) which on the 25th of February 2020 successfully docked with the Intelsat satellite IS-901 to correct its orbit (acting as an external engine) and augmented the life of the spacecraft of an additional five years [2].

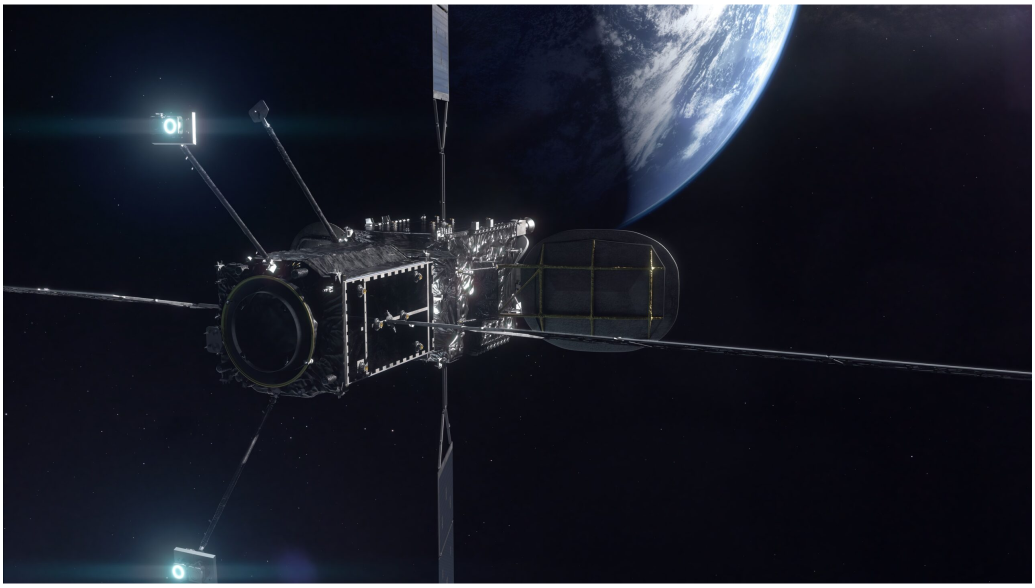


Figure 1.1: Artist Render of a MEV Unit Attached to a Client Satellite, (*Credits*: Northrop Grumman)

The possibility to repair and act mechanically on the existing satellites and not only on

the software side would allow to greatly reduce costs as the need to replace them would practically vanish.
Moreover, following the same reasoning, many of the activities carried out nowadays by astronauts on the International Space Station (ISS) could be performed by specific platforms, both tele-operated and autonomous (following the trend of increasing development of artificial intelligence applications in the form of neural networks), by the means of robotic arms such as the newly designed CAESAR [3] from DLR (Figure 1.2). This would increase the safety of such operations and also their number since machines are not subjected to the same restrictions that humans face in space.

Figure 1.2: Compliant Assistance and Exploration SpAce Robot (CAESAR), (*Credits*: DLR)

The use of robots in space is slightly different from the on Earth equivalent due to the absence of gravity. There are typically two main categories in which a satellite equipped with a robotic arm can be employed, namely in free-flying mode or free-floating mode. The first one foresees the active control of both the center of mass position of the satellite and its attitude, by exploiting reaction control systems on board such as thrusters for the translational dynamics and reaction wheels or control moment gyroscopes for the rotational ones, in order to compensate for the platform reaction to the arm movement during operations. This strategy in the long run brings to consumption of thrusters fuel, reducing in this way the length of the mission, which on the other hand is not the case for a free-floating mode. In fact, for this specific mode all reaction control systems are kept switched off during the arm motion, leaving the base body free to tumble by conserving the total momentum of the system since there are nor external forces nor moments acting on it. Of course the control algorithms on board have to deal with the robotic arm floating base to permit to its end-effector to fulfill the assigned task, coping with problems that do not exist on ground such as dynamic singularities [4], [5].

This framework requires the precise knowledge of the inertial parameters of the whole platform, to be given as input to the dynamics driving the control loop, as for example in [6], [7], [8], [9].
For operations such as in orbit capture of a cooperative or uncooperative target (as would be the case for a debris removal scenario) the requirements on the grasping process are in fact tight and can be achieved only by exploiting an on board model of the system of high fidelity.
For this reason the parametric identification of robotic space manipulators is a research topic that has been studied in the past thirty years and it is still in development [10], as the recent ESA founded studies *OBSIdian* also confirm [11].

1.2. Thesis Contribution

This master thesis work focuses on the inertial parameter identification of a spacecraft equipped with a robotic arm (a space robot). The conservation of angular momentum is exploited with a novel approach, which, in contrast to the most recent works (see [12], [13]), does not require the presence of reaction wheels on board the spacecraft. The lack of reaction wheels on board may in fact well be the case for a robotic satellite operating in outer space, given that these are very bulky and energy-consuming. It is also possibly going to be the case for future robotic free-flyers operating on board of a space station (see for example NASA's ASTROBEE system [14]).
As reported by Ayusawa, Venture and Nakamura in [15] the total mass of a free-floating system, for which it is assumed that no external forces act on the system, is not identifiable: in the literature this problem is implicitly solved by assuming as known the inertial properties of on-board reaction wheels, given that their angular momentum is considered known and not null. In this work, instead, an object of known inertial properties (the load), positioned at the end-effector, is employed to solve the identification problem, satisfying in this way the conditions for the system identifiability. This strategy also has the operational advantage of allowing to assume a null total angular momentum during the

robotic arm exciting manoeuvre.
Moreover, the present research treats a robotic arm with a seven degrees of freedom (DoF) geometry, whereas the aforementioned texts dealt with arms of maximum three DoF, thus providing an extension to the currently available data. It was decided to set as a goal a seven DoF manipulator to be able to exploit DLR's On-Orbit Servicing - Simulator (OOS-SIM), described in Chapter 6. This is a great addition to the topic given the difficulty of reproducing the in orbit environment (in which these manoeuvres would take place) and in particular the absence of gravity on the ground, coupled with the complexity of the geometry of the considered robotic arm.

1.3. Thesis Outline

The structure of this thesis is divided into seven main chapters. After Chapter 1, which gives a broad overview of the topics treated throughout the whole text and the main contribution of this work, Chapter 2 outlines the most common practices in the field of fixed robotic arms inertial parameters identification and then focuses on the same subject but applied to free-floating space robots.
Chapter 3, instead, presents all the theoretical aspects covered by the problem under study: starting from a brief description of manipulator kinematics, free-floating dynamics are then introduced and in particular the expressions for the conservation of linear and angular momentum for this kind of systems. Afterwards, the regressor form of the angular momentum is treated in two different formulations, and the steps to pass from one framework, which requires the knowledge of the base body position and velocity referred to an inertial frame, to the other, for which these quantities are not necessary, are described. Lastly, this chapter treats the mathematical formulation of the optimization problem exploited to generate exciting trajectories for the robotic arm to fulfill the identification task.
Chapter 4 starts with a description of the Simulink© model employed for the simulation of the free-floating dynamics of the system under study in response to the selected exciting trajectory performed by the manipulator. The developed model includes also blocks to reproduce noisy measurements from the exploited sensors, namely gyroscopes and joints motors encoders, to make the simulation environment closer to reality. It then proceeds to show the statistical outputs of simulations trying to reproduce the identification results for a space manipulator in presence of non zero angular momentum obtained by Christidi et al. in [12]. This paper is the starting point for this work and its results are thus treated as a baseline for further experiments. It then introduces a possible source for the non zero angular momentum such as reactions wheels and simulates the system under this new

condition, starting from zero base body angular velocity and zero joint angular rates.
Chapter 5 is the core chapter of this thesis. Here the novel framework for the parametric identification of inertial parameters for a space robot is carefully explained and simulations with a free-floating seven DoF robotic arm are performed and their results commented. Moreover, a validation procedure is presented and outlined, and successively used to check the consistency of the simulations results.
Chapter 6 gives a brief introduction to the experimental facility in DLR's Oberpfaffenhofen site: the OOS-SIM. It also presents the changes which had to be done on the model used in simulations in order to perform the experiments on the platform, with particular attention to the additional mechanical constraints of the system. It finally provides the results of the new framework tested on the experimental set-up with a hardware-in-the-loop (HIL) approach, which is the great contribution of this thesis, since it is difficult to reproduce the orbital environment and the underlying dynamics on Earth.
Finally Chapter 7 concludes this thesis by commenting the results obtained, making comparisons with the existing methodologies, and proposing possible topics to extend the present work.

2 | State Of The Art

2.1. Inertial Parameters Identification Methods

Since this work is focused on the inertial parameters identification of a robotic arm (and its base body given the free-floating dynamics), a brief review of the state-of-the-art techniques to identify the said values is treated in this section, starting from the identification methods themselves applied to fixed base manipulators, for which examples can be found in [16], [17]. The research is then extended in the next section to methodologies applied to the specific problem under study, namely the field of free-floating space manipulators. The work followed in this first part of the review is the one by Leboutet et al. in [18], which gives a great overview of the topic.

2.1.1. Inverse Dynamic Identification With Least-Squares

The most common approach to identify the inertial parameters of a rigid robot is the Inverse Dynamic Identification Model with ordinary Least-Squares estimation and its variants (namely Weighted Least-Squares or Total Least-Squares), because they are efficient and easy to code. These methods have intrinsic downsides related to being sensitive to data noise and to being greatly influenced by the conditioning of the trajectory of the robot. The framework in which this method is applied requires to start from the inverse dynamic model of the manipulator, given by Equation 2.1

$$\mathbf{M}\left(\boldsymbol{\pi},\mathbf{q}\right)\ddot{\mathbf{q}}+\mathbf{C}\left(\boldsymbol{\pi},\mathbf{q},\dot{\mathbf{q}}\right)\dot{\mathbf{q}}+\mathbf{g}\left(\boldsymbol{\pi},\mathbf{q}\right)+\boldsymbol{\zeta}\left(\boldsymbol{\pi},\dot{\mathbf{q}}\right)=\boldsymbol{\tau_{idm}} \tag{2.1}$$

which relates the joint space quantities $\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}}$ to the the generalized forces $\boldsymbol{\tau_{idm}}$ exerted on the system. The equation is dependent on the quantity $\boldsymbol{\pi}$, which concatenates all the inertial parameters of the links (with $i = 1,...,n$) composing the system (Equation 2.2)

$$\pi_i = [I_{xx_i}, I_{xy_i}, I_{xz_i}, I_{yy_i}, I_{yz_i}, I_{zz_i}, MX_i, MY_i, MZ_i, M_i, Ia_i, Fv_i, Fc_i]^T \tag{2.2}$$

This formulation of the problem has the advantage of being linear in $\boldsymbol{\pi}$ and can thus be re-written as in Equation 2.3:

$$\boldsymbol{\tau_{idm}} = \mathbf{Y}_{\pi}\left(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}\right)\boldsymbol{\pi} \tag{2.3}$$

The quantity $\mathbf{Y}_{\pi}$ is typically known as the model regressor. If this matrix is filled with samples taken from a trajectory performed by the arm as well as measurements of the generalized forces, the equation given above becomes an over-determined system of linear equations in $\boldsymbol{\pi}$, and a unique solution can be found given that matrix $\mathbf{Y}_{\pi}$ is not rank deficient, which unfortunately is hardly ever the case.

Two types of rank deficiency can be present: data rank deficiency and structural rank deficiency of the matrix $\mathbf{Y}_{\pi}$. The first one is related to an improper collection of data, i.e. by following an identification trajectory that does not excite properly the system parameters. The structural deficiency is instead bounded to the system parameters that do not influence the dynamics (no matter the trajectory), and to parameters that act combined on it. To solve this issue it is necessary to find the linear coefficients which are underlying the columns of the regressor matrix and rearrange them consequently, leading to a set of base parameters (or minimal parameters) which is lower than the initial bunch. These parameters are called base inertial parameters because they permit to fully describe the constrained-robot dynamics with the minimal set of parameters [19].

Having rearranged the vector of inertial parameters $\boldsymbol{\pi}$ in its base version $\boldsymbol{\beta}$, Equation 2.3 becomes Equation 2.4

$$\mathbf{y}_{\boldsymbol{\tau}} = \mathbf{W}\left(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}\right)\boldsymbol{\beta} + \boldsymbol{\varepsilon} \tag{2.4}$$

where the changed symbols represent the sampled quantities (N times) and $\boldsymbol{\varepsilon}$ the sampled error.

The most common practice to solve Equation 2.4 consists of computing the weighted least-squares estimate $\hat{\boldsymbol{\beta}}_{WLS}$ and associated covariance matrix $\boldsymbol{\Sigma}_{WLS}$ in Equation 2.5

$$\begin{aligned} \hat{\boldsymbol{\beta}}_{WLS} &= \left(\mathbf{W}^T\boldsymbol{\Sigma}^{-1}\mathbf{W}\right)^{-1}\mathbf{W}^T\boldsymbol{\Sigma}^{-1}\mathbf{y}_{\boldsymbol{\tau}} \\ \boldsymbol{\Sigma}_{WLS} &= \left(\mathbf{W}^T\boldsymbol{\Sigma}^{-1}\mathbf{W}\right)^{-1} \end{aligned} \tag{2.5}$$

For what concerns the Total Least-Squares (TLS) approach, a good explanation is given in [20]. This method can be used to mitigate the effects of a noisy regressor matrix. The mathematical formulation reads the following: the TLS estimate of $\hat{\boldsymbol{\beta}}_{TLS}$ can be obtained by exploiting the singular value decomposition of the augmented matrix $\mathbf{X} = [\mathbf{W}\,\mathbf{y}_{\boldsymbol{\tau}}]$ as in Equation 2.6

$$\mathbf{X} = \begin{bmatrix} \mathbf{U}_W & \mathbf{U}_y \end{bmatrix} \begin{bmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{0}^T & S \end{bmatrix} \begin{bmatrix} \mathbf{V}_{WW} & \mathbf{V}_{Wy} \\ \mathbf{V}_{yW} & V_{yy} \end{bmatrix} \tag{2.6}$$

and Equation 2.7 gives the expression for the parameters estimate

$$\hat{\boldsymbol{\beta}}_{TLS} = -\mathbf{V}_{Wy} V_{yy}^{-1} \tag{2.7}$$

2.1.2. Instrumental Variable And Maximum Likelihood

The Instrumental Variable (IV) method applied to parameter identification consists in introducing an instrument matrix $\mathbf{Z}$ which is well correlated with the regressor matrix $\mathbf{W}$ and uncorrelated with the error $\boldsymbol{\varepsilon}$, and the corresponding identification formulation reads (Equation 2.8)

$$\mathbf{Z}^T \mathbf{y}_{\boldsymbol{\tau}} = \mathbf{Z}^T \mathbf{W}\left(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}\right) \boldsymbol{\beta} + \mathbf{Z}^T \boldsymbol{\varepsilon} \tag{2.8}$$

for which the estimate $\hat{\boldsymbol{\beta}}_{IV}$ is given by Equation 2.9

$$\hat{\boldsymbol{\beta}}_{IV} = \left(\mathbf{Z}^T \mathbf{W}\right)^{-1} \mathbf{Z}^T \mathbf{y}_{\boldsymbol{\tau}} \tag{2.9}$$

The problem with this approach is the difficulty to identify the instrument matrix $\mathbf{Z}$ with the properties reported above. An optimal guess for this matrix is another regressor matrix filled with synthetic data generated by an auxiliary model which, for this specific case, is represented by the direct dynamic formulation of the problem. This auxiliary model should be simulated with the same controller and exciting trajectories employed for the IV framework, using the last iteration estimate of the base parameters $\hat{\boldsymbol{\beta}}_{IV}^{i-1}$, setting up in this way an iterative scheme.

The Maximum Likelihood (ML) methods were specifically tailored to address the intrinsic noise of the measurements used to feed the regressor matrix of the system, which in this case are the torque and joint angles values. The mathematical formulation of the estimate $\hat{\boldsymbol{\beta}}_{ML}$ is reported in Equation 2.10

$$\hat{\boldsymbol{\beta}}_{ML} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2} \sum_{k=1}^{N} \boldsymbol{\varepsilon}^T\left(t_k\right) \left(\mathbf{G}_k \boldsymbol{\sigma}_k^2 \mathbf{G}_k^T\right)^{-2} \boldsymbol{\varepsilon}\left(t_k\right) \tag{2.10}$$

where $\boldsymbol{\sigma}_k^2$ is the diagonal variance matrix of the k-th sample and $\mathbf{G}_k$ is the Jacobian matrix of the error at the current toque sample with respect to the measurement vector which comprehends the joints positions, velocities, accelerations and the torque itself.

2.1.3. Output Error And Input Error

The Output Error (OE) method is an identification method in which the parameter estimate $\boldsymbol{\beta}$ is the one that minimizes the cost function $J(\boldsymbol{\beta})$, which is in turn the L2-squared norm of the error between the output $\mathbf{y}$ of the system under study and the output $\mathbf{y}_s$ of the model related to it (Equation 2.11)

$$J(\boldsymbol{\beta}) = ||\mathbf{y} - \mathbf{y}_s||^2 \tag{2.11}$$

where for a robot $\mathbf{y}$ and $\mathbf{y}_s$ comprehend the measured and simulated joints positions. The minimization of the function $J(\boldsymbol{\beta})$ is carried out by non linear least squares optimization algorithms. The unknown base parameters are then updated iteratively to match the output of the "real" system

$$\hat{\boldsymbol{\beta}}_{i+1} = \hat{\boldsymbol{\beta}}_i + \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_i \tag{2.12}$$

with $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_i$, which is called the innovation vector. As for the Instrumental Variable method, the direct dynamical model is used to obtain the outputs of the simulated system by exploiting the parameters estimate at the last iteration as its input (using the same controller and the same exciting trajectories).
In an Input Error (IE) method instead the quantities to be tracked are the measured input torques to the system and the simulated ones. The authors of [18] report that this method gives better results in terms of accuracy with respect to ordinary Least Squares and Output Error method, because they have the same computational cost as the OE but better convergence properties, thanks to their higher sensibility to changes in the parameters.

2.1.4. Kalman Filters and Neural Networks

For the sake of completeness a brief overview of Kalman Filters and Neural Networks applied to parameter identification is here reported, again referring to the work of Leboutet et al. in [18].

For what concerns Kalman filters they are widely used for state estimation purposes, but can also be exploited to identify model parameters. A common approach in the field of robotics, as documented by [18], is to estimate the state and the parameters simultaneously in a method which is referred to as joint method. This formulation requires to stack the state of the system and the actual base parameters estimates in the same vector, composing in this way a higher dimensional vector $\mathbf{z}_k = \left[\mathbf{x}_k^T \; \hat{\boldsymbol{\beta}}_{KF}^{k-1\,T}\right]^T$. This in turn is to be used in Equation 2.13

$$\begin{aligned} \mathbf{z}_{k+1} &= \mathbf{\Gamma}\left(\mathbf{z}_k\right) + \mathbf{v}_k \\ \mathbf{y}_k &= \mathbf{S}\mathbf{z}_k + \mathbf{w}_k \end{aligned} \tag{2.13}$$

which represent update equations where the non linear state transition function is given by Equation 2.14

$$\mathbf{\Gamma}\left(\mathbf{z}_k\right) = \begin{bmatrix} \dot{\mathbf{q}}\left(t_k\right) \\ \mathbf{q}\left(t_k\right) \\ \hat{\boldsymbol{\beta}}^k_{KF} \end{bmatrix} + \begin{bmatrix} \mathrm{DDM}\left(\mathbf{z}_k\right) \\ \dot{\mathbf{q}}\left(t_k\right) \\ \mathbf{0} \end{bmatrix} \cdot \delta t \tag{2.14}$$

DDM is the direct dynamic model, which is dependent on the current state and on the parameter estimate $\hat{\boldsymbol{\beta}}^{k-1}_{KF}$. The fact that the DDM is non linear can be dealt with several methods, resorting for example to the use of Extended Kalman Filters (EKF), which employ a first order linearization of the robot dynamics.

A possible neural network to be exploited for parameter identification purposes is the Adaline (ADAptive LInear NEuron), which uses a stochastic gradient learning to converge to an estimate given by the Inverse Dynamic Identification Model with Ordinary Least Squares. The parameter estimate at the sample time t_k is given recursively by Equation 2.15

$$\hat{\boldsymbol{\beta}}^k_{AdaNN} = \hat{\boldsymbol{\beta}}^{k-1}_{AdaNN} + \eta \mathbf{Y}^{k\,T}\left(\boldsymbol{\tau}(t_k) - \mathbf{Y}^k \hat{\boldsymbol{\beta}}^{k-1}_{AdaNN}\right) \tag{2.15}$$

where $\mathbf{Y}$ is the regressor matrix introduced in the Least Squares section and η is the learning rate. A single "neuron" can be exploited to fully identify the model, but several cycles should be employed to achieve convergence.

Another neural network architecture to be used for parameter identification is represented by the Hopfield-Tank Recurrent Neural Networks (HTRNN), which are dynamic systems composed by a set of N interconnected neurons. Each neuron i is an integrator intertwined with a non linear activation function *f*. A discrete formulation of the neuron state equation for the HTRNN given by Abe is here reported (Equation 2.16)

$$u_i^{k+1} = u_i^k + \eta \sum_{j=1}^{N} \psi_{ij} \alpha_j \tanh\left(\frac{u_j^k}{\vartheta_j}\right) + k_i \tag{2.16}$$

It can be demonstrated that this specific dynamical system is asymptotically Lyapounovstable and converges to the minimum of the energy function in Equation 2.17

$$E = -\frac{1}{2}\mathbf{s}^T \boldsymbol{\Psi} \mathbf{s} + \mathbf{s}^T \mathbf{k} \tag{2.17}$$

Given this formulation of the problem the parameter identification process can be correlated to match the L2-norm of the parameter error given in Equation 2.3 with the energy function of the HTRNN. If one selects properly the quantities introduced in Equation 2.3 this correlation can be achieved. The selection is provided in Equation 2.18.

$$\begin{cases} \boldsymbol{\Psi} & = -\mathbf{W}^T \mathbf{W} \\ \mathbf{k} & = -\mathbf{W}^T \mathbf{y}_{\boldsymbol{\tau}} \\ \mathbf{s} & = \hat{\boldsymbol{\beta}}_{HTRNN} \end{cases} \tag{2.18}$$

The neural network will then have the same number of neurons as the base parameters to be estimated and will converge to the estimate given by the Inverse Dynamic Identification Model with Ordinary Least-Squares as the Adaline Neural Network.

2.2. Parametric Identification Of Free-Floating Robots

Having treated the most common algorithms to deal with the inertial parameter identification of fixed base manipulators, the focus has to be shifted to the specific case under study, thus space manipulators in free-flying or free-floating mode. This section will present a review of the methods and techniques developed over the recent years to address this problem.

The review article by Papadopoulous, Aghili, Ma and Lampariello in [10] presents a section dedicated to the identification of satellites and tumbling objects, from which interesting articles on the subject are briefly introduced. Some of them will be shortly described in this section as well as other relevant papers on the topic.
Lampariello and Hirzinger in [21] propose a method to identify the inertial parameters of the base body of a space manipulator directly in space, by moving the origin of the inertial reference frame from Earth to a local point directly in orbit. The other bodies inertial quantities composing the system are assumed to be known. The identification framework presented in the paper consists in the comparison between measured variables of the base motion and those obtained by integration of the equations of motion of the system. The parameter identification is then treated as an optimisation problem for which the cost function κ is the sum of the differences among the simulated and measured velocity functions at specific sampling points, as in Equation 2.19, which puts this method in the

Output Error (OE) family.

$$\kappa = \sum_{i=1}^{n} \left(v_{m,i}^{11} - v_{s,i}^{11}\right)^2 + \left(v_{m,i}^{12} - v_{s,i}^{12}\right)^2 + \left(\omega_{m,i}^{1} - \omega_{s,i}^{1}\right)^2 \tag{2.19}$$

The identification results are deemed satisfactory by the authors, however the methodology is applied only to a bidimensional system and posed problems in the convergence of the solution, which was limited by the noise of the sensors and by a high mass ratio between the two bodies of the system.

Yoshida and Abiko in their work in [22] deal with the identification of inertial parameters of a free-flying space robot by exploiting the conservation of momentum, by not requiring in this way joints torques or base body acceleration measurements. The algorithms are verified with the flight data of a Japanese satellite, the ETS-VII. In principle all the inertial parameters of each body composing the system are to be identified.

As stated above the identification algorithm uses the conservation of the angular momentum, to which the accumulated momentum in the reaction wheels is explicitly added to the equation (Equation 2.20)

$$\tilde{\mathbf{H}}_b \boldsymbol{\omega}_b + \tilde{\mathbf{H}}_{bm} \dot{\boldsymbol{\phi}} + \mathbf{L}_{RW} = 0 \tag{2.20}$$

The difference between the simulated value for $\mathbf{L}_{RW}$ and the one obtained with measurements of the kinematic quantities and a guess of the inertial parameters is given in Equation 2.21

$$\Delta \mathbf{L} = \mathbf{L}_{RW} - \hat{\mathbf{L}}_{RW} = f\left(\mathbf{M} + \Delta \mathbf{M}\right) - f\left(\mathbf{M}\right) \tag{2.21}$$

where f($\mathbf{M}$) is a function of the inertial parameters. By linearising the above equation, its solution is then represented by Equation 2.22

$$\Delta \mathbf{M} = \left\{ \frac{\partial f}{\partial \mathbf{M}} \right\}^{+} \{\Delta \mathbf{L}\} \tag{2.22}$$

However, the resulting matrix is rank deficient and shall be manipulated to cope with this issue, by eliminating the parameters having minimal effect on the system dynamics.

The algorithms are validated with flight data of the ETS-VII Japanese satellite, giving good results and requiring only measurements which are typically registered on board a satellite.

Again Lampariello and Hirzinger published on the topic in [23], proposing a method to identify the parameters of the base body and of a load at the end-effector, while also covering the issue of full body identification in a theoretical manner.

Given the equations of motion in Equation 2.23

$$\mathbf{M}\dot{\bar{\mathbf{y}}}_{II} = \mathbf{C} + \boldsymbol{\Lambda}^0 + \boldsymbol{\chi} \tag{2.23}$$

these can be rewritten in regressor form by exploiting the fact that are linear in the inertial parameters

$$\boldsymbol{\Phi}\left(\bar{\mathbf{y}}_I, \bar{\mathbf{y}}_{II}, \dot{\bar{\mathbf{y}}}_{II}\right)\mathbf{p} = \boldsymbol{\Lambda}^0 \tag{2.24}$$

At this point this formulation can be solved in the least squares sense. The problem is that the vector $\boldsymbol{\Lambda}^0$ of external forces acting on the base body is nearly zero and thus gives rise to an ill-posed problem. The authors then resorted to a least squares method applied to the integral of the equations of motion as in Equation 2.25 as an alternative way of solving it

$$\min_{\mathbf{p}} \sum_{i=1}^{N} ||\bar{y}_{II}^i - \bar{y}_{II,exp}^i||^2 \tag{2.25}$$

where the quantity $\bar{y}_{II}$ is dependent on the vector of parameters $\mathbf{p}$ to be identified, namely to the base body parameters in the first case, to the load parameters for the second case and to the whole system parameters for the third case.

The algorithms are tested with the real inertial data of the ETS-VII Japanese satellite, and then with different sets of data. The algorithm works well with systems of different mass ratios (manipulator/spacecraft), spanning from 1/25 of the ETS-VII up to 1/7.

Ma, Dang and Pham in [24], [25] address the topic of the identification of the inertial parameters of the base body by exploiting a robotic arm and the conservation of momentum. The problem is treated as a linear identification one, and is divided in two steps: firstly the mass and center of mass of the spacecraft are identified, and secondly its inertia tensor.

For the first part of the method Equation 2.26 reads

$$\frac{m_R}{m_S}\left(\mathbf{v}_S + \boldsymbol{\omega}_S \times \boldsymbol{\rho}_R - \frac{\mathbf{p}_C(0)}{m_R}\right) + \boldsymbol{\omega}_S \times \boldsymbol{\rho}_S = -\mathbf{v}_S \tag{2.26}$$

being linear in the ratio of the masses (but the robotic arm mass is known and constant) and the center of mass position of the spacecraft $\boldsymbol{\rho}_S$. If the arm is moved into N different positions and the linear and angular velocities of the base are registered by on-board sensors, a linear system of regression equations can be built and solved for the aforementioned variables. After these two quantities are known, the inertia tensor can be identified through the simple equation

$$\mathbf{B}\mathbf{y} = \mathbf{c} \tag{2.27}$$

with the quantity $\mathbf{c}$ containing the known angular momentum values along the robotic arm trajectory and the matrix $\mathbf{B}$ the measured spacecraft angular velocities.
The simulations carried out to verify the methodology output as results that, for the second case, an initial spin is necessary to identify the inertia tensor. Moreover the mass of the arm shall not be less than the 5% of the spacecraft mass to obtain acceptable identification values.
The work carried out by Rackl, Lampariello and Albu-Schäffer in [26] treats the inertial parameter identification of free-floating robots equipped with torque sensors by the use of two different methods, one requiring the base body accelerations and one not requiring it, and compare them with other methods as the momentum conservation method or the energy conservation method.
The first method uses the modified linear form of the recursive Newton-Euler equations as in

$$\boldsymbol{\tau} = \begin{bmatrix} \mathbf{\Pi}_b & \mathbf{\Pi}_m \end{bmatrix} \begin{bmatrix} \mathbf{\Phi}_b \\ \mathbf{\Psi}_m \end{bmatrix} = \mathbf{\Pi}\mathbf{\Lambda} \tag{2.28}$$

which in regressor form can be written as

$$\mathbf{Y} = \mathbf{\Pi}\mathbf{\Lambda} + \boldsymbol{\rho} \tag{2.29}$$

The vector $\mathbf{Y}$ contains the torque measurements, whereas the matrix $\mathbf{\Pi}$ is the regressor matrix, and the solution to the over determined problem can be tackled with ordinary least squares optimization, by making sure that the regressor matrix is not rank deficient. In this specific case, the manipulator arm is considered to be known and thus no rank deficiency is present.
The second method, instead, stems from the general inverse dynamics equations of motion for free floating robots, in the form of the Reduced Dynamics Algorithm (RDA) which permits to eliminate the dependency of such equations from the base body acceleration. This framework is not linear anymore in the inertial parameters and must solved with a non-linear optimization algorithm (placing it in the category of the Input Error (IE) methods)

$$\Gamma = \sum_{i=1}^{n} \sum_{j=1}^{N} \tau_{i,j} - \tau_{i,j,msr} \tag{2.30}$$

The results of the simulations show that the method proposed in the paper gives results which are deemed acceptable and with good accuracy, with the second method (the one not requiring the base body acceleration measurements) giving the best results, which is consistent with what is reported in [18], being it an Input Error method. Moreover it seems to give better results with respect to the other ones explored, namely the conservation of momentum and energy.

Rackl and Lampariello in [27] tackle the problem of inertial parameter identification in the presence of fuel sloshing and flexible appendages.

The Reduced Dynamics Algorithm is again exploited and modified to account for these two additional effects. Equation 2.31 is used to analyse the system under the first or second case introduced above

$$\mathbf{H}\ddot{\mathbf{y}} + \mathbf{C} + \mathbf{K}\delta\mathbf{y} + \mathbf{D}\dot{\mathbf{y}} = \boldsymbol{\tau} \tag{2.31}$$

For what concerns the flexible appendages, rigid solar panels connected by flexible rotational joints are considered. The parameters to be identified, by knowing the manipulator ones, are the natural frequencies, mass, center of mass position and inertia of both the bas body and the appendages, the stiffness and the damping matrices.

The natural frequencies are retrieved in the first step of the method which uses a Fast Fourier Transform (FFT) on the measured joints torques and base angular velocity, whereas the remaining parameters are obtained with Equation 2.30.

The sloshing is instead treated with an equivalent pendulum model, composed of a fixed mass and slosh masses connected to the pendulum with springs and damper elements. The quantities to identify are the same as previous case, having this time substituted the appendages ones with values related to the equivalent pendulum. The robot parameters are again assumed to be known. The results in both cases show how the identification process improves by taking into account these two disturbances.

Christidi-Loumpasefski, Nanos and Papadopoulos in [12] introduce a new methodology for the inertial parameter identification of a free-floating space manipulator capable of obtaining all the parameters of the system in linear combinations, thus not only the base body ones, exploiting a non-null initial angular momentum stored in reaction wheels with known inertia. Such parameters clusters are then to be used in a model-based control framework. The method does not require noisy base body acceleration measurements but only joints angles and rates, base body angular velocity and base body attitude.

Having expressed the angular momentum of the system linearly with respect to the inertial parameters (Equation 2.32)

$$\mathbf{h}_{CM} = \mathbf{Y}_{\mathbf{h}}\left(\dot{\mathbf{q}}, \mathbf{q}, {}^{0}\boldsymbol{\omega}_{0}, \boldsymbol{\varepsilon}, \eta\right)\boldsymbol{\pi} \tag{2.32}$$

the resulting system, after being filled with measurements of the required quantities, is an over-determined system of linear equations which can be solved in a least squares sense

(Equation 2.33)

$$\hat{\boldsymbol{\pi}} = \left(\hat{\mathbf{Y}}_{\mathbf{h}}^{T}\hat{\mathbf{Y}}_{\mathbf{h}}\right)^{-1}\hat{\mathbf{Y}}_{\mathbf{h}}^{T}\,\mathbf{h}_{CM} \tag{2.33}$$

However, the regressor matrix $\mathbf{Y_h}$ is rank deficient and must be manipulated in order to perform such operation. The authors of [12] do not give an explicit method to treat with this problem but provide only a generic procedure to be tailored case by case.

The authors confront this new method with a commonly used one which makes use of the full equations of motion (EOM) of a space manipulator requiring base body acceleration measurements: the new formulation of the problem outperforms the EOM based one due to fact that the base body acceleration measurements are not needed to fill the regressor matrix, measurements which are typically afflicted by a non-negligible noise level. It has to be noted again that the non-null angular momentum is given by actuating reaction wheels, but their explicit contribution in the angular momentum equation is not present.

Christidi-Loumpasefski and Papadopoulos in [28] extend the previous work to free-flying space manipulators systems, by proposing a simple 2D system to be tested both in simulation and in an experimental setup.

Resembling the work in [25] the method is divided in two steps: firstly the linear momentum is used to identify the total mass of the system by exploiting pulses of on-board thrusters, and secondly a non-null angular momentum is given to the system in the same fashion of what was done in [12], with the introduction of additional kinematics to retrieve all the parameters needed for free-flying model-based control. In contrast to the work in [25] all the inertial parameters, both for the base and the manipulator, are identified remembering that they are obtained in linear combinations between each other.

As mentioned above a novelty in this article, a part from the additional identified kinematic quantities, are the obtained experimental results gathered with the *Cepheus* platform, capable of reproducing 2D free-flying/floating dynamics of a simple space manipulator. The algorithms are tested on the experimental platform and then verified by imposing a randomly parametrised joint angles trajectory to the system and subsequently comparing the simulated and measured responses. The output quantities are tracked with good accuracy after the identification process confirming the validity of the newly proposed method.

Christidi-Loumpasefski, Rekleitis and Papadopoulos in [29] continue the research topic by focusing on the development of a controller to reject the disturbances exerted on the system by accumulated reaction wheels momentum. The part which is interesting in this work for the present thesis is the explicit contribution of the reaction wheels accumulated momentum in the system total angular momentum.

The angular momentum given by the reaction wheels is (Equation 2.34)

$$\mathbf{H}_{RW} = \sum_{i=1}^{N_{RW}} m_{RW,i} \boldsymbol{\rho}_{RW,i} \times \dot{\boldsymbol{\rho}}_{RW,i} + \mathbf{I}_{RW,i} \left(\boldsymbol{\omega}_0 + \mathbf{R}_0 \ {}^{0}\mathbf{R}_{RW,i} \ {}^{RW,i}\mathbf{z}_{RW,i} \dot{q}_{RW,i} \right) \tag{2.34}$$

in which the first two terms are related to the spacecraft motion and the last one to the relative motion between the spacecraft and the reaction wheels. This is the additional contribution to be considered in the total angular momentum of the system leading to Equation 2.35

$$\mathbf{H}_{rs}(t) = \mathbf{Y}(t)\boldsymbol{\pi} = \mathbf{H}_{TOT} - \mathbf{R}_0(t)^{\mathbf{0}}\mathbf{H}_{RW/rs} \tag{2.35}$$

At this point the framework obtained is formally equal to the one in [12] with a more rigorous expression.

Naveen et al. in [13] propose a framework for inertial parameter identification that makes use of the momentum equations, and also in this case the assumption is of a non-null angular momentum given by on-board reaction wheels.

An interesting addition to the topic made by this article is the explicit presentation of an automatic procedure to obtain the minimal parameters of the system given a user-defined geometry for the manipulator, provided that it belongs to the family of open-chain kinematic trees.

Referring to the angular momentum equations, their linear formulation with respect to the inertial parameters is given by (after a simple manipulation) Equation 2.36

$$\mathbf{H}_i = \left({}^{I}\mathbf{R}_i \left[\left({}^{I}\mathbf{R}_i^T \boldsymbol{\omega}_i \right) \bullet \right] \right) \left[\bullet {}^{i}\mathbf{I}_i \right] + \left(\tilde{\mathbf{r}}_i \mathbf{v}_i \right) M_i + \left(\left(\tilde{\mathbf{r}}_i \tilde{\boldsymbol{\omega}}_i - \tilde{\mathbf{v}}_i \right) {}^{I}\mathbf{R}_i \right) M_i \ {}^{i}\mathbf{a}_i \tag{2.36}$$

As can be seen in contrast to the methodology proposed in [12], [28], and [29] the resulting regressor matrix is dependent on both the base body inertial position and linear velocity. In any case following the procedure given in [13] one can easily obtain the minimal form of the inertial parameters and the corresponding regressor matrix which coupled with a non-null angular momentum provided by the on-board reaction wheels form the classic system of over-determined linear equations to be solved with a least squares algorithm.

The article presents also a different approach to obtain exciting trajectories for the system, in opposition to the truncated Fourier Series used in [12].

Xu et al. in [30] and in [31] treat a method for identifying the complete inertia parameters of a space robotic system. The methodology is similar to the one employed in [28] and it is divided in two steps: an equivalent single-body identification, and an equivalent two-body identification. For the first step, all joints are blocked into a selected configuration, and the thrusters are used to translate and rotate the center of mass of the equivalent single-body system to excite its inertial parameters. The objective function for this first part is dependent from the of acceleration and velocity of the equivalent single-body system. For the second step, only one joint is unlocked and commanded to follow the exciting trajectory under the free-floating mode. In this case, instead, the linear and angular momentum equations are used to define the objective function.

At this point the parameter identification problem is transformed into a non-linear optimization problem, and the PSO algorithm is used to determine the optimal parameters. This algorithm is selected because it is suitable for solving unconstrained nonlinear problems and because it is easy to set-up, having to tweak few parameters. By sequentially unlocking the 1^{st} to n^{th} joint, the mass properties of body 0 to n are completely identified. The identification results on the first part of the method give very small errors ($O \sim 10^{-10}$), results which are then employed in second step, namely for the equivalent two-body system identification. In the second part, the unlocked joint follows exciting trajectories modeled with 3^{rd} order splines. Also for the second part promising results are obtained, however the authors do not provide any information regarding the noise models employed to model the sensors for the simulations.

Finally Wei et al. in [32] work on a topic which is slightly different from the main focus of this thesis: the identification of the inertial parameters of a grasped target in orbit, by exploiting information coming from the contact between the chaser gripper and the unknown target itself. This to increase the safety of the grasping process, by having a priori data to better model the control law to complete the grasping manoeuvre. Also in this case the momentum conservation of the chaser-target couple is assumed, and used to derive the linear identification model for the target inertial parameters.

The identification results of all the treated cases (i.e. of a target for which a priori information are available, a target and a heavy target for which no information on the inertial parameters are present) show that the identification of the target mass is quicker to converge with respect to to the target inertia and center of mass, which is in agreement with the theoretical studies performed for the research. The numerical results indicate that all of the target inertial parameters can be identified based on the linear identification model presented in the paper, given that at least three collision between the gripper and the target occur. By increasing the number and variety of collisions the of identification efficiency can be improved, leading to faster convergence to the real inertial parameters of the target.

3 | Mathematical Formulation Of The Problem

3.1. Kinematics Of A Space Manipulator

3.1.1. Geometrical Description Of The Manipulator

The geometry of a robotic arm is defined by assigning coordinate frames to each link, and the most convenient way to perform this task is to adhere to one of the existing conventions. In this thesis the one adopted is the one described by Khalil et al. in [19] and [33], also referred as "Modified Denavit-Hartenberg (DH)" convention, which is briefly summarized below.

This convention allows to address the position of one frame with respect to the adjacent one with just four parameters instead of the normally required six. The numbering of the links is such that the base link is labelled with 0 and the terminal link with n. Joint i is placed in between link i and $i-1$, and the variable q_i is associated to it. The frame associated with link i is placed such as:

- The $\hat{\mathbf{z}}_i$ axis is located along the axis of joint i;
- The $\hat{\mathbf{x}}_i$ axis is placed along the common normal between $\hat{\mathbf{z}}_i$ and $\hat{\mathbf{z}}_{i+1}$;
- The $\hat{\mathbf{y}}_i$ axis completes a right-hand rule coordinate system.

At this point the transformation matrix between the frame $i-1$ to the frame i can be expressed as function of the following four parameters:

- a_i is the distance from $\hat{\mathbf{z}}_{i-1}$ to $\hat{\mathbf{z}}_i$ along $\hat{\mathbf{x}}_{i-1}$;
- α_i is the angle from $\hat{\mathbf{z}}_{i-1}$ to $\hat{\mathbf{z}}_i$ about $\hat{\mathbf{x}}_{i-1}$;
- d_i is the distance from $\hat{\mathbf{x}}_{i-1}$ to $\hat{\mathbf{x}}_i$ along $\hat{\mathbf{z}}_i$;
- q_i is the angle from $\hat{\mathbf{x}}_{i-1}$ to $\hat{\mathbf{x}}_i$ about $\hat{\mathbf{z}}_i$.

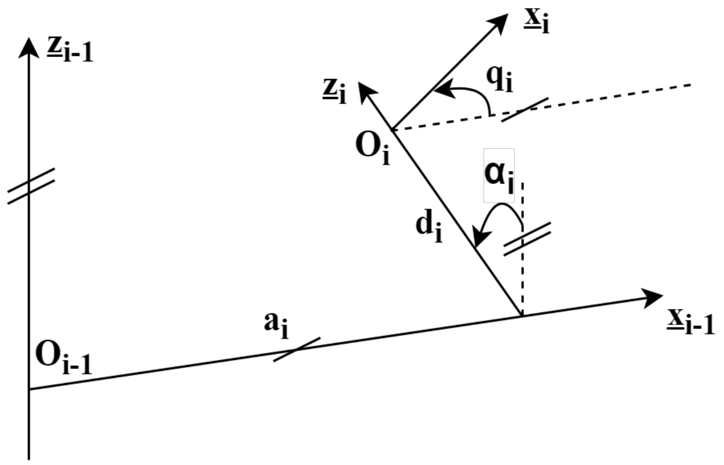


Figure 3.1: Modified DH Parameters

Having introduced the four variables (shown in Figure 3.1), the transformation matrix ${}^{i-1}\mathbf{T}_i$ defining frame i with respect to frame $i-1$ is reported in Equation 3.1

$$ {}^{i-1}\mathbf{T}_i = \begin{bmatrix} \cos q_i & -\sin q_i & 0 & a_i \\ \sin q_i \cos\alpha_i & \cos q_i \cos\alpha_i & -\sin\alpha_i & -\sin\alpha_i d_i \\ \sin q_i \sin\alpha_i & \cos q_i \sin\alpha_i & \cos\alpha_i & \cos\alpha_i d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.1} $$

3.1.2. Positions And Velocities

The center of mass position of each link ($\mathbf{c}_i$) with respect to an inertial reference frame[1] can be described with Equation 3.2 [34]

$$ \mathbf{c}_i = \mathbf{c}_0 + \sum_{j=1}^{i} \mathbf{a}_j + \sum_{j=0}^{i-1} \mathbf{b}_j \tag{3.2} $$

where $i = 1, ..., n$ represents the link index, $\mathbf{c}_0$ is the spacecraft center of mass position (from now on the spacecraft will be referred as base body), $\mathbf{a}_j$ is the vector of the center of mass of link j from joint j and $\mathbf{b}_j$ is the vector from the center of mass of link j to joint $j+1$. The quantity $\mathbf{b}_0$ represents the vector from the base center of mass to the point where link 1 is attached to it.

By taking the derivative of $\mathbf{c}_i$ with respect to time, every link center of mass velocity $\mathbf{v}_i$

[1] All the quantities referred to the inertial reference frame are denoted with no left superscript.

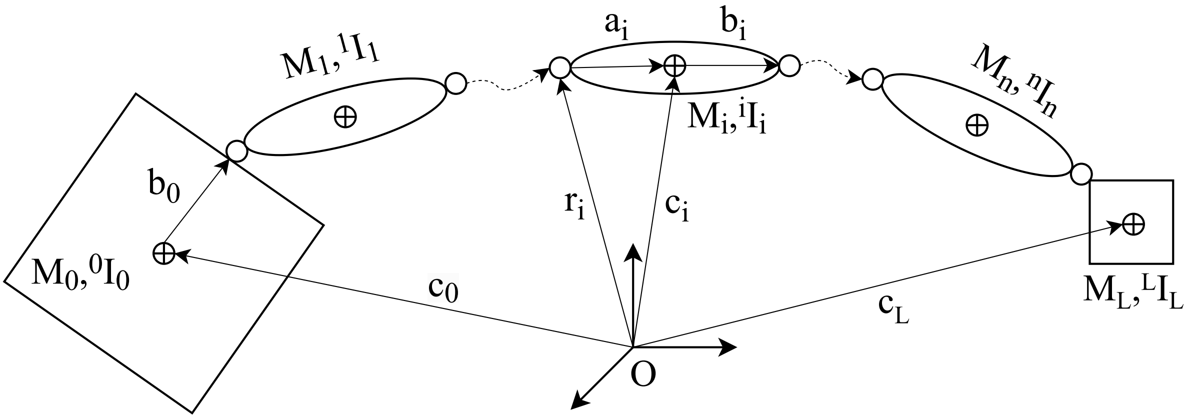


Figure 3.2: Space Robot Representation

can be written as in Equation 3.3

$$\mathbf{v}_i = \mathbf{v}_0 + \boldsymbol{\omega}_0 \times (\mathbf{c}_i - \mathbf{c}_0) + \sum_{j=1}^{i} \mathbf{z}_j \times (\mathbf{c}_i - \mathbf{r}_j) \cdot \dot{q}_j \tag{3.3}$$

where $\mathbf{v}_0$ and $\boldsymbol{\omega}_0$ are respectively the base body velocity and angular rate, $\mathbf{z}_j$ represent joint j axis of rotation, $\mathbf{r}_j$ is the position vector of joint j and $\dot{q}_j$ is its angular rate magnitude (namely the time derivative of joint j relative angular coordinate q_j). It has to be noted that both the base body linear and angular velocity are present in the description of links velocities, since for a space manipulator in free-floating mode (thus left free to tumble) the base body will react to the manipulator movement and for this reason this quantities are not null. The angular velocity of link i is instead computed as in Equation 3.4

$$\boldsymbol{\omega}_i = \boldsymbol{\omega}_0 + \sum_{j=1}^{i} \mathbf{z}_j \cdot \dot{q}_j \tag{3.4}$$

3.2. Free Floating Dynamics

The free floating dynamics for a space manipulator in absence of external forces/moments (except for forces/moments applied at the end effector $\mathbf{F}_h$) are expressed by the following set of equations of motion (Equation 3.5 [5])

$$\begin{bmatrix} \mathbf{M}_b & \mathbf{M}_{bm} \\ \mathbf{M}_{bm}^T & \mathbf{M}_m \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{x}}_b \\ \ddot{\mathbf{q}} \end{Bmatrix} + \begin{Bmatrix} \mathbf{c}_b \\ \mathbf{c}_m \end{Bmatrix} = \begin{Bmatrix} \mathbf{0} \\ \boldsymbol{\tau} \end{Bmatrix} + \begin{Bmatrix} \mathbf{J}_b^T \\ \mathbf{J}_m^T \end{Bmatrix} \mathbf{F}_h \tag{3.5}$$

where the matrices $\mathbf{M}_b$, $\mathbf{M}_m$ and $\mathbf{M}_{bm}$ represent respectively the composite inertia matrices for the base body, the manipulator and the coupling terms between the two, $\ddot{\mathbf{x}}_b$ is the vector of base acceleration and angular acceleration, $\ddot{\mathbf{q}}$ is the vector of joints accelerations, $\mathbf{c}_b$ and $\mathbf{c}_m$ are vectors that contain the terms related to centrifugal and Coriolis forces, $\boldsymbol{\tau}$ is the vector of joints torques, and finally $\mathbf{J}_b$ and $\mathbf{J}_m$ are Jacobian matrices that transmit forces/moments applied at the end effector to the base body and the other links. An explicit and extensive formulation for all the terms composing these equations of motion can be found in [35].

By integrating the first row in Equation 3.5 with respect to time [5], the total momentum of the system can be obtained as in Equation 3.6

$$\mathcal{L} = \mathbf{M}_b\dot{\mathbf{x}}_b + \mathbf{M}_{bm}\dot{\mathbf{q}} \tag{3.6}$$

For the purposes of this work the quantities of interest are the linear and angular momentum, for which the expressions will be described more in detail in the next section.

3.3. Linear And Angular Momenta

Equation 3.6 is a composition of linear and angular momenta. The linear part of it is reported in Equation 3.7

$$\mathbf{P} = \hat{\mathbf{M}}_b\mathbf{v}_0 + \hat{\mathbf{M}}_{bm}\dot{\mathbf{q}} \tag{3.7}$$

where the equations for $\hat{\mathbf{M}}_b$ and $\hat{\mathbf{M}}_{bm}$ are given below in Equation 3.8 and Equation 3.9

$$\hat{\mathbf{M}}_b = M_{TOT}\mathbf{E}_{3x3} \tag{3.8}$$

$$\hat{\mathbf{M}}_{bm} = \sum_{i=1}^{n} M_i \sum_{j=1}^{i} \mathbf{J}_{Pj} \tag{3.9}$$

$$\mathbf{J}_{Pi} = \left[\mathbf{z}_1 \times (\mathbf{c}_i - \mathbf{r}_1), ..., \mathbf{z}_i \times (\mathbf{c}_i - \mathbf{r}_i), 0, ..., 0\right] \tag{3.10}$$

in which M_i and M_{TOT} are respectively the mass of each body (excluding the mass of the base) and the total mass of the system, and $\mathbf{E}_{3x3}$ is a three by three identity matrix.

Concerning the expression for the angular momentum, it is reported in Equation 3.11

$$\mathbf{H} = \tilde{\mathbf{M}}_b\boldsymbol{\omega}_0 + \tilde{\mathbf{M}}_{bm}\dot{\mathbf{q}} \tag{3.11}$$

and the definitions of $\tilde{\mathbf{M}}_b$ and $\tilde{\mathbf{M}}_{bm}$ are given in Equation 3.12 and Equation 3.13

$$\tilde{\mathbf{M}}_b = \mathbf{I}_0 + \sum_{i=1}^{n} \left(\mathbf{I}_i + M_i \tilde{\mathbf{c}}_i \left(\tilde{\mathbf{c}}_{0i} \right)^T \right) \tag{3.12}$$

$$\tilde{\mathbf{M}}_{bm} = \sum_{i=1}^{n} \left(\mathbf{I}_i \mathbf{J}_{Hi} + M_i \tilde{\mathbf{c}}_i \mathbf{J}_{Pi} \right) \tag{3.13}$$

$$\mathbf{J}_{Hi} = [\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_i, 0, ..., 0] \tag{3.14}$$

(where $\mathbf{c}_{0i} = \mathbf{c}_i - \mathbf{c}_0$). If the system has zero linear momentum, the base velocity can be retrieved from Equation 3.7 as reported below

$$\mathbf{v}_0 = -\hat{\mathbf{M}}_{\mathbf{b}}^{-1} \hat{\mathbf{M}}_{\mathbf{bm}} \dot{\mathbf{q}} \tag{3.15}$$

This assumption will be used in Section 3.4 to remove the dependency of the regressor matrix from this quantity. From Equation 3.11, instead, the angular velocity of the base can be computed, both in case of zero angular momentum (as will be the case in Chapter 5) or with a constant value (as in [12] and [13])

$$\boldsymbol{\omega}_0 = -\tilde{\mathbf{M}}_b^{-1} \tilde{\mathbf{M}}_{bm} \dot{\mathbf{q}} \tag{3.16}$$

$$\boldsymbol{\omega}_0 = \tilde{\mathbf{M}}_b^{-1} \left(\mathbf{H} - \tilde{\mathbf{M}}_{bm} \dot{\mathbf{q}} \right) \tag{3.17}$$

3.4. Regressor Form Of The Angular Momentum Equations

The regressor form of the angular momentum for a multibody system (as the one under study in this thesis) consists in rearranging the equations of such physical quantity by separating the measurable variables from the dynamical parameters, making the new formulation linear with respect to the latter. A possible methodology to obtain this form for a space manipulator is the one proposed in [13].

The vector of parameters $\boldsymbol{\pi}$ to be identified comprehends ten inertial parameters for each body ($i = 0, ..., n$) and it is sorted out as follows:

$$\boldsymbol{\pi}_i = \left[{}^{i}I_{xx_i}, {}^{i}I_{yy_i}, {}^{i}I_{zz_i}, {}^{i}I_{xy_i}, {}^{i}I_{yz_i}, {}^{i}I_{xz_i}, M_i, M_i\, {}^{i}\mathbf{a}_{x_i}, M_i\, {}^{i}\mathbf{a}_{y_i}, M_i\, {}^{i}\mathbf{a}_{z_i} \right]^T \tag{3.18}$$

where ${}^{i}I_{pq_i}$, $p, q = x, y, z$ are the independent components of the inertia tensor for the *i-th* body, M_i is the mass of the *i-th* body and $M_i\, {}^{i}\mathbf{a}_{p_i}$, $p = x, y, z$ are the first moments of inertia of the *i-th* body.

The equations for the angular momentum of each body have to be rearranged in order to be linear in the inertial parameters, starting from Equation 3.19

$$\mathbf{H}_i = {}^{I}\mathbf{R}_i \, {}^{i}\mathbf{I}_i^{(c)} \, {}^{I}\mathbf{R}_i^T \boldsymbol{\omega}_i + \mathbf{c}_i \times M_i \, \dot{\mathbf{c}}_i \tag{3.19}$$

where $\mathbf{c}_i$ is the position of the center of mass of the *i-th* body in the inertial frame, ${}^{I}\mathbf{R}_i$ is the rotation matrix from the *i-th* reference frame to the inertial frame, and ${}^{i}\mathbf{I}_i^{(c)}$ is the inertia tensor of the *i-th* body referred to its center of mass. Vectors that do not present the left superscript are referred to the inertial frame.

By writing the position of the center of mass and its velocity for each body as

$$\mathbf{c}_i = \mathbf{r}_i + \mathbf{a}_i \tag{3.20}$$

$$\dot{\mathbf{c}}_i = \mathbf{v}_i + \boldsymbol{\omega}_i \times \mathbf{a}_i \tag{3.21}$$

where $\mathbf{r}_i$ is the position of the origin of the reference frame of the *i-th* body (placed in the center of mass for the base body, and at the joints for the robotic arm links), $\boldsymbol{\omega}_i$ is the angular velocity of the *i-th* body, and $\mathbf{a}_i$ is the position of the center of mass from the *i-th* body reference frame, one obtains the angular momentum as in Equation 3.22

$$\mathbf{H}_i = {}^{I}\mathbf{R}_i \, {}^{i}\mathbf{I}_i^{(c)} \, {}^{I}\mathbf{R}_i^T \boldsymbol{\omega}_i + M_i \left(\tilde{\mathbf{r}}_i \mathbf{v}_i + \left(\tilde{\mathbf{r}}_i \tilde{\boldsymbol{\omega}}_i - \tilde{\mathbf{v}}_i \right) \mathbf{a}_i + \mathbf{a}_i \times \boldsymbol{\omega}_i \times \mathbf{a}_i \right) \tag{3.22}$$

where the term $\tilde{\mathbf{r}}_i \tilde{\boldsymbol{\omega}}_i \mathbf{a}_i$ comes from $\mathbf{r}_i \times \boldsymbol{\omega}_i \times \mathbf{a}_i$ (the notation $\tilde{\mathbf{r}}_i$ stands for the skew symmetric matrix associated to the vector). The equation above is still not linear in the position of the center of mass $\mathbf{a}_i$ due to the term $\mathbf{a}_i \times \boldsymbol{\omega}_i \times \mathbf{a}_i$. By making the substitution in Equation 3.23

$$\mathbf{a}_i \times \boldsymbol{\omega}_i \times \mathbf{a}_i = \left(\mathbf{a}_i^T \mathbf{a}_i \mathbf{E}_{3x3} - \mathbf{a}_i \mathbf{a}_i^T \right) \boldsymbol{\omega}_i \tag{3.23}$$

from vector algebra, one can recognize the term in Equation 3.24

$${}^{I}\mathbf{R}_i \, {}^{i}\mathbf{I}_i \, {}^{I}\mathbf{R}_i^T = {}^{I}\mathbf{R}_i \, {}^{i}\mathbf{I}_i^{(c)} \, {}^{I}\mathbf{R}_i^T + M_i \left(\mathbf{a}_i^T \mathbf{a}_i \mathbf{E}_{3x3} - \mathbf{a}_i \mathbf{a}_i^T \right) \tag{3.24}$$

which is the inertia tensor of the *i-th* body in its reference frame, obtained by exploiting the parallel axis theorem. At this point, the angular momentum of the *i-th* body can be written in linear form with respect to its inertial parameters as shown in Equation 3.25

$$\mathbf{H}_i = \left({}^{I}\mathbf{R}_i \left[\left({}^{I}\mathbf{R}_i^T \boldsymbol{\omega}_i \right) \bullet \right] \right) \left[\bullet {}^{i}\mathbf{I}_i \right] + \left(\tilde{\mathbf{r}}_i \mathbf{v}_i \right) M_i + \left(\left(\tilde{\mathbf{r}}_i \tilde{\boldsymbol{\omega}}_i - \tilde{\mathbf{v}}_i \right) {}^{I}\mathbf{R}_i \right) M_i \, {}^{i}\mathbf{a}_i \tag{3.25}$$

$$[\boldsymbol{\omega}_i \bullet] = \begin{bmatrix} \omega_x & 0 & 0 & \omega_y & 0 & \omega_z \\ 0 & \omega_y & 0 & \omega_x & \omega_z & 0 \\ 0 & 0 & \omega_z & 0 & \omega_z & \omega_x \end{bmatrix} \qquad [\bullet\, \mathbf{I}] = [I_{xx}, I_{yy}, I_{zz}, I_{xy}, I_{yz}, I_{xz}]^T \tag{3.26}$$

The matrix containing the geometrical and measurable quantities (reference frames positions, S/C attitude, angular velocities, joint positions and velocities) can be grouped into a matrix $\mathbf{Y}_i$ (Equation 3.27)

$$\mathbf{Y}_i = \left[\left({}^I\mathbf{R}_i \left[\left({}^I\mathbf{R}_i^T \boldsymbol{\omega}_i\right) \bullet\right]\right) \;\; \tilde{\mathbf{r}}_i \mathbf{v}_i \;\; \left(\left(\tilde{\mathbf{r}}_i \tilde{\boldsymbol{\omega}}_i - \tilde{\mathbf{v}}_i\right)^I \mathbf{R}_i\right)\right] \tag{3.27}$$

The issue with the full matrix $\mathbf{Y}$ (obtained by appending the $\mathbf{Y}_i$, $i = 0, ..., n$) is that it is not full rank, thus it can not be inverted to perform identification in a least squares framework. To cope with this problem, [13] gives a systematic procedure to remove the linearly dependent columns from $\mathbf{Y}$ and then regroup the vector of inertial parameters $\boldsymbol{\pi}$ into parameter clusters accordingly. The linearly dependent columns which are removed are the ones with the highest index (which is consistent with literature and in particular with the identification chapter in the book by Khalil in [19] and with [36]) and are $[{}^iI_{yy_i},\ M_i,\ M_i\,{}^ia_{i_z}]$, for an open kinematic tree space robot connected with revolute joints.

As stated in the paper, a parent-child link pair creates redundant column in the child $\mathbf{Y}_i$ matrix. Following this reasoning, since the base link has no parent, its columns are independent. On the other hand, since terminal links have no children, they do not contribute to the formation of redundant columns in the other links, but, by having a parent, they have linearly dependent columns.

At this point, the vector of parameters can be divided in three categories:

1. Links without children (terminal links);
2. Links having both a parent and children;
3. Links having no parent (base link).

Starting from the terminal links, by combining the parameters with the linear coefficients obtained from the dependent columns of the $\mathbf{Y}_i$ matrices, and by removing the parameters associated with these dependent columns, one obtains the vector of minimal (or base) inertial parameters[2]. At the end of the process, the minimal $\mathbf{Y}_m$ matrix and the minimal

[2] The full expressions and a pseudo algorithm for the removal of the linearly dependent columns of $\mathbf{Y}$ can be found in [13].

$\boldsymbol{\pi}_m$ have dimensions

$$\mathbf{Y}_m \in \mathbb{R}^{3\ x\ (10+7n)}$$
$$\boldsymbol{\pi}_m \in \mathbb{R}^{(10+7n)\ x\ 1}$$

where n is the number of links that compose the space manipulator. To simplify the problem under study, by considering that for a seven DoF manipulator (which is the aimed system to study) the parameters clusters would be 59, all the products of inertia are set to zero as well as two out of three of the first moments of inertia for each body, by assuming that only the direction in which the link geometry develops is not null. This allows the dimension of the parameter clusters vector to be greatly reduced by almost half.

3.4.1. Formulation Independent From Base Body Position and Velocity

An issue with this formulation is that the regressor matrix requires the values of $\mathbf{r}_0$ and $\mathbf{v}_0$, which are the base body position and velocity referred to an inertial reference frame. While the velocity of the base body could be retrieved by visually tracking a point on the surface of the base, its position in the inertial frame can only be obtained by integrating twice the measurements obtained by accelerometers on board. This kind of procedure could result in an amplification of the noise of the original data considering that all position vectors of the system are computed from this value, thus worsening the condition number of the regressor matrix and consequently the identification results.
Christidi et al. propose a different formulation of the problem in [12] that does not require this kind of measurements but only the base body attitude, base body angular velocity, joints angular positions and angular rates (Equation 3.28)

$$\mathbf{H} = \mathbf{Y}\left(\mathbf{d}, \boldsymbol{\omega}_0, \mathbf{q}, \dot{\mathbf{q}}\right)\boldsymbol{\pi} \tag{3.28}$$

where $\mathbf{d}$ stands for the quaternions[3] which represent the base attitude.
To pass from the framework proposed in the article by Naveen et al. in [13] to the one of Christidi et al. in [12] a few steps have to be followed. Firstly all the position vectors have to be referred to the center of mass of the system, which under the assumption of zero linear momentum is fixed in inertial space and it is chosen as the inertial reference frame origin. Under this assumption it is possible to remove the dependence of the regressor

[3] In this work the quaternion is intended to be a four dimensional vector in which the "scalar" part is the fourth component, following the JPL convention.

matrix from the base velocity $\mathbf{v}_0$, by substituting this quantity from Equation 3.29

$$\mathbf{v}_0 = -\hat{\mathbf{M}}_{\mathbf{b}}^{-1}\hat{\mathbf{M}}_{\mathbf{bm}}\dot{\mathbf{q}} \tag{3.29}$$

which represents the equation of linear momentum conservation, as was explained in Section 3.3. Then substitute the first moments of inertia of the links as the product of each body mass times the its center of mass position, and use the parallel axis theorem to transport the inertia from each link reference frame to its center of mass.

At this point the regressor matrix contains again the inertial parameters of the space manipulator, in particular in the columns related to the parameter clusters of the base body mass and base body first moments of inertia, and the columns related to the first moments of inertia of the links. By multiplying these columns with the corresponding parameter clusters one can then collect the linear combinations of inertial parameters which are multiplied by the same kinematic expressions. Unfortunately this passage depends on the manipulator geometry, thus the combinations are specific of the system under study and have to be provided manually. Nevertheless some guidelines arose while analysing different open kinematic chains, in particular for a series of pitch joints after an initial roll joint (a typical geometry and also the one treated in [12]), and a configuration reproducing the one of DLR's Light-Weight Robot (LWR), thus a repeated series of the couple pitch-roll joints, after an initial roll joint. It has to be recalled that the products of inertia and two out of three first inertia moments were set to zero to simplify the problem. Concerning the first structure the following scheme has been found:

- Six (6) terms are related to the inertia components of the base even though the products of inertia are set to zero;
- Two (2) terms for each link related to principal inertia moments;
- Three (3) terms for each link comprehending the product between the geometrical parameters of the link and the coordinates of the attachment point of the first joint (thus one term for x, one for y, one for z) not considering the first one, due to the fact its inertial parameters are included in the base ones;
- Mixed terms between the links, not comprehending the first one (e.g. with four links, only the last three combine, for a total of three additional terms).

The number of combinations is different from the ones which would result by following the procedure in [13], due to the fact that velocity of the base body has been substituted to eliminate the regressor matrix dependency from it. In mathematical terms this can be written as (Equation 3.30):

$$N = 6 + (2 \cdot n_{links}) + (3 \cdot (n_{links} - 1)) + n_{comb} \tag{3.30}$$

Giving an example: for the 3 DoF case, after the recombination process and after setting to zero all the products of inertia and the desired first moments of the links (depending on the configuration of the manipulator), the number of independent columns of the regressor matrix is 18 (from the 40 original inertial parameters). Instead, after substituting the velocity of the base the independent columns become 19 which is consistent with [12]:

$$N = 6 + (2 \cdot 3) + (3 \cdot 2) + 1 = 6 + 6 + 6 + 1 = 19$$

in which $n_{comb} = 1$ because only the last two links combine between each other.
Thus, for a 4 DoF case (where for [13] and after the said simplifications the independent columns are 21) this becomes:

$$N = 6 + (2 \cdot 4) + (3 \cdot 3) + 3 = 6 + 8 + 9 + 3 = 26$$

in which $n_{comb} = 3$ in this case.
The parameter clusters with an additional DoF parallel to the previous ones maintain the same structure as the one presented in [12], with additional terms due to a further link.

For the second case, instead, the combinations behave as reported below:

- Six (6) terms are related to the inertia components of the base even though the products of inertia are set to zero (this remains equal to the previous case since the first joint is of the same type);
- Two (2) terms for each link related to principal inertia moments, but the geometrical parameters regroup with the ones of the pitch joint for every pitch-roll joints pairs, meaning that the inertias of the roll joints are not combined with other inertial parameters;
- Three (3) terms for each pair pitch-roll comprehending the product between the geometrical parameters of both links and the coordinates of the attachment point of the first joint (thus one term for x, one for y, one for z);
- Mixed terms between the couples, not comprehending the first one (e.g. with seven links, 2-3 combine with 4-5, 2-3 with 6-7, 4-5 with 6-7, for additional three combinations).

In a formula this becomes (Equation 3.31)

$$N = 6 + (2 \cdot n_{links}) + (3 \cdot n_{pair}) + n_{comb} \tag{3.31}$$

where n_{pair} represents the pitch-roll pairs excluding the first joint, and n_{comb} the combinations between links. For a 5 DoF geometry R-P-R-P-R (P: pitch joint, R: roll joint) the formula outputs

$$N = 6 + (2 \cdot 5) + (3 \cdot 2) + 1 = 6 + 10 + 6 + 1 = 23$$

since there are two pairs of P-R joints and hence $n_{comb} = 1$. The situation for a 6 DoF geometry R-P-R-P-R-P reads instead

$$N = 6 + (2 \cdot 6) + (3 \cdot 3) + 3 = 6 + 12 + 9 + 3 = 30$$

because the last pitch joint has to be considered a pair by itself and thus the mixed combinations are $n_{comb} = 3$ (2-3 with 4-5, 2-3 with 6, 4-5 with 6).

3.4.2. Simplification For 2D Systems

For 2D systems the problem further simplifies: after performing the same passages described in subsection 3.4.1 the regressor matrix $\boldsymbol{Y}^{2D}$ is dependent only on (Equation 3.32)

$$\mathbf{H}^{2D} = \mathbf{Y}^{2D} \left(\boldsymbol{\omega}_0,\ \mathbf{q},\ \dot{\mathbf{q}}\right) \boldsymbol{\pi}^{2D} \tag{3.32}$$

noting that now the system is not influenced by the base body attitude. Moreover, the vector of minimal inertial parameters for a 2D system consists of four parameters for the base

$$[RI_{zz_0},\ RM_0,\ RM_0 a_{x_0},\ RM_0 a_{y_0}]$$

where the R before the parameter indicates that that parameter has been combined with the other inertial parameters, having removed $[RI_{xx_0},\ RI_{yy_0},\ RI_{xy_0},\ RI_{yz_0},\ RI_{xz_0},\ RM_0 a_{z_0}]$ since they are null in 2D space (thus $10 - 6 = 4$ parameters), and three parameters for each link

$$[RI_{zz_i},\ RM_i a_{x_i},\ RM_0 a_{y_i}]$$

having removed $[RI_{xx_0},\ RI_{xy_0},\ RI_{yz_0},\ RI_{xz_0}]$, remembering that the inertial parameters that were associated with linearly dependent columns of the $\mathbf{Y}_i$ have already been removed (thus $7-4 = 3$ parameters for each link). Lastly, assuming that for each link the position of the center of mass in the local reference frame is given as

$$^{i}\mathbf{a}_i = \left[{}^{i}a_{x_i},\ 0,\ 0\right]^T$$

the terms $RM_i a_{y_i}$ are null and thus the columns of $\mathbf{Y}^{2D}$ corresponding to them can be removed. Finally, the dimensions of the bidimensional regressor matrix $\mathbf{Y}^{2D}$ and of the bidimensional parameters clusters vector $\boldsymbol{\pi}^{2D}$ are respectively

$$\mathbf{Y}^{2D} \in \mathbb{R}^{1\ x\ (4+2n)}$$
$$\boldsymbol{\pi}^{2D} \in \mathbb{R}^{(4+2n)\ x\ 1}$$

Thus for a 2D system composed of a base body and two links connected with revolute joints the number of minimal inertial parameters is 8, which is consistent with the result given in [28].

3.4.3. Symbolic Implementation Of The Angular Momentum Equations In Regressor Form

In order to obtain the analytical expressions needed to fill Equation 3.28, and thus the regressor matrix $\mathbf{Y}$ and the vector of parameter clusters $\boldsymbol{\pi}$, the whole procedure described in subsection 3.4.1 was implemented in MATLAB© to exploit its Symbolic Toolbox. An exhaustive summary on the analytic derivation of the generalized equations of motion of a free-floating spacecraft-manipulator system can be found in the article by Wilde et al. in [37].

Firstly, the model of the studied configuration has to be translated in symbolic form (for both geometrical and inertial parameters) through the `sym` function provided by the Toolbox, as well as the kinematic variables $\mathbf{c}_0$, $\mathbf{v}_0$, $\boldsymbol{\omega}_0$, $\mathbf{d}$ (for the base body), $\mathbf{q}$ and $\dot{\mathbf{q}}$ (for the manipulator). Having all these quantities it is then possible to compute the kinematics of the system in symbolic form, and obtain center of mass positions and velocities for all the links.

The next step is to obtain the regressor form of the angular momentum equations in the formulation proposed in [13] since their procedure is fully automated and permits to compute in a straightforward manner both $\mathbf{Y}_m$ and $\boldsymbol{\pi}_m$. The list below reports the main steps for this first part of the algorithm:

1. Compute the total regressor matrix of the system using Equation 3.27;
2. Build the symbolic coefficients related to the linear combinations between the columns

of the regressor matrix $\mathbf{Y}$, which are dependent on the geometrical description of the selected model;

3. Append all the inertial parameters of system in one vector (considering a system having n links, it has dimensions $[10 \cdot (n+1)]$), which will be successively treated by combining the said parameters into clusters;

4. Exploiting the coefficients obtained before, regroup the inertial parameters in the vector $\boldsymbol{\pi}$ and remove the rows related to the linearly dependent columns in the regressor matrix, from which the said columns are also removed.

Having evaluated the symbolic expressions for $\mathbf{Y}_m$ and $\boldsymbol{\pi}_m$, the second part of the algorithm transforms the current formulation in the one proposed in [12]. The starting point is to make the simplifications introduced at the end of Section 3.4, thus setting to zero the inertia products and keeping just one first moment of inertia for each link, namely the one associated with the link main direction. Consequently, all the rows of the parameter clusters vector which are now zero can be removed together with the associated columns in the regressor matrix. Then the inertias of each body composing the system have to be referred to the body center of mass, exploiting the parallel axes theorem, and the symbolic first moments of inertia are replaced by the product between each mass times the only non null component of the local center of mass position.

Building the composite inertia matrices of the system (with the expressions presented in Section 3.3 and in particular with Equation 3.7 and Equation 3.8) in symbolic form permits to exploit Equation 3.29 after the assumption of zero linear momentum to obtain the base body velocity as a function of the variables of the problem. It is now possible to substitute this expression inside the regressor matrix and thus eliminate its dependency from it, filling it again with inertial parameters (in particular masses and local center of mass positions) in the columns associated with the combinations between the first moments of inertia.

Arrived at this point with the algorithm, the columns of the regressor matrix associated with the principal moments of inertia of the each body do not have to be modified since they are already in their final form. The remaining columns (introduced in the paragraph above), instead, have to be multiplied with the associated rows in the parameter cluster vector, obtaining in such way expressions containing both kinematic quantities and combinations of inertial parameters. The discriminant factor to transform these complex symbolic strings as a sum of individual quantities is to isolate the combinations between the local center of mass positions of the same link or of different links (e.g. a_1^2 or $a_1 \cdot b_1$ or $a_1 \cdot b_2$), together with masses combinations and the associated kinematic expressions (for an example see Appendix A). Once all these individual sets are separated from each

other, it is possible to combine them by selecting the terms sharing identical kinematic expressions. The latter quantity is put inside the new regressor matrix (already filled with the columns associated with each body inertias) and these new parameter clusters are instead inserted in the new $\boldsymbol{\pi}$ vector, being careful to associate the columns of $\mathbf{Y}$ with the correct clusters π_i.
Unfortunately, as said in subsection 3.4.1, these combinations should be input by the user a priori, but in any case the guidelines presented in the same section might help with the process.

3.5. Exciting Trajectories Optimization

The same structure of the joint trajectories given in [12] are retained, thus a truncated Fourier series with N_h harmonics, to which a polynomial expression of the fifth order is added in order to impose boundary conditions on joint positions, velocities and accelerations:

$$q_i(t) = \sum_{j=1}^{N_h} \frac{{}^i a_j}{\omega_f \cdot j} \sin(\omega_f \cdot j \cdot t) - \frac{{}^i b_j}{\omega_f \cdot j} \cos(\omega_f \cdot j \cdot t) + \sum_{k=0}^{5} {}^i c_k \cdot t^k \tag{3.33}$$

where $i = 1, ..., n$ and $\omega_f = 2\pi/t_f$ (t_f is the total time of the exciting manoeuvre). For what concerns the evaluation of the polynomial expression coefficients, MATLAB© Symbolic Toolbox was exploited to obtain the analytical expression of such variables in order to speed up the computations. For each joint the free parameters are given by

$$N_{free} = (N_h \cdot 2)N_J$$

where N_h is the number of harmonics, and N_J is the number of joints. The coefficients of the truncated Fourier series are the optimization variables for the minimization problem stated in Equation 3.34, for which the condition number of the regressor matrix is the cost function:

$$\min_{{}^i a_j, {}^i b_j} \quad \text{cond}(\mathbf{Y}) \quad \text{s.t.} \begin{cases} q_{i_{min}} \leq q_i \leq q_{i_{max}} \\ \dot{q}_{i_{min}} \leq \dot{q}_i \leq \dot{q}_{i_{max}} \\ \ddot{q}_{i_{min}} \leq \ddot{q}_i \leq \ddot{q}_{i_{max}} \end{cases} \quad i = 1, ..., N_J \quad j = 1, ..., N_h \tag{3.34}$$

in which ${}^i a_j$, ${}^i b_j$ are the free coefficients of the series, and constraints are added to joints positions, velocities and accelerations to resemble mechanical constraints. The optimization loop proceeds as follows:

1. A guess of the free coefficients is given (from a uniform random distribution between

-1 and 1), from which the trajectories for joints positions and velocities are evaluated for a specified total manoeuvre time and number of sampling points.

2. For each time instant (thus for each sample) the forward kinematics of the manipulator is evaluated, and the composite inertia matrices of the system are built.

3. The angular velocity of the base body referred to the inertial frame is evaluated, such that the total angular momentum of the system remains constant/equal to zero (depending on the treated case). The base body velocity is also evaluated, in order to propagate also the base body position in the inertial frame.

4. Having obtained the values of joint positions and velocities from the given trajectories, the base body attitude represented by quaternions and the base body angular velocity, the reduced regressor matrix at the current time instant can be filled.

5. Update the base body attitude for the next cycle by rotating the base body angular velocity in the body frame and by integrating the associated quaternions. Notice that for the optimization loop, to speed up calculations, the integration is made by adding to the current quaternion the product between its derivative and a delta time.

Once the regressor matrix is filled for each time step, its condition number is evaluated and the optimization can move to its next iteration.
The optimized trajectories are evaluated without taking into account noisy measures. To check the validity of the found solution, the pseudo inverse of the regressor matrix is computed (after having verified that it is full rank) in the least squares sense, and the linear combinations of the inertial parameters are identified via non recursive least squares identification. The relative errors of this process between the true parameters and the identified ones are in the order of $O \sim (10^{-10})$, as one would expect since there is no random noise added to the measures. The formula in Equation 3.35 reports the expression for the relative errors evaluation

$$e_{rel,i} = \frac{\hat{\pi}_i - \tilde{\pi}_i}{\hat{\pi}_i} \cdot 100 \ \ [\%] \tag{3.35}$$

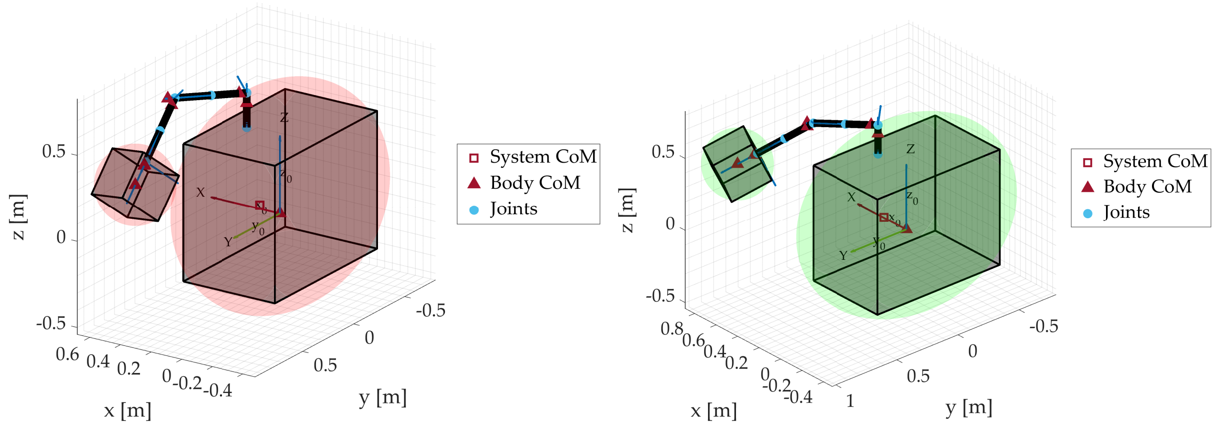
where $\hat{\pi}_i$ represents the true value of the i-th parameter, whereas $\tilde{\pi}_i$ the identified one.

3.5.1. Collision Avoidance

To make the simulations more realistic, the optimized trajectories for the robotic arm shall be collision free. A conservative approach has been followed, by encapsulating the base body in a ellipse as tight as possible to its physical dimensions, and the end-effector in a sphere based on the same considerations: a collision is effectively happening when the distance between the ellipse surface and the center of the sphere is less than the sphere radius.

Unfortunately, this kind of approach depends on the refinement with which the surface of the ellipse is discretized to be sure that the two fictious volumes are not colliding, thus a fine grid should be provided. Given a parameter n, the matrices which describe the X, Y, Z coordinates have dimensions $[n+1,\ n+1]$, meaning that at every time step the number of points which have to be checked are $(n+1)^2$, increasing in this way the size of the non linear constraints vector, slowing down the simulation to non feasible waiting times given the available hardware. It was decided to verify if the optimized trajectory is effectively collision free a posteriori and force the process to avoid collisions by tightening the constraints on the joints allowed motion.

Figures 3.3a and 3.3b display both situations: a red coloured surface signifies that a collision is taking place, whereas a green shaded surface means collision free. It has to be stressed out that a conservative approach has been followed: to check the contact in 3D space two surfaces with simple analytic formulation have been selected (an ellipse for the base body and a sphere for the end-effector), and in fact in the shown image representing a crash between the end effector volume and the base one it can be clearly seen that the two objects are not actually touching. Nevertheless, further studies could provide a more reliable collision model, which was not the scope of this thesis.



(a) Collision Example

(b) No Collision Example

Figure 3.3: Collision/No Collision Visualization

4 | Preliminary Research: Baseline Simulations

4.1. Simulink Model Description

As introduced in Section 1.3, a Simulink model was developed with the aim of simulating the free-floating dynamics of the system in response to the exciting trajectories performed by the robotic arm and to reproduce noisy measurements from sensors in order to perform a statistical analysis on the quality of the identification process. The tracked quantities are the mean relative errors for each parameter cluster, and also the maximum mean relative error, which should be contained within a reasonable value (as reported in the *OBSIdian* studies in [11]) of the original one for each cluster.
The model is composed by various blocks (displayed in Figure 4.1)

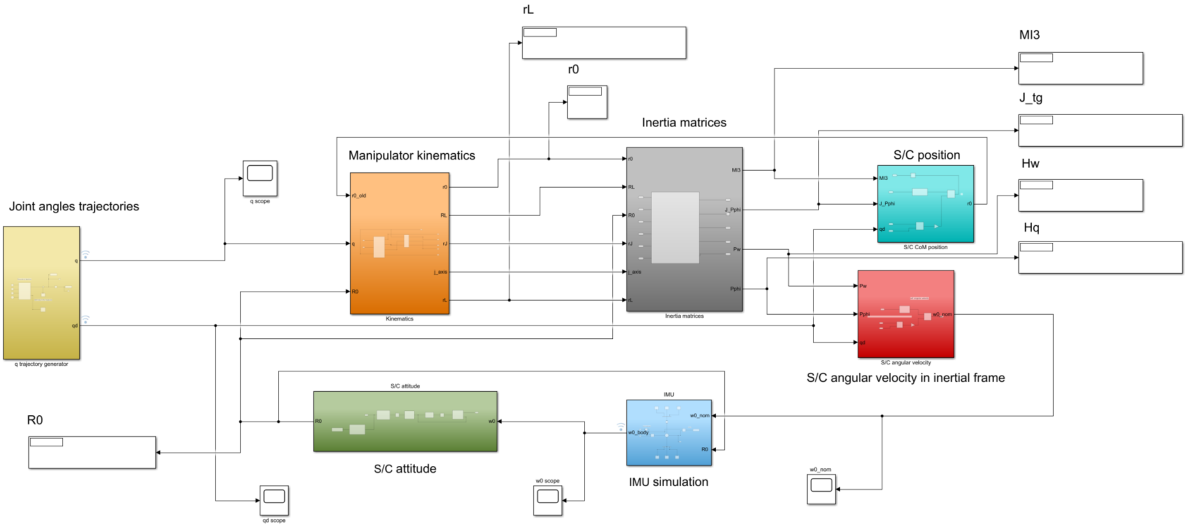


Figure 4.1: Simulink Model

and in particular:

- The block "Joints angles trajectories" evaluates the joints angles for each time step given the truncated Fourier series coefficients (coming from the exciting trajectory optimization explained in Section 3.5) and adds white noise to the simulated measurements. Moreover the joints angles are numerically differentiated to obtain the joints angular rates, as would be done in reality.

- The block "Manipulator kinematics" computes the manipulator forward kinematics with respect to an inertial frame whose origin is in the system center of mass (and thus has to be evaluated at each time step) as described in Section 3.1.
- The block "Inertia matrices", that computes the composite inertia matrices of the system, namely for the base, the manipulator and the coupling terms between base and manipulator (Section 3.2).
- The block "S/C position" computes the base body velocity from linear momentum conservation equation, which was treated in Section 3.3, and integrates it to obtain the base center of mass position in the inertial reference frame.
- The block "S/C angular velocity" that evaluates the base body angular velocity in the inertial frame from angular momentum conservation equation, reported in Section 3.3.
- The block "IMU simulation" reproduces IMU measurements (only gyroscope measurements, since the framework does not require base body acceleration measurements), after having rotated the angular velocity vector in the base body frame. The measurements are obtained by exploiting the gyroscope noise model presented in subsection 4.1.1.
- The block "S/C attitude" is needed to simulate the base body attitude measurements by integrating quaternions (since this attitude parametrization is singularity-free), given that for the sake of simplicity no attitude sensors and thus attitude estimation algorithms are treated in the present work.

4.1.1. Noise Models

The measures needed to fill the regressor matrix given in subsection 3.4.1 are joints angles and angular rates, the base body angular velocity and the base body attitude. To be consistent with the work presented in [12], the available measures on board of the robotic servicer are the joints angles coming from encoders and the base body angular velocity from an IMU. No sensors for the base body attitude are treated in the cited work, thus the base body attitude is obtained by integrating quaternions exploiting the knowledge of the base body angular velocity in the body frame. In an experimental set up attitude sensors should be used (if possible) to estimate the base body attitude, and use the IMU just for the evaluation of angular velocity, to keep the two measures independent from each other and obtain in this way a more realistic simulation.

4.1.2. Joints Angles & Angular Rates

The nominal measurements for joints angles and angular rates come from the optimized trajectories computed previously. White noise with zero mean and standard deviation σ_q [rad] is added at each time step to each joint angle. Joints angular rates are instead computed by differentiating numerically with finite differences the joints angles. This operation amplifies the measurements noise: in an experimental set up the joints angles obtained with motor encoders should be filtered and then differentiated, and the obtained joints angular rates be filtered as well to obtain smooth signals.

4.1.3. Gyroscope

The mathematical model of the gyroscope measurements is taken from the work of Crassidis and Markley in [38] and is here reported (Equation 4.1):

$$\begin{cases} {}^{0}\tilde{\boldsymbol{\omega}}_0 &= {}^{0}\boldsymbol{\omega}_0 + \mathbf{n}_\omega + \mathbf{b} \\ \\ \dot{\mathbf{b}} &= \mathbf{n}_{\mathbf{b}\omega} \end{cases} \tag{4.1}$$

where ${}^{0}\boldsymbol{\omega}_0$ is the true angular velocity, ${}^{0}\tilde{\boldsymbol{\omega}}_0$ is the "measured" angular velocity, $\mathbf{n}_\omega$ and $\mathbf{n}_{\mathbf{b}\omega}$ are white noises with zero mean and standard deviations σ_b [rad/s] and $\sigma_{b\omega}$ [rad/s^2] for every angular velocity component. The bias $\mathbf{b}$ [deg/hr] has instead an initial value equal for every component. σ_b and $\sigma_{b\omega}$ are evaluated from the ARW (Angle Random Walk) and RRW (Rate Random Walk), from the following relations (Equation 4.2)

$$\begin{cases} \text{ARW} &= \sigma_b\sqrt{T_s} \quad \left[\frac{rad}{s^{0.5}}\right] \\ \\ \text{RRW} &= \sigma_{b\omega}\sqrt{T_s} \quad \left[\frac{rad}{s^{1.5}}\right] \end{cases} \tag{4.2}$$

where T_s is the rate integrating gyroscope sampling time in [s]. The true angular velocity ${}^{0}\boldsymbol{\omega}_0$ is computed by rotating in the base body frame the angular velocity vector obtained from the equation of conservation of the total angular momentum, after having computed the inertia matrices.

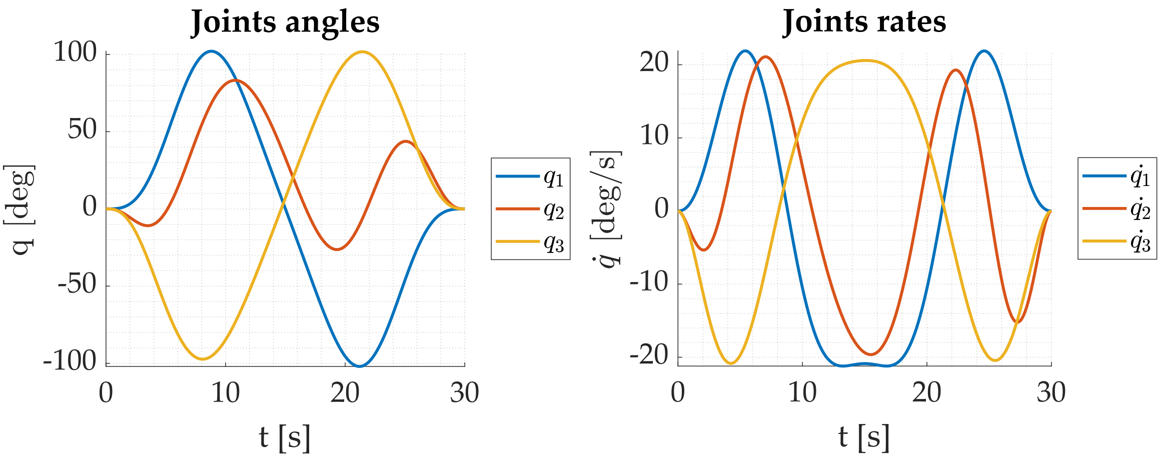
4.1.4. Random Numbers Generation in Simulink

An issue with random generated numbers in the Simulink environment is that the seed of the random generator block has to be manually changed for each simulation. To overcome this problem, given the number of random numbers to be produced, a set of seeds is previously obtained with the `randperm.m` MATLAB© function, which permits to generate arrays of randomly picked numbers from 1 to a user selected number. In this way each time a new simulation starts the seed numbers are changed and are different and random every time. Moreover, this methodology accounts also for repeatability, since given a specified seed number the random number generator block produces always the same output.

The random number generator block selects the value from a Gaussian distribution and requests in input the variance (thus the standard deviation squared) of the considered statistical distribution. It is also a discrete time block, meaning that a sampling time has to be chosen for the block to output the signal, and it forces the simulation to follow its sampling time.

4.2. Baseline Simulations Results: Case Study 1

To check the consistency and the validity of the developed simulation environment, both for the free floating dynamics and the exciting trajectories optimization, the same space manipulator data and sensor noise parameters provided in the work by Christidi et al. in [12] were used. It has to be recalled that this methodology requires an initial non-null angular momentum for the identification procedure.



(a) Case Study 1: Joints Angles

(b) Case Study 1: Joints Angular Rates

Figure 4.2: Case Study 1: Optimized Joints Trajectories

Firstly, the pseudo-algorithm presented in Section 3.5 is exploited to obtain a trajectory with minimum condition number by satisfying the imposed constraints on joints angles and rates. Table 4.1 reports the optimized coefficients of the truncated Fourier series, whereas Figure 4.2 displays the joints angles time series and the joints rates time histories. The found coefficients return a condition number $cond(\mathbf{Y}) = 42$, which is comparable with the one reported in [12].

After the optimized coefficients are obtained, the seed numbers for random numbers block generators for each run are computed. Afterwards the results of 2500 simulations (this number is justified by looking at the identified parameters mean and standard deviation convergence shown in Figure 4.4) are used to compute the regressor matrix for each of them and exploit the non recursive least squares formula (Equation 4.3) to identify the inertial parameters of the system, which are then confronted with the true ones to obtain absolute and relative errors. In the formula $\hat{\mathbf{Y}}$ indicates that the regressor matrix $\mathbf{Y}$ is filled with noisy measurements.

Table 4.1: Case Study 1: Optimized Fourier Coefficients

$^{i}a_{j}$		$^{i}b_{j}$	
$^{1}a_{1}$	-3.4282	$^{1}b_{1}$	-0.0015
$^{1}a_{2}$	-0.4343	$^{1}b_{2}$	-0.0003
$^{1}a_{3}$	-0.1726	$^{1}b_{3}$	-0.0002
$^{2}a_{1}$	-20.9963	$^{2}b_{1}$	0.9185
$^{2}a_{2}$	-1.5940	$^{2}b_{2}$	0.1559
$^{2}a_{3}$	-0.2083	$^{2}b_{3}$	0.0655
$^{3}a_{1}$	-9.0245	$^{3}b_{1}$	-0.1206
$^{3}a_{2}$	-0.4094	$^{3}b_{2}$	-0.0276
$^{3}a_{3}$	0.0098	$^{3}b_{3}$	-0.0166

$$\boldsymbol{\pi} = \left(\left(\hat{\mathbf{Y}}^{T}\right) \cdot \hat{\mathbf{Y}}\right)^{-1} \cdot \hat{\mathbf{Y}}^{T} \cdot \mathbf{H} \tag{4.3}$$

As introduced before the convergence of the mean and standard deviation of each parameter is checked, to see if the results stabilize or are still oscillating for the given number of simulations. Figure 4.4 displays the two quantities for the first parameter cluster π_1, confirming that full convergence for both the mean and standard deviation in achieved. The maximum relative error for each simulation is tracked, staying below 2.5% for all runs (Figure 4.3), together with the mean value ($\bar{\boldsymbol{\pi}}_{est}$) and standard deviation ($\boldsymbol{\sigma}_{est}$) for each parameter, the best estimate[1] ($\boldsymbol{\pi}_{best}$) and finally the

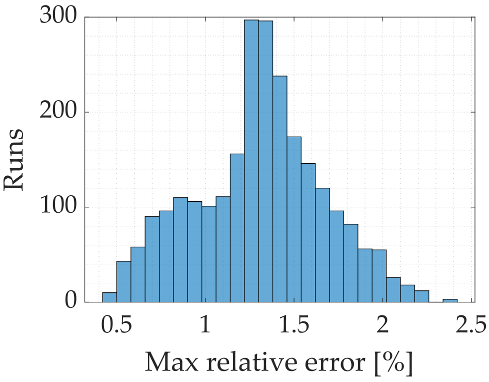
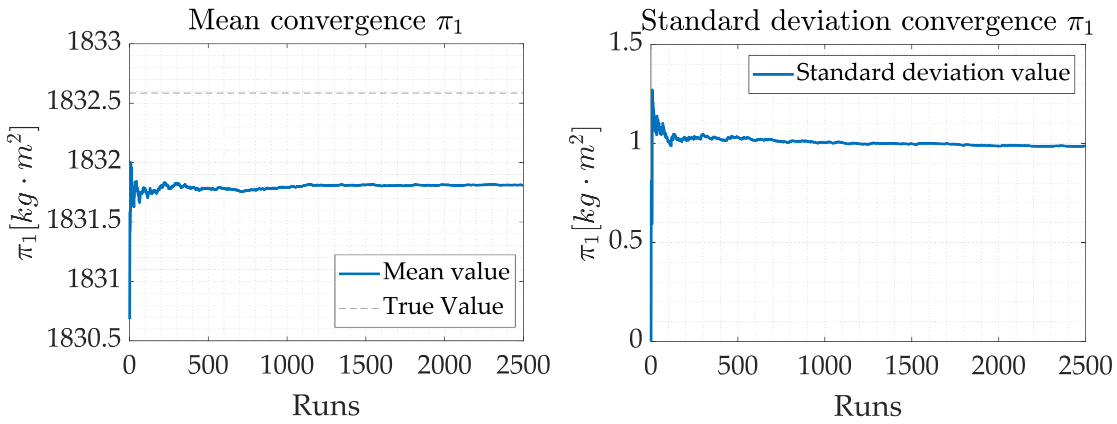


Figure 4.3: Case Study 1: Maximum Relative Error

[1] The best estimate is selected as the one which presents the minimum mean value of the absolute relative errors between all the runs.



(a) Case Study 1: π_1 Mean Convergence

(b) Case Study 1: π_1 Standard Deviation Convergence

Figure 4.4: Case Study 1: π_1 Convergence Analysis

mean relative error ($\bar{\boldsymbol{\pi}}_{err}$). The mentioned quantities are reported in Table 4.2, opposed to the "real" value of each parameter cluster ($\boldsymbol{\pi}_{true}$). All the entries of the table have units $[kg \cdot m^2]$, unless otherwise specified.

Table 4.2: Case Study 1: Identification Results

$\boldsymbol{\pi}$	$\boldsymbol{\pi}_{true}$	$\bar{\boldsymbol{\pi}}_{est}$	$\boldsymbol{\pi}_{best}$	$\boldsymbol{\sigma}_{est}$	$\bar{\boldsymbol{\pi}}_{err}$ [%]
π_1	1832.5856	1831.8101	1832.6132	0.9871	0.0423
π_2	-104.2654	-103.7782	-104.0855	0.5830	0.4672
π_3	-154.0284	-154.0395	-154.6870	0.6283	-0.0072
π_4	1832.5856	1828.5426	1832.8003	2.1302	0.2206
π_5	-154.0284	-152.6908	-153.9799	0.5901	0.8684
π_6	1708.5308	1707.9252	1708.5874	0.7879	0.0354
π_7	321.5317	318.5664	320.5653	1.0874	0.9222
π_8	321.5417	320.2949	321.4419	0.4768	0.3877
π_9	-255.9765	-255.1116	-256.0284	0.5119	0.3379
π_{10}	256.0265	255.7413	256.0515	0.2791	0.1114
π_{11}	-65.4552	-64.8485	-65.4134	0.2575	0.9268
π_{12}	65.5052	65.6798	65.7098	0.1974	-0.2666
π_{13}	142.1801	141.4940	141.8448	0.2100	0.4826
π_{14}	213.4479	211.7359	213.0494	0.4616	0.8020
π_{15}	142.1801	141.4882	142.1729	0.2685	0.4866
π_{16}	47.3934	47.7219	47.4340	0.1414	-0.6933
π_{17}	71.1493	70.6140	71.0168	0.2470	0.7523

π_{18}	47.3934	46.8342	47.1637	0.2071	1.1799
π_{19}	96.4455	96.3094	96.4125	0.1056	0.1411

In the column of the mean relative errors it can be seen that the maximum value is achieved by one of the parameters clusters having the smallest value, which is in line with what was expected. A part from that particular entry, all the other mean relative errors remain below 1%, confirming the validity of the implemented method and of the optimization process, as well as having obtained results which are comparable with [12].

4.3. Reaction Wheels Accumulated Momentum Case

In the work performed until now there was no information regarding how the system had a non null angular momentum. The continuation of the baseline analysis requires to include a possibile source for that, namely the angular momentum accumulated in reaction wheels (RW). To achieve that the work presented in [29] is followed.
Equation 4.4 reports the total angular momentum of the system by considering the contribution of RW, in the absence of external forces and moments

$$\mathbf{H}_{tot} = \mathbf{H}_{rs} + \mathbf{R}_0 {}^{\mathbf{0}}\mathbf{H}_{RW/rs} = \text{const.} \tag{4.4}$$

where the angular momentum of the robotic servicer is given by Equation 4.5

$$\mathbf{H}_{rs} = \tilde{\mathbf{M}}_b \boldsymbol{\omega}_0 + \tilde{\mathbf{M}}_{bm} \dot{\mathbf{q}} \tag{4.5}$$

and ${}^{\mathbf{0}}\mathbf{H}_{RW/rs}$ is the angular momentum that the RW exchange with the base, expressed in the base frame. The full expression for the RW angular momentum is (Equation 4.6)

$$\mathbf{H}_{RW} = \sum_{i=1}^{N_{RW}} m_{RW,i} \mathbf{c}_{RW,i} \times \dot{\mathbf{c}}_{RW,i} + \mathbf{I}_{RW,i} \left(\boldsymbol{\omega}_0 + \mathbf{R}_0 \, {}^{0}\mathbf{R}_{RW,i} \, {}^{RW,i}\mathbf{z}_{RW,i} \dot{q}_{RW,i} \right) \tag{4.6}$$

where the first two terms are due to the base movement, whereas the last term is the one exchanged by the RW with the base. If the inertial parameters of the RW are assumed to be known, by considering that the system is left free to tumble without any external forces or moments (as before), and if the RW are left to rotate freely, their angular momentum ${}^{\mathbf{0}}\mathbf{H}_{RW/rs}$ remains constant over time. If the inertial parameters of the RW are included in the base inertial parameters, the identification framework is the same as the previously considered one, with the only difference of the presence of this extra term. This means that the regressor matrix $\mathbf{Y}$ and the vector of parameters clusters $\boldsymbol{\pi}$ are the same as treated in Section 4.2.

Making the assumption of zero initial base angular velocity (e.g. obtained with thrusters) and joints initially at rest, but with accumulated angular momentum in the RW, the initial angular momentum of the system is equal to (Equation 4.7)

$$(\mathbf{H}_{tot})_{in} = (\mathbf{R}_0)_{in}{}^{\mathbf{0}}\mathbf{H}_{RW/rs} = \mathbf{H}_{tot} \tag{4.7}$$

and remains constant over time in the absence of external forces and moments.

The identification process remains formally the same as the previous case, with the difference that the robotic servicer angular momentum is now equal to (for each time step Equation 4.8)

$$\mathbf{H}_{rs}(t) = \mathbf{Y}(t)\boldsymbol{\pi} = \mathbf{H}_{tot} - \mathbf{R}_0(t)^{\mathbf{0}}\mathbf{H}_{RW/rs} \tag{4.8}$$

It is expected that the optimized trajectories will achieve a higher condition number than in the previous case, given the fact that in the first instants of the trajectory the base body angular velocity is null or very close to zero.

4.4. Baseline Simulations Results: Case Study 2

As introduced in Section 4.3 the second case study (Case Study 2) deals with RW accumulated angular momentum, which in absence of external forces and moments remains constant over time. The system data (S/C and manipulator geometrical and dynamical parameters) are the same exploited in Section 4.2 with the addition of RW initial angular momentum in the base frame ${}^{\mathbf{0}}\mathbf{H}_{RW/rs}$. As foreseen the condition number of the regressor matrix $\mathbf{Y}$ found after the optimization process is almost doubled ($cond(\mathbf{Y}) = 78$) and the Fourier coefficients for the exciting trajectory are reported in Table 4.3. The joint angles and angular rates time histories are shown in Figure 4.5. The formula which has been used for the identification task is

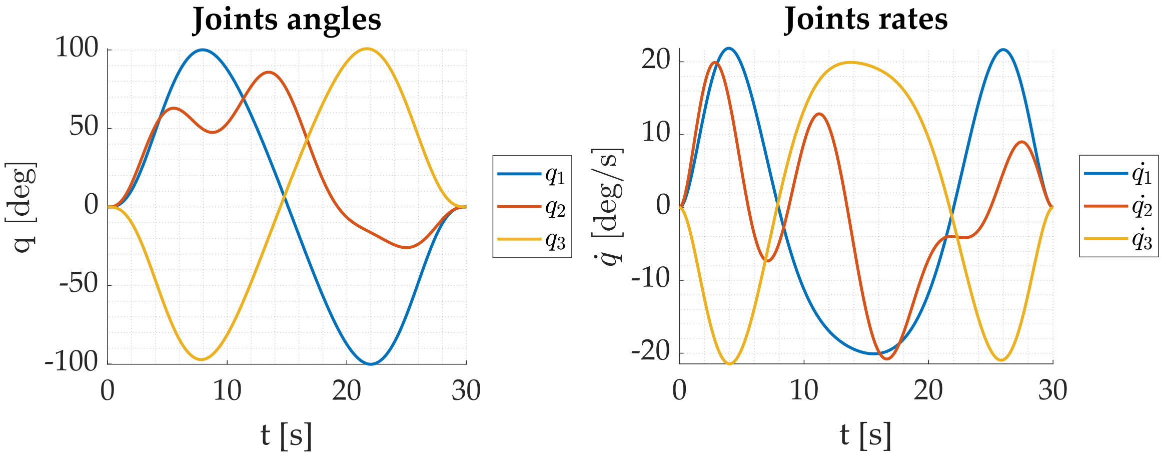
Table 4.3: Case Study 2: Optimized Fourier Coefficients

${}^{\mathrm{i}}\mathrm{a}_{\mathrm{j}}$		${}^{\mathrm{i}}\mathrm{b}_{\mathrm{j}}$	
${}^{1}\mathrm{a}_{1}$	14.0691	${}^{1}\mathrm{b}_{1}$	0.0384
${}^{1}\mathrm{a}_{2}$	0.7551	${}^{1}\mathrm{b}_{2}$	0.0010
${}^{1}\mathrm{a}_{3}$	0.0494	${}^{1}\mathrm{b}_{3}$	0.0064
${}^{2}\mathrm{a}_{1}$	26.1470	${}^{2}\mathrm{b}_{1}$	1.1791
${}^{2}\mathrm{a}_{2}$	1.6338	${}^{2}\mathrm{b}_{2}$	0.0632
${}^{2}\mathrm{a}_{3}$	0.3579	${}^{2}\mathrm{b}_{3}$	0.1567
${}^{3}\mathrm{a}_{1}$	-12.3359	${}^{3}\mathrm{b}_{1}$	-0.0679
${}^{3}\mathrm{a}_{2}$	-0.6417	${}^{3}\mathrm{b}_{2}$	-0.0283
${}^{3}\mathrm{a}_{3}$	-0.0273	${}^{3}\mathrm{b}_{3}$	-0.0075

(Equation 4.9)

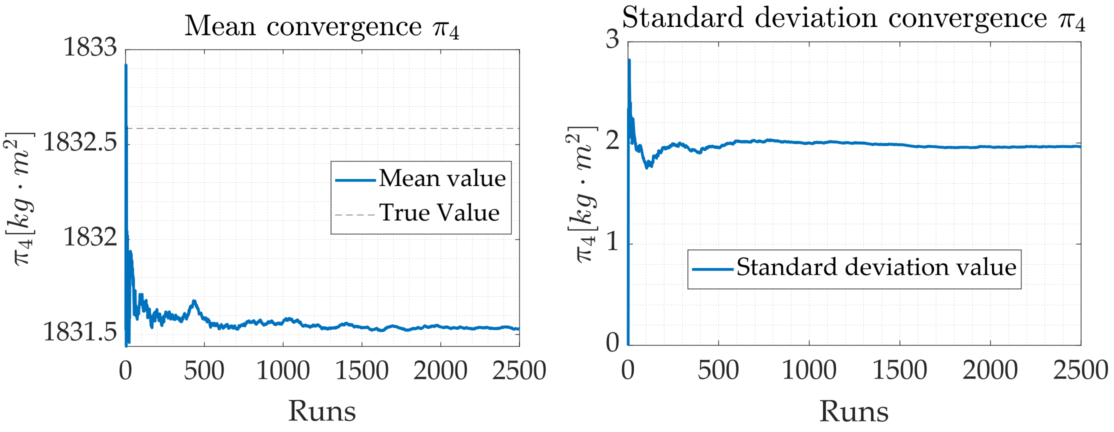
$$\begin{aligned}\boldsymbol{\pi} &= \left(\left(\hat{\mathbf{Y}}^T\right)\cdot\hat{\mathbf{Y}}\right)^{-1}\cdot\hat{\mathbf{Y}}^T\cdot\mathbf{H}_{rs}(t) = \\ &= \left(\left(\hat{\mathbf{Y}}^T\right)\cdot\hat{\mathbf{Y}}\right)^{-1}\cdot\hat{\mathbf{Y}}^T\cdot\left(\mathbf{H}_{tot}-\mathbf{R}_0(t)^{\mathbf{0}}\mathbf{H}_{RW/rs}\right)\end{aligned} \tag{4.9}$$

for which the quantity $\mathbf{H}_{rs}$ has to be evaluated at every time step (hence $\mathbf{H}_{rs}(t)$) since the base body attitude is changing in response to the exciting trajectory carried out by the robotic arm, in contrast to the previous framework where $\mathbf{H}$ was constant over time.



(a) Case Study 2: Joints Angles (b) Case Study 2: Joints Angular Rates

Figure 4.5: Case Study 2: Optimized Joints Trajectories



(a) Case Study 2: π_4 Mean Convergence (b) Case Study 2: π_4 Standard Deviation Convergence

Figure 4.6: Case Study 2: π_4 Convergence Analysis

For the sake of completeness the convergence analysis concerning one of the parameters clusters after having run the model in Simulink is shown in Figure 4.6, where it can be seen that almost full convergence is achieved in this case as well.

The only difference with the simulation model exploited in Section 4.2 is the formula to compute the base body angular velocity $\boldsymbol{\omega}_0$, due to the fact that to conserve the total angular momentum the term associated with the RW must be taken into account (Equation 4.10)

$$\boldsymbol{\omega}_0 = \tilde{\mathbf{M}}_b^{-1}\left(\mathbf{H}_{tot} - \mathbf{R}_0(t)^{\mathbf{0}}\mathbf{H}_{RW/rs} - \tilde{\mathbf{M}}_{bm}\dot{\mathbf{q}}\right) \tag{4.10}$$

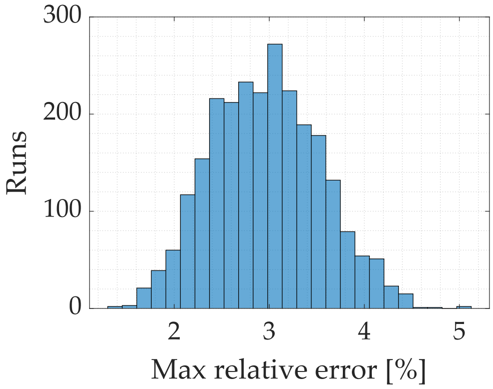


Figure 4.7: Case Study 2: Maximum Relative Error

Two additional charts are interesting in the results analysis: Figure 4.7 shows that the maximum relative error stays below 5% for almost all runs, whereas Figure 4.8 displays each parameter clusters relative error versus time (just the initial seconds of the trajectory). This to show that the algorithm needs a very small amount of time (less than 0.5 [s]) to bring the error below the 5% threshold, confirming what is stated in the work of Christidi et al. in [29], that just a little motion is required to perform the parametric identification. It has to be remarked that this chart was obtained by filling the regressor matrix with measurements without noise.

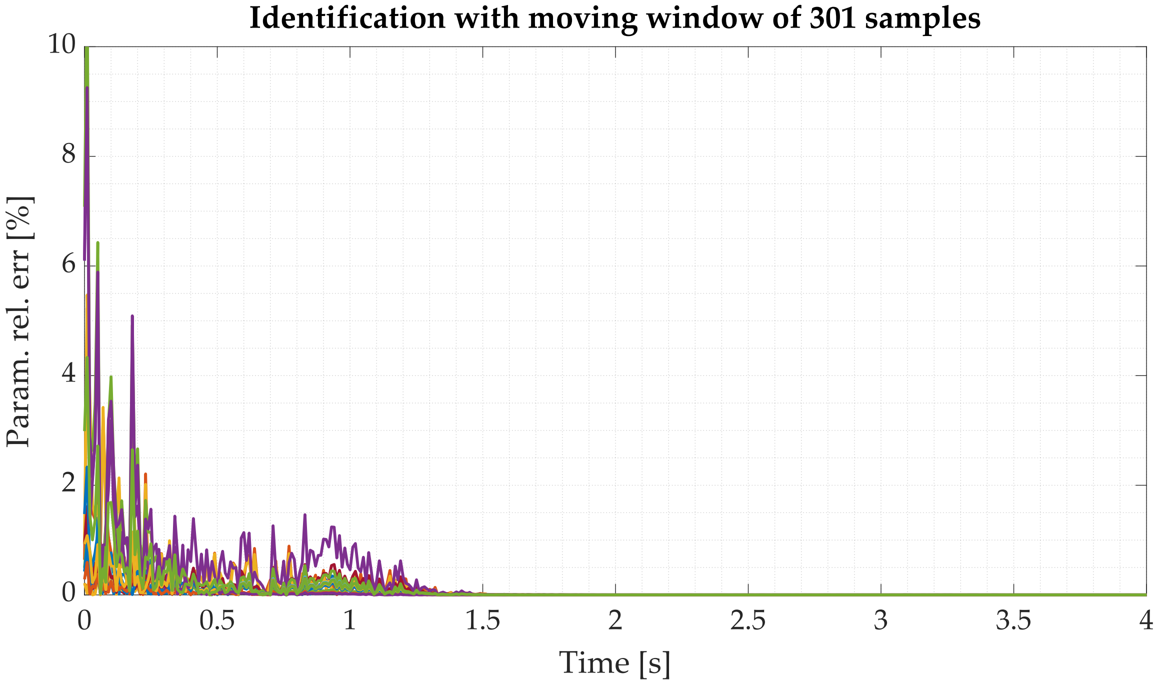


Figure 4.8: Case Study 2: Identification with Moving Window

Table 4.4 lists the results of the parametric identification for Case Study 2. As expected the highest value of the mean relative error ($\bar{\boldsymbol{\pi}}_{err}$) achieves a greater magnitude with respect to Case Study 1, where the condition number of the regressor matrix $\mathbf{Y}$ was almost half. Nonetheless $\bar{\boldsymbol{\pi}}_{err}$ remains below the 5% threshold for all the parameters, confirming the validity of the optimized exciting trajectory and of the methodology.
Case Study 2 is important to set the baseline for simulations that deal with motions starting from zero base body angular velocity $\boldsymbol{\omega}_0$ as will be the framework explained in Chapter 5.

Table 4.4: Case Study 2: Identification Results

$\boldsymbol{\pi}$	$\boldsymbol{\pi}_{true}$	$\bar{\boldsymbol{\pi}}_{est}$	$\boldsymbol{\pi}_{best}$	$\boldsymbol{\sigma}_{est}$	$\bar{\boldsymbol{\pi}}_{err}$ [%]
π_1	1832.5856	1838.5405	1837.1829	2.4447	-0.3249
π_2	-104.2654	-104.6941	-104.1749	1.4499	-0.4112
π_3	-154.0284	-156.2170	-154.3746	1.3371	-1.4209
π_4	1832.5856	1831.5298	1833.2639	1.9614	0.0576
π_5	-154.0284	-154.4942	-154.2327	1.9307	-0.3024
π_6	1708.5308	1708.3293	1708.8584	2.9204	0.0118
π_7	321.5317	312.1383	314.4503	1.9941	2.9214
π_8	321.5417	320.8746	321.1394	0.7577	0.2074
π_9	-255.9765	-254.5401	-254.9634	0.5381	0.5611
π_{10}	256.0265	257.0343	256.8073	0.5685	-0.3936
π_{11}	-65.4552	-65.5991	-65.7166	0.3348	-0.2199
π_{12}	65.5052	65.7464	65.7263	0.2277	-0.3682
π_{13}	142.1801	141.3774	141.6737	0.2745	0.5645
π_{14}	213.4479	211.1216	211.7739	0.6278	1.0899
π_{15}	142.1801	141.6325	141.7087	0.2485	0.3851
π_{16}	47.3934	47.9065	47.5560	0.3838	-1.0828
π_{17}	71.1493	70.9564	70.8480	0.3252	0.2711
π_{18}	47.3934	47.1950	47.0641	0.1591	0.4185
π_{19}	96.4455	96.2408	96.3529	0.2005	0.2123

5 | Novel Identification Framework And Simulation Results

5.1. Methodology Description

Systems such as the Astrobee free-flying robots (Figure 5.1) on the ISS do not have reaction wheels on board to provide a non zero angular momentum needed to perform the parametric identification of inertial parameters treated up until now. They can translate and rotate inside the space station thanks to a set of 12 nozzles which exhaust air accelerated by an electrically moved impeller, capable of guaranteeing holonomic control. Nonetheless they have a preferred forward motion and a preferred orientation with the top side facing up, as described by Smith et al. in [14].

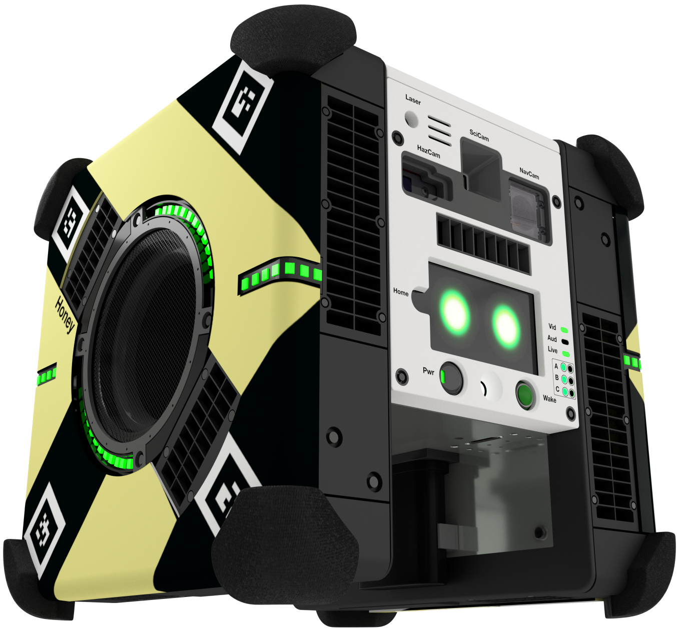


Figure 5.1: Render of the Astrobee Free-Flying Robot (*Credits:* NASA)

To treat this kind of framework for which it is an issue to obtain a non null angular momentum given the absence of on board reaction wheels, a different methodology is proposed. The idea is to exploit an object of known mass, inertia and center of mass position (thus all the inertial parameters, and from now on renamed as load) placed at the end-effector of a robotic arm, coupled with the conservation of linear and angular momenta. The movement of the known load at the end-effector would be granted by the torques applied to the robotic arm joints. On the Astrobee this could be achieved by installing on the platform a robotic arm like DLR's TINA (This Is Not an Arm) [39] with four DoF, positioning a load with known inertial properties at the end-effector.

The presence of the known load at the end-effector modifies the expression of the total angular momentum as in Equation 5.1

$$\mathbf{H}_{tot} = \mathbf{H}_{rs} + \mathbf{I}_L \boldsymbol{\omega}_L + M_L \mathbf{c}_L \times \mathbf{v}_L = \mathbf{0} \tag{5.1}$$

where $\mathbf{H}_{rs}$ is the angular momentum of the robotic spacecraft (namely the base body and the robotic arm), whereas the other two terms are given by the motion of the load at the end-effector. All the quantities are referred to an inertial frame with its origin positioned in the center of mass of the system[1].

The end-effector is assumed to have known inertial parameters, or to be substituted with a known load by using a suitable mechanical interface at the link joint (see for example [40]).

At this point the equations of conservation of the angular momentum can be written in regressor form since they are linear in the inertial parameters of the bodies composing the system. To tackle the rank deficiency of the regressor matrix and to obtain the minimal form of the inertial parameter clusters, the procedure described in Section 3.4 is exploited. It is then possible to rewrite Equation 5.1 as

$$\mathbf{H}_{tot} = \mathbf{Y}\left(\mathbf{d}, \boldsymbol{\omega}_0, \mathbf{q}, \dot{\mathbf{q}}\right) \boldsymbol{\pi} = \mathbf{0} \tag{5.2}$$

where $\mathbf{Y}$ is the new system regressor matrix and $\boldsymbol{\pi}$ is the vector of minimal inertial parameter clusters. It can be shown that $\mathbf{Y}$ is only dependent on the quaternions representing the base attitude, $\mathbf{d}$, the base body angular velocity, $\boldsymbol{\omega}_0$ and the joints angles and angular rates, $\mathbf{q}, \dot{\mathbf{q}}$. Given that the right-hand side of this equation is zero, it cannot be used in this form for identification with an ordinary least squares algorithm.

The principle of the presented method is to extract from the vector of parameter clusters $\boldsymbol{\pi}$ the terms composed only of the inertial parameters related to the known load. Once

[1] This is the same assumption made in subsection 3.4.1, for which in absence of external forces and moments the system center of mass position remains fixed in inertial space.

these are multiplied together with the associated kinematic expressions, made of measurable quantities and stored in the system regressor matrix, they form a part of the total angular momentum which is only due to the load and, for this reason, it is known:

$$\mathbf{H}_L = \mathbf{H}_L\left(\mathbf{d}, \boldsymbol{\omega}_0, \mathbf{q}, \dot{\mathbf{q}}, M_L, M_L{}^L\mathbf{c}_L, {}^L\mathbf{I}_L\right) \tag{5.3}$$

where the superscript L indicates that the quantity is referred to the reference frame of the known load. The known part of angular momentum is clearly not constant over time but varies during the manoeuvre. Equation 5.2 can now be reformulated as

$$\mathbf{Y}_{red}(t) \cdot \boldsymbol{\pi}_{red} = -\mathbf{H}_L(t), \tag{5.4}$$

where the subscript *red* represents the vector of inertial parameter clusters composed only of unknown values and the associated columns of the regressor matrix.

The regressor matrix of the reduced system and the vector of the angular momentum due to the known load can then be filled with measurements of the needed quantities taken at N time steps. If these are appended in a matrix and a vector of appropriate dimensions ($[3N, N_m]$ for the matrix, with N_m number of minimal inertial parameter clusters, $3N$ for the vector), Equation 5.4 forms a system of over-determined equations which can be solved with a simple ordinary least squares algorithm as

$$\boldsymbol{\pi}_{red} = \left(\hat{\mathbf{Y}}_{red}^T \hat{\mathbf{Y}}_{red}\right)^{-1} \hat{\mathbf{Y}}_{red}^T \left(-\hat{\mathbf{H}}_L\right) \tag{5.5}$$

where the superscript $(\hat{})$ refers to measured quantities. The full vector of parameter clusters $\boldsymbol{\pi}$ for the seven DoF space robot under study in this thesis is reported in Appendix A.

A 2D Example Consider a planar system of 3 DoF with 2 links and the known load replacing the end-effector. This system is used to evaluate all the kinematic quantities (positions and velocities), and all vectors are referred to center of mass of the system, computed by considering the masses of the robotic spacecraft and the known load at the end-effector. The procedure is formally identical to the one exploited in the case of non zero angular momentum (subsection 3.4.1): having obtained the linear combinations of parameters, the velocity of the base is evaluated from the conservation of linear momentum (equal to zero) and then substituted in the regressor matrix $\mathbf{Y}^{2D}$ to eliminate its dependency from it. At this point, the parameters and the kinematic quantities are re-arranged in the form presented in [12] and [28]. Finally, to complete the procedure, the term of the parameter clusters vector which contains only the inertia of the last link[2] (hence of the known load) can be extracted together with the related kinematic quantities to give Equation 5.6

$$H_L(t) = \left(I_L + M_L {}^L a_L^2 \left(1 - \frac{M_L}{M_{tot}} \right) \right) \cdot (\dot{q}_1(t) + \dot{q}_2(t) + \dot{q}_3(t) + \omega_0(t)) \tag{5.6}$$

in which the inertia, mass and position of the center of mass of the load (in its own reference frame) are assumed to be known as stated before, whereas the base body angular velocity and joint angular rates are given by on board sensors and compose the term extracted from the regressor matrix. The quantity H_L here evaluated is just a part of the angular momentum due to the known load, since the term composed by the cross product between the position of the center of mass of the load and its momentum is included in the robotic spacecraft contribution. This is due to the fact that the velocity of the base body was substituted in the regressor matrix to eliminate the system dependency from it.

5.1.1. Considerations On The Symbolic Implementation

It is worth mentioning at this point that the baseline research to test the developed algorithms only dealt with three DoF manipulators, for which a reference of the parameters clusters was given in [12], but not for the regressor matrix due to its complexity. As stated in Chapter 1, the goal of this thesis was to treat systems composed of robotic arms up to seven DoF, as the one employed on the OOS-SIM platform (described in Section 6.1). Using this same geometry allows to perform experimental tests and further verify the implemented identification procedure. Thus, the algorithm that computes the analytical expressions for $\mathbf{Y}$ and $\boldsymbol{\pi}$ in symbolic form had to deal with systems of increasing difficulty: in particular, every time a DoF was added to model the length of the kinematic expressions increased considerably, slowing down the symbolic computations and making MATLAB© built-in functions practically unusable to simplify the expressions. This happened in particular with the built-in routine `simplify` when dealing with trigonometric functions, since this function tries to rearrange them following known trigonometric relations which, applied to too long strings made of numerous characters, consistently reduces the computation speed to unbearable waiting times. For this reason this function could not be used anymore, which in turn created an issue when exploiting the MATLAB© tool to transform symbolic functions into numeric ones (`matlabfunction.m`): the software cannot handle expressions which contain too many characters. For instance, in the

[2] It has to be recalled that there is only one term related to inertia for every body in the system in a bidimensional case since the I_{xx} and I_{yy} are null.

case for the seven DoF robotic arm, the non-simplified kinematic expression of the last term of the regressor matrix had more than 500000 characters.
To overcome the problem the following strategy was pursued: after computing the matrices in the form of [13] all the symbolic trigonometric functions were substituted with normal symbolic variables, given that as mentioned above the expressions were too complex to perform trigonometric simplifications between them. Fortunately MATLAB© gives the possibility to assume additional rules and restrictions for its symbolic variables to ease their handling during the simplification process or for outputting the correct results when dealing with symbolic equations (e.g. discarding a negative solution for a variable for which the square root is requested, if one wants only the real result). For this specific application is was helpful to keep the trigonometric rule $\sin\vartheta^2 + \cos\vartheta^2 = 1$, since it was a recurrent situation in the expressions and thus it helped to reduce considerably their size. This proved to be a valid solution for the encountered issue, which permitted to obtain the sought analytical forms for the regressor matrix $\mathbf{Y}$ and the parameter clusters vector $\boldsymbol{\pi}$. Before performing the conversion from symbolic to numeric functions, the trigonometric functions were re-substituted given the fact the expressions were at this point already simplified.

Another shortstopper during the symbolic computations proved to be during the separation between clusters of inertial parameters and kinematic terms: the complexity of the latter did not allow to obtain a correct subdivision (e.g. the correct sign for the kinematic term). In this sense the author found to be a good practice to compute the system regressor form without the base body attitude at first (by substituting the base attitude with an identity matrix), since the resulting kinematic expressions were simplified and shorter and thus more manageable by MATLAB©, whereas the parameter clusters vector retained the same form. This procedure simplified the computations with the base body attitude since the form of $\boldsymbol{\pi}$ was already known at that point and the only quantity to be evaluated remained the full regressor matrix. Moreover, this method permitted to verify the results gathered with attitude against the ones obtained without it, by replacing the identity matrix in the former and thus checking that the two were identical.

5.2. Methodology Validation

To validate the methodology one should be able to use the identified parameter clusters to reconstruct the dynamics of the system under study given that they are a set minimal parameters. In particular, in the equation of conservation of angular momentum the composite inertia matrices (i.e. the one of the base body and the one related to the mixed terms of the base body and the robotic arm) should be written as the product of measurable quantities times the parameter cluster vector. In mathematical terms this translates into Equation 5.7

$$\tilde{\mathbf{M}}_b\left(\mathbf{d}, \mathbf{q}, \boldsymbol{\pi}\right)\boldsymbol{\omega}_0 + \tilde{\mathbf{M}}_{bm}\left(\mathbf{d}, \mathbf{q}, \boldsymbol{\pi}\right)\dot{\mathbf{q}} = \mathbf{0} \tag{5.7}$$

where the measurable quantities are the base body attitude and the joints angles. Of course the parameter clusters vector is the full vector containing also the inertial parameters of the known load.

All the information regarding the kinematic quantities are stored in the full regressor matrix $\mathbf{Y}$ in which the base body angular velocity $\boldsymbol{\omega}_0$ and the joints angular rates $\dot{\mathbf{q}}$ appear linearly in the complex kinematic equations (as can be also inferred by Equation 5.7). Thus, the regressor matrix shall be manipulated and then rearranged in a form in which the base body angular velocity and the joints angular rates are multiplied by a set of coefficients formed by trigonometric expressions of the base body attitude and the joints angles, in order for these coefficients to be extracted and consequently combined with the vector of inertial parameter clusters to reconstruct in this way the composite inertia matrices.

This recombination process is not trivial as it is in the work of Christidi et al. [12], since their dynamical model stems directly from this particular formulation where each term of the inertia matrices is composed by a measurable quantity times a known quantity (the inertial parameters). In this case the manipulation is done on the symbolic formulation of the terms of the regressor matrix to extract the coefficients of the base body angular velocity and of the joints angular rates. Given for example the seven DoF configuration, even though the kinematic expression are complex, the fact that the sought quantities appear linearly facilitates the process.

The main idea behind the validation process in the current framework is to exploit the equation of conservation of angular momentum with the composite inertia matrices built as explained above and thus dependant on the identified parameter clusters to reconstruct the base body angular velocity given a randomly parametrised joints angles and angular rates trajectory (Equation 5.8)

$$\boldsymbol{\omega}_0(t) = -\tilde{\mathbf{M}}_b^{-1}\left(\mathbf{d}(t), \mathbf{q}(t), \boldsymbol{\pi}\right)\tilde{\mathbf{M}}_{bm}\left(\mathbf{d}(t), \mathbf{q}(t), \boldsymbol{\pi}\right)\dot{\mathbf{q}}(t) \tag{5.8}$$

An applied example of the validation process is reported in subsection 5.3.2.

5.3. Case Study N1: Free-Floating Seven Degrees Of Freedom Robotic Arm

The methodology is tested with a case study (renamed N1) which takes into account a satellite equipped with a seven DoF robotic arm and a load with known inertial properties placed at the end-effector. The idea is to get closer to DLR's experimental facility setup: for this reason the inertias and the masses of the robotic arm resemble the ones of the LWR mounted on the OOS-SIM platform (described in Section 6.1). All the modified DH parameters and inertial properties of each body composing the system are reported in Table 5.1. As can be noticed from the Table, the satellite mass has been reduced of an order of magnitude with respect to the one used by Christidi et al. in [12] (i.e. 2000 [kg]) to obtain meaningful results. The mass and inertia of the load have been selected in relation to the base body inertial properties in order to be able to excite them during the trajectory with null total angular momentum. Indeed, the mass of the robotic arm used for this simulation (in theory the 14 [kg] of the LWR) is consistently less than the one used in the paper (namely 110 [kg]). Moreover, it can be seen that the seventh link represents the load with known inertial properties.

Table 5.1: Case Study N1: 7 DoF Simulation Model Parameters

Par.	**Unit**	**L0**	**L1**	**L2**	**L3**	**L4**	**L5**	**L6**	**L7**
a_{DH}	$[m]$	0	0	0	0	0	0	0	0
α_{DH}	$[rad]$	0	0	$\pi/2$	-$\pi/2$	-$\pi/2$	$\pi/2$	$\pi/2$	-$\pi/2$
d_{DH}	$[m]$	0	0.20	0	0.40	0	0.39	0	0.28
I_{xx}	$[kg\, m^2]$	27.3333	0.0232	0.0262	0.0142	0.0285	0.0261	0.0034	0.4000
I_{yy}	$[kg\, m^2]$	16.6667	0.0237	0.0053	0.0142	0.0049	0.0255	0.0030	0.4000
I_{zz}	$[kg\, m^2]$	22.6667	0.0050	0.0263	0.0049	0.0237	0.0041	0.0030	0.4000
M	$[kg]$	200	2.7082	2.7100	2.5374	2.5053	1.3028	1.5686	25
a_x	$[m]$	0	0	0	0	0	0	0	0
a_y	$[m]$	0	0	0.0420	0	-0.0435	0	0.0180	0
a_z	$[m]$	0	0.1409	0	0.2393	0	0.2126	0	0.1400
b_x	$[m]$	0.15	0	0	0	0	0	0	0
b_y	$[m]$	0.10	0	0.1580	0	-0.1565	0	-0.0180	0
b_z	$[m]$	0.5	0.591	0	-0.0393	0	-0.0226	0	0.1400

For what concerns the trajectory optimization process the following joints angles constraints are imposed:

$$\begin{array}{lll} -100^\circ \leq q_1 \leq 100^\circ & -50^\circ \leq q_2 \leq 50^\circ & -100^\circ \leq q_3 \leq 100^\circ \\ -50^\circ \leq q_4 \leq 50^\circ & -100^\circ \leq q_5 \leq 100^\circ & -90^\circ \leq q_6 \leq 90^\circ \\ -90^\circ \leq q_7 \leq 90^\circ & & \end{array}$$

together with the reported joints angular rates constraints:

$$\begin{array}{lll} -8^\circ/s \leq \dot{q}_1 \leq 8^\circ/s & -8^\circ/s \leq \dot{q}_2 \leq 8^\circ/s & -8^\circ/s \leq \dot{q}_3 \leq 8^\circ/s \\ -8^\circ/s \leq \dot{q}_4 \leq 8^\circ/s & -8^\circ/s \leq \dot{q}_5 \leq 8^\circ/s & -8^\circ/s \leq \dot{q}_6 \leq 8^\circ/s \\ -10^\circ/s \leq \dot{q}_7 \leq 10^\circ/s & & \end{array}$$

The pitch joints 2 and 4 are restricted to ±50° to avoid collisions between the load placed at the end-effector and the satellite external envelope, whereas pitch joint 6 is bounded between ±90° to avoid contact between the load and the robotic arm itself. The remaining roll joints are instead left with more freedom of movement since the links attached to them revolve around themselves and are thus less likely to cause a collision.

The limitations on the joints angular rates are instead all set to $\pm 8^\circ/s$ except the last joint, which is the one that effectively moves the load and it is thus allowed to rotate slightly faster. Nevertheless the constraints on the velocities are less than in [12], and are selected thinking to the safety of the equipment during the experimental tests.

The other free optimization parameters are the trajectory time and the sampling frequency: a trade-off analysis and the guidelines laid by the ESA *OBSIdian* study lead to select a total time of $t_f = 150$ [s] and a sampling frequency of $f_s = 3$ [Hz], resulting in N $=$ 451 samples to feed $\mathbf{Y}_{red}$ and $\mathbf{H}_L$.

The trajectory optimization for case study N1 considering the geometrical and inertial properties of the system (Table 5.1), and the constraints just discussed achieved a condition number of $cond(\mathbf{Y}_{red}) = 297$. It is almost four times higher than the one obtained in the baseline simulations for Case Study 2 (Section 4.4), but it has to be remembered that the model complexity has been increased from three to seven DoF and that the trajectory has been designed to keep a null angular momentum for all the time of the manoeuvre. This is in contrast to the mentioned case for which the system starts from zero angular velocities but then reacts to match and thus conserve the reaction wheels accumulated angular momentum.

Figure 5.2 shows the norm of the angular momentum for the robotic spacecraft (RS) and the part of the angular momentum due to the load, demonstrating that the total angular momentum remains null along the whole trajectory.

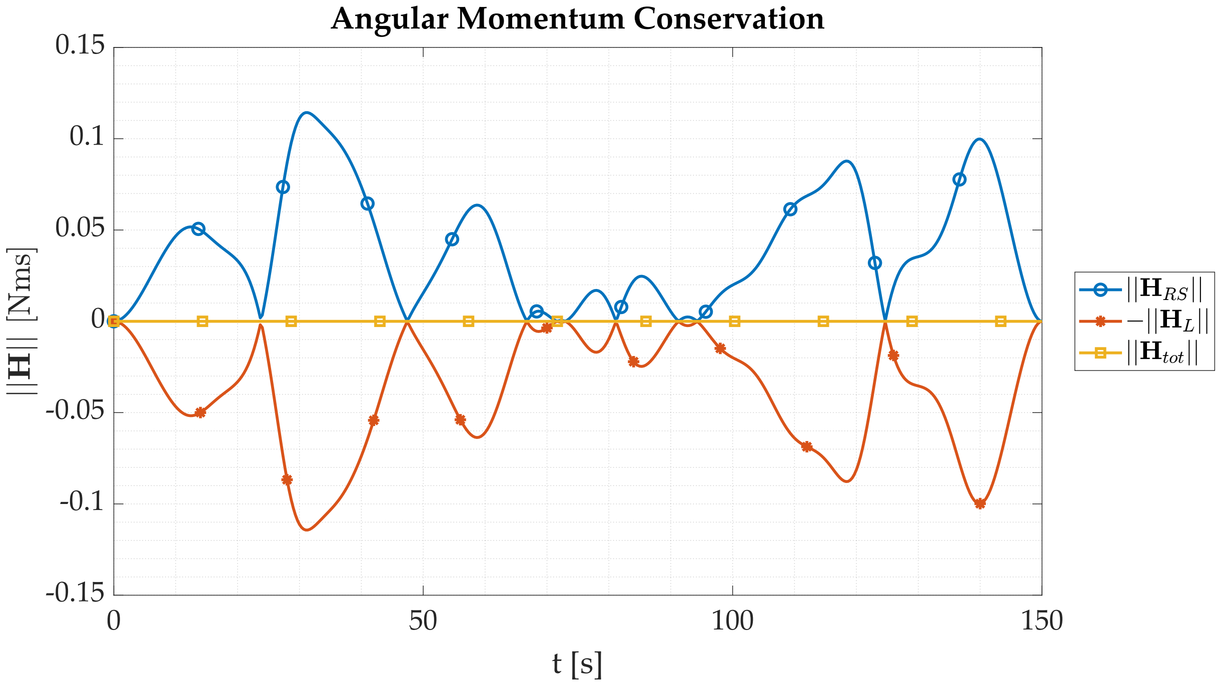


Figure 5.2: Case Study N1: Angular Momentum Norm

The Figure also shows that the norm of the angular momentum due to the load (or of the robotic spacecraft) remains always below 0.15 [Nms], which is two orders of magnitude less than the one considered for Case Study 2 (around 17 [Nms]). This helps to explain the difference in the condition number between the two cases: the motion constraints imposed to the system due to physical limitations put a boundary on the achievable excitation of the systems parameters, which can not be overcome since this framework considers zero total angular momentum as a constitutive hypothesis.

The optimized Fourier coefficients for the joints angles trajectories, resembling that four harmonics were employed for the truncated Fourier series which are used to parametrise them, are reported in Table 5.2.

Table 5.2: Case Study N1 Trajectory: Optimized Fourier Coefficients

$^{i}a_{j}$		$^{i}b_{j}$		$^{i}a_{j}$		$^{i}b_{j}$	
$^{1}a_{1}$	1.5328	$^{1}b_{1}$	1.9437	$^{5}a_{1}$	13.1082	$^{5}b_{1}$	-1.7408
$^{1}a_{2}$	0.0928	$^{1}b_{2}$	0.2102	$^{5}a_{2}$	0.8071	$^{5}b_{2}$	-0.2050
$^{1}a_{3}$	0.0260	$^{1}b_{3}$	0.0902	$^{5}a_{3}$	0.1403	$^{5}b_{3}$	-0.0684
$^{1}a_{4}$	-0.0339	$^{1}b_{4}$	0.1413	$^{5}a_{4}$	0.0862	$^{5}b_{4}$	-0.1280
$^{2}a_{1}$	-8.5861	$^{2}b_{1}$	1.2872	$^{6}a_{1}$	-1.7417	$^{6}b_{1}$	2.3577
$^{2}a_{2}$	-0.5231	$^{2}b_{2}$	0.1496	$^{6}a_{2}$	-0.0968	$^{6}b_{2}$	0.3367

${}^{2}\mathrm{a}_{3}$	-0.0831	${}^{2}\mathrm{b}_{3}$	0.0306	${}^{6}\mathrm{a}_{3}$	-0.0271	${}^{6}\mathrm{b}_{3}$	0.1165
${}^{2}\mathrm{a}_{4}$	-0.1050	${}^{2}\mathrm{b}_{4}$	0.1112	${}^{6}\mathrm{a}_{4}$	-0.0149	${}^{6}\mathrm{b}_{4}$	0.1395
${}^{3}\mathrm{a}_{1}$	-7.8434	${}^{3}\mathrm{b}_{1}$	-2.1984	${}^{7}\mathrm{a}_{1}$	5.3135	${}^{7}\mathrm{b}_{1}$	-1.5825
${}^{3}\mathrm{a}_{2}$	-0.4974	${}^{3}\mathrm{b}_{2}$	-0.3404	${}^{7}\mathrm{a}_{2}$	0.3507	${}^{7}\mathrm{b}_{2}$	-0.1588
${}^{3}\mathrm{a}_{3}$	-0.0907	${}^{3}\mathrm{b}_{3}$	-0.1061	${}^{7}\mathrm{a}_{3}$	0.0619	${}^{7}\mathrm{b}_{3}$	-0.2046
${}^{3}\mathrm{a}_{4}$	-0.0594	${}^{3}\mathrm{b}_{4}$	-0.1200	${}^{7}\mathrm{a}_{4}$	0.0381	${}^{7}\mathrm{b}_{4}$	-0.0275
${}^{4}\mathrm{a}_{1}$	-3.5324	${}^{4}\mathrm{b}_{1}$	1.8720				
${}^{4}\mathrm{a}_{2}$	-0.2196	${}^{4}\mathrm{b}_{2}$	0.2219				
${}^{4}\mathrm{a}_{3}$	-0.0410	${}^{4}\mathrm{b}_{3}$	0.0595				
${}^{4}\mathrm{a}_{4}$	-0.0306	${}^{4}\mathrm{b}_{4}$	0.1488				

The above coefficients generate the joints angles time histories shown in Figure 5.3 and their derivatives in Figure 5.4. From the first chart it can be appreciated that the joints angles trajectories respect the imposed constraints. Moreover the maxima and minima of each curve are close to the boundaries, meaning that the optimization algorithm worked properly by pushing the curves to their allowable limit while minimizing the condition number of the manoeuvre.

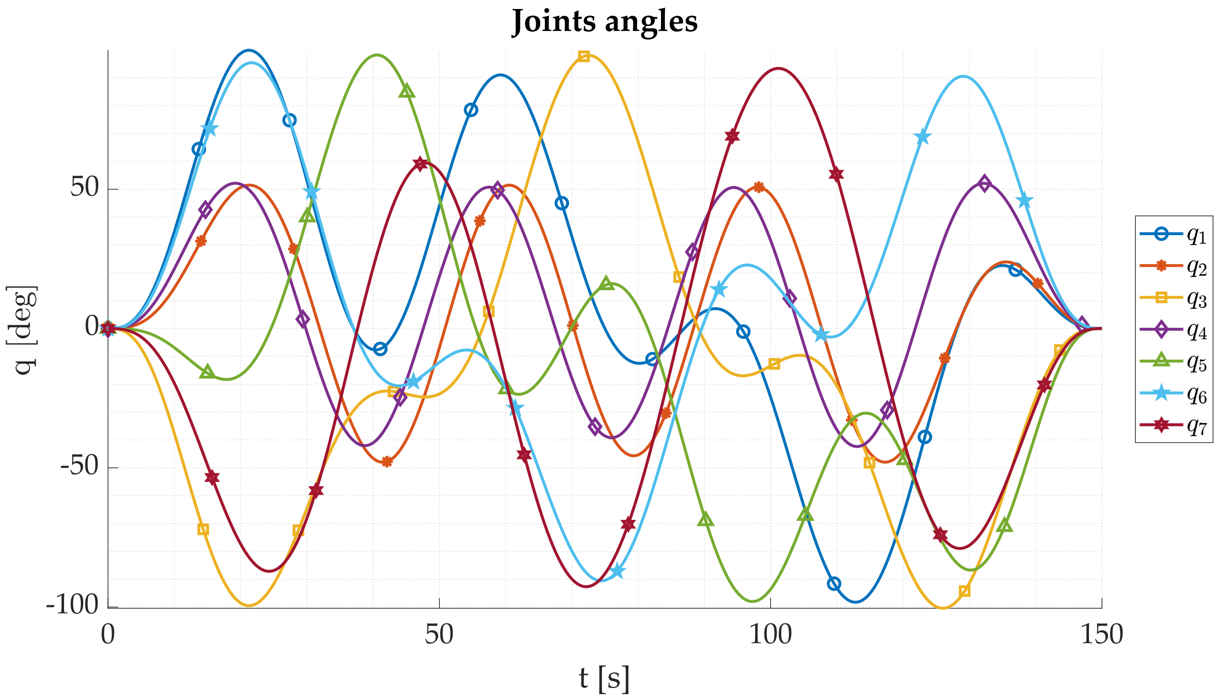


Figure 5.3: Case Study N1: Joints Angles

The same reasoning applies to the joints angular rates chart, where it can be clearly seen that the curves are within the limits and that the only joint which reaches $\pm 10°/s$ is the seventh one as mentioned above.

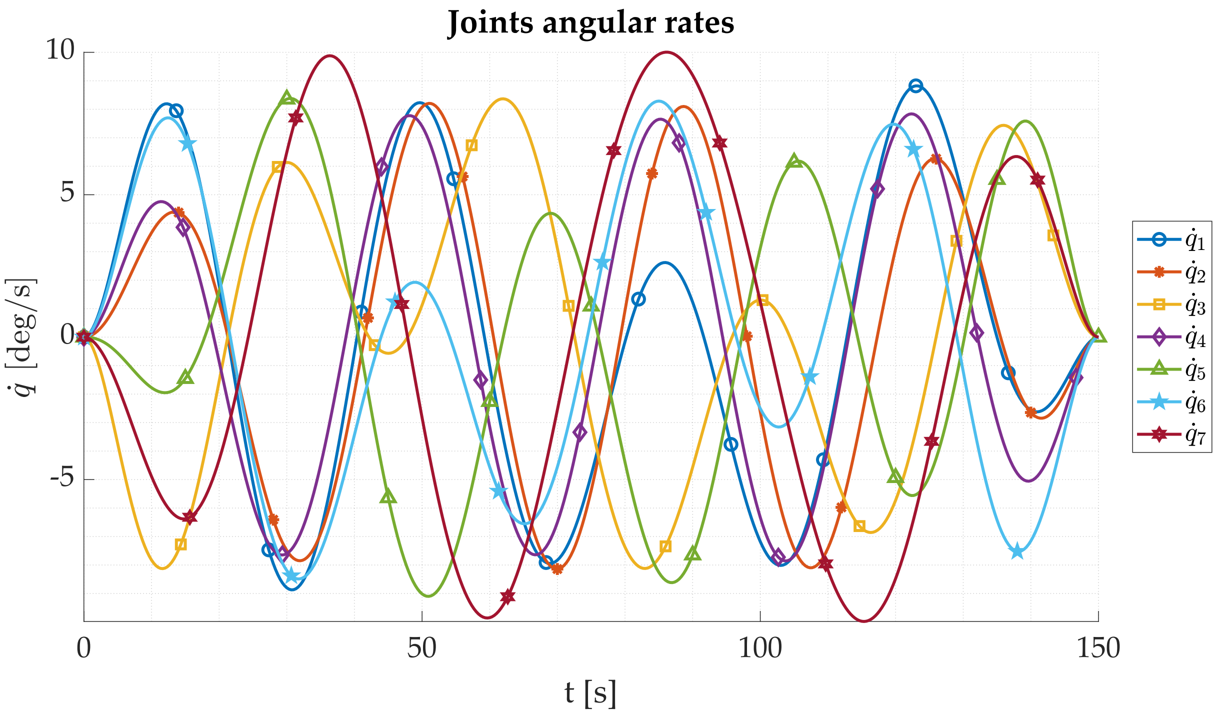


Figure 5.4: Case Study N1: Joints Angular Rates

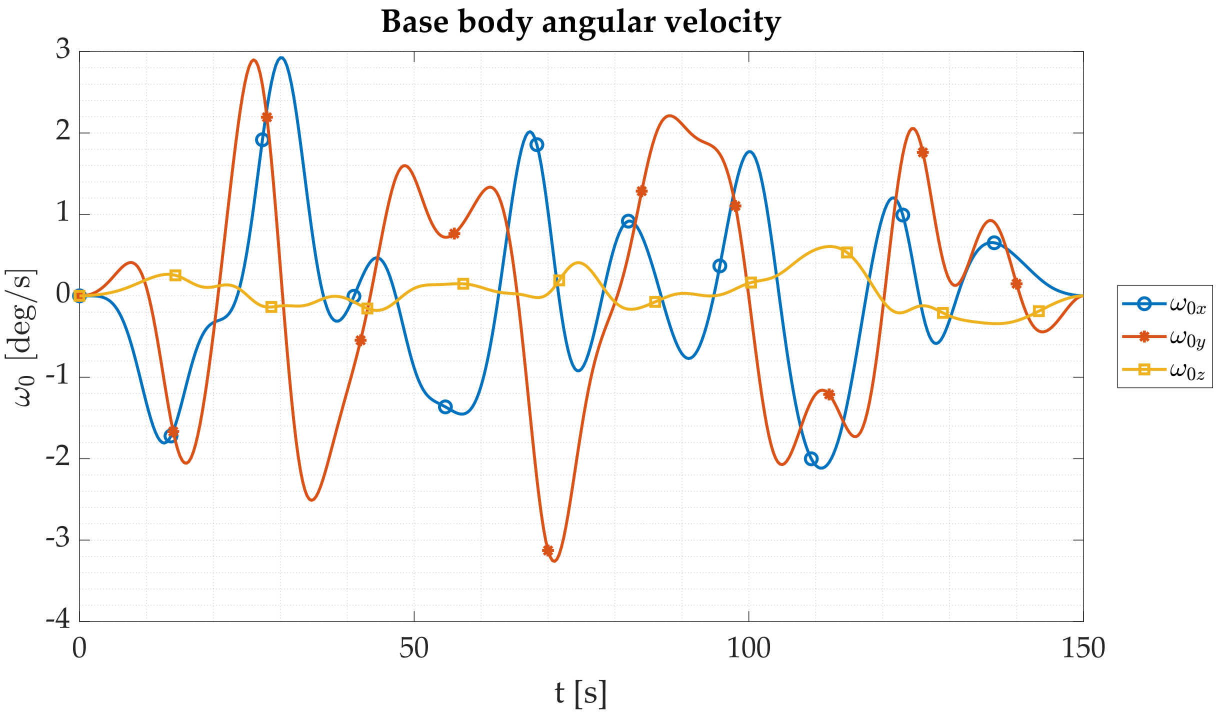


Figure 5.5: Case Study N1: Base Body Angular Velocity

These specific trajectories for the joints angles and angular rates result in the reaction of the base body which is governed by the free-floating dynamics, in order to maintain a null total angular momentum. Figure 5.5 displays the base angular velocities with respect to the inertial frame placed in the system center of mass. The component ω_{0z} and thus the rotation along the z axis is considerably lower than the other two components: this is because the rest position of the robotic arm lays along the z axis and, given the limits imposed on the first pitch joint and that the initial base body attitude starts from identity, the arm never fully extends in the $x-y$ plane, thus to keep a null angular momentum it has to react less in that direction. In any case all the components stay below ±3.5 [deg/s] and all the angular velocities are contained, which is good by thinking to an experimental setup.

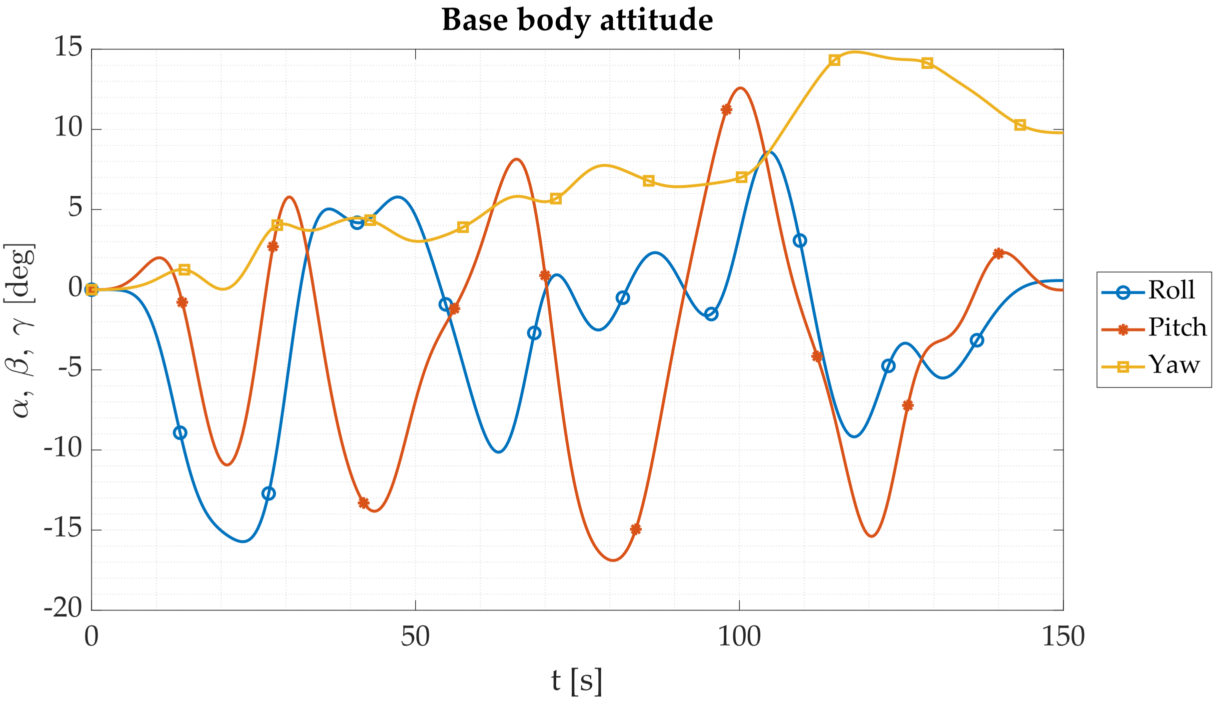
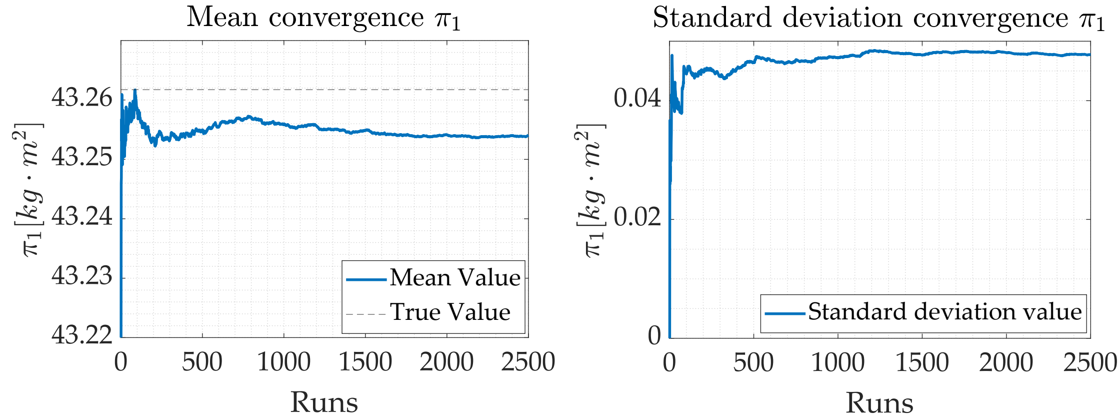


Figure 5.6: Case Study N1: Base Body Attitude

The chart in Figure 5.6 instead shows the base body attitude along the whole manoeuvre. The roll and yaw angles show a similar oscillating behaviour which is caused by the x and y angular velocity components (since the base body motion is limited), whereas the yaw angle presents a slightly increasing trend to be attributed mainly to the z component. Also in this case the base body angular motion is contained, which will be useful for the trajectories to be generated for the experimental tests, as will be explained in Section 6.1.

5.3.1. Identification Results

Given the optimized trajectory the same simulation model described in Section 4.1 is exploited to simulate the trajectory with the noise of the sensors. To be consistent with the results obtained in the baseline the noise parameters are kept equal.



(a) Case Study N1: π_1 Mean Convergence

(b) Case Study N1: π_1 Standard Deviation Convergence

Figure 5.7: Case Study N1: π_1 Convergence Analysis

The total number of simulations is 2500 as in the baseline study, always for comparison reasons. The convergence analysis for the first identified parameter cluster is shown for completeness in Figure 5.7: the charts show that both the mean and the standard deviation are almost fully settled to a constant value.

Table 5.3 reports the results of the identification process with the same formalism of Section 4.2 and Section 4.4. The full expressions for the parameter clusters for a seven DoF robotic arm with the given geometry are listed in Appendix A. In the list given π_{19} and π_{20} are the parameter clusters removed from the vector to form with the related columns of the regressor matrix the part of the angular momentum due to the load at the end-effector. This explains why Table 5.3 reports only 30 parameter clusters, given that those two are already known[3]. From the listed identification results it can be seen that the maximum relative error stays below 7%, and that the parameters clusters identified with the highest relative errors are the ones related to those with a smaller absolute value of at least two orders of magnitude less

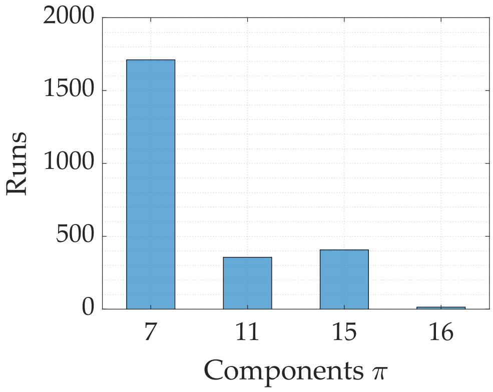


Figure 5.8: Case Study N1: π_i Components Identified With The Highest Relative Errors In Each Run

[3] Thus to be consistent with the list in Appendix A, starting from π_{19}, each identified cluster corresponds to the one in the list by adding 2 (e.g. π_{19} in the table corresponds at π_{21} in the list and so on).

than the rest. This is confirmed by Figure 5.8 which displays the components that are identified with the highest relative errors versus the whole 2500 runs of the noisy trajectory. These clusters (better explained in Appendix A) are the ones related to the roll joints, for which only the inertias are combined and, given the small values from the LWR real properties, this result in a worse identification performance with respect to the other clusters for which the relative error stays always below 1%.

Table 5.3: Case Study N1: Identification Results

$\boldsymbol{\pi}$	$\boldsymbol{\pi}_{true}$	$\bar{\boldsymbol{\pi}}_{est}$	$\boldsymbol{\pi}_{best}$	$\boldsymbol{\sigma}_{est}$	$\bar{\boldsymbol{\pi}}_{err}$ [%]
π_1	43.2617	43.2541	43.3370	0.0478	0.0177
π_2	-0.4825	-0.4824	-0.4855	0.0019	0.0175
π_3	-3.3574	-3.3581	-3.3678	0.0073	-0.0223
π_4	32.9972	32.9914	33.0517	0.0366	0.0178
π_5	-2.2383	-2.2377	-2.2395	0.0056	0.0247
π_6	23.7121	23.7045	23.7129	0.0169	0.0324
π_7	0.0049	0.0046	0.0050	0.0048	5.2472
π_8	0.0103	0.0105	0.0100	0.0007	-2.2973
π_9	4.6388	4.6374	4.6458	0.0048	0.0307
π_{10}	4.6442	4.6432	4.6542	0.0056	0.0217
π_{11}	0.0049	0.0046	0.0050	0.0017	6.1038
π_{12}	0.0099	0.0097	0.0092	0.0010	1.6258
π_{13}	3.8082	3.8080	3.8124	0.0040	0.0076
π_{14}	3.8083	3.8078	3.8118	0.0035	0.0149
π_{15}	0.0036	0.0039	0.0037	0.0013	-6.6727
π_{16}	0.0071	0.0072	0.0073	0.0008	-0.6595
π_{17}	0.8387	0.8385	0.8413	0.0013	0.0284
π_{18}	0.8413	0.8410	0.8437	0.0014	0.0362
π_{19}	1.1227	1.1224	1.1237	0.0023	0.0334
π_{20}	7.8681	7.8674	7.8840	0.0106	0.0089
π_{21}	1.6841	1.6839	1.6876	0.0024	0.0096
π_{22}	-0.9238	-0.9237	-0.9243	0.0018	0.0078
π_{23}	-6.4738	-6.4726	-6.4894	0.0094	0.0185
π_{24}	-1.3857	-1.3854	-1.3884	0.0022	0.0211
π_{25}	0.2961	0.2959	0.2948	0.0008	0.0581
π_{26}	2.0749	2.0751	2.0778	0.0032	-0.0101

π_{27}	0.4441	0.4440	0.4451	0.0010	0.0279
π_{28}	-3.7853	-3.7847	-3.7900	0.0046	0.0182
π_{29}	1.2132	1.2128	1.2141	0.0015	0.0339
π_{30}	-1.2130	-1.2126	-1.2155	0.0016	0.0340

Nevertheless, by reconstructing the base angular velocity as explained in Section 5.2, it can be seen from Figure 5.9 that these parameter clusters have very little influence on the dynamics, given that the absolute errors between the baseline angular velocity curve ($\boldsymbol{\omega}_0$) and the one computed using the mean of the identified parameter clusters ($\boldsymbol{\omega}_{0C}$) are in the order of 10^{-5} [deg/s].

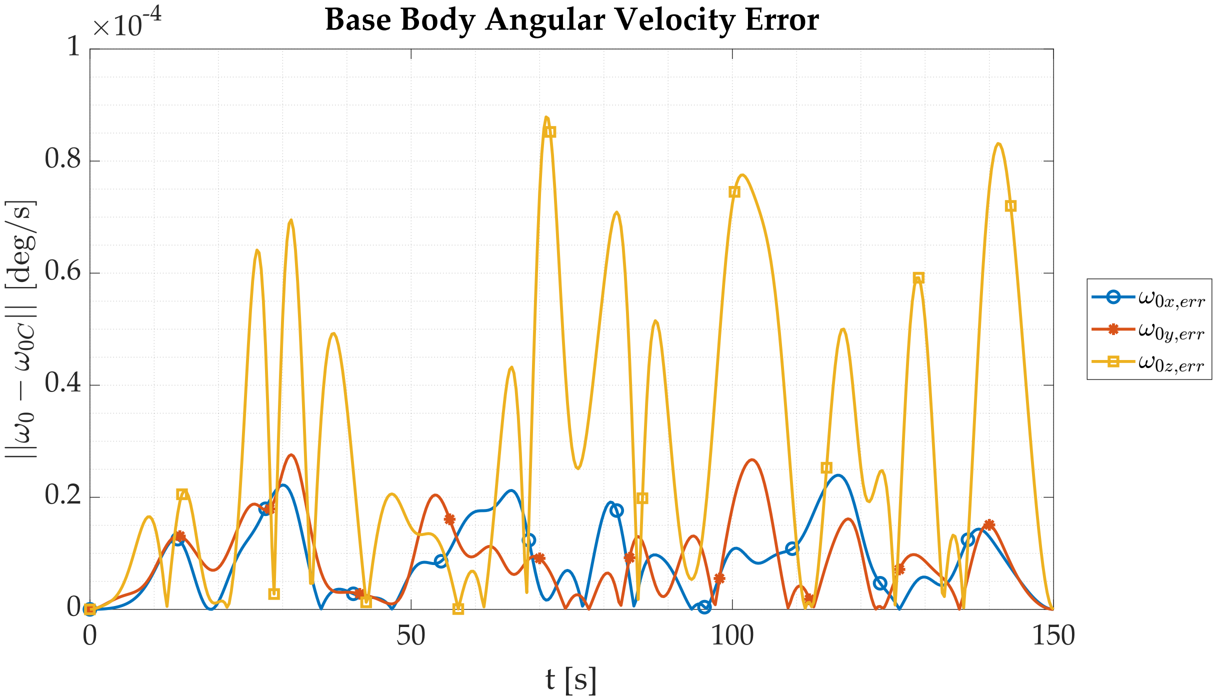


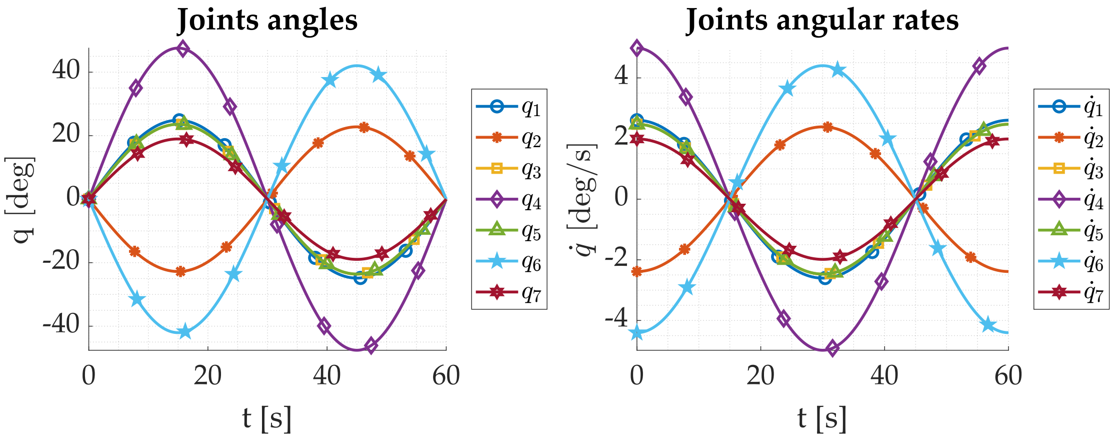
Figure 5.9: Case Study N1: Base Body Angular Velocity Error

5.3.2. Validation Tests

To further prove the validity of the methodology and of the identification process, random joint angles trajectories are generated with Equation 5.9

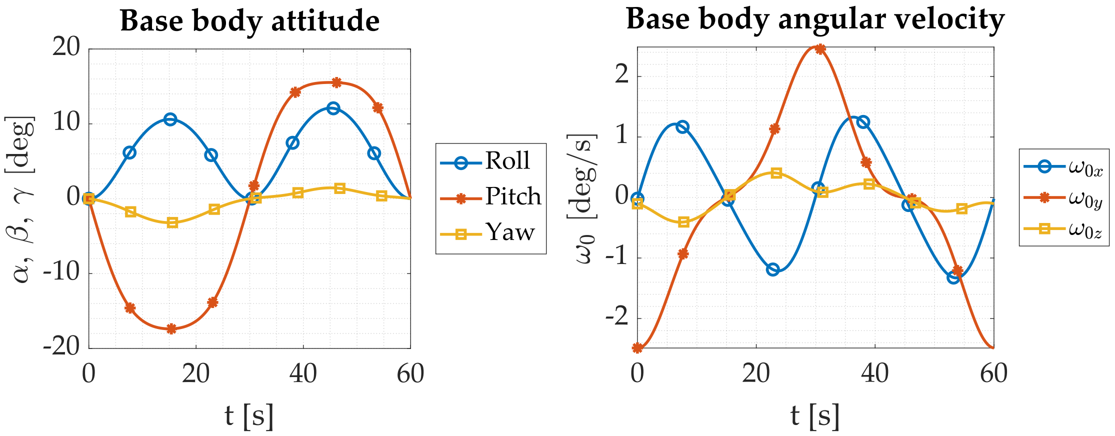
$$q_i\left(t\right) = A_{rand} \cdot \sin\left(\frac{2\pi t}{t_f}\right) \; [rad] \tag{5.9}$$

with which the base body angular velocity is reconstructed with Equation 5.8. An example of a randomly generated trajectory is given in Figure 5.10, together with the base body reaction to the robotic arm motion in Figure 5.11. All the trajectories are generated by considering a constant null total angular momentum. The base body angular velocity does not start necessarily with a null value since the condition of zero initial joint angular rates is not imposed, as it is clear by looking at Equation 3.16.



(a) Validation: Joints Angles (b) Validation: Joints Angular Rates

Figure 5.10: Validation: Joints Angles & Angular Rates



(a) Validation: Base Body Attitude (b) Validation: Base Body Angular Velocity

Figure 5.11: Validation: Base Body Attitude & Angular Velocity

The mean value of the identified parameter clusters vector (in Table 5.3, with the addition of the rows removed to build the angular momentum of the known load) is used to reconstruct the composite inertia matrices and thus the base body angular velocity. The tracked quantity is the maximum absolute value of the error between the base body angular velocity trajectory generated with the real parameters and the trajectory obtained with the said identified parameter clusters vector.

For the tests a total time of the manoeuvre $t_f = 60$ [s] and a sampling frequency of $f_s = 10$

[Hz] is selected, for a total number of simulations $n = 1000$. The results are reported in Figure 5.12, which shows the distribution of the absolute value of the maximum error for each component and the corresponding mean value.

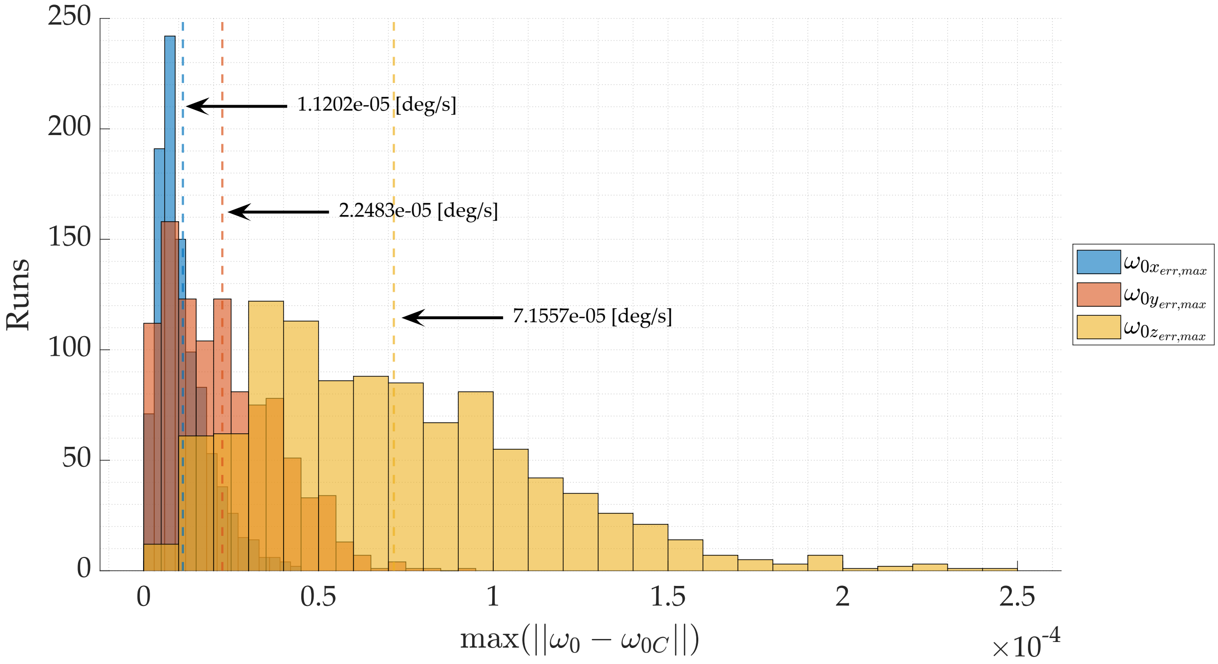


Figure 5.12: Validation: Distribution of the Absolute Value of the Maximum Error for the Base Body Angular Velocity

From the chart it can be seen that the z component is the one which is tracked with the lowest performances by the identified parameters clusters vector, followed by the y component and finally the x component. This can be attributed to the base body reaction to the robotic arm trajectory: given the joint angles constraints, there was less excitation of the base body inertial parameters in the z component, which was in turn reflected in poorer identification quality for the clusters describing the dynamics around that axis. In any case, the mean values reported on the chart are all in the order of magnitude of 10^{-5} [deg/s], which is consistent with the results obtained in Figure 5.9. This proves the validity of the methodology which is thus deemed reliable given the small error value in the reconstructed base body angular velocity.

6 Simulations On DLR Experimental Facility

6.1. OOS-SIM Description

The DLR's OOS-SIM (On Orbit Servicing - Simulator) testing facility in the Institute of Robotics and Mechatronics in Oberpfaffenhofen is composed by a chaser and a target, to reproduce in-orbit capture scenarios with the aid of a seven DoF robotic arm (the LWR) and cameras/LiDAR mounted on the chaser [41], as shown in Figure 6.1. Both the satellite mock-ups are mounted on six DoF industrial robots which are position controlled.

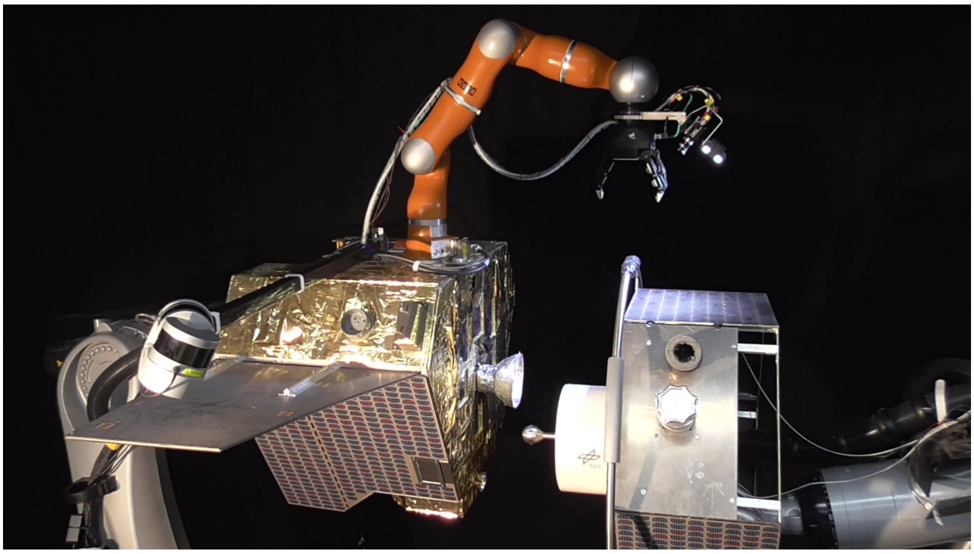


Figure 6.1: OOS-SIM in DLR Institute of Robotics and Mechatronics, (*Credits*: DLR)

The simulation which runs in the experimental facility relies on a mathematical model that reproduces the free-floating or free-flying dynamics which govern the two spacecrafts motion in orbit. Since it is a mathematical framework, the masses and inertias of the model can be set accordingly to the type of experiment which has to be performed, and do not need to be representative of the physical components of the system.

This is a great advantage for the current thesis work, since it permits to test different combinations of base body mass - mass at the end-effector and see the effects on the condition number of the regressor matrix. Unfortunately, as all physical systems also the one under study has mechanical constraints which can not be overcome as in simulations, where these can be relaxed and thus reproduce with more freedom the motion of the free-floating satellite. In particular, the most stringent constraints are the ones on the chaser satellite orientation, given below

$$\begin{cases} -40\ [deg] \leq & \mathrm{x-rotation\ (roll)} & \leq 40\ [deg] \\ -20\ [deg] \leq & \mathrm{y-rotation\ (pitch)} & \leq 15\ [deg] \\ -15\ [deg] \leq & \mathrm{z-rotation\ (yaw)} & \leq 10\ [deg] \end{cases} \tag{6.1}$$

This imposes a strong limitation on the allowable motion of the base body, making the optimization process even more complex. For example choosing a high mass ratio between the satellite and the couple robotic arm - known load reduces the satellite attitude changes, but in return worsens the condition number, since the base body parameters are not properly excited. On the contrary, seeking for a lower condition number by reducing the aforementioned mass ratio leads to a violation of such constraints because of the free-floating dynamics governing the problem.

6.2. OOS-SIM: IMU Calibration

To perform experiments on the DLR's OOS-SIM, the IMU mounted on the chaser satellite shall be calibrated first. The work presented in [42] outlines a simple calibration algorithm that does not require external equipment to complete this kind of procedure. This also due to the fact that the IMU is already mounted on the facility, making it difficult to place it on dedicated machines.

The non-availability of the factory calibration parameters makes it more challenging since there is not a precise reference to check results with, but the validity of the calibrated measurements depends on the confidence with which the magnitude of the local gravity is retrieved. In any case, the authors of the paper provide a set of data with which the algorithm can be tested.

Table 6.1: Calibration Parameters

Accelerometers		**Gyroscopes**	
α_{yz}	0.0004	γ_{yz}	-0.0003
α_{zy}	-0.0051	γ_{zy}	-0.0262
α_{zx}	0.0094	γ_{xz}	0.0043
s_x^a	1.0136	γ_{zx}	-0.0029
s_y^a	1.0128	γ_{xy}	0.0077
s_z^a	1.0305	γ_{yx}	0.0321
b_x^a	0.0020	s_x^g	1.0159
b_y^a	0.0633	s_y^g	1.0255
b_z^a	-0.2803	s_z^g	0.8808

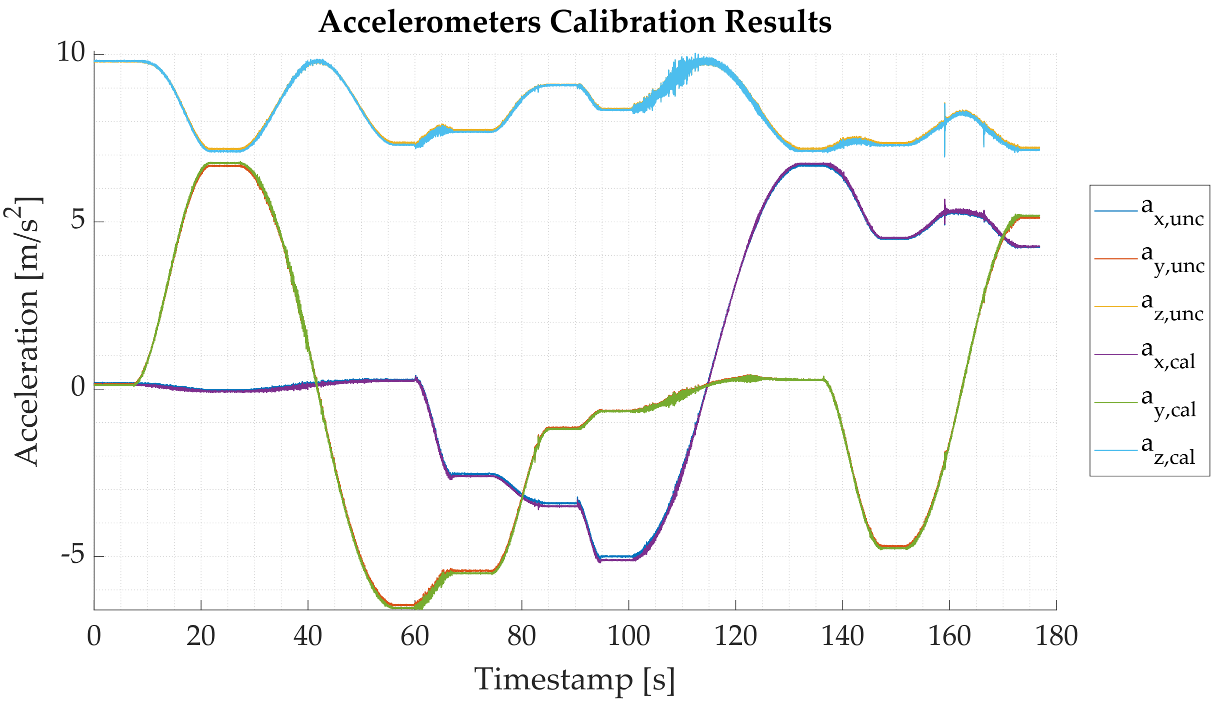


Figure 6.2: Uncalibrated vs Calibrated Accelerometers Measurements

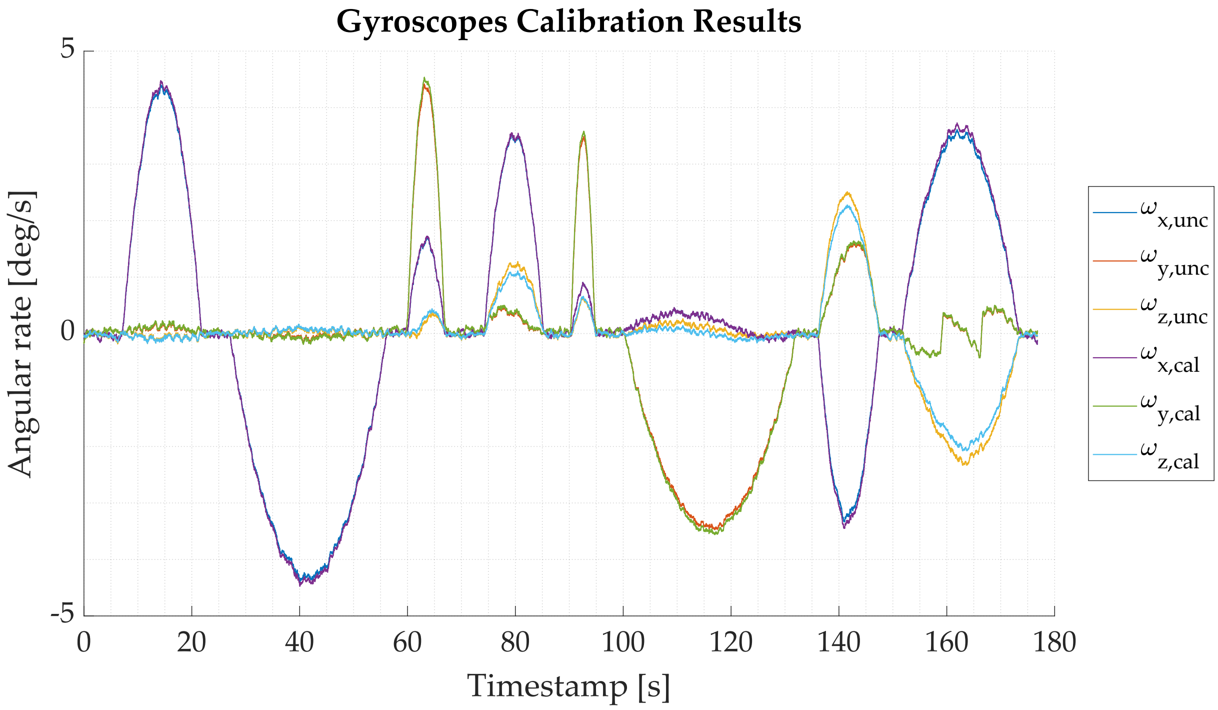


Figure 6.3: Uncalibrated vs Calibrated Gyroscopes Measurements

The calibration pseudo-algorithm and the sensors (accelerometers and gyroscopes) error models can be found in [42], whereas the calibration parameters are reported in Table 6.1. Figure 6.2 and Figure 6.3 display the results of the calibration procedure, both for the accelerometers and the gyroscopes.

6.3. Required Measurements For The Identification Experiments

The data needed to fill the regressor matrix is reported in Table 6.2, in which the distinction between measured and computed quantities is underlined.

Table 6.2: Regressor Matrix Required Measurements

Quantity	Units	Measured	Computed
Base body attitude	[rad]	✗	✓
Base body angular velocity	[rad/s]	✓	✗
Joints positions	[rad]	✓	✗
Joints angular rates	[rad/s]	✗	✓

It has to be recalled that the regressor matrix has to be filled with measurements of the base body angular velocity in the inertial frame, thus the data retrieved from the gyroscopes will have to be rotated in that reference frame in order to make the algorithm work properly.

In the absence of other sensors to measure the attitude of the base body, quaternion integration will be exploited, having the knowledge of the base body angular velocity in the base body frame[1] (thus directly from the IMU gyroscopes) as in Equation 6.2

$$\dot{\mathbf{q}} = \frac{1}{2} \cdot \boldsymbol{\Omega}(\boldsymbol{\omega}) \cdot \mathbf{q} \qquad \text{with} \qquad \boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix} \tag{6.2}$$

The formulation given in Equation 6.2 considers an attitude quaternion for which the scalar part is placed in the fourth component of the vector (JPL notation).

For what concerns the joints positions and joints angular rates, the first ones are directly recovered from the motors encoders, while the latter have to be obtained via numerical differentiation.

Signal processing has to be performed on every set of measurements, in particular on the

[1] This it is typically done in AOCS simulations, which employ a mathematical model to retrieve this quantity and use it as "reference" for control purposes.

ones that are not obtained in a direct manner, thus the base body attitude and joints angular rates, since numerical integration and derivation propagate the intrinsic noise of the "parent" sensors (i.e. the gyroscopes and the motor encoders).

6.3.1. Signals Processing

The data retrieved from sensors have different sampling frequencies, namely around 200 [Hz] for IMU gyroscopes and 1000 [Hz] for the joints motors encoders. This means that the latter have to be downsampled in order to be used to fill the regressor matrix $\mathbf{Y}_{red}$ and be consistent with the other data. Moreover, following the guidelines given by the ESA *OBSIdian* studies [11], providing data at 2-3 [Hz] should be sufficient to obtain good identification results, thus all the signals shall be further downsampled to meet this specific requirement.

Given an intrinsic physical delay between the start of the manoeuvre on the OOS-SIM and the data gathering (which begins some seconds prior to the actual motion and stops some seconds after), to confront the simulations with the experimental curves all the signals have to be cropped to retain only the effective movement section. This is the first step in the signals conditioning, which is then followed by filtering the measurements of gyroscopes and motors encoders with low pass filters. In particular, for the first variable the cutting frequency is set 0.2 [Hz], whereas for the second one the chosen cutting frequency is 0.05 [Hz]. These values are selected after performing Fourier analysis on the signals with Fast Fourier Transforms (FFT), knowing their sampling frequency.

At this point the base body attitude values can be obtained by exploiting the gyroscope measurements, as explained in the previous section, via quaternion integration, through a Runge-Kutta integration scheme of the fourth order with normalization at every step to keep the quaternion unitary norm, from which the direct cosine matrix derivation is straightforward.

To compute the joints angular rates, instead, the MATLAB© function `gradient` is used to numerically differentiate the joints angles. This function was selected over the `diff` function since it the keeps the output vector of the same length as the original one, in contrast to the latter which shortens the resulting vector by one. Given that numerical differentiation amplifies the intrinsic noise of the parent data, low pass filtering is applied also to the joint angular rates with the same cutting frequency window employed for the joints angles.

The final step is to downsample all the data on an equally spaced time vector of a given frequency (which in this case it is selected to be 3 [Hz]): the data can now be used to fill the regressor matrix and to perform the identification task.

6.4. Trajectory Optimization For OOS-SIM Experiments

The optimization process for the experiments to be run on the OOS-SIM facility has to take into account tighter constraints (e.g. on the base body attitude), as was briefly introduced in Section 6.1. Limiting the base motion to satisfy the mechanical capabilities of the experimental set-up should worsen the condition number of the regressor matrix and make the optimization algorithm struggle to find a feasible solution. This happens because the software is given contrasting requests: excite the system to perform the identification of the inertial parameters while keeping the movements contained. Some other difficulties have to be taken into account for the experimental setup, which are not present in simulation: a camera is mounted on the LWR gripper to guide the in-orbit capture manoeuvres, and cables connect it to a power source as well as cables to transmit images to the facility hardware (see Figure 6.1). The motion of the robotic arm shall be limited not to entangle the said cables and break them during the motion of the servicer mock-up. The constraints on the joints angles are reported below for completeness (noting that some of them are well below the actual capabilities of the LWR of 120°):

$$\begin{aligned}
-90^\circ &\leq q_1 \leq 90^\circ & -40^\circ &\leq q_2 \leq 40^\circ & -97^\circ &\leq q_3 \leq 97^\circ \\
-40^\circ &\leq q_4 \leq 40^\circ & -97^\circ &\leq q_5 \leq 97^\circ & -65^\circ &\leq q_6 \leq 65^\circ \\
-90^\circ &\leq q_7 \leq 90^\circ
\end{aligned}$$

In particular, the constraints on the pitch joints 2 and 4 is set to 40° to avoid collisions with the base body, whereas the pitch joint 6 is set to 65° degrees to avoid contact between the camera on the gripper and the arm itself. Instead the limits on the roll joints (1-3-5-7) are more relaxed and are set in order to allow a reasonable movement of the camera cables without weaving them.

6.5. Experimental Results

To run experiments on the OOS-SIM the dynamic model had to be simplified to make it compatible with the algorithms currently running on it. The simulation method is based on a Lagrange-Poincaré formulation of the equations of motion, for which noisy joint acceleration measurements are avoided in the computation of the spacecraft motion due to manipulator interaction [43]. The resulting physical consistency is to the advantage for the parameter identification. The current development of this methodology requires that the links masses are lumped in the joints, resulting in a slightly different dynamical model from the one employed in the simulations. This means under a practical point of view that the space manipulator input files had to be simplified since the algorithms have not yet been tested for a complete geometrical and dynamical description of the satellite and its robotic arm.

As said above, the major simplification to be made to be compliant with the simulation software is to lump all the link masses in the joints, since the program is not yet capable (meaning that its validity has not been proven yet) of dealing with non-null links centers of mass. Moreover, the inertia of each link have to be expressed with respect to the link reference frame (where all the mass is now concentrated) instead of the classical link center of mass.

The last thing to be mentioned is that the input files used in simulation follow the formalism of the SpaceDyn library [44], whereas the newer algorithms running on the OOS-SIM require `.urdf` files. The conversion between one input file to the other proved to be not trivial, mainly because of how the end-effector description is treated in `.urdf`, a fundamental point in this methodology since it is where the known load is placed.

The geometrical and dynamical parameters used for the model are reported in Table 6.3, where the modified DH parameters are firstly reported and followed by the dynamical parameters which are namely the inertia, mass and position of the center of mass. As discussed in Chapter 3 the products of inertia and two out of three centers of mass positions are set to zero to simplify the problem.

Table 6.3: Experimental Model Parameters

Par.	**Unit**	**L0**	**L1**	**L2**	**L3**	**L4**	**L5**	**L6**	**L7**
a_{DH}	$[m]$	0	0	0	0	0	0	0	0
α_{DH}	$[rad]$	0	0	$\pi/2$	-$\pi/2$	-$\pi/2$	$\pi/2$	$\pi/2$	-$\pi/2$
d_{DH}	$[m]$	0	0.20	0	0.40	0	0.39	0	0.28
I_{xx}	$[kg \cdot m^2]$	27.3333	0.0770	0.0310	0.1595	0.0333	0.0850	0.0039	0.8900
I_{yy}	$[kg \cdot m^2]$	16.6667	0.0774	0.0053	0.1595	0.0049	0.0844	0.0030	0.8900
I_{zz}	$[kg \cdot m^2]$	22.6667	0.0050	0.0311	0.0049	0.0284	0.0041	0.0035	0.4000
M	$[kg]$	200	2.7082	2.7100	2.5374	2.5053	1.3028	1.5686	25
a_x	$[m]$	0	0	0	0	0	0	0	0
a_y	$[m]$	0	0	0	0	0	0	0	0
a_z	$[m]$	0	0	0	0	0	0	0	0
b_x	$[m]$	0.15	0	0	0	0	0	0	0
b_y	$[m]$	0.10	0	0.20	0	-0.20	0	0.28	0

b_z	$[m]$	0.5	0.20	0	0.20	0	0.19	0	0

6.5.1. Optimized Trajectory For OOS-SIM

The optimized trajectory for the experimental test, obtained by using the simplified model presented above, achieved a condition number $cond(\mathbf{Y}_{red}) = 288$. The joints angles trajectory and angular rates are shown in Figure 6.4 and Figure 6.5, whereas the optimized Fourier coefficients are reported in Table 6.4.

Table 6.4: Experimental Trajectory: Optimized Fourier Coefficients

${}^{i}a_{j}$		${}^{i}b_{j}$		${}^{i}a_{j}$		${}^{i}b_{j}$	
${}^{1}a_{1}$	0.9036	${}^{1}b_{1}$	0.4098	${}^{5}a_{1}$	-1.8812	${}^{5}b_{1}$	-0.3846
${}^{1}a_{2}$	0.0361	${}^{1}b_{2}$	0.0293	${}^{5}a_{2}$	-0.1135	${}^{5}b_{2}$	-0.1056
${}^{1}a_{3}$	0.0705	${}^{1}b_{3}$	0.1111	${}^{5}a_{3}$	-0.0028	${}^{5}b_{3}$	-0.0641
${}^{1}a_{4}$	-0.0037	${}^{1}b_{4}$	-0.0280	${}^{5}a_{4}$	-0.0553	${}^{5}b_{4}$	0.0248
${}^{2}a_{1}$	0.6283	${}^{2}b_{1}$	-0.9730	${}^{6}a_{1}$	-0.1911	${}^{6}b_{1}$	0.5028
${}^{2}a_{2}$	0.0204	${}^{2}b_{2}$	-0.1126	${}^{6}a_{2}$	-0.0453	${}^{6}b_{2}$	0.0747
${}^{2}a_{3}$	-0.0298	${}^{2}b_{3}$	-0.0077	${}^{6}a_{3}$	0.0864	${}^{6}b_{3}$	0.0106
${}^{2}a_{4}$	0.0676	${}^{2}b_{4}$	-0.0952	${}^{6}a_{4}$	-0.0355	${}^{6}b_{4}$	0.0385
${}^{3}a_{1}$	2.5221	${}^{3}b_{1}$	0.5771	${}^{7}a_{1}$	-1.4211	${}^{7}b_{1}$	-0.1959
${}^{3}a_{2}$	0.1664	${}^{3}b_{2}$	0.1561	${}^{7}a_{2}$	-0.0047	${}^{7}b_{2}$	0.0195
${}^{3}a_{3}$	0.0320	${}^{3}b_{3}$	-0.0167	${}^{7}a_{3}$	-0.0612	${}^{7}b_{3}$	-0.0397
${}^{3}a_{4}$	0.0369	${}^{3}b_{4}$	0.0364	${}^{7}a_{4}$	-0.0658	${}^{7}b_{4}$	-0.0042
${}^{4}a_{1}$	-0.6289	${}^{4}b_{1}$	-0.6272				
${}^{4}a_{2}$	-0.0352	${}^{4}b_{2}$	-0.0825				
${}^{4}a_{3}$	-0.0732	${}^{4}b_{3}$	-0.0641				
${}^{4}a_{4}$	0.0538	${}^{4}b_{4}$	-0.0153				

As can be seen from the joints angles charts the most stringent constraints, namely the ones on the pitch joints, are respected. For what concerns the joints rates instead the maximum value stays always below 10 [deg/s] which is good for the safety of the experiment itself, since the situation of the cables can be checked and monitored during the trajectory given that the system is moving at a slow pace. This is also confirmed by the computed base body angular velocity (Figure 6.6), which remains below 4 [deg/s] in all components along the whole trajectory. Finally Figure 6.7 displays the base body RPY

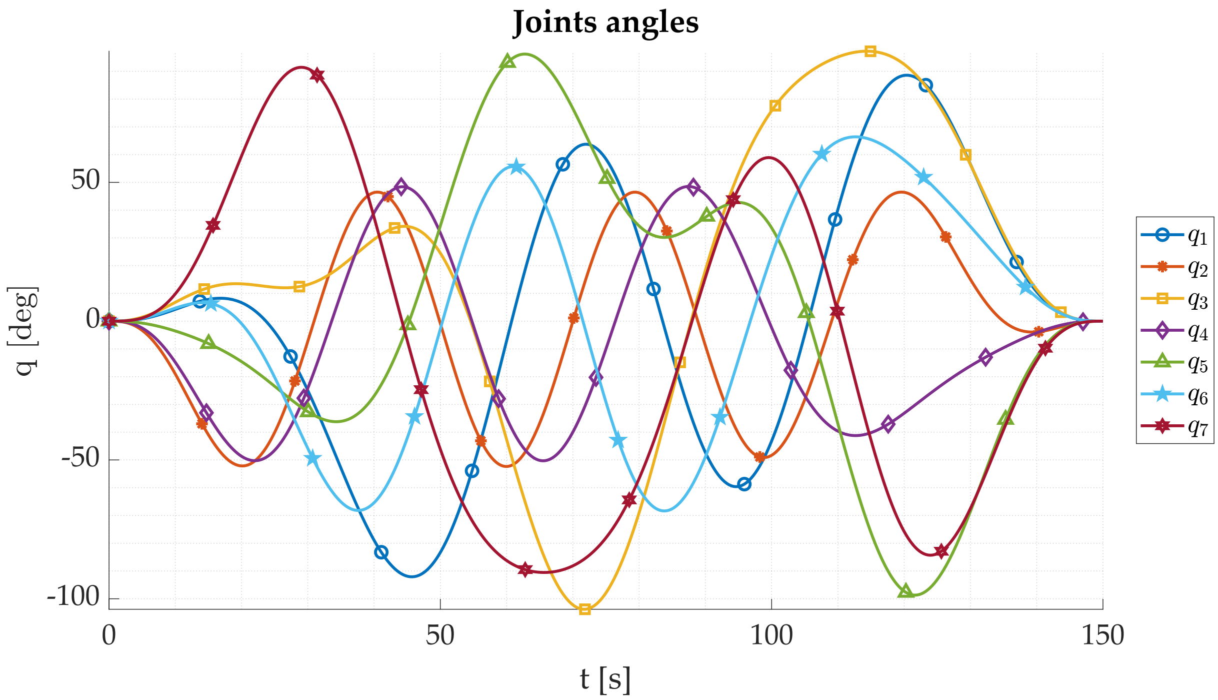


Figure 6.4: Joints Angles For Experimental Test

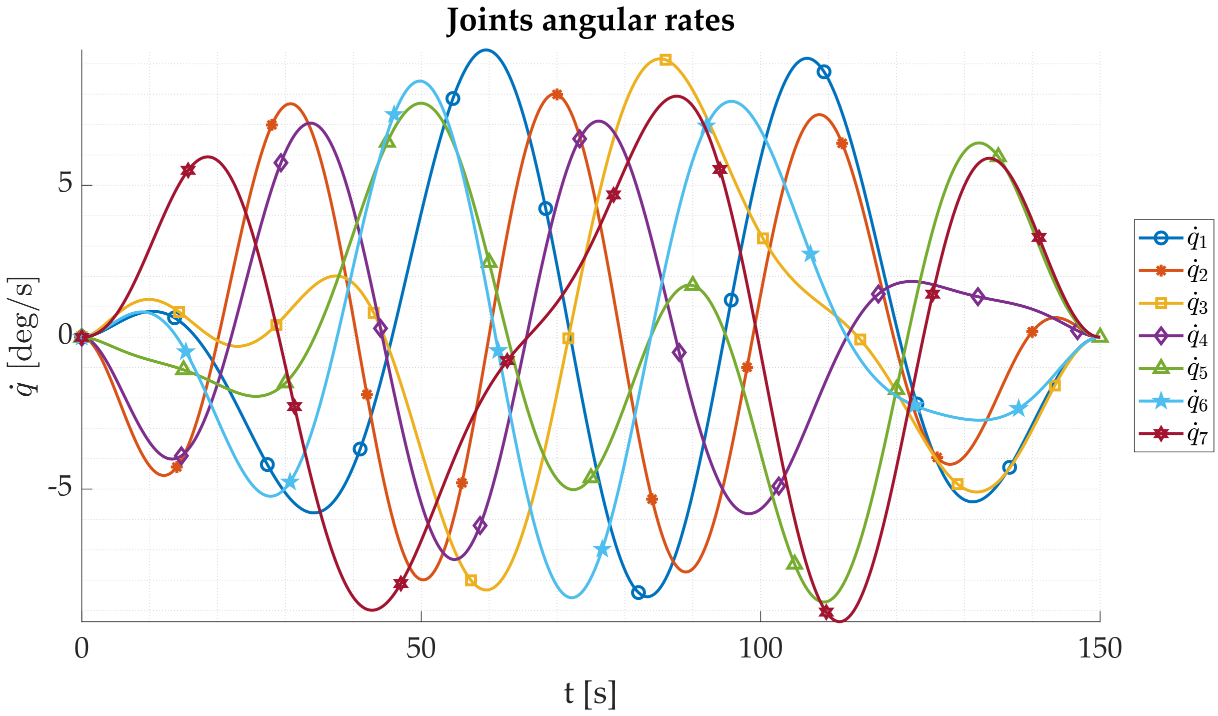


Figure 6.5: Joints Angular Rates For Experimental Test

angles and the given experimental facility limits. It can be clearly seen that the base body angular motion respects these constraints, while giving also a reasonable condition number confirming the validity of the optimization process.

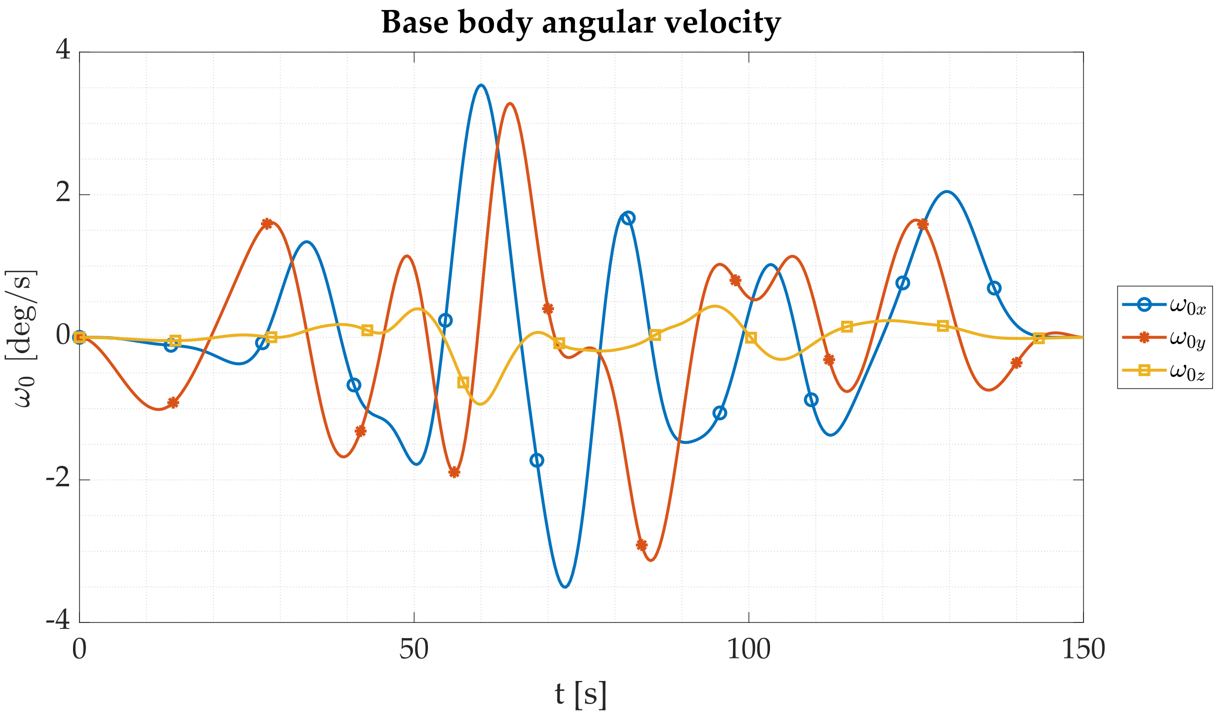


Figure 6.6: Base Body Angular Velocity For Experimental Test

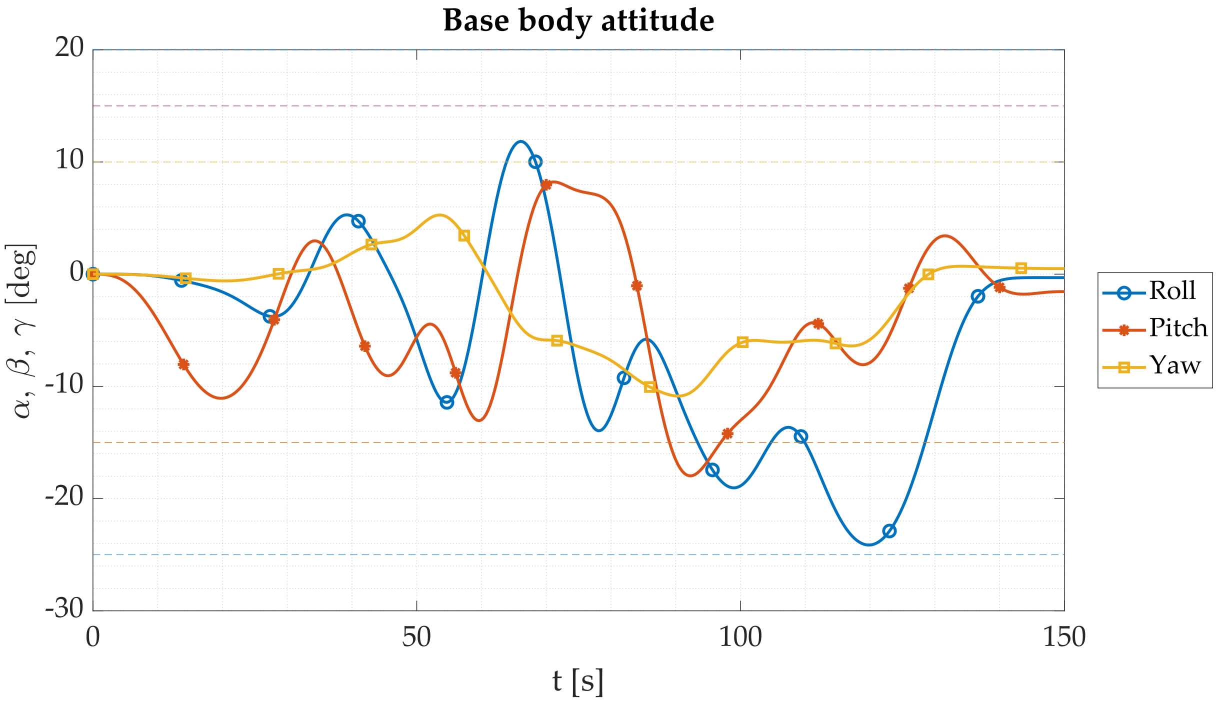


Figure 6.7: Base Body RPY Angles For Experimental Test

6.5.2. Sensors Gathered Data

The following section presents the charts related to the data gathered by the sensors on board the OOS-SIM.

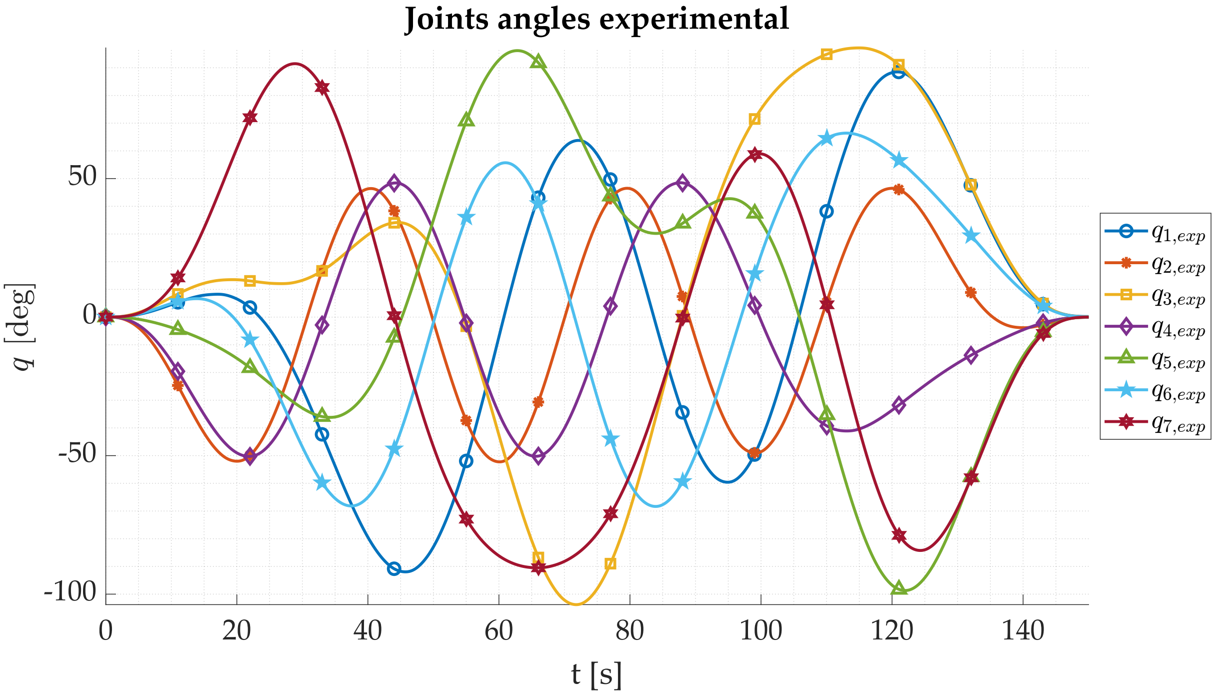


Figure 6.8: Joints Angles Registered On Board (Filtered)

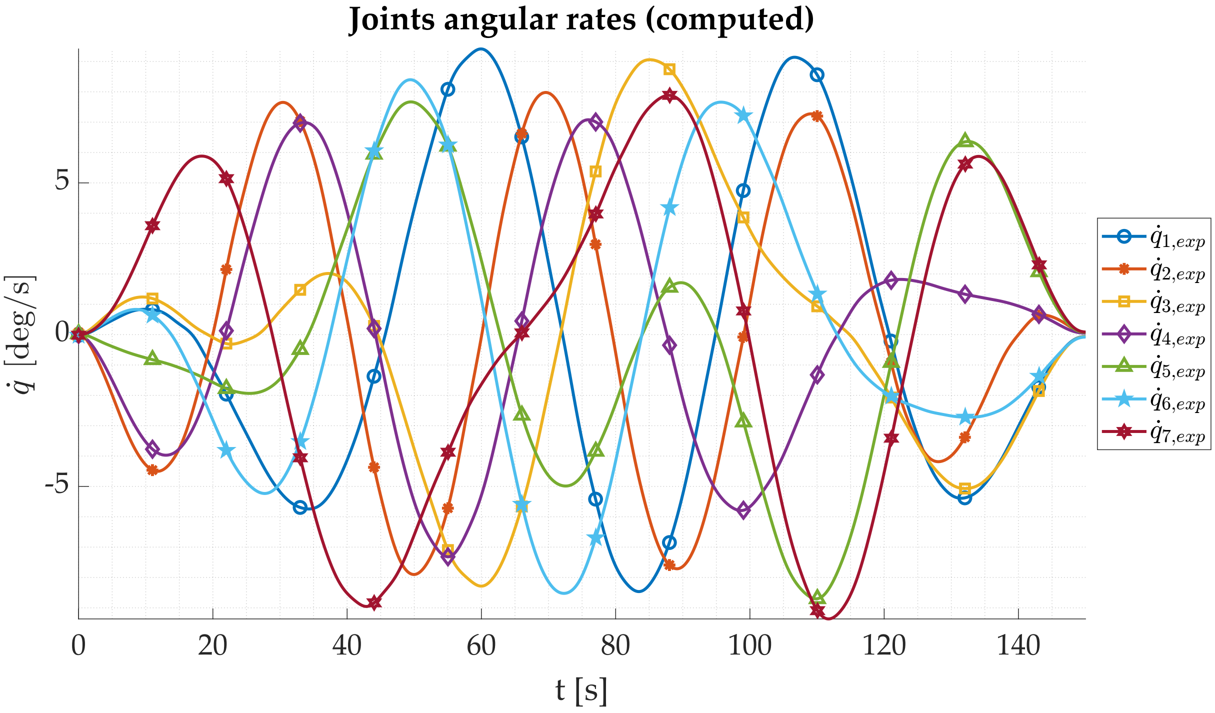


Figure 6.9: Joints Angular Rates Computed (Filtered)

Starting from the joints angles retrieved by the robotic arm motor encoders in Figure 6.8, where the simulated curves (Figure 6.4) are not shown to make the chart readable, it can be seen that the LWR real joints followed the optimized trajectory closely. The chart reports the filtered curves obtained with the signal processing technique described in subsection 6.3.1. The curves in Figure 6.9 are instead computed with numerical differentiation of the joints angles trajectories, process which amplifies the noise of the measurements. The joints angular rates are consequently filtered to mitigate this effect: the results of the filtering process can be appreciated in the chart where the curves are smooth and are almost identical to the ones obtained with the analytical solution shown in Figure 6.5. For what concerns the base body angular velocity obtained with the on-board IMU sensor, the chart is reported in Figure 6.10. Also in this case the validity of the filtering process is noticeable. It is more interesting to look at the same quantity but referred to the inertial reference frame so that it is comparable with the one obtained in simulation.

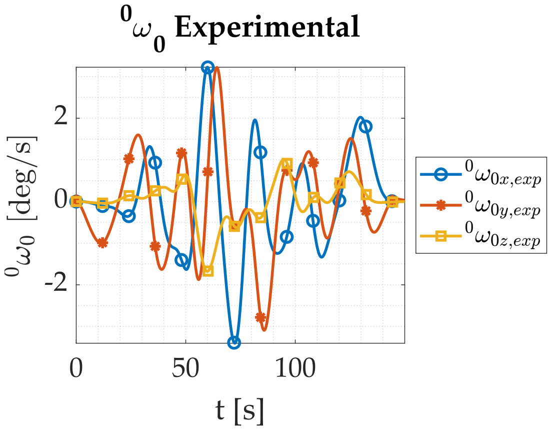


Figure 6.10: Base Body Angular Velocity Registered On Board (Filtered)

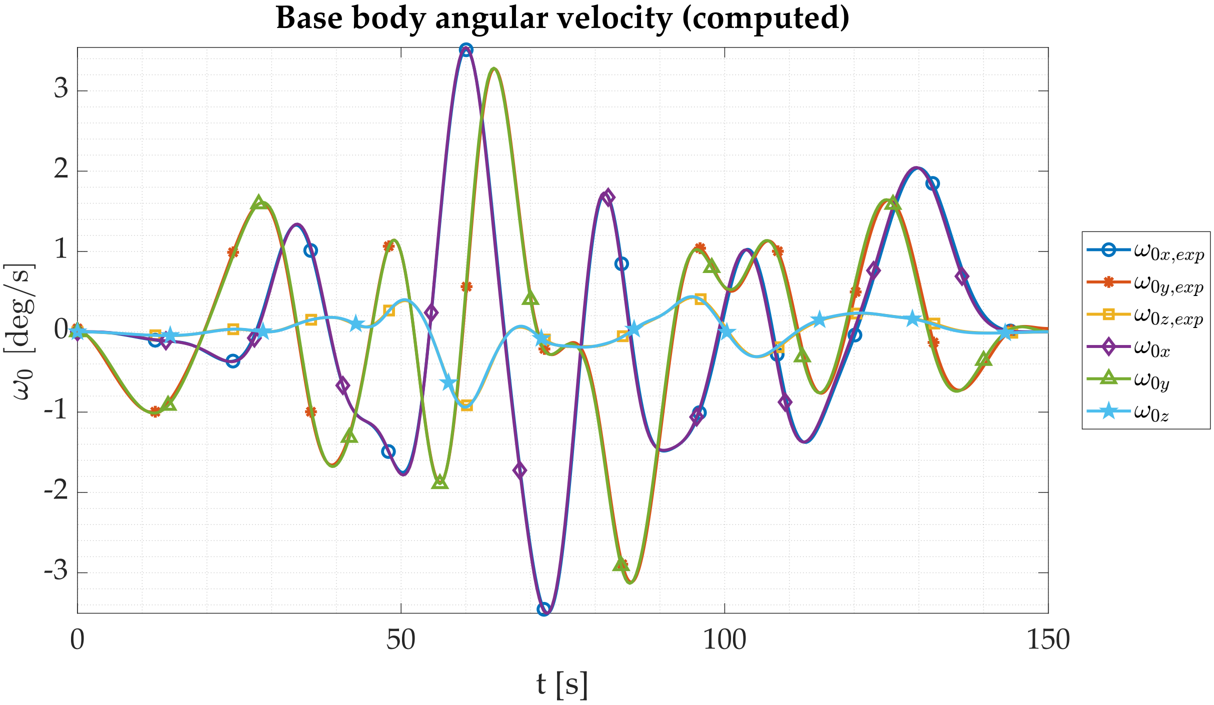


Figure 6.11: Base Body Body Angular Velocity In The Inertial Reference Frame (Filtered)

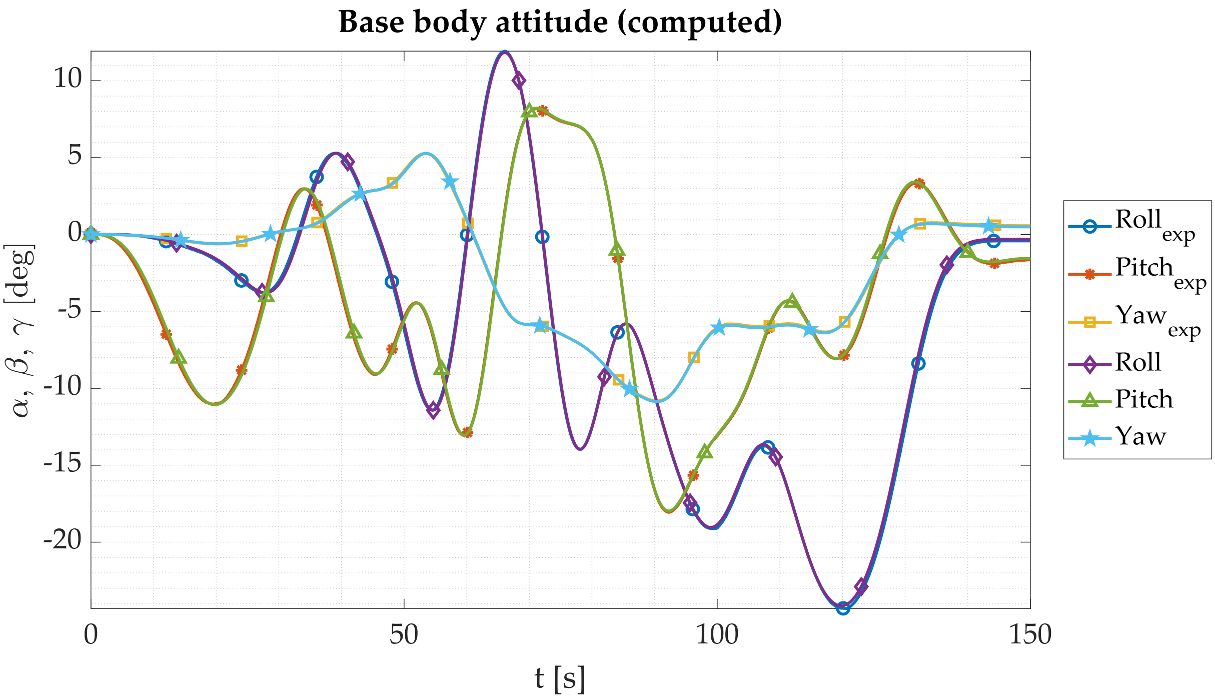


Figure 6.12: Base Body Attitude Computed (Filtered)

Figure 6.11 displays the comparison of the base body angular velocity in the inertial reference frame and the one coming as an output from the simulation process. It can be noticed that the experimental curve is early at the beginning of the trajectory with respect to the simulated one and it is instead late towards the end of the trajectory. This phenomenon was registered also for the joints angles, for which the experimental trajectory was early at the start of the manoeuvre and then trailed in the final seconds.

The same comment can be made by looking at Figure 6.12, where the base body attitude Roll-Pitch-Yaw angles computed from the gyroscope measurements through quaternions integration are confronted with the trajectories coming from simulations. Even in this case the curves are very similar, with the same problem of non constant delay encountered with the other data. In any case also the base body attitude is tracked with precision confirming that the experiment was carried out with success.

The delay can be attributed to the admittance control used for the industrial robots of the facility, which has its own settling time. This explanation is reinforced by analysis of the total angular momentum of the system simulated by the hardware-in-the-loop (HIL) facility (Figure 6.13), which is not perfectly constant (see also [43]). This issue interferes with the identification process as will be discussed in the next section. Nevertheless the two sets of curves are, apart from the mentioned problem, almost superposed confirming in this way that the base body followed the correct dynamics during the manoeuvre.

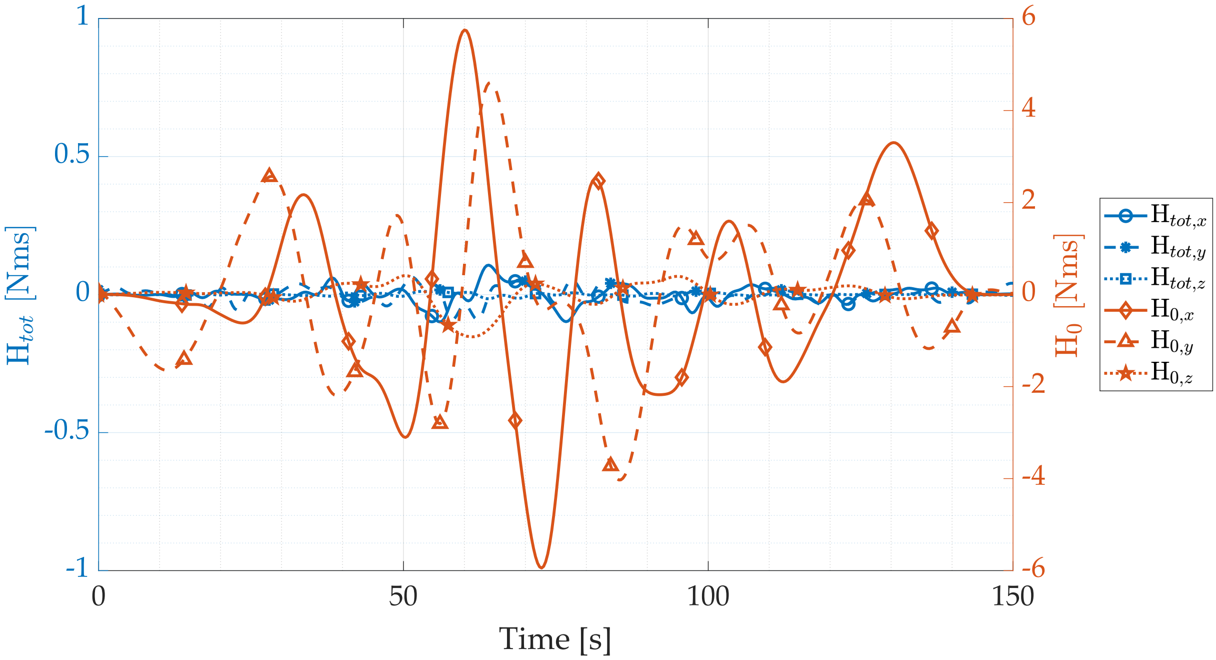


Figure 6.13: Experimental total angular momentum showing error from expected zero value (left y-axis), compared with the base body angular momentum (right y-axis) to show the magnitude of the error. The two y-axis have different scales to make the chart readable.

6.5.3. Identification Results

With the measured and computed data (Table 6.2) the regressor matrix of the system $\mathbf{Y}_{red}$ and the angular momentum due to the load at the end-effector $\mathbf{H}_L$ are evaluated to perform the identification test. Due to the noisy nature of the data a Total Least Squares (TLS) identification algorithm is used (see Section 2.1). Table 6.5 lists the true parameter clusters and the parameter clusters identified with the experimental data. It can be clearly seen that the problem encountered with the HIL simulation, described in the previous section, affects the identification results. It has to be recalled that these results come from just one experimental run of the trajectory. Due to lack of time for the experiments for logistic problems this issue could not be further addressed and will be the subject of future works.

Table 6.5: OOS-SIM: Identification Results

$\boldsymbol{\pi}$	$\boldsymbol{\pi}_{true}$	$\boldsymbol{\pi}_{est}$	$\boldsymbol{\pi}$	$\boldsymbol{\pi}_{true}$	$\boldsymbol{\pi}_{est}$
π_1	42.9650	42.3694	π_{16}	0.0071	0.0533
π_2	-0.4825	-0.2589	π_{17}	2.6453	2.7066

π_3	-3.3094	-3.5431	π_{18}	2.6479	2.6090
π_4	32.7005	32.6860	π_{19}	1.0622	0.7592
π_5	-2.2062	-1.1827	π_{20}	7.4644	7.2920
π_6	23.7121	21.0432	π_{21}	1.5933	1.5859
π_7	0.0049	-1.4504	π_{22}	-0.8914	-0.7235
π_8	0.0103	-0.0253	π_{23}	-6.2638	-6.1733
π_9	4.4747	4.5955	π_{24}	-1.3371	-1.3778
π_{10}	4.4801	5.1243	π_{25}	0.5874	0.4720
π_{11}	0.0049	-0.2708	π_{26}	4.1278	4.1949
π_{12}	0.0099	-0.1453	π_{27}	0.8811	0.7512
π_{13}	3.7325	4.4889	π_{28}	-3.6848	-3.1095
π_{14}	3.7326	3.8146	π_{29}	2.4282	2.3964
π_{15}	0.0036	-0.3527	π_{30}	-2.4180	-2.3198

To check the quality of the identification process, the same test performed for the simulations is carried out. The vector of identified parameter clusters (with the addition of the two terms related to the known load at the end-effector) is used to reconstruct the base body angular velocity to check if the experimental trajectory and the computed one are comparable between each other, even if the identified vector of inertial parameter clusters differ from the true parameters used as input in the model.

Figure 6.14 displays the two sets of angular velocity curves. The difference between the two is barely appreciable, confirming that the identification process correctly found the vector of parameters clusters that reproduces the dynamics represented by the data gathered by the sensors. This result highlights that the data collection is of fundamental importance given that the algorithm is capable of identifying the system described by the sensor measurements.

The last chart in Figure 6.15 shows the absolute error between the base body experimental angular velocity and the reconstructed base body angular velocity (Equation 5.8). The error stays below a maximum value of 0.05 [deg/s] for all the components, a value which is deemed acceptable given its small magnitude and the length of the manoeuvre. This value confirms the validity of the method considering that it gave good results in the model reconstruction with just one experiment, and could be improved by focusing on the data acquisition and by making more runs in order to have a statistical basis in support of the tests.

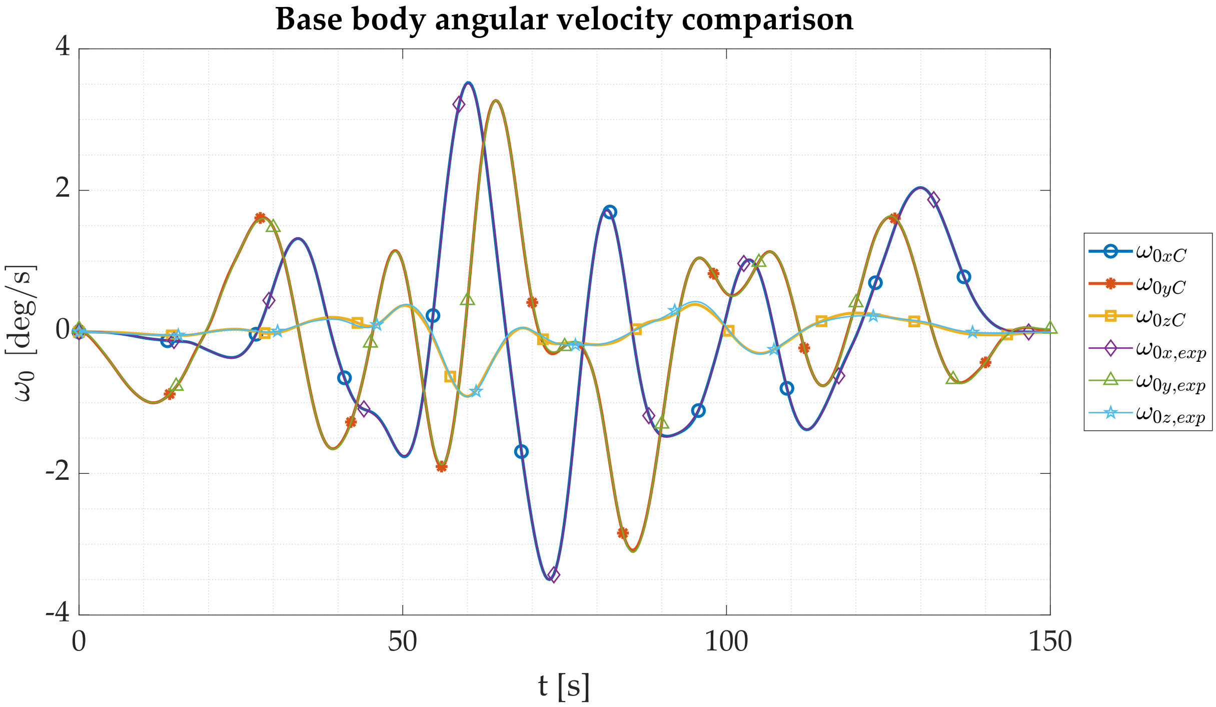


Figure 6.14: OOS-SIM: Comparison between the Base Body Experimental Angular Velocity and the Reconstructed Angular Velocity

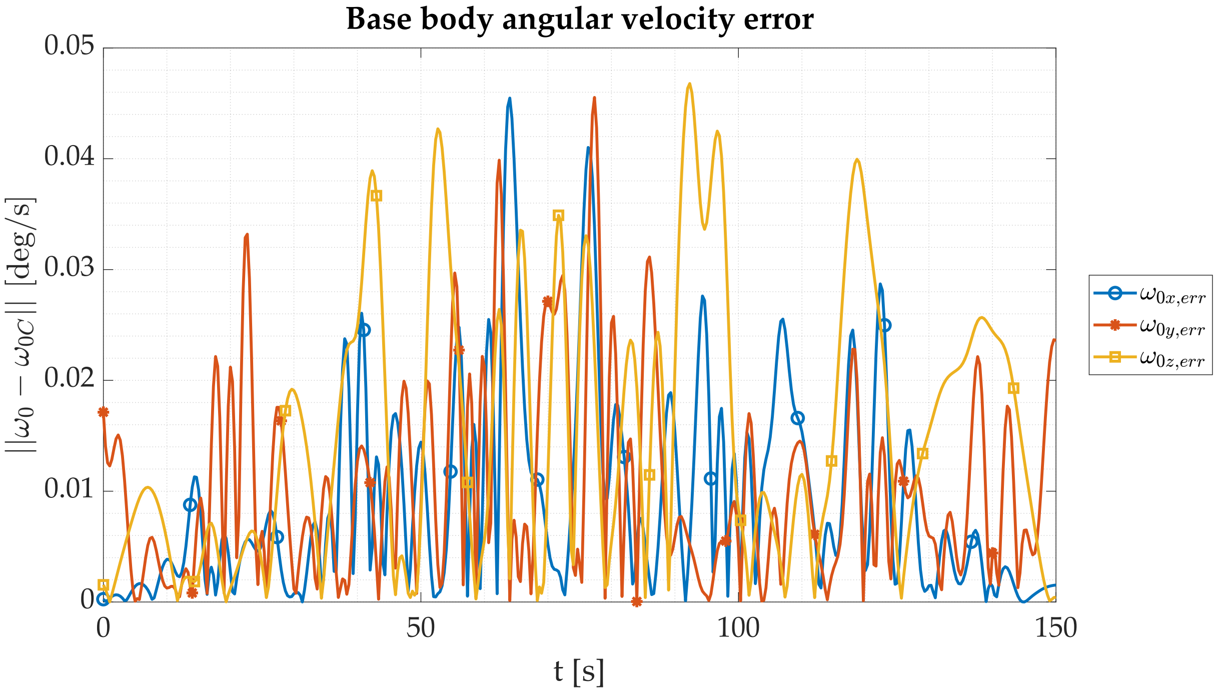


Figure 6.15: OOS-SIM: Absolute Error Between Experimental Angular Velocity And Reconstructed Angular Velocity

7 | Conclusions And Future Developments

7.1. Conclusions

This thesis demonstrated that the identification of the whole free-floating system with a known load at the end-effector is feasible. This methodology is suitable for platforms which are not equipped with reaction wheels, as may be the case in outer space and in robotic intravehicular activities. The identification procedure is carried out under the assumption of a null total angular momentum, made possible by the newly developed framework. This is in opposition with the most recent works which assumed a non null angular momentum condition as was done in [12], the starting point of this research. The validity of this assumption had to be proven with simulated and experimental data, since to the knowledge of the author of this thesis it was never done before for this topic.

The toughest part of this research work proved to be the computation of the full rank regressor matrix in symbolic form for the free-floating seven DoF space robot, treated in Chapter 5, the core of this thesis. In fact, this matrix is composed only of measurable quantities combined between each other with complex trigonometric functions, which are difficult to handle for the symbolic computation engine. This was expected since in literature this specific matrix is never reported explicitly, but only in its implicit form, and thus only with its dependency from other quantities.

Another bottleneck, which is strictly related to the previous point, was the computational time requested to complete the optimization loop needed to find the exciting trajectories for the robotic arm. This specific value grew exponentially from the baseline case, which treated the three DoF robotic arm, to the current case. This has to be attributed to the increased complexity of the geometrical and dynamical model of the space robot, which was reflected in the reduced system regressor matrix, as well as to the incremented number of optimization variables due to the higher number of joints. The available hardware did not permit to overcome this issue, which prevented from testing different geometries and from performing sensitivity analyses.

Although it was not predicted for the values of the condition number of the reduced regressor matrix to be steadily higher with respect to the values found in literature (see e.g. [12]), the simulations made with sensor noise showed that the method achieves results which are comparable with the ones of the baseline. These higher values for the condition number translate eventually in slightly lower performances in the identification of the inertial parameter clusters having an evident smaller value with respect to the others. Nevertheless, it has been shown that the clusters which are identified with the highest relative errors have minimal influence on the dynamics, since the reconstructed base angular velocity is superposed almost exactly with the curve computed in simulation. The discrepancy between this thesis and the baseline in terms of condition number can be explained by taking into account the different framework between the two: zero angular momentum for the first due to the imposed condition of a platform not having reaction wheels on board, non zero angular momentum for the second. The latter scenario permits to have a non null initial base body angular velocity, which has been proven in literature (see e.g. [29]) to be a great advantage for identification purposes.
Finally, the penultimate chapter of this thesis (Chapter 6) presented the results of the new methodology applied to an experimental setup, the DLR's OOS-SIM in Oberpfaffenhofen. Due to lack of time for the experimental tests for logistic problems, only one test was performed and could actually be used to check the new framework with a case study and with real sensors data. The experimental test gave good results in terms of reconstruction of the base body angular velocity after the identification procedure. The treatment of this identification task has also underlined the detrimental effect of very small time delays in the simulation of the system dynamics with the hardware-in-the-loop approach, which results in significant changes in the identified parameters. In any case worse identification results were foreseen from the experimental campaign, in view of the use of real sensors which give measurements composed of real random noise and other effects which were not modelled in Simulink. However, the problem encountered with the delay of the experimental trajectories did not permit to make a proper comparison with the results obtained in simulation.

7.2. Future Developments

7.2.1. Optimization of the Computational Times

As stated in the previous section, computational times for the optimization loop were a bottleneck for this research. This issue could be overcome by acting on two separate problems. The first and more obvious would be to increase the computational power of the machine on which the optimizations are ran, using a multi-core CPU to exploit MATLAB Parallel Computing Toolbox. The second one would be to optimize the developed algorithms, from the computational time point of view, starting for example by further simplifying the complex trigonometric expressions composing the reduced system regressor matrix.

7.2.2. Sensitivity Analyses

A way to increase the number of available data in relation to the topic treated in this master thesis would be to perform sensitivity analyses on different aspects. The first and more interesting would be to test diverse ratios of satellite mass to load at the end-effector mass, to observe how the condition number of the reduced system regressor matrix would change with respect to this parameter and, with it, the quality of the identification results. The selection of the ratio should not be arbitrary, but it should respect upper and lower constraints for the end-effector load, i.e. the maximum allowable mass which could be carried by the embarked robotic arm as well as a minimum mass for the load, to be able to excite the inertial parameters of the base body.

The second group of sensitivity analyses should focus on finding the inertial parameter clusters which contribute the most and the least to the system free-floating dynamics, in order to better understand the results of the identification process. This should be done with particular attention to the relative errors of the identified inertial parameter clusters with respect to the expected ones, as was the case with the results of the simulations. This process could be helpful to guide the generation of the manoeuvres done by the robotic arm, by tailoring them to excite specific clusters, i.e. the ones driving the dynamics and that, for this reasons, are requested to have a higher confidence.

7.2.3. Experimental Tests

As mentioned in subsection 6.5.3 only one experimental test was carried out for logistic reasons. An obvious development to further verify the new methodology described in this thesis would be to gather more experimental data by performing additional tests on the OOS-SIM facility. This would allow to complete also a statistical analysis of the results, but in this case with measurements collected by real sensors with real noise models, and thus closer to what the scenario would be in orbit. The problem encountered with the delay of the experimental trajectories with respect to the simulated trajectories shall be investigated, to obtain consistency between the two and check if the identification process with real sensors measurements keeps the same performances in a more realistic environment.

Moreover, a geometrical and dynamical model of the same complexity for the space robot as the one used in the simulations could be implemented on the facility if the algorithms running on the OOS-SIM will be updated. This would grant the possibility to have a better comparison between the simulated and experimental data.

7.2.4. Model Complexity

Finally, to further extend this research topic the guidelines set by the ESA *OBSIdian* studies could be a good starting point. In particular the focus should go the physical disturbances that are typically acting on board of a satellite such as fuel sloshing inside of tanks and the presence of flexible appendages such as solar panels, as well as the flexibility of the robotic arm itself. Neglecting this additional noisy sources and treating them as non modeled dynamics could worsen the identification results on a real system (see e.g. [27]). Having the knowledge of the natural frequencies of flexible objects and of the sloshing modes of a fluid inside a tank, identified in the same processes where the inertial parameters are retrieved, could permit a more accurate description of the system and consequently a more accurate control which is vital for operations requiring a high precision in the motion of the end-effector, as in a mission for in orbit capture of an uncooperative satellite.

Bibliography

[1] *On-Orbit Satellite Servicing Study Project Report.* Tech. rep. NASA (National Aeronautics and Space Administration), Oct. 2010.

[2] URL: `https://www.northropgrumman.com/space/space-logistics-services/`.

[3] Alexander Beyer et al. "CAESAR: Space Robotics Technology for Assembly, Maintenance, and Repair". In: *Proceedings of the International Astronautical Congress (IAC)*. Bremen, Germany, Oct. 2018.

[4] Evangelos Papadopoulos. "On the Dynamics and Control of Space Manipulators". PhD thesis. MIT (Massachussets Instute of Technology), Oct. 1990.

[5] B. Siciliano and O. Khatib. *Handbook of Robotics*. Springer, 2016. Chap. 55. ISBN: 978-3-319-32550-7.

[6] Kazuya Yoshida and Yoji Umetani. "Control of Space Manipulators with Generalized Jacobian Matrix". In: *Space Robotics: Dynamics and Control*. Ed. by Yangsheng Xu and Takeo Kanade. Boston, MA: Springer US, 1993, pp. 165–204. ISBN: 978-1-4615-3588-1. DOI: `10.1007/978-1-4615-3588-1_7`.

[7] Evangelos G. Papadopoulos. "Nonholonomic Behavior in Free-floating Space Manipulators and its Utilization". In: *Nonholonomic Motion Planning*. Boston, MA: Springer US, 1993, pp. 423–445. ISBN: 978-1-4615-3176-0. DOI: `10.1007/978-1-4615-3176-0_11`.

[8] S. Dubowsky and E. Papadopoulos. "The kinematics, dynamics, and control of free-flying and free-floating space robotic systems". In: *IEEE Transactions on Robotics and Automation* 9.5 (1993), pp. 531–543. DOI: `10.1109/70.258046`.

[9] Jin-Ho Shin and Jujang Lee. "Dynamic control with adaptive identification for free-flying space robots in joint space". In: *Robotica* 12 (1994), pp. 541–551.

[10] Evangelos Papadopoulos et al. "Robotic Manipulation and Capture in Space: A Survey". In: *Frontiers in Robotics and AI* 8:686723 (July 2021). DOI: `10.3390/frobt.2021.686723`.

[11] Olga-Orsalia Christidi-Loumpasefski et al. "Towards On-Board Identification Of Space Systems". In: *International Symposium on Artificial Intelligence, Robotics and Automation in Space i-SAIRAS 2020*. Pasadena, CA, Oct. 2020.

[12] Olga-Orsalia Christidi-Loumpasefski, Kostas Nanos, and Evangelos Papadopoulos. “On Parameter Estimation of Space Manipulator Systems Using the Angular Momentum Conservation”. In: *Proceedings of ICRA 2017*. Singapore, 2017, pp. 5453–5458.

[13] B. Naveen, Suril V. Shah, and Arun K. Misra. “Momentum Model-Based Minimal Parameter Identification of a Space Robot”. In: *Journal of Guidance, Control, and Dynamics* 42.3 (Mar. 2019), pp. 508–523. DOI: 10.2514/1.g003541. URL: https://doi.org/10.2514%5C%2F1.g003541.

[14] Trey Smith et al. “Astrobee: A New Platform for Free-Flying Robotics Research on the International Space Station”. In: *International Symposium on Artificial Intelligence, Robotics and Automation in Space i-SAIRAS 2016*. Beijing, China, June 2016.

[15] Ko Ayusawa, Gentiane Venture, and Yoshihiko Nakamura. “Identifiability and identification of inertial parameters using the underactuated base-link dynamics for legged multibody systems”. In: *The International Journal of Robotics Research IJRR* 33.3 (2014), pp. 446–468. DOI: 10.1177/0278364913495932.

[16] Christopher G. Atkeson, Chae H. An, and John M. Hollerbach. “Estimation of Inertial Parameters of Manipulator Loads and Links”. In: *The International Journal of Robotics Research* 5.3 (1986), pp. 101–119. DOI: 10.1177/027836498600500306.

[17] G. Antonelli, F. Caccavale, and P. Chiacchio. “A systematic procedure for the identification of dynamic parameters of robot manipulators”. In: *Robotica* 17.4 (1999), pp. 427–435. DOI: 10.1017/S026357479900140X.

[18] Q. Leboutet et al. “Inertial Parameter Identification in Robotics: A Survey.” In: *Applied Sciences* 11, 4303 (May 2021). DOI: https://doi.org/10.3390/app11094303.

[19] W. Khalil and E. Dombre. *Modeling, Identification and Control of Robots*. Oxford: Butterworth-Heinemann, 2002. Chap. 12. ISBN: 978-1-903996-66-9.

[20] *Total Least Squares*. Course Notes. CEE 629. System Identification, Duke University. 2017.

[21] Roberto Lampariello and G. Hirzinger. “Freeflying Robots -Inertial Parameters Identification and Control Strategies”. In: Jan. 2000.

[22] Kazuya Yoshida and Satoko Abiko. “Inertia Parameter Identification For a Free-Flying Space Robot”. In: *AIAA Guidance, Navigation, and Control Conference and Exhibit*. Monterey, California, Aug. 2002.

[23] Roberto Lampariello and Gerard Hirzinger. “Modeling and experimental design for the on-orbit inertial parameter identification of free-flying space robots”. In: *Proceedings of IDETC/CIE 2005*. ASME 2005 IDETC/CIE. Long Beach, California, USA, Sept. 2005.

[24] Ou Ma and Hung Dang. *On-Orbit Identification of Inertia Properties of Spacecraft using Robotics Technology.* Tech. rep. Dec. 2006.

[25] Ou Ma, Hung Dang, and Khanh D. Pham. "On-Orbit Identification of Inertia Properties of Spacecraft Using a Robotic Arm". In: *Journal of Guidance, Navigation, Control and Dynamics* 31 (6) (Nov. 2008). DOI: `10.2514/1.35188`.

[26] Wolfgang Rackl, Roberto Lampariello, and Alin Albu-Schäffer. "Parameter Identification Methods for Free-Floating Space Robots with direct Torque Sensing". In: *IFAC Proceedings Volumes* 46.19 (2013). 19th IFAC Symposium on Automatic Control in Aerospace, pp. 464–469. ISSN: 1474-6670. DOI: `https://doi.org/10.3182/20130902-5-DE-2040.00121`. URL: `https://www.sciencedirect.com/science/article/pii/S1474667015363679`.

[27] Wolfgang Rackl and Roberto Lampariello. "Parameter identification of free-floating robots with flexible appendages and fuel sloshing". In: *Proceedings of 2014 International Conference on Modelling, Identification & Control.* 2014, pp. 129–134. DOI: `10.1109/ICMIC.2014.7020740`.

[28] Olga-Orsalia Christidi-Loumpasefski and Evangelos Papadopoulos. "On Parameter Estimation of Multi-Arm Free-Flying Space Manipulator Systems". 2018.

[29] Olga-Orsalia Christidi-Loumpasefski and Evangelos Papadopoulos. "Concurrent Parameter Identification and Control for Free-Floating Robotic Systems During On-Orbit Servicing". In: *Proceedings of ICRA 2020.* Paris, 2020, pp. 6014–6020.

[30] Wenfu Xu et al. "A Practical and Effective Method for Identifying the Complete Inertia Parameters of Space Robots". In: *International Conference on Intelligent Robots and Systems (IROS).* Hamburg, Germany, Sept. 2015.

[31] Wenfu Xu et al. "On-orbit identifying the inertia parameters of space robotic systems using simple equivalent dynamics". In: *Acta Astronautica* (2017), pp. 131–142. DOI: `10.1016/j.actaastro.2016.12.031`.

[32] Cheng Wei et al. "Inertia parameter identification of space floating target during robotic exploratory grasping". In: *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 233.11 (2019), pp. 4247–4260. DOI: `10.1177/0954410018819567`. URL: `https://doi.org/10.1177/0954410018819567`.

[33] Wisama Khalil and J.F. Kleinfinger. "A new geometric notation for open and closed-loop robots". In: vol. 3. May 1986, pp. 1174–1179. DOI: `10.1109/ROBOT.1986.1087552`.

[34] Y. Li et al. "Constrained Motion Planning of 7-DOF Space Manipulator via Deep Reinforcement Learning Combined with Artificial Potential Field". In: *Aerospace* 9, 163 (2022). DOI: `https://doi.org/10.3390/aerospace9030163`.

[35] S.A.A. Moosavian and Evangelos Papadopoulos. "Explicit dynamics of space free-flyers with multiple manipulators via SPACEMAPLE". In: *Advanced Robotics* 18 (Jan. 2004), pp. 223–244. DOI: 10.1163/156855304322758033.

[36] Maxime Gautier and Wisama Khalil. "Direct calculation of minimum set of inertial parameters of serial robots". In: *IEEE Trans. Robotics Autom.* 6 (1990), pp. 368–373.

[37] Markus Wilde et al. "Equations of Motion of Free-Floating Spacecraft-Manipulator Systems: An Engineer's Tutorial". In: *Frontiers in Robotics and AI* 5 (2018). ISSN: 2296-9144. DOI: 10.3389/frobt.2018.00041. URL: https://www.frontiersin.org/articles/10.3389/frobt.2018.00041.

[38] John L. Crassidis and F. Landis Markley. "Unscented Filtering for Spacecraft Attitude Estimation". In: *AIAA Guidance, Navigation, and Control Conference and Exhibit*. Austin, Texas, 2003.

[39] Maximilian Maier et al. "TINA: Small Torque Controlled Robotic Arm For Exploration And Small Satellites". In: *70th International Astronautical Congress (IAC)*. Washington D.C., USA, Oct. 2019.

[40] J. Vinals, J. Gala, and G. Guerra. "Standard Interface For Robotic Manipulation (SIROM): SRC H2020 OG5 Final Results - Future Upgrades And Applications". In: *i-SAIRAS 2020 Papers*. Oct. 2020.

[41] Jordi Artigas et al. "The OOS-SIM: An on-ground simulation facility for on-orbit servicing robotic operations". In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. 2015, pp. 2854–2860. DOI: 10.1109/ICRA.2015.7139588.

[42] David Tedaldi, Alberto Pretto, and Emanuele Menegatti. "A Robust and Easy to Implement Method for IMU Calibration without External Equipments". In: *IEEE International Conference on Robotics and Automation ICRA*. Honk Hong Convention and Exhibition Center, China, June 2014.

[43] M. Hrishik et al. "Inertia-decoupled equations for hardware-in-the-loop simulation of an orbital robot with external forces". In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2020, pp. 1879–1886.

[44] K. Yoshida. "The SpaceDyn: a MATLAB toolbox for space and mobile robots". In: *Proceedings 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human and Environment Friendly Robots with High Intelligence and Emotional Quotients (Cat. No.99CH36289)*. Vol. 3. 1999, 1633–1638 vol.3. DOI: 10.1109/IROS.1999.811712.

A | Appendix A

A.1. Free-Floating 7 DOF Robotic Arm Inertial Parameter Clusters

$$
\begin{aligned}
\boldsymbol{\pi_1} = {} & I_{0xx} + I_{1yy} + \frac{M_0\, {a_1}^2\, \left(M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_0\, \left({b_{02}}^2 + {b_{03}}^2\right)\, \left(M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{{b_1}^2\, \left(M_0 + M_1\right)\, \left(M_2 + M_3 + M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{2\, M_0\, a_1\, b_{03}\, \left(M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{2\, M_0\, a_1\, b_1\, \left(M_2 + M_3 + M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{2\, M_0\, b_1\, b_{03}\, \left(M_2 + M_3 + M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
\end{aligned}
$$

$$
\boldsymbol{\pi_2} = -\frac{M_0\, b_{01}\, b_{02}\, \left(M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
$$

$$
\begin{aligned}
\boldsymbol{\pi_3} = {} & -\frac{M_0\, a_1\, b_{01}\, \left(M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& - \frac{M_0\, b_{01}\, b_{03}\, \left(M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& - \frac{M_0\, b_{01}\, b_1\, \left(M_2 + M_3 + M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
\end{aligned}
$$

$$
\boldsymbol{\pi_4} = I_{0yy} + I_{1yy} + \frac{M_0\, {a_1}^2\, \left(M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
$$

$$+\frac{M_0\,\left({b_{02}}^2+{b_{03}}^2\right)\,\left(M_1+M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{{b_1}^2\,\left(M_0+M_1\right)\,\left(M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{2\,M_0\,a_1\,b_{03}\,\left(M_1+M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{2\,M_0\,a_1\,b_1\,\left(M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{2\,M_0\,b_1\,b_{03}\,\left(M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$\boldsymbol{\pi_5}=-\frac{M_0\,a_1\,b_{02}\,\left(M_1+M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$-\frac{M_0\,b_{02}\,b_{03}\,\left(M_1+M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$-\frac{M_0\,b_1\,b_{02}\,\left(M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$\boldsymbol{\pi_6}=I_{0zz}+\frac{M_0\,\left({b_{01}}^2+{b_{02}}^2\right)\,\left(M_1+M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$\boldsymbol{\pi_7}=I_{1xx}-I_{1yy}+I_{2yy}$$

$$\boldsymbol{\pi_8}=I_{1zz}+I_{2yy}$$

$$\boldsymbol{\pi_9}=I_{2xx}-I_{2yy}+I_{3yy}+\frac{{a_2}^2\,\left(M_0+M_1\right)\,\left(M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{{a_3}^2\,\left(M_0+M_1+M_2\right)\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{{b_2}^2\,\left(M_0+M_1+M_2\right)\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{{b_3}^2\,\left(M_0+M_1+M_2+M_3\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{2\,a_2\,a_3\,\left(M_0+M_1\right)\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{2\,a_2\,b_2\,\left(M_0+M_1\right)\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{2\,a_3\,b_3\,\left(M_0+M_1+M_2\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{2\,b_2\,b_3\,\left(M_0+M_1+M_2\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{2\,a_3\,b_2\,\left(M_0+M_1+M_2\right)\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{2\,a_2\,b_3\,\left(M_0+M_1\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$\boldsymbol{\pi_{10}}=I_{2zz}+I_{3yy}+\frac{{a_2}^2\,\left(M_0+M_1\right)\,\left(M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{{a_3}^2\,\left(M_0+M_1+M_2\right)\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{{b_2}^2\,\left(M_0+M_1+M_2\right)\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{{b_3}^2\,\left(M_0+M_1+M_2+M_3\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{2\,a_2\,a_3\,\left(M_0+M_1\right)\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{2\,a_2\,b_2\,\left(M_0+M_1\right)\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{2\,a_3\,b_3\,\left(M_0+M_1+M_2\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{2\,b_2\,b_3\,\left(M_0+M_1+M_2\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{2\,a_3\,b_2\,\left(M_0+M_1+M_2\right)\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{2\,a_2\,b_3\,\left(M_0+M_1\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$\boldsymbol{\pi_{11}}=I_{3xx}-I_{3yy}+I_{4yy}$$

$$\boldsymbol{\pi_{12}}=I_{3zz}+I_{4yy}$$

$$\boldsymbol{\pi_{13}}=I_{4xx}-I_{4yy}+I_{5yy}+\frac{{a_4}^2\,\left(M_0+M_1+M_2+M_3\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{{a_5}^2\,\left(M_5+M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$
\begin{aligned}
&+ \frac{{b_4}^2\,\left(M_5+M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{{b_5}^2\,\left(M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3+M_4+M_5\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&- \frac{2\,a_4\,a_5\,\left(M_5+M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{2\,a_4\,b_4\,\left(M_5+M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{2\,a_5\,b_5\,\left(M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&- \frac{2\,b_4\,b_5\,\left(M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&- \frac{2\,a_5\,b_4\,\left(M_5+M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&- \frac{2\,a_4\,b_5\,\left(M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\pi_{14}} = {} & I_{4zz}+I_{5yy}+\frac{{a_4}^2\,\left(M_0+M_1+M_2+M_3\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{{a_5}^2\,\left(M_5+M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{{b_4}^2\,\left(M_5+M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{{b_5}^2\,\left(M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3+M_4+M_5\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&- \frac{2\,a_4\,a_5\,\left(M_5+M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{2\,a_4\,b_4\,\left(M_5+M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{2\,a_5\,b_5\,\left(M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&- \frac{2\,b_4\,b_5\,\left(M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&- \frac{2\,a_5\,b_4\,\left(M_5+M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&- \frac{2\,a_4\,b_5\,\left(M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}
\end{aligned}
$$

$$\boldsymbol{\pi_{15}} = I_{5xx} - I_{5yy} + I_{6yy}$$

$$\boldsymbol{\pi_{16}} = I_{5zz} + I_{6yy}$$

$$\begin{aligned}\boldsymbol{\pi_{17}} &= I_{6xx} - I_{6yy} + I_{7yy} + \frac{M_7\, {a_7}^2\ (M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\ &+ \frac{M_7\, {b_6}^2\ (M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\ &+ \frac{{a_6}^2\ (M_6 + M_7)\ (M_0 + M_1 + M_2 + M_3 + M_4 + M_5)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\ &+ \frac{2\, M_7\, a_7\, b_6\ (M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\ &+ \frac{2\, M_7\, a_6\, a_7\ (M_0 + M_1 + M_2 + M_3 + M_4 + M_5)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\ &+ \frac{2\, M_7\, a_6\, b_6\ (M_0 + M_1 + M_2 + M_3 + M_4 + M_5)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}\end{aligned}$$

$$\begin{aligned}\boldsymbol{\pi_{18}} &= I_{6zz} + I_{7yy} + \frac{M_7\, {a_7}^2\ (M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\ &+ \frac{M_7\, {b_6}^2\ (M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\ &+ \frac{{a_6}^2\ (M_6 + M_7)\ (M_0 + M_1 + M_2 + M_3 + M_4 + M_5)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\ &+ \frac{2\, M_7\, a_7\, b_6\ (M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\ &+ \frac{2\, M_7\, a_6\, a_7\ (M_0 + M_1 + M_2 + M_3 + M_4 + M_5)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\ &+ \frac{2\, M_7\, a_6\, b_6\ (M_0 + M_1 + M_2 + M_3 + M_4 + M_5)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}\end{aligned}$$

$$\boldsymbol{\pi_{19}} = I_{7xx} - I_{7yy}$$

$$\boldsymbol{\pi_{20}} = I_{7zz}$$

$$\boldsymbol{\pi_{21}} = \frac{M_0\, b_{02}\, b_3\ (M_4 + M_5 + M_6 + M_7)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}$$

$$+\frac{M_0\,a_3\,b_{02}\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{M_0\,b_{02}\,b_2\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{M_0\,a_2\,b_{02}\,\left(M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$\boldsymbol{\pi_{22}}=\frac{a_3\,b_1\,\left(M_0+M_1\right)\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{b_1\,b_2\,\left(M_0+M_1\right)\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{M_0\,a_1\,b_3\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{M_0\,b_{03}\,b_3\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{a_2\,b_1\,\left(M_0+M_1\right)\,\left(M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{M_0\,a_1\,a_3\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{M_0\,a_1\,b_2\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{M_0\,a_3\,b_{03}\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{M_0\,b_2\,b_{03}\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{M_0\,a_1\,a_2\,\left(M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{M_0\,a_2\,b_{03}\,\left(M_2+M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{b_1\,b_3\,\left(M_0+M_1\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$\boldsymbol{\pi_{23}}=\frac{M_0\,b_{01}\,b_3\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{M_0\,a_3\,b_{01}\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+\frac{M_0\,b_{01}\,b_2\,\left(M_3+M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$+ \frac{M_0\, a_2\, b_{01}\, \left(M_2 + M_3 + M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}$$

$$\boldsymbol{\pi_{24}} = \frac{M_0\, a_4\, b_{02}\, \left(M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
- \frac{M_0\, b_{02}\, b_5\, \left(M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
- \frac{M_0\, a_5\, b_{02}\, \left(M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
+ \frac{M_0\, b_{02}\, b_4\, \left(M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}$$

$$\boldsymbol{\pi_{25}} = \frac{M_0\, a_1\, a_4\, \left(M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
+ \frac{M_0\, a_4\, b_{03}\, \left(M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
- \frac{b_1\, b_5\, \left(M_0 + M_1\right)\, \left(M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
- \frac{a_5\, b_1\, \left(M_0 + M_1\right)\, \left(M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
+ \frac{b_1\, b_4\, \left(M_0 + M_1\right)\, \left(M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
- \frac{M_0\, a_1\, b_5\, \left(M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
- \frac{M_0\, b_{03}\, b_5\, \left(M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
+ \frac{a_4\, b_1\, \left(M_0 + M_1\right)\, \left(M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
- \frac{M_0\, a_1\, a_5\, \left(M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
+ \frac{M_0\, a_1\, b_4\, \left(M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
- \frac{M_0\, a_5\, b_{03}\, \left(M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
+ \frac{M_0\, b_{03}\, b_4\, \left(M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}$$

$$
\begin{aligned}
\boldsymbol{\pi_{26}} = & \frac{M_0\, a_4\, b_{01}\, \left(M_4 + M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& - \frac{M_0\, b_{01}\, b_5\, \left(M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& - \frac{M_0\, a_5\, b_{01}\, \left(M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_0\, b_{01}\, b_4\, \left(M_5 + M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\pi_{27}} = & \frac{M_0\, a_6\, b_{02}\, \left(M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_0\, M_7\, a_7\, b_{02}}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_0\, M_7\, b_{02}\, b_6}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\pi_{28}} = & \frac{a_6\, b_1\, \left(M_0 + M_1\right)\, \left(M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_0\, a_1\, a_6\, \left(M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_7\, a_7\, b_1\, \left(M_0 + M_1\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_0\, a_6\, b_{03}\, \left(M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_7\, b_1\, b_6\, \left(M_0 + M_1\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_0\, M_7\, a_1\, a_7}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_0\, M_7\, a_1\, b_6}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_0\, M_7\, a_7\, b_{03}}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_0\, M_7\, b_{03}\, b_6}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\pi_{29}} = & \frac{M_0\, a_6\, b_{01}\, \left(M_6 + M_7\right)}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7} \\
& + \frac{M_0\, M_7\, a_7\, b_{01}}{M_0 + M_1 + M_2 + M_3 + M_4 + M_5 + M_6 + M_7}
\end{aligned}
$$

$$+\frac{M_0\,M_7\,b_{01}\,b_6}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$\begin{aligned}
\boldsymbol{\pi_{30}} = {} & \frac{a_3\,a_4\,\left(M_0+M_1+M_2\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& + \frac{a_4\,b_2\,\left(M_0+M_1+M_2\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& - \frac{a_5\,b_3\,\left(M_5+M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& + \frac{b_3\,b_4\,\left(M_5+M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& - \frac{a_2\,b_5\,\left(M_0+M_1\right)\,\left(M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& + \frac{a_4\,b_3\,\left(M_0+M_1+M_2+M_3\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& - \frac{a_2\,a_5\,\left(M_0+M_1\right)\,\left(M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& - \frac{a_3\,b_5\,\left(M_6+M_7\right)\,\left(M_0+M_1+M_2\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& + \frac{a_2\,b_4\,\left(M_0+M_1\right)\,\left(M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& - \frac{b_2\,b_5\,\left(M_6+M_7\right)\,\left(M_0+M_1+M_2\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& + \frac{a_2\,a_4\,\left(M_0+M_1\right)\,\left(M_4+M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& - \frac{a_3\,a_5\,\left(M_0+M_1+M_2\right)\,\left(M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& + \frac{a_3\,b_4\,\left(M_0+M_1+M_2\right)\,\left(M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& - \frac{a_5\,b_2\,\left(M_0+M_1+M_2\right)\,\left(M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& + \frac{b_2\,b_4\,\left(M_0+M_1+M_2\right)\,\left(M_5+M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7} \\
& - \frac{b_3\,b_5\,\left(M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}
\end{aligned}$$

$$\boldsymbol{\pi_{31}} = \frac{M_7\,a_7\,b_3\,\left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$

$$
\begin{aligned}
&+ \frac{M_7\, b_3\, b_6\, \left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{a_2\, a_6\, \left(M_0+M_1\right)\, \left(M_6+M_7\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{a_3\, a_6\, \left(M_6+M_7\right)\, \left(M_0+M_1+M_2\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{a_6\, b_2\, \left(M_6+M_7\right)\, \left(M_0+M_1+M_2\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{M_7\, a_2\, a_7\, \left(M_0+M_1\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{M_7\, a_2\, b_6\, \left(M_0+M_1\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{a_6\, b_3\, \left(M_6+M_7\right)\, \left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{M_7\, a_3\, a_7\, \left(M_0+M_1+M_2\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{M_7\, a_3\, b_6\, \left(M_0+M_1+M_2\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{M_7\, a_7\, b_2\, \left(M_0+M_1+M_2\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{M_7\, b_2\, b_6\, \left(M_0+M_1+M_2\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\pi_{32}} = &\frac{a_6\, b_4\, \left(M_6+M_7\right)\, \left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&- \frac{a_5\, a_6\, \left(M_6+M_7\right)\, \left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{M_7\, a_4\, a_7\, \left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&+ \frac{M_7\, a_4\, b_6\, \left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&- \frac{a_6\, b_5\, \left(M_6+M_7\right)\, \left(M_0+M_1+M_2+M_3+M_4+M_5\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&- \frac{M_7\, a_5\, a_7\, \left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}\\
&- \frac{M_7\, a_5\, b_6\, \left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}
\end{aligned}
$$

$$+\frac{M_7\,a_7\,b_4\,\left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{M_7\,b_4\,b_6\,\left(M_0+M_1+M_2+M_3+M_4\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$-\frac{M_7\,a_7\,b_5\,\left(M_0+M_1+M_2+M_3+M_4+M_5\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$-\frac{M_7\,b_5\,b_6\,\left(M_0+M_1+M_2+M_3+M_4+M_5\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$
$$+\frac{a_4\,a_6\,\left(M_6+M_7\right)\,\left(M_0+M_1+M_2+M_3\right)}{M_0+M_1+M_2+M_3+M_4+M_5+M_6+M_7}$$