MANIPULATION OF UNCOOPERATIVE ROTATING OBJECTS IN SPACE WITH A MODULAR SELF-RECONFIGURABLE ROBOT

by

Thibault Lacourtablaise







Submitted for the degree of Doctor of Philosophy

Department of Computer Science School of Mathematical and Computer Sciences Heriot-Watt University

March 2023

The copyright in this thesis is owned by the author. Any quotation from the thesis or use of any of the information contained in it must acknowledge this thesis as the source of the quotation or information.

Abstract

The following thesis is a feasibility study for the controlled deployment of robotic scaffolding structures on randomly tumbling objects with low-magnitude gravitational field for use in space applications such as space debris removal, spacecraft maintenance and asteroids capture and mining. The proposed solution is based on the novel use of self-reconfigurable modular robots performing deployments on randomly tumbling objects as a task-driven reconfiguration or manipulation through reconfiguration. The robot design focused on its control strategy which used a decentralised modular controller with two levels. One high-level behaviour-based component and one low-level component generating commands via a constrained optimisation using either a linear or a non-linear model predictive control approach and constituting a novel control method for rotating objects via angular momentum exchanges and mass distribution changes. The controller design relied on modelling the robot modules and the object as a rotating discretised deformable continuum whose rigid part, the object, was an ellipsoid. All parameters were normalised when possible and disturbances, sensors and actuator errors were modelled respectively as biased white noises and coloured noises. The correctness of the overall control algorithm was ensured. The main objective of the MPC controllers was to control the deployment of a module from the tip of the spinning axis to the plane containing the object's centre of mass while coiling around the spinning axis and ensuring the object's rotational state tracked a reference state. Simulations showed that the nonlinear MPC controller should be preferred over a linear one and that, for a mass ratio of the object's to the module's equal to 10000, the nonlinear MPC controller is best suited to stability maintenance and meets the deployment requirement, suggesting that the proposed solution would be acceptable for medium-size objects such as asteroids.

Acknowledgements

My profound gratitude goes to all the people who supported me throughout this PhD. I cannot mention everyone but would like to thank those whose help was critical for me to reach the end of this long journey:

- Prof. Nick Taylor, my supervisor, for agreeing to take me on half-way through my PhD and for all of his encouragement, guidance and patience which really helped me put one foot in front of the other.
- Anne Murphy, the CDT Centre Manager, for all the administrative support which made all the difference to my ability to reach the finishing line.
- Hans-Nikolai Vießmann, dear friend and fellow ECR student, whose assistance with the setting up of my simulations made the production of all my data possible.
- My wife Rachel for her resilience, her advice and affection.
- My daughters, Arianwen and Agnès who have endured the many absences of their father.
- My mother for "the future of a child is the work of his mother." (Napoléon Bonaparte)



Research Thesis Submission

Please note this form should be bound into the submitted thesis.

Name:	Thibault LACOURTABLAISE				
School:	EPS (RAS C	CDT)		- تر	
Version: (i.e. First, Resubmission, Final)	Final	Degree Sought:	PhD		

Declaration

In accordance with the appropriate regulations I hereby submit my thesis and I declare that:

- 1. The thesis embodies the results of my own work and has been composed by myself
- 2. Where appropriate, I have made acknowledgement of the work of others
- 3. The thesis is the correct version for submission and is the same version as any electronic versions submitted*.
- 4. My thesis for the award referred to, deposited in the Heriot-Watt University Library, should be made available for loan or photocopying and be available via the Institutional Repository, subject to such conditions as the Librarian may require
- 5. I understand that as a student of the University I am required to abide by the Regulations of the University and to conform to its discipline.
- 6. I confirm that the thesis has been verified against plagiarism via an approved plagiarism detection application e.g. Turnitin.

ONLY for submissions including published works

Please note you are only required to complete the Inclusion of Published Works Form (page 2) if your thesis contains published works)

- 7. Where the thesis contains published outputs under Regulation 6 (9.1.2) or Regulation 43 (9) these are accompanied by a critical review which accurately describes my contribution to the research and, for multi-author outputs, a signed declaration indicating the contribution of each author (complete)
- 8. Inclusion of published outputs under Regulation 6 (9.1.2) or Regulation 43 (9) shall not constitute plagiarism.
- * Please note that it is the responsibility of the candidate to ensure that the correct version of the thesis is submitted.

Signature of Candidate:	A	Date:	12/03/2023	
----------------------------	---	-------	------------	--

Submission

Submitted By (name in capitals):	THIBAULT LACOURTABLAISE
Signature of Individual Submitting:	
Date Submitted:	12/03/2023

For Completion in the Student Service Centre (SSC)

Limited Access	Requested	Yes	No	Approved	Yes	No
E-thesis Submitted (mandatory for final theses)						
Received in the SSC by (name in capitals):				Date:		

Page 1 of 1 RDC Clerk/Apr 2019

Contents

1	Intr	oducti	ion		1
	1.1	Motiv	ation		1
	1.2	Study	aims and	objectives	4
		1.2.1	Study A	ims	4
		1.2.2	Contribu	itions and Achievements	5
		1.2.3	Solution	Outline and Objectives	7
			1.2.3.1	Chapter 2: The Origin of the Concept	7
			1.2.3.2	Chapter 3: Modelling the Problem	7
			1.2.3.3	Chapter 4: Designing the Robot Controller	8
			1.2.3.4	Chapter 5: Simulation Results	8
			1.2.3.5	Conclusion:	8
2	$\operatorname{Lit}\epsilon$	erature	Review		9
	2.1	Scope	of the Lit	terature Review	9
	2.2	Dynar	mics of As	teroids	10
		2.2.1	Asteroid	s' Origin	11
		2.2.2	A Classi	fication of Asteroids' Rotation State	12
		2.2.3	Perturba	ations Affecting the Dynamics of Asteroids	13
			2.2.3.1	Gravitational Perturbations Effects on Orbits About	
				an Asteroid	13
			2.2.3.2	Solar Radiation Perturbations Effect on Asteroids'	
				Rotation State	15
		2.2.4	Approac	h to Landing Site Choice	16
		2.2.5	Approac	h to Dynamics, Perturbations and Noise Modelling	16

2.3	Spaced	eraft design basics and requirements	18
	2.3.1	Spacecraft Design Constraints versus Robotics	18
	2.3.2	Whole-system Design Methodology	20
	2.3.3	Attitude Control Hardware	21
	2.3.4	Active Attitude Control Actuators	21
2.4	Attitue	de Control and De-spin of an Asteroid:	22
2.5	Active	space debris capturing and removal methods	24
	2.5.1	Stiff connection capturing	24
	2.5.2	Flexible connection capturing	25
2.6	Reconf	figurable robots	27
	2.6.1	Pack Robots (Tens of Modules)	28
	2.6.2	Herd Robots (Hundreds of Modules)	28
	2.6.3	Swarm Robots	28
	2.6.4	Desired Properties of SR Robots	29
	2.6.5	Types of SR robots	30
	2.6.6	Computing and Communication Infrastructure	32
	2.6.7	The Self-Reconfiguration problem	32
	2.6.8	Task-driven self-reconfiguration the way forward	33
	2.6.9	Gaits and Manipulations: Introductions to Control of SR Robots	34
		2.6.9.1 Gaits Control	34
		2.6.9.2 Manipulations Control	35
	2.6.10	Designing an SR Robot Controller: the Challenges	35
	2.6.11	From Basic Functionalities to Behaviours	36
	2.6.12	Low-Level Control and Interaction with the Environment	38
		2.6.12.1 Principle of Model Predictive Control	38
		2.6.12.2 Advantages and Drawbacks of MPC	39
		2.6.12.3 Comparison between Linear and Nonlinear MPC $$	39
		2.6.12.4 Stability of Linear and Nonlinear MPC	40
		2.6.12.5 MPC Conclusion	42
2.7	Gap a	nalysis and Solution Rational	43

3	Dyı	namic Model of a Rotating Ellipsoid with Relatively Rotating						
	Ado	litiona	l Surface	e Masses and Landing Site Selection	47			
3.1		Introd	uction: th	ne Yo-Yo de-spin mechanism towards continuum mod-				
		elling			48			
		3.1.1	The Yo-	Yo de-spin mechanism model and limitations	48			
		3.1.2	The Yo-	Yo de-spin mechanism from a continuum perspective .	50			
	3.2	System	n Modelli	ng Using Rotational Dynamics of a Deformable Medium	52			
		3.2.1	System	definition	53			
		3.2.2	Physical	Hypotheses	54			
		3.2.3	Frames	of References and Relative Angular Velocity	55			
			3.2.3.1	Frames of References	55			
			3.2.3.2	Non-Rigid Body Rotation Frame $[\mathbf{O}'']$	56			
			3.2.3.3	Relative Angular Velocity	57			
		3.2.4	Continu	ous model	58			
			3.2.4.1	The Model	58			
			3.2.4.2	Derivation valid in any vector basis	59			
		3.2.5	Discrete	Normalised Model with Point Mass Non-Rigid Parts .	60			
		3.2.6	Modellin	ng the Randomly Tumbling Object as an Ellipsoid	61			
	3.3	Non-C	ontributi	ng Rotation of an Additional Mass and Point Mass				
		Hypot	hesis		63			
	3.4	Landii	ng Site Se	election	65			
		3.4.1	Modellin	ng the Landing	65			
		3.4.2	Systems	Definitions and Properties	66			
		3.4.3	Physical	Assumptions	66			
		3.4.4	Angular	Momentum Change During Landing	67			
		3.4.5	Error Ve	ector and Sizing Method	68			
		3.4.6	Landing	Site Location Impact on the Rotational State of the				
			Target A	Asteroid	69			
			3.4.6.1	Landing Error Simulation Set Up	69			
			3.4.6.2	Landing Error Simulation Results Analysis	70			

			3.4.6.3	Landing Location: Conclusion	79
	3.5	Ellipso	oid's and l	Robot's Simulation Set Up	80
		3.5.1	Paramet	rization	80
			3.5.1.1	Normalised parameters	80
			3.5.1.2	Angular Velocity and Kinetic Energy Sizing	81
		3.5.2	Size of I	ntegration Steps	82
		3.5.3	Reference	e Angular Velocity of a Torque-Free Rotating Ellipsoid	83
			3.5.3.1	Symmetric Bodies	83
			3.5.3.2	Asymmetric Bodies	84
		3.5.4	Initial C	ondition	85
		3.5.5	ODE Fin	st Order Integration	86
		3.5.6	Perturba	tions and Noise Modelling	86
			3.5.6.1	Gyroscope Noise Model	87
			3.5.6.2	Modelling White Noise	87
			3.5.6.3	Radiation Perturbation Noise Model	88
	3.6	Conclu	usion		89
4	3.6 Ro b	Conclu	ision sign and	Control	89 90
4	3.6 Rob 4.1	Conclu oot Des Engine	usion sign and eering Ap	Control plication: Definition Requirements and Assumptions .	89 90 90
4	3.6 Rob 4.1	Conclu oot Des Engine 4.1.1	usion sign and eering Ap Robot D	Control plication: Definition Requirements and Assumptions . esign and Assumed Properties	89 90 90
4	3.6 Rob 4.1	Conclu oot Des Engine 4.1.1 4.1.2	usion sign and eering Ap Robot D Task Dea	Control plication: Definition Requirements and Assumptions . esign and Assumed Properties	89 90 90 92
4	3.6 Rob 4.1	Conclu pot Des Engine 4.1.1 4.1.2 4.1.3	usion sign and eering Ap Robot D Task De Stability	Control plication: Definition Requirements and Assumptions . esign and Assumed Properties	 89 90 90 92 93
4	3.6 Rob 4.1	Conclu oot Des Engine 4.1.1 4.1.2 4.1.3 4.1.4	ision sign and eering Ap Robot D Task De Stability Measure	Control plication: Definition Requirements and Assumptions . resign and Assumed Properties	 89 90 90 92 93 94
4	3.6 Rob 4.1	Conclu bot Des Engine 4.1.1 4.1.2 4.1.3 4.1.4 4.1.5	ision sign and eering Ap Robot D Task Des Stability Measure Low-Lev	Control plication: Definition Requirements and Assumptions . esign and Assumed Properties	 89 90 90 92 93 94 96
4	3.6 Rob 4.1	Conclu bot Des Engine 4.1.1 4.1.2 4.1.3 4.1.4 4.1.5	ision sign and eering Ap Robot D Task Des Stability Measure Low-Lev 4.1.5.1	Control plication: Definition Requirements and Assumptions . esign and Assumed Properties	 89 90 90 92 93 94 96 96
4	3.6 Rob 4.1	Conclu oot Des Engine 4.1.1 4.1.2 4.1.3 4.1.4 4.1.5	ision sign and eering App Robot D Task Des Stability Measures Low-Lev 4.1.5.1 4.1.5.2	Control plication: Definition Requirements and Assumptions . esign and Assumed Properties	 89 90 90 92 93 94 96 96 96 96
4	3.6 Rob 4.1	Conclu oot Des Engine 4.1.1 4.1.2 4.1.3 4.1.4 4.1.5	ision sign and eering Ap Robot D Task Des Stability Measure Low-Lev 4.1.5.1 4.1.5.2 4.1.5.3	Control plication: Definition Requirements and Assumptions . esign and Assumed Properties	 89 90 90 92 93 94 96 96 97
4	3.6 Rob 4.1	Conclu bot Des Engine 4.1.1 4.1.2 4.1.3 4.1.4 4.1.5	ision sign and eering Ap Robot D Task Des Stability Measure Low-Lev 4.1.5.1 4.1.5.2 4.1.5.3 4.1.5.4	Control plication: Definition Requirements and Assumptions . esign and Assumed Properties . scription Goals and Objectives . Definition . before the Robot Task Performance . el Module Controller Derivation Hypotheses . System Definition . Physical Hypotheses . Hypotheses for the Object . Landing and Set Up Prior to Deployment .	 89 90 90 92 93 94 96 96 96 97 98
4	3.6 Rob 4.1	Conclu bot Des Engine 4.1.1 4.1.2 4.1.3 4.1.4 4.1.5	usion sign and eering App Robot D Task Dea Stability Measurea Low-Lev 4.1.5.1 4.1.5.2 4.1.5.3 4.1.5.4 4.1.5.5	Control plication: Definition Requirements and Assumptions . esign and Assumed Properties	 89 90 90 92 93 94 96 96 96 97 98 98
4	3.6 Rob 4.1	Conclu oot Des Engine 4.1.1 4.1.2 4.1.3 4.1.4 4.1.5	asion sign and eering App Robot D Task Des Stability Measure Low-Lev 4.1.5.1 4.1.5.2 4.1.5.3 4.1.5.4 4.1.5.5 4.1.5.6	Control plication: Definition Requirements and Assumptions . esign and Assumed Properties	 89 90 90 92 93 94 96 96 96 97 98 98 99

		4.1.5.8 Deployment Illustrations
4.2	Low-L	evel Module Controller's Plant State Space Modelling \ldots . 102
	4.2.1	State Definition
	4.2.2	Relative Angular Velocity Definition
	4.2.3	Control Commands Definition
	4.2.4	Reference State Trajectory and Reference Control Command . 104
	4.2.5	Nonlinear State Space Model
	4.2.6	Linear State Space Model
	4.2.7	Explicit Linear State Space Model
		4.2.7.1 Single Module Case
		4.2.7.2 Multiple Modules Case
4.3	Modul	le Controller Design
	4.3.1	Unicity of Each Trajectory
	4.3.2	Admissibility
	4.3.3	Viability
	4.3.4	Feasibility
	4.3.5	Stabilisability Controllability and Choice of Control Method . 113
		4.3.5.1 Stability of the Nonlinear Model via Linearisation 113
		4.3.5.2 Controllability of the Linear Model
		4.3.5.3 Choice of a Model Predictive Control Approach 126
	4.3.6	Model Predictive Control Formulation
	4.3.7	Cost Function
	4.3.8	Optimal Control Commands Existence and Stability 130
	4.3.9	Observability of the State
	4.3.10	UKF: Unscented Kalman Filter
	4.3.11	Matlab Implementation
		4.3.11.1 MPC in Matlab
		4.3.11.2 NMPC in Matlab
4.4	Robot	Control Algorithm Design
	4.4.1	Task-Driven Self-Reconfiguration Approach

		4.4.2	Behaviours Based Algorithm Design Principles	139
		4.4.3	Robot Behaviours Based Algorithm	139
		4.4.4	Correctness of the Controller	144
			4.4.4.1 Initialisation Sequence	144
			4.4.4.2 Principle of Deployment	144
			4.4.4.3 Deployment: Getting out of the Lattice	145
			4.4.4.4 Deployment: Follow the Chain	145
			4.4.4.5 Deployment: Go On the Surface and Anchor the Chain	n145
			4.4.4.6 Conclusion	146
	4.5	Exper	imental Setup	146
	4.6	Concl	usion	147
5	Fea	sibility	Study and Validation: Simulation of the Deployment o	f
0	a Si	nglo B	Potet Module	1/8
	5 1	Object	tives and Simulations Parameters	1/8
	5.1	5 1 1	Objectives	140
		5.1.2	Scenarios and Simulations Parameters	140
		0.1.2	5.1.2.1 Simulations	149
			5.1.2.2 Controller	151
		513	Verification and Validation Metrics	152
		514	Data Processing	152
		515	Organisation of Results	155
	5.2	Contro	ol Commands: A Comparison	157
	0.2	521	Origin of the Data	157
		5.2.2	Data Description	158
		5.2.3	Conclusion	160
	5.3	Surfac	e Trajectory and Reaching the Target Location	162
	0.0	5.3.1	Origin of the Data	162
		5.3.2	Data Description	163
		5.3.3	Conclusion	167

5.4	Linear	Model Pr	redictive Control Law Performance and Stability: Spin				
	Tracki	ng Case .					
	5.4.1	Origin o	f the Data				
	5.4.2	Symmet	Symmetric Ellipsoids				
	5.4.3	Asymme	etric Ellipsoids and Limit Cases				
	5.4.4	Conclusi	on				
5.5	Linear	Model Pr	redictive Control Law: Stabilisation of the Spin Track-				
	ing Ca	se					
	5.5.1	Origin o	f the Data				
	5.5.2	Data De	scription				
	5.5.3	Sphere C	Case				
	5.5.4	Symmet	ric Case $(a = 0.5, b = 0.5, c = 1)$				
		5.5.4.1	Initial Condition Wzinit = $0.5 \dots \dots \dots \dots 197$				
		5.5.4.2	Initial Condition Wzinit = 0.75				
		5.5.4.3	Initial Condition Wzinit = 1 $\dots \dots $				
	5.5.5	Symmet	ric Case $(a = 2, b = 2, c = 1)$				
		5.5.5.1	Initial Condition Wzinit = $0.5 \dots \dots \dots \dots 206$				
		5.5.5.2	Initial Condition Wzinit = $0.75 \dots 209$				
		5.5.5.3	Initial Condition Wzinit = 1 $\dots \dots $				
	5.5.6	Asymme	etric Case $(a = 0.1, b = 0.8, c = 1)$				
		5.5.6.1	Initial Condition $\frac{h^2}{2T} = 130 \dots \dots$				
		5.5.6.2	Initial Condition $\frac{h^2}{2T} = 166 \dots \dots$				
		5.5.6.3	Initial Condition $\frac{h^2}{2T} = 202 \dots \dots$				
		5.5.6.4	Initial Condition $\frac{h^2}{2T} = 265 \dots \dots$				
		5.5.6.5	Initial Condition $\frac{h^2}{2T} = 328 \dots \dots$				
	5.5.7	Asymme	etric Case $(a = 0.8, b = 0.9, c = 1)$				
		5.5.7.1	Initial Condition $\frac{h^2}{2T} = 290 \dots 230$				
		5.5.7.2	Initial Condition $\frac{h^2}{2T} = 309 \dots \dots \dots \dots \dots \dots \dots \dots 233$				
		5.5.7.3	Initial Condition $\frac{h^2}{2T} = 328 \dots \dots \dots \dots \dots \dots \dots \dots \dots 236$				
		5.5.7.4	Initial Condition $\frac{h^2}{2T} = 345 \dots 239$				

		5.5.7.5 Initial Condition $\frac{h^2}{2T} = 362 \dots \dots$
	5.5.8	Conclusions
5.6	Comp	arisons Between Linear and Nonlinear Model Predictive Control
	Laws	With a Convergence to a State of Spin Objective
	5.6.1	Origin of the Data
	5.6.2	Data Description
	5.6.3	Sphere Case
	5.6.4	Symmetric Case $(a = 0.5, b = 0.5, c = 1)$
		5.6.4.1 Initial Condition Wzinit = $0.5 \dots \dots \dots \dots 251$
		5.6.4.2 Initial Condition Wzinit = 0.75
		5.6.4.3 Initial Condition Wzinit = $1 \dots \dots \dots 257$
	5.6.5	Symmetric Case $(a = 2, b = 2, c = 1)$
		5.6.5.1 Initial Condition Wzinit = $0.5 \dots \dots \dots 260$
		5.6.5.2 Initial Condition Wzinit = 0.75
		5.6.5.3 Initial Condition Wzinit = $1 \dots \dots \dots 266$
	5.6.6	Symmetric Case $(a = 0.1, b = 0.8, c = 1)$
		5.6.6.1 Initial Condition $\frac{h^2}{2T} = 130 \dots \dots$
		5.6.6.2 Initial Condition $\frac{h^2}{2T} = 166 \dots \dots$
		5.6.6.3 Initial Condition $\frac{h^2}{2T} = 202 \dots \dots$
		5.6.6.4 Initial Condition $\frac{h^2}{2T} = 265 \dots \dots$
		5.6.6.5 Initial Condition $\frac{h^2}{2T} = 328 \dots \dots$
	5.6.7	Symmetric Case $(a = 0.8, b = 0.9, c = 1)$
		5.6.7.1 Initial Condition $\frac{h^2}{2T} = 290 \dots \dots$
		5.6.7.2 Initial Condition $\frac{h^2}{2T} = 309 \dots 287$
		5.6.7.3 Initial Condition $\frac{h^2}{2T} = 328 \dots \dots$
		5.6.7.4 Initial Condition $\frac{h^2}{2T} = 345 \dots 293$
		5.6.7.5 Initial Condition $\frac{h^2}{2T} = 362 \dots \dots$
	5.6.8	Conclusions
5.7	Conclu	usions and Recommendations
	5.7.1	Influence of the Mass Ratio

		5.7.2	Influenc	e of the Object's Shape	302
			5.7.2.1	Influence of the Object Shape in Relation to The	
				Sphere Limit case	302
			5.7.2.2	Influence of the Symmetric Shapes	303
			5.7.2.3	Influence of the Asymmetric Shapes	303
		5.7.3	Influenc	e of the Initial Conditions	304
		5.7.4	Energy	Levels and Controller Validation	305
		5.7.5	Low-Lev	vel Controller Performance and Final Recommendations	306
			5.7.5.1	Low-Level Controller Design Limitations	307
			5.7.5.2	Low-Level Controller Performance Assessment	307
			5.7.5.3	Final Recommendations	309
6	Con	clusio	n		312
	6.1	Work	Undertak	en, Contributions and Conclusions	312
	6.2	Limita	ations and	l Future Work	320
		6.2.1	Further	Evaluations	320
		6.2.2	Further	Explorations	321
\mathbf{A}	Phy	sical N	/Iodel De	erivation with Consistent Vector Dimension Con-	-
	ven	tion			323
	A.1	First 7	Ferm		324
	A.2	Second	d Term		326
	A.3	Third	Term .		327
	A.4	Fourth	n Term .		328
	A.5	Conclu	usion		328
в	Jaco	obian d	of the St	ate Function	329
	B.1	Main	Notations	and Definitions	330
	B.2	Jacobi	ians of Po	osition Vectors and Angular Velocities	331
		B.2.1	Jacobia	n of Basis Vectors	331
		B.2.2	Jacobia	n of r the Radius of the Particle's Position	332
		B.2.3	Jacobia	n of Relative Velocities	334

		B.2.4	Jacobian of Relative Angular Velocities	335
	B.3	Jacobi	an of g with Respect to the State Variables $\ldots \ldots \ldots \ldots$	336
		B.3.1	Jacobian of the Particles State Variables	336
			B.3.1.1 $\dot{\vec{\Theta}}$ with respect to $\vec{\Omega}$:	336
			B.3.1.2 $\dot{\vec{\Theta}}$ with respect to $\vec{\Theta}$:	336
			B.3.1.3 $\dot{\vec{\Theta}}$ with respect to $\dot{\vec{\Theta}}$:	336
			B.3.1.4 $\ddot{\vec{\Theta}}$ with respect to $\vec{\Omega}$:	337
			B.3.1.5 $\ddot{\vec{\Theta}}$ with respect to $\vec{\Theta}$:	337
			B.3.1.6 $\ddot{\vec{\Theta}}$ with respect to $\dot{\vec{\Theta}}$:	337
		B.3.2	Jacobian of f With Respect to the Rigid Angular Velocity $\ . \ .$	337
		B.3.3	Jacobian of f With Respect to the Angular Positions of the	
			Particles	337
			B.3.3.1 $\begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}} \end{bmatrix}_{MoI}$	338
			B.3.3.2 $\begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}} \end{bmatrix}$	338
		B.3.4	Jacobian of f With Respect to the Angular Velocities of the	
			Particles	339
	B.4	Jacobi	an of g with Respect to the Manipulated Variables \ldots \ldots	340
	B.5	Jacobi	an of the Moment of Inertia Matrices	341
		B.5.1	Jacobian of the Moment of Inertia $[\mathbf{I}_n]$ \hdots	341
		B.5.2	Jacobian of the Inverse of the Moment of Inertia $[\mathbf{I}_n]^{-1}$	342
	B.6	Jacobi	an of h at the Origin $(0,0)$	342
С	Yo-y	yo de-s	spin mechanism model verification	345
П	Hol	onomi	c Constraint for Ellipsoidal Objects with Consistent Vec-	_
D	tor	Dimen	ision Convention	351
		2		001
\mathbf{E}	\mathbf{Exp}	licit S	olution of the Euler equation for Asymetric Body with	l
	Zero	o Exte	rnal Moment	355
	E.1	Genera	al solution of the Euler equations for asymmetric bodies \ldots .	356
	E.2	Mappi	ng k with the values of $\frac{h^2}{2T}$	359
	E.3	Euler o	equation solution	361

E.4	Proof of the Abramowitz-Stegun formulae	 	 	3	363
Bibliog	graphy			3	64

List of Symbols and Units

Symbol	Description	Units
$\overrightarrow{\mathbf{v}}$	Vector Notation	
$\overrightarrow{\mathbf{v}}^{T}$	Vector Transpose Notation	
[A]	Matrix Notation	
$[\mathbf{A}]^T$	Matrix Transpose Notation	
\mathbf{I}_{z}	Moment of Inertia about Z axis	$kg\cdot m^2$
[I]	Moment of Inertia Matrix	$kg\cdot m^2$
det	Matrix Determinant	
$\overrightarrow{\mathbf{L}}_{[Total]}$	System Total Angular Momentum	$kg\cdot m^2\cdot s^{-1}$
$L ext{ or } h$	Magnitude of the Angular Momentum	$kg\cdot m^2\cdot s^{-1}$
$(O, \overrightarrow{\mathbf{X}}, \overrightarrow{\mathbf{Y}}, \overrightarrow{\mathbf{Z}})$	Inertial Frame with Origin O	
$(O, \overrightarrow{\mathbf{i}}_{Body}, \overrightarrow{\mathbf{j}}_{Body}, \overrightarrow{\mathbf{k}}_{Body})$	Body Frame with Origin O	
$(O, \overrightarrow{\mathbf{e}}_r, \overrightarrow{\mathbf{e}}_\theta, \overrightarrow{\mathbf{e}}_\phi)$	Module Frame with Origin O	
$(heta,\phi)$	Spherical Coordinates Angles	rad
T	Symbol for Energy	$J = kg \cdot m^2 \cdot s^{-2}$
•	Scalar Product	
\wedge	Vector Cross Product	
\otimes	Dyadic Product	
•	Euclidean Norm	
$\overrightarrow{\mathbf{x}}$	Particle Position Vector in the Continuum	m
$\overrightarrow{\mathbf{x}}' = r \overrightarrow{\mathbf{e}_{\mathbf{r}}}$	Spherical Coordinate Notation	
r	Radial Length	m
$\overrightarrow{\mathbf{\Omega}}(\overrightarrow{\mathbf{x}},t)$	Total Angular Velocity of the Continuum Configurations	$rad \cdot s^{-1}$
$ \overrightarrow{\mathbf{\Omega}}(t)$	Rigid Body Angular Velocity	$rad \cdot s^{-1}$
$ec{\Psi}(\overrightarrow{\mathbf{x}},t)$	Particle's Additional Angular Velocity	$rad \cdot s^{-1}$

Symbol	Description	Units
$[\mathbf{I}](\overrightarrow{\mathbf{x}},\mathbf{t})$	Moment of Inertia Matrix over the Entire Continuum	$kg\cdot m^2$
$\frac{\dot{\mathbf{x}}_{0}}{\mathbf{x}_{0}''}$	Particle Velocity in Frame $[O'']$	$m \cdot s^{-1}$
$\frac{\ddot{\mathbf{x}}_{0}}{\mathbf{x}_{0}''}$	Particle Acceleration in Frame $[O'']$	$m \cdot s^{-2}$
$\overrightarrow{\mathbf{M}}_{Body}$	Moment due to Body Forces	$N \cdot m$
$\overrightarrow{\mathbf{M}}_{Stresses}$	Moment due to Material Stress	$N \cdot m$
a, b, c MR	Half Length of an Ellipsoid Principal Axes Mass Ratio between the Object and a Robot Module	m
$\overrightarrow{\varepsilon}_t$	Error Vector Added to the Target Angular Velocity	$rad \cdot s^{-1}$
	Sampling Feriod	S II
<i>JS</i>	Components of the Dirid Angular Velocity	Πz
$\omega_x, \omega_y, \omega_z$	Nutation Angle	raa · s ·
ν f	Plant Non Linear Model Function	744
J	State Space Model Function	
y I	Cost Function	
$\vec{\mathbf{X}} = \begin{bmatrix} \vec{\Omega} \\ \vec{\Theta} \\ \vec{\Theta} \end{bmatrix}$	Model State	
$\overrightarrow{\boldsymbol{\Theta}} = \begin{bmatrix} \theta_1 \\ \phi_1 \\ \vdots \\ \theta_{n_r} \\ \phi_{n_r} \end{bmatrix}$	Coordinates of the Robot Modules on the Surface of the Object	rad
$\overrightarrow{\mathbf{u}}=\overset{\ddot{\mathbf{\Theta}}}{\mathbf{\Theta}}$	Control Commands of All Modules	$rad\cdot s^{-2}$
$[\Lambda_{\overrightarrow{\Omega}}], [\Lambda_{\overrightarrow{\Theta}}], [\Lambda_{\overrightarrow{\Theta}}], [\Lambda_{\overrightarrow{u}}]$	Positive Semi-Definite Matrices Weights of the Cost Function	
$\overrightarrow{\mathbf{X}}_{ref}$	Reference Trajectory	
$\overrightarrow{\mathbf{u}}_{ref}$	Reference Control Commands	
X	State Set	
$\mathbb{U}(x)$	Control Constraint Set	
X	Set of All Possible States	
U	Set of All Possible Control Commands	

List of Tables

3.1	Ellipsoids Dimensions Normalised by the Body's Z Semi-Axis Length 70
3.2	Dimensionless Mass Ratios Ellipsoid to Robot
3.3	Dimensionless $\frac{h^2}{2T}$ Parameter Values
3.4	Optimal Landing Site in Spherical Angles (θ, ϕ) in Degrees 70
3.5	Bias and Noise Magnitudes
4.1	Influence of the Object's Shape on the Stability of the System's Spin 115
4.2	Rank of the Controllability Matrix for the Sphere
4.3	Rank of the Controllability Matrix for the Ellipsoid $a=\frac{1}{2} b=\frac{1}{2} c=1$. 120
4.4	Rank of the Controllability Matrix for the Ellipsoid $a=\frac{1}{10} b=\frac{1}{10} c=1$. 121
4.5	Rank of the Controllability Matrix for the Ellipsoid $a=\frac{8}{10}$ $b=\frac{8}{10}$ $c=1$. 122
4.6	Rank of the Controllability Matrix for the Ellipsoid a=10 b=10 c=1 . 122
4.7	Rank of the Controllability Matrix for the Ellipsoid a=2 b=2 c=1 123
4.8	Rank of the Controllability Matrix for the Ellipsoid a= $\frac{5}{4}$ b= $\frac{5}{4}$ c=1 123
4.9	Rank of the Controllability Matrix for the Ellipsoid a= $\frac{1}{10}$ b= $\frac{8}{10}$ c=1 . 124
4.10	Rank of the Controllability Matrix for the Ellipsoid a= $\frac{1}{3}$ b= $\frac{1}{2}$ c=1 124
4.11	Rank of the Controllability Matrix for the Ellipsoid a= $\frac{8}{10}$ b= $\frac{9}{10}$ c=1 $$. 125
5.1	Ellipsoids Normalised Semi-Axes Lengths
5.2	Mass Ratios: Object to Module
5.3	$\omega_{Z_{init}}$
5.4	Asymmetric Bodies Parameters $\frac{h^2}{2T}$
5.5	Ellipsoids Normalised Semi-Axes Lengths
5.6	Ellipsoids Normalised Semi-Axes Lengths
5.7	Ellipsoids Normalised Semi-Axes Lengths

List of Figures

2.1	A Chain Type Robot: the Polybot [23]	30
2.2	A Lattice Type Robot: the ATRON [13]	31
2.3	A Hybrid Type Robot: the M-TRAN [1] $\ldots \ldots \ldots \ldots \ldots$	31
3.1	Yo-Yo De-spin Mechanism Diagram.	49
3.2	System Diagram.	54
3.3	Example of a System Deployment	54
3.4	Oblate Ellipsoid.	62
3.5	Prolate Ellipsoid.	62
3.6	Sphere Landing Normalised Error	72
3.7	Prolate Ellipsoid Landing Normalised Error.	73
3.8	Prolate Ellipsoid Landing Normalised Error	74
3.9	Prolate Ellipsoid Landing Normalised Error	74
3.10	Oblate Ellipsoid Landing Normalised Error	75
3.11	Oblate Ellipsoid Landing Normalised Error	76
3.12	Oblate Ellipsoid Landing Normalised Error	76
3.13	Asymmetric Ellipsoid Landing Normalised Error	77
3.14	Asymmetric Ellipsoid Landing Normalised Error	78
3.15	Asymmetric Ellipsoid Landing Normalised Error	78
4.1	Robots at Landing Sites (the Two Tips of the Spin Axis) in Lattice	
	Form	100
4.2	Symmetrical Deployment of the Two Robots from Both Tips of the	
	Spin Axis.	101
4.3	Stability Limit vs. Mass Ratio (Small MR).	116

4.4	Stability Limit vs. Mass Ratio (Medium MR)
4.5	Stability Limit vs. Mass Ratio (Large MR)
4.6	Robot Algorithm Top Level
4.7	Robot Algorithm Initialisation
4.8	Robot Algorithm Deployment
5.1	Linear Results for $\ddot{\theta}$
5.2	Linear Results for $\ddot{\phi}$
5.3	Non Linear Results for $\ddot{\theta}$
5.4	Non Linear Results for $\ddot{\phi}$
5.5	Comparison Between the Control Commands Generated by the Lin-
	ear and Nonlinear Controllers
5.6	a=0.1 b=0.1 c=1
5.7	a=0.8 b=0.8 c=1
5.8	a=2 b=2 c=1
5.9	a=0.5 b=0.5 c=1
5.10	a=1.25 b=1.25 c=1
5.11	a=10 b=10 c=1
5.12	Module Surface Trajectory with Direction of Travel for Symmetric
	Ellipsoids
5.13	a=0.1 b=0.8 c=1
5.14	a=0.8 b=0.9 c=1
5.15	a=0.33 b=0.5 c=1
5.16	Sphere
5.17	Module Surface Trajectory with Direction of Travel for Asymmetric
	Ellipsoids and the Spherical Stability Limit Case
5.18	Phase Diagram
5.19	Nutation vs. Time
5.20	Kinetic Energy vs. Time
5.21	Nutation vs. Kinetic Energy
5.22	Stability for the Prolate Case (a=0.1 b=0.1 c=1) $\dots \dots \dots$

5.23 Phase Diagram
5.24 Nutation vs. Time $\ldots \ldots 172$
5.25 Kinetic Energy vs. Time
5.26 Nutation vs. Kinetic Energy
5.27 Stability for the Prolate Case (a=0.5 b=0.5 c=1) $\dots \dots \dots$
5.28 Phase Diagram
5.29 Nutation vs. Time $\ldots \ldots 174$
5.30 Kinetic Energy vs. Time
5.31 Nutation vs. Kinetic Energy
5.32 Stability for the Prolate Case (a=0.8 b=0.8 c=1) $\dots \dots \dots$
5.33 Phase Diagram
5.34 Nutation vs. Time $\ldots \ldots 176$
5.35 Kinetic Energy vs. Time
5.36 Nutation vs. Kinetic Energy
5.37 Stability for the Oblate Case (a=1.25 b=1.25 c=1) $\dots \dots \dots$
5.38 Phase Diagram
5.39 Nutation vs. Time
5.40 Kinetic Energy vs. Time
5.41 Nutation vs. Kinetic Energy
5.42 Stability for the Oblate Case (a=2 b=2 c=1)
5.43 Phase Diagram
5.44 Nutation vs. Time $\ldots \ldots 180$
5.45 Kinetic Energy vs. Time
5.46 Nutation vs. Kinetic Energy
5.47 Stability for the Oblate Case (a=10 b=10 c=1) $\dots \dots \dots$
5.48 Phase Diagram
5.49 Nutation vs. Time
5.50 Kinetic Energy vs. Time
5.51 Nutation vs. Kinetic energy
5.52 Stability for the Sphere Case

5.53 Phase Diagram
5.54 Nutation vs. Time $\ldots \ldots 184$
5.55 Kinetic Energy vs. Time
5.56 Nutation vs. Kinetic Energy
5.57 Stability for the Asymmetric Case (a=0.1 b=0.8 c=1)
5.58 Phase Diagram
5.59 Nutation vs. Time
5.60 Kinetic Energy vs. Time
5.61 Nutation vs. Kinetic Energy
5.62 Stability for the Asymmetric Case (a=0.33 b=0.5 c=1) \hdots
5.63 Phase Diagram
5.64 Nutation vs. Time
5.65 Kinetic Energy vs. Time
5.66 Nutation vs. Kinetic Energy
5.67 Stability for the Asymmetric Case (a=0.8 b=0.9 c=1)
5.68 Phase Diagrams for the Z Axis (Sphere)
5.69 Phase Diagrams for the Z Axis (Sphere)
5.70 Phase Diagrams for the Z Axis (Sphere)
5.71 Phase Diagrams for the Z Axis (Sphere)
5.72 Sphere Z Axis Stability Free Motion
5.73 Wzinit = 0.5
5.74 Wzinit = 0.5
5.75 Wzinit = 0.5
5.76 Wzinit = 0.5
5.77 Phase Diagrams for the Z Axis (a=0.5 b=0.5 c=1)
5.78 Wzinit = 0.5
5.79 Wzinit = 0.5
5.80 Wzinit = 0.5
5.81 Wzinit = 0.5
5.82 Phase Diagrams for the Z Axis (a=0.5 b=0.5 c=1)

5.83 Wzinit = 0.75
5.84 Wzinit = 0.75
5.85 Wzinit = 0.75
5.86 Wzinit = 0.75
5.87 Phase Diagrams for the Z Axis (a=0.5 b=0.5 c=1)
5.88 Wzinit = 0.75
5.89 Wzinit = 0.75
5.90 Wzinit = 0.75
5.91 Wzinit = 0.75
5.92 Phase Diagrams for the Z Axis (a=0.5 b=0.5 c=1)
5.93 Wzinit = 1 $\dots \dots $
5.94 Wzinit = 1 $\dots \dots $
5.95 Wzinit = 1 $\dots \dots $
5.96 Wzinit = 1 $\dots \dots $
5.97 Phase Diagrams for the Z Axis (a=0.5 b=0.5 c=1)
5.98 Wzinit = 1 $\dots \dots $
5.99 Wzinit = 1 $\dots \dots $
$5.100 Wzinit = 1 \dots \dots$
5.101Wzinit = 1
5.102 Phase Diagrams for the Z Axis (a=0.5 b=0.5 c=1)
5.103Wzinit = 0.5
5.104Wzinit = 0.5
5.105Wzinit = 0.5
5.106Wzinit = 0.5
5.107 Phase Diagrams for the Z Axis (a=2 b=2 c=1) $\dots \dots \dots$
5.108Wzinit = 0.5
5.109Wzinit = 0.5
5.110 Wzinit = 0.5
5.111Wzinit = 0.5
5.112Phase Diagrams for the Z Axis (a=2 b=2 c=1) $\dots \dots \dots$

5.113Wzinit = 0.75
5.114Wzinit = 0.75
5.115Wzinit = 0.75
5.116Wzinit = 0.75
5.117Phase Diagrams for the Z Axis (a=2 b=2 c=1) $\dots \dots \dots$
5.118Wzinit = 0.75
5.119Wzinit = 0.75
5.120Wzinit = 0.75
5.121Wzinit = 0.75
5.122 Phase Diagrams for the Z Axis (a=2 b=2 c=1) $\dots \dots \dots$
5.123Wzinit = 1
5.124Wzinit = 1
5.125Wzinit = 1
5.126Wzinit = 1
5.127 Phase Diagrams for the Z Axis (a=2 b=2 c=1) $\dots \dots \dots$
5.128Wzinit = 1
5.129Wzinit = 1
5.130Wzinit = 1
5.131Wzinit = 1
5.132 Phase Diagrams for the Z Axis (a=2 b=2 c=1) $\dots \dots \dots$
$5.133 \frac{h^2}{2T} = 130 \dots 216$
$5.134 \frac{h^2}{2T} = 130 \dots 216$
$5.135 \frac{h^2}{2T} = 130 \dots 216$
$5.136 \frac{h^2}{2T} = 130 \dots 216$
5.137 Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)
$5.138 \frac{h^2}{2T} = 130 \dots 217$
$5.139\frac{h^2}{2T} = 130$
$5.140\frac{h^2}{2T} = 130$
$5.141\frac{h^2}{2T} = 130$
5.142Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)

$5.143\frac{h^2}{2T} = 166$
$5.144 \frac{h^2}{2T} = 166 \dots 219$
$5.145 \frac{h^2}{2T} = 166 \dots 219$
$5.146 \frac{h^2}{2T} = 166 \dots 219$
5.147Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)
$5.148 \frac{h^2}{2T} = 166 \dots 220$
$5.149\frac{h^2}{2T} = 166$
$5.150\frac{h^2}{2T} = 166$
$5.151 \frac{h^2}{2T} = 166 \dots 220$
5.152Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)
$5.153\frac{h^2}{2T} = 202$
$5.154\frac{h^2}{2T} = 202$
$5.155 \frac{h^2}{2T} = 202$
$5.156 \frac{h^2}{2T} = 202$
5.157 Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)
$5.158 \frac{h^2}{2T} = 202$
$5.159 \frac{h^2}{2T} = 202$
$5.160\frac{h^2}{2T} = 202$
$5.161\frac{h^2}{2T} = 202$
5.162Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)
$5.163\frac{h^2}{2T} = 265$
$5.164\frac{h^2}{2T} = 265$
$5.165 \frac{h^2}{2T} = 265 \dots 225$
$5.166 \frac{h^2}{2T} = 265 \dots 225$
5.167 Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)
$5.168 \frac{h^2}{2T} = 265$
$5.169\frac{h^2}{2T} = 265$
$5.170\frac{h^2}{2T} = 265$
$5.171\frac{h^2}{2T} = 265$
5.172Phase Diagrams for the Z Axis $(a=0.1 b=0.8 c=1) \dots 226$

$5.173\frac{h^2}{2T} = 328$
$5.174\frac{h^2}{2T} = 328$
$5.175 \frac{h^2}{2T} = 328 \dots 228$
$5.176\frac{h^2}{2T} = 328$
5.177Phase Diagrams for the Z Axis $(a=0.1 b=0.8 c=1) \dots 228$
$5.178 \frac{h^2}{2T} = 328 \dots 229$
$5.179\frac{h^2}{2T} = 328$
$5.180\frac{h^2}{2T} = 328$
$5.181\frac{h^2}{2T} = 328$
5.182Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)
$5.183 \frac{h^2}{2T} = 290 \dots 231$
$5.184 \frac{h^2}{2T} = 290 \dots 231$
$5.185 \frac{h^2}{2T} = 290 \dots 231$
$5.186 \frac{h^2}{2T} = 290 \dots 231$
5.187 Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)
$5.188 \frac{h^2}{2T} = 290 \dots 232$
$5.189\frac{h^2}{2T} = 290 \dots 232$
$5.190\frac{h^2}{2T} = 290 \dots 232$
$5.191\frac{h^2}{2T} = 290$
5.192Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)
$5.193\frac{h^2}{2T} = 309$
$5.194 \frac{h^2}{2T} = 309 \dots 234$
$5.195 \frac{h^2}{2T} = 309 \dots 234$
$5.196\frac{h^2}{2T} = 309$
5.197 Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)
$5.198 \frac{h^2}{2T} = 309 \dots 235$
$5.199\frac{h^2}{2T} = 309$
$5.200 \frac{h^2}{2T} = 309 \dots 235$
$5.201\frac{h^2}{2T} = 309$
5.202Phase Diagrams for the Z Axis $(a=0.8 b=0.9 c=1) \dots 235$

$5.203\frac{h^2}{2T} = 328$
$5.204 \frac{h^2}{2T} = 328$
$5.205 \frac{h^2}{2T} = 328$
$5.206\frac{h^2}{2T} = 328$
5.207 Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)
$5.208 \frac{h^2}{2T} = 328$
$5.209 \frac{h^2}{2T} = 328$
$5.210\frac{h^2}{2T} = 328$
$5.211\frac{h^2}{2T} = 328$
5.212 Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)
$5.213 \frac{h^2}{2T} = 345 \dots 240$
$5.214 \frac{h^2}{2T} = 345 \dots 240$
$5.215 \frac{h^2}{2T} = 345$
$5.216\frac{h^2}{2T} = 345$
5.217 Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)
$5.218 \frac{h^2}{2T} = 345$
$5.219\frac{h^2}{2T} = 345$
$5.220 \frac{h^2}{2T} = 345$
$5.221\frac{h^2}{2T} = 345$
5.222 Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)
$5.223 \frac{h^2}{2T} = 362 \dots 243$
$5.224 \frac{h^2}{2T} = 362 \dots 243$
$5.225 \frac{h^2}{2T} = 362 \dots 243$
$5.226 \frac{h^2}{2T} = 362 \dots 243$
5.227Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)
$5.228 \frac{h^2}{2T} = 362 \dots 244$
$5.229 \frac{h^2}{2T} = 362 \dots 244$
$5.230 \frac{h^2}{2T} = 362 \dots 244$
$5.231\frac{h^2}{2T} = 362$
5.232Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)

5.233Sphere
5.234Sphere
5.235Sphere
5.236Sphere
5.237 Stability Diagrams for the Sphere (a=1 b=1 c=1) \ldots
5.238Sphere
5.239Sphere
5.240Sphere
5.241Sphere
5.242 Stability Diagrams for the Sphere (a=1 b=1 c=1) \hdots
5.243Wzinit = 0.5
5.244Wzinit = 0.5
5.245Wzinit = 0.5
5.246Wzinit = 0.5
5.247 Phase Diagrams for the Z Axis Wzinit = 0.5 (a = 0.5 b= 0.5 c=1) $$ 252
5.248Wzinit = 0.5
5.249Wzinit = 0.5
5.250Wzinit = 0.5
5.251Wzinit = 0.5
5.252 Phase Diagrams for the Z Axis Wzinit = 0.5 (a = 0.5 b= 0.5 c=1) $$ 253
5.253Wzinit = 0.75
5.254Wzinit = 0.75
5.255Wzinit = 0.75
5.256Wzinit = 0.75
5.257 Phase Diagrams for the Z Axis Wzinit = 0.75 (a = 0.5 b= 0.5 c=1) $$. 255
5.258Wzinit = 0.75
5.259Wzinit = 0.75
5.260Wzinit = 0.75
5.261Wzinit = 0.75
5.262 Phase Diagrams for the Z Axis Wzinit = 0.75 (a = 0.5 b= 0.5 c=1) $$. 256

5.263Wzinit = 1
5.264Wzinit = 1
5.265Wzinit = 1
5.266Wzinit = 1
5.267 Phase Diagrams for the Z Axis Wzinit = 1 (a= 0.5 b= 0.5 c= 1) $\hfill\h$
5.268Wzinit = 1
5.269Wzinit = 1
5.270Wzinit = 1
5.271Wzinit = 1
5.272 Phase Diagrams for the Z Axis Wzinit = 1 (a= 0.5 b= 0.5 c= 1) $\ .$ 259
5.273Wzinit = 0.5
5.274Wzinit = 0.5
5.275Wzinit = 0.5
5.276Wzinit = 0.5
5.277 Phase Diagrams for the Z Axis Wzinit = 0.5 (a= 2 b= 2 c= 1) $\ . \ . \ . \ . \ 261$
5.278Wzinit = 0.5
5.279Wzinit = 0.5
5.280 Wzinit = 0.5
5.281Wzinit = 0.5
5.282 Phase Diagrams for the Z Axis Wzinit = 0.5 (a= 2 b= 2 c= 1) $\ .$ 262
5.283Wzinit = 0.75
5.284Wzinit = 0.75
5.285Wzinit = 0.75
5.286Wzinit = 0.75
5.287 Phase Diagrams for the Z Axis Wzinit = 0.75 (a= 2 b= 2 c= 1) $\ .$ 264
5.288Wzinit = 0.75
5.289Wzinit = 0.75
5.290 Wzinit = 0.75
5.291Wzinit = 0.75
5.292 Phase Diagrams for the Z Axis Wzinit = 0.75 (a = 2 b= 2 c=1) $\ .$ 265

5.293Wzinit = 1
$5.294 Wzinit = 1 \dots \dots \dots \dots \dots \dots \dots \dots \dots $
5.295Wzinit = 1
5.296 Wzinit = 1
5.297 Phase Diagrams for the Z Axis Wzinit = 1 (a= 2 b= 2 c= 1) 267
5.298Wzinit = 1
5.299Wzinit = 1
5.300 Wzinit = 1
5.301Wzinit = 1
5.302 Phase Diagrams for the Z Axis Wzinit = 1 (a= 2 b= 2 c=1) 268
$5.303\frac{h^2}{2T} = 130$
$5.304\frac{h^2}{2T} = 130$
$5.305 \frac{h^2}{2T} = 130 \dots 270$
$5.306\frac{h^2}{2T} = 130$
5.307 Phase Diagrams for the Z Axis HT = 130 (a= 0.1 b= 0.8 c= 1) 270
$5.308 \frac{h^2}{2T} = 130 \dots 271$
$5.309\frac{h^2}{2T} = 130$
$5.310\frac{h^2}{2T} = 130$
$5.311\frac{h^2}{2T} = 130$
5.312 Phase Diagrams for the Z Axis HT = 130 (a= 0.1 b= 0.8 c= 1) 271
$5.313\frac{h^2}{2T} = 166$
$5.314\frac{h^2}{2T} = 166$
$5.315 \frac{h^2}{2T} = 166 \dots 273$
$5.316\frac{h^2}{2T} = 166$
5.317 Phase Diagrams for the Z Axis HT = 166 (a= 0.1 b= 0.8 c= 1) 273
$5.318 \frac{h^2}{2T} = 166 \dots 274$
$5.319\frac{h^2}{2T} = 166$
$5.320\frac{h^2}{2T} = 166$
$5.321\frac{h^2}{2T} = 166$
5.322 Phase Diagrams for the Z Axis HT = 166 (a= 0.1 b= 0.8 c= 1) 274

$5.323\frac{h^2}{2T} = 202$
$5.324 \frac{h^2}{2T} = 202 \dots 276$
$5.325 \frac{h^2}{2T} = 202 \dots 276$
$5.326 \frac{h^2}{2T} = 202 \dots 276$
5.327 Phase Diagrams for the Z Axis HT = 202 (a= 0.1 b= 0.8 c= 1) 276
$5.328 \frac{h^2}{2T} = 202 \dots 277$
$5.329 \frac{h^2}{2T} = 202 \dots 277$
$5.330 \frac{h^2}{2T} = 202 \dots 277$
$5.331\frac{h^2}{2T} = 202$
5.332 Phase Diagrams for the Z Axis HT = 202 (a= 0.1 b= 0.8 c= 1) 277
$5.333 \frac{h^2}{2T} = 265 \dots 279$
$5.334 \frac{h^2}{2T} = 265 \dots 279$
$5.335 \frac{h^2}{2T} = 265 \dots 279$
$5.336 \frac{h^2}{2T} = 265 \dots 279$
5.337 Phase Diagrams for the Z Axis HT = 265 (a= 0.1 b= 0.8 c= 1) 279
$5.338 \frac{h^2}{2T} = 265 \dots 280$
$5.339 \frac{h^2}{2T} = 265 \dots 280$
$5.340\frac{h^2}{2T} = 265$
$5.341\frac{h^2}{2T} = 265$
5.342 Phase Diagrams for the Z Axis HT = 265 (a= 0.1 b= 0.8 c= 1) 280
$5.343 \frac{h^2}{2T} = 328 \dots 282$
$5.344 \frac{h^2}{2T} = 328 \dots 282$
$5.345 \frac{h^2}{2T} = 328 \dots 282$
$5.346\frac{h^2}{2T} = 328$
5.347 Phase Diagrams for the Z Axis HT = 328 (a= 0.1 b= 0.8 c= 1) 282
$5.348 \frac{h^2}{2T} = 328 \dots 283$
$5.349 \frac{h^2}{2T} = 328 \dots 283$
$5.350\frac{h^2}{2T} = 328$
$5.351\frac{h^2}{2T} = 328$
5.352 Phase Diagrams for the Z Axis HT = 328 (a= 0.1 b= 0.8 c= 1) 283

$5.353 \frac{h^2}{2T} = 290 \dots 285$
$5.354 \frac{h^2}{2T} = 290 \dots 285$
$5.355 \frac{h^2}{2T} = 290 \dots 285$
$5.356 \frac{h^2}{2T} = 290 \dots 285$
5.357 Phase Diagrams for the Z Axis HT = 290 (a= 0.8 b = 0.9 c=1) 285
$5.358 \frac{h^2}{2T} = 290 \dots 286$
$5.359 \frac{h^2}{2T} = 290 \dots 286$
$5.360 \frac{h^2}{2T} = 290 \dots 286$
$5.361\frac{h^2}{2T} = 290$
5.362 Phase Diagrams for the Z Axis HT = 290 (a= 0.8 b= 0.9 c= 1) 286
$5.363\frac{h^2}{2T} = 309$
$5.364 \frac{h^2}{2T} = 309 \dots 288$
$5.365 \frac{h^2}{2T} = 309 \dots 288$
$5.366 \frac{h^2}{2T} = 309 \dots 288$
5.367 Phase Diagrams for the Z Axis HT = 309 (a= 0.8 b= 0.9 c= 1) 288
$5.368 \frac{h^2}{2T} = 309 \dots 289$
$5.369\frac{h^2}{2T} = 309$
$5.370\frac{h^2}{2T} = 309$
$5.371\frac{h^2}{2T} = 309$
5.372 Phase Diagrams for the Z Axis HT = 309 (a= 0.8 b= 0.9 c= 1) 289
$5.373 \frac{h^2}{2T} = 328 \dots \dots$
$5.374\frac{h^2}{2T} = 328$
$5.375\frac{h^2}{2T} = 328$
$5.376\frac{h^2}{2T} = 328$
5.377 Phase Diagrams for the Z Axis HT = 328 (a= 0.8 b= 0.9 c= 1) 291
$5.378 \frac{h^2}{2T} = 328 \dots \dots$
$5.379\frac{h^2}{2T} = 328$
$5.380\frac{h^2}{2T} = 328$
$5.381\frac{h^2}{2T} = 328$
5.382 Phase Diagrams for the Z Axis HT = 328 (a= 0.8 b= 0.9 c= 1) 292

$5.383\frac{h^2}{2T} = 345$
$5.384 \frac{h^2}{2T} = 345 \dots 294$
$5.385 \frac{h^2}{2T} = 345 \dots 294$
$5.386 \frac{h^2}{2T} = 345 \dots 294$
5.387 Phase Diagrams for the Z Axis HT = 345 (a = 0.8 b= 0.9 c=1) 294
$5.388 \frac{h^2}{2T} = 345 \dots 295$
$5.389 \frac{h^2}{2T} = 345 \dots 295$
$5.390 \frac{h^2}{2T} = 345 \dots 295$
$5.391\frac{h^2}{2T} = 345$
5.392 Phase Diagrams for the Z Axis HT = 345 (a = 0.8 b= 0.9 c=1) 295
$5.393\frac{h^2}{2T} = 362$
$5.394\frac{h^2}{2T} = 362$
$5.395 \frac{h^2}{2T} = 362 \dots 297$
$5.396\frac{h^2}{2T} = 362$
5.397 Phase Diagrams for the Z Axis HT = 362 (a= 0.8 b = 0.9 c=1) 297
$5.398 \frac{h^2}{2T} = 362 \dots 298$
$5.399\frac{h^2}{2T} = 362$
$5.400\frac{h^2}{2T} = 362$
$5.401\frac{h^2}{2T} = 362$
5.402Phase Diagrams for the Z Axis $\frac{h^2}{2T} = 362$ (a= 0.8 b= 0.9 c=1) 298
C.1 Yo-Yo De-spin Mechanism Diagram

Chapter 1

Introduction

1.1 Motivation

On 25/11/2015, the US congress passed the U.S. Commercial Space Launch Competitiveness Act authorising U.S. citizens to explore and recover space resources for commercial purposes [39]. This reflects that the space industry now represents a significant and strategic part of the global economy. In 2014, it was worth 330 billion dollars worldwide. This amount had doubled over the ten year between 2005 and 2014 growing at an average 7 percents compound annual rate. Three quarters of this output was the result of commercial activities and revenues from commercial space products and services accounted for more than a third of the global industry's worth [22].

Mining space resources borne by near-Earth asteroids (NEA) could offer substantial impact and benefits: a secure access to as well as an increase of the supply of critical mineral resources providing geopolitical stability, promoting economic growth and fostering technological development in many fields like artificial intelligence and robotics [61].

Asteroids are expected to contain excellent and easy to mine ores such as precious metals like platinum group metal (PGM), gold or germanium as well as metallic industrial resources like nickel-iron and cobalt [61]. Moreover, some asteroids harbour significant quantities of water which could be used in space either as fuel for space engines or for supporting life [20] and may harbour other volatiles such as ammonia, carbon dioxide or methane [61]. The available quantity of those resources can only be statistically estimated [48], but precious metal are expected to be in higher concentrations in asteroids than in the Earth crust [61]. In order to illustrate the economic value of these NEAs resources, it is estimated that they collectively harbour 37E15 kg of iron worth 11000 trillion dollars at current Earth prices [34] and one 100m-diameter PMG-bearing asteroid alone would provide a tenth of the 2011 world platinum production [11].

Most asteroids are located in the asteroids belt between Mars and Jupiter out of immediate reach as the energy expenditure involved would be, at least for now, prohibitive. Near Earth asteroids (NEAs) are therefore considered as the prime targets for the first space mining operations. These NEAs have orbits which near Earth on a regularly basis. As of 2015, the number of known NEAs is 10,337 according to the international astronomical union [61]. 861 of them have a diameter larger than one km [61] and if other objects, such as comets, are included the population of near-Earth objects (NEOs) increases significantly. For diameters larger than 100m, the number of NEOs nears 20,000 [36] a number which goes up to 10 millions for diameters larger than 20m [36].

Asteroids are uncontrolled and uncooperative objects devoid of any docking device [61]. Their shapes are irregular, in particular for small ones (i.e. with diameter lower than 100km). Due to their their irregular shape, their rotational motion cannot be described in terms of principal axis of rotation but rather as a tumbling motion i.e. a rotation around a dynamic axis [20, 34]. For the biggest asteroids, the equatorial velocity can be of the order of tens of km/s [61] and the magnitude of the gravitation force at their surface is extremely small. This combination of lack of gravitation and centrifugal forces and relative velocity at the surface renders any attempt to land, maintain contact with and deploy on the surface very challenging.

There have already been one successful attempt to reach asteroids. The Hayabusa mission successfully landed on the surface and collected surface sample of the asteroid Itokawa number 25143 in 2005. However, contact was brief and not secured [34]. Secure attachment to the surface of an asteroid by a probe remains to be tested [34].

Currently, NASA is in the planning of yet another mission: the Asteroid Redirect Mission (ARM) which proposes to bring back a captured asteroid in a lunar orbit for further human exploration [61].

Among the engineering challenges faced by such missions, the attitude control problem is of particular interest for it constrains critical tasks such as the rendezvous and docking to an asteroid, the mining process itself and the accuracy of the trajectory should this asteroid be retrieved to where it should be mined or consumed.

In [61, 34] various strategies are envisaged to de-spin rotating asteroids. In [34], it is estimated that in order to de-spin a rapidly rotating 100m-diameter asteroid whose mass is of the order of a few million tonnes, about one tonne of chemical propellant would be required to do so completely, assuming the propellent would be readily available on the asteroid itself. However, these methods may prove impractical or energy expensive. For one, finding propellent material on the spot and in sufficient quantity assumes accuracy of its estimation process and ability to retrieve it easily in other words without too involved a mining process. It also assumes autonomous, accurate and secure anchoring of devices on a very uneven surface under the quasi absence of light, landmarks and gravitation force (of the order of $\frac{mm}{s^2}$) [34]. The most common devices controlling attitude are internal torquers like reaction wheels and external torquers like thrusters. For practical scaling reasons, only thrusters could realistically be used. These devices use respectively mass movements and reaction mass as a way to actuate. Usually, any changes of mass distribution such as appendages is considered as disturbances however, leaving it as an unexplored method of attitude control.

With all of the above in mind, using a modular self-reconfigurable robot offers an original alternative for tumbling objects manipulation or attitude control. The basic robot architecture under consideration is one where each module remains in communication with and is attached to at least one other module. The reconfiguration process sees modules move on top of one another changing the robot shape volume and mass distribution. This deployment is coupled with exchange of angular momentum between the modules and the tumbling object. The potential benefits
of such an approach are multiple. It could circumvent the risks associated with multiple landings in micro-gravity by enveloping the tumbling object instead of potentially bouncing off it. Moreover, the strong argument for enveloping a tumbling object with a self-reconfigurable robot is that it provides a "scaffolding" structure or "tracks" along which other robot could latch on to perform tasks and have accurately placed propulsion systems for instance. Considering the importance of the mass payback ratio (i.e. the mass retrieved to the mass sent) as a key driver to maximise when designing a space mining missions, this "scaffolding" solution would certainly help maximise the ratio of area or volume covered to the mass deployed. Finally, one could even imagine that a modular self-reconfigurable robot could deploy in such a way that it alters the mass distribution of a tumbling object so as to make it more symmetrical. In doing so, maintaining a constant spinning rate around an axis of symmetry would be much easier rendering the overall attitude of the system more stable. One practical application of this could be the possibility of generating and maintaining an artificial gravitation force on the largest objects with an appropriate centrifugal force. It is with consideration of all of the above that the research reported in this thesis has endeavoured to explore the feasibility of using a self-reconfigurable robotic attitude control system.

1.2 Study aims and objectives

1.2.1 Study Aims

The specific aims of the research carried out were:

- 1. Model the dynamic interactions between an uncooperative random-shaped tumbling object with low-magnitude gravitation field and a modular device moving on the surface of this object in order to describe the effects of dynamic changes in mass distribution to the overall system and the resulting exchanges of angular momentum between the object and the device.
- 2. Derive from the above model a simple behaviour-based decentralized algorithm controlling the deployment of a modular self-reconfigurable robot over the

surface of the object as a continuous chain of modules circling around the main spin axis. The problem is akin to the resolution of a constrained optimization or tracking problem in the sense that the deployment should make the object's rotational state converge to a reference rotational state.

3. Verify and validate the robot controller concept and correctness at the lower module level and at the higher robot level through computer simulations.

1.2.2 Contributions and Achievements

This PhD thesis met the objectives of the preceding section through two main original contributions which give ground to the further exploration a self-reconfigurable robotic structure concept for space application:

- 1. The design of a combined behaviour-based and model predictive control algorithm for the module of a decentralised modular self-reconfigurable robotic structure which controls the deployment of this structure on the surface of an ellipsoidal object rotating in a low-gravity field while manipulating the rotational state of this object.
- 2. The partial validation of the above approach through simulations of the deployment of one module only. These simulations showed that a module can successfully deploy to a target surface location on the object's surface while partially driving the object's rotational state to a pure spin or at least to a state close to its initial rotational state avoiding divergence to degenerate uncontrollable rotational states.

These contributions are based on the following original work:

1. A physical model of the modular self-reconfigurable robot and its interactions with the environment based on a deformable rotating continuum with variable mass approach. This physical model contains all the relevant dynamics, perturbations, constraints and parameters to simulate the robot accurately and provide the basis upon which a model predictive controller for the robot's modules was designed.

- 2. The design of a basic decentralised hardware architecture concurrently with a task-focused two-level decentralised control algorithm for each robot module enabling the robot to perform a manipulation through reconfiguration of the entire robot body via changes of mass distribution. The top-level controller controls the reconfiguration and deployment of the robot on the object's surface using task-driven behaviours organised in a tree structure. The low-level controller uses a state space representation and a model predictive control approach to track a reference state trajectory representing the ideal final robot configuration and ideal final robot was proven to be either at best neutrally stable or unstable and its controllability could not be proven. However, under the hypothesis of partial controllability via the model predictive controller, the correctness of the overall two-level controller was proven.
- 3. The performance evaluation of the low-level module controller through the extensive simulation of the linear and nonlinear model predictive controllers of one module deploying on the object's surface under 32 initial conditions parametrised by the object geometry, the mass ratio of the object's to the module's and the object's initial rotational state. The simulations showed that the nonlinear model predictive controller should be chosen over the linear one. The simulations also validated the nonlinear model predictive controller for driving the module's deployment to a target location but not for controlling the system's rotational state which never converge to the reference rotational state. However, the simulations showed that the nonlinear model predictive controller could be used effectively for a strategy focusing on stabilising the system's initial rotational state. This strategy trades off controllability for stability by combining the controller with a stiffening of the system through an increase of the mass ratio which is shown to be the main stabilising parameter. The benefit of stability outweighs the risk of divergence while trying to converge to a reference rotational state and in this case, the proposed self-reconfigurable robot is suited for objects involving a mass ratio of at least 1000 or more.

The outline of the work carried out to produce these original contributions will be found in the next section 1.2.3, while the details of the contributions will be found in the conclusion in chapter 6.

1.2.3 Solution Outline and Objectives

1.2.3.1 Chapter 2: The Origin of the Concept

After a presentation of the space environment the deployable robotic "scaffolding" structure would have to operate in, an introduction to basic spacecrafts requirements and design was conducted in chapter 2 and combined with a review of the currently planned methods for active space debris capturing and removal to gain an overview of the current and potential ways of manipulating non-cooperative object in space. After an introduction to reconfigurable robots, an analysis of their potential use in space application was performed with a focus on how they provide another method of dealing with uncooperative randomly tumbling object with benefits of their own.

1.2.3.2 Chapter 3: Modelling the Problem

The study then delved into the physical modelling of the interactions between a randomly tumbling object and another object, or surface mass, moving at its surface. This in order to size and derive design guidelines and parameters for a modular robot whose primary objective would be to deploy all over the surface of such an object. The problem was approached by considering the tumbling object and the surface mass as one rotating continuum with a rigid part, the object and a deformable part, the surface mass. The discretisation of an existing model of rotational dynamics of a deformable medium led to the building of a simulation of the rotation of an object with multiple surface masses at its surface. All simulation parameters are presented in the chapter. A brief analysis of potential landing sites was also conducted while all important mathematical derivations can be found in appendices.

1.2.3.3 Chapter 4: Designing the Robot Controller

The study then moved on to the design of the robot. While spanning the whole spectrum of design requirements, the chapter focused on the control aspect of the design. The choice of approach was driven by concerns for fault recovery and led to the design of a decentralised behaviour-based algorithm where each module makes a decision based on communication with its immediate neighbours, each module passing information along. The behaviour-based algorithm is seconded by a Linear (MPC) or Non-Linear Model Predictive Controller (NMPC), based on chapter 3's model, which ensures that a module deploys while tracking a reference rotational state of the object. The objective of the control algorithm is to lead to the deployment of modules as a continuous chain over the surface of the object so as to cover a continuous trajectory circling around the main spin axis.

1.2.3.4 Chapter 5: Simulation Results

The results of the simulation of a single mass deploying on the rotating object are reported here leaving the simulation of the entire robot for future work.

1.2.3.5 Conclusion:

Finally a summary of findings and contributions is provided in the conclusion.

Chapter 2

Literature Review

2.1 Scope of the Literature Review

As this thesis is covering a multi-disciplinary field, this section will define the scope of the literature review which sets the boundaries of the thesis.

- The first part of the literature review is dedicated to understanding of the environment the robot is to operate in and focusses on asteroid dynamics. This introduction constitutes the basis upon which a physical model of the rotational interaction between the robot and the asteroid is to be derived. This first part introduces:
 - (a) the origin of asteroid motion and the types of rotations they experienced,
 - (b) the perturbations affecting asteroids motion, in particular their rotation,
 - (c) the approach to planning the landing phase and the choice of landing site,
 - (d) the approach to modelling:
 - the dynamics of the rotational interaction between the robot and the asteroid,
 - the perturbations,
 - the sensors and actuators noise.

- 2. The second part of the literature review is dedicated to review the methods and challenges of spacecraft design which condition the robot design. It is completed by a presentation of current asteroid and spaced debris capture methods. This second part introduces:
 - (a) the basics of spacecraft design and engineering requirements for space operations,
 - (b) attitude control and asteroids and debris de-spin methods,
 - (c) the current designs envisaged for tackling asteroid and space debris capture.
- 3. The third part of the literature review is dedicated to the presentation of Self-Reconfigurable robots with a focus on control. The discussion of the various control methods emphasises the need for a flexible and easy-to-implement controller with a capability to "understand" its environment and produce actuation commands accordingly. This third part introduces:
 - (a) a Self-Reconfigurable robots taxonomy,
 - (b) Self-Reconfigurable robots hardware capabilities,
 - (c) the self-reconfiguration problem
 - (d) why a task-base self-reconfiguration is the method of choice for designing a controller and its translation into
 - (e) a need for a high-level behaviour-based controller coupled with
 - (f) a lower-level controller based on a physical model of the environment which lends itself well to an implementation of a linear and nonlinear model predictive controller.

2.2 Dynamics of Asteroids

The design drivers and parameters of the "scaffolding" robotic system to be designed in this PhD study are dependent on the wider mission parameters prior to and during deployment on an asteroid. These mission parameters include the approach and rendezvous orbits of the spacecraft carrying the robot as well as the robot optimal landing site. Moreover, the rotational dynamics of the target asteroid needs to be understood to ensure that the physical hypotheses conditioning the planning of the robot deployment are valid, that the robot deployment is feasible and that the chances of mission failures are lowered. This section provides a briefs review of the Near Earth Asteroids (NEAs) and main-belt asteroids dynamic properties which are fed into the robot design.

2.2.1 Asteroids' Origin

NEAs are assumed to originate from the main asteroid belt between Jupiter and Mars under the influence of orbital resonances with Jupiter [41]. A brief introduction of the origin and dynamics of the belt asteroids gives important insights into the problem of reaching, landing and deploying a device on the surface of an asteroid. This subsection and the following highlight the challenges faced by such a mission.

The main asteroid belt is a very large collection of solid bodies orbiting the sun between Jupiter and Mars and occupying a torus-shaped volume. These bodies have irregular shapes and are much smaller than planets. They are assumed to have been formed during the process of gravitational accretion of dust and gas which led to the formation of the Solar System and its planets. This hypothesis states that, in the main asteroid belt region, the perturbing influence of the gravitation force exerted by Jupiter on these small bodies prevented the formation of planet-sized bodies by increasing their relative velocities to such a level that shattering post collision dominated over accretion [18, 43, 55, 62].

Broadly speaking, asteroids can be found as stand alone and, in this case, are usually large (10 km diameter or more) or in families or clusters of asteroids. These families originate from catastrophic collisions undergone by a large body and cluster around it while moving together on an average orbit. Family member of 1 km in diameter or less contribute the most to the population of these families [18, 43, 55, 62].

2.2.2 A Classification of Asteroids' Rotation State

This section lists the types of rotations that asteroids experience in order to show the diversity of situations the robot controller is to face during deployment.

According to the asteroid lightcurve database [67] containing data for more than 15,000 bodies, asteroids can be classified by rotation period in four categories: typical rotators, slow rotators, fast rotators and tumblers. The rotational period can be sidereal, i.e. with respect to fixed stars, or synodic, i.e. with respect to the Sun. In the case of main-belt asteroids, the difference between the two measures can be neglected.

Typical Rotators Typical rotators have a period of rotation between 2 and 20 hours [67].

Slow Rotators Slow rotators have an exceptionally long period of rotation, longer than 100 hours up to longer than 1000 hours in a few cases. These asteroids typically measure between 1 and 20 km in diameter [67].

Fast Rotators Fast rotators have an exceptionally short period of rotation: 2.2 hours or less. Typically, they have a diameters measuring less than 1 km, the majority of them being shorter than 100 meters. The rotation period has a lower limit corresponding to the disintegration of the asteroid which occurs when the centrifugal force is greater than the gravitational force holding it together. Consequently, asteroids with a diameter over 100 meters are possibly conglomerations of smaller pieces loosely kept together by gravitation [67].

Tumblers Tumblers are bodies with unequal moments of inertia which do not rotate in a constant manner with a constant period. Tumbling can be caused by the YORP effect [67].

2.2.3 Perturbations Affecting the Dynamics of Asteroids

This section introduces the main perturbations to asteroid dynamics which affect key mission parameters: the target asteroid's orbit in the Solar System, the target asteroid's rendezvous orbits and the target asteroid's rotational motion.

The orbital and rotational motions of asteroids are both strongly perturbed by two main types of disturbances [62]:

- 1. <u>gravitational perturbations</u> which are due to the gravitational forces exerted by the many bodies present in the Solar System, particularly the Sun and the major planets.
- 2. <u>solar radiation perturbations</u> which are due to the heat, light and other radiations given off by the Sun. These can be divided into:
 - (a) the Yarkovsky effect
 - (b) the YORP (Yarkovsky–O'Keefe–Radzievskii–Paddack) effect
 - (c) the Poynting–Robertson effect

2.2.3.1 Gravitational Perturbations Effects on Orbits About an Asteroid

Gravitational perturbations are mostly concerned with the orbit of the bodies under their influence and engender resonance phenomena. This section will focus on their influence on a spacecraft orbit about an asteroid. These orbital perturbations have to be taken into account for choosing the robot landing site on the surface of the asteroid. Indeed, the impact of robot landing sets the initial rotational state of the asteroid on which the robot will start to deploy.

In [51], the author evaluates the stability of near-synchronous orbits about celestial bodies modelled as ellipsoids. This gives rise to a two-type ellipsoid classification based on whether these ellipsoids possess two stable and two unstable near-synchronous orbits (defined as Type 1) or only unstable near-synchronous orbits (defined as Type 2). The classification is not fixed and depends on three main parameters:

- 1. <u>the ellipsoid shape</u>: oblate ellipsoids are of Type 1 and prolate ellipsoids tend to be of Type 2.
- 2. **the ellipsoid mass distribution and density**: the more dense the ellipsoid is, the more stable are its two near-synchronous orbits,
- 3. <u>the ellipsoid rotational state</u>: an increasing rotational velocity increases instability.

The two main sources of perturbations are gravitational attraction and radiation pressure from the Sun. Two regimes can be defined from this dichotomy: a body dominated regime and the solar regime [53].

Body Dominated Regime occurs when the gravitational attraction of the asteroid is larger than the Sun's perturbations. If an equilibrium orbit can be found, i.e. a periodic orbit, methods exist to determine a motion around the asteroid if such a motion can be proven to remain stable. This is mostly possible for uniformly rotating asteroids. In the general case, however, asteroids may only have unstable equilibria in the vicinity of their equatorial plane, depending on their shape, mass distribution and rotational state. Finally, specific resonances of the considered orbit with the asteroid are also to be taken into account [53].

Solar Regime occurs when the Sun perturbations are larger than the gravitational attraction of the asteroid which is usually the case except for objects with very low area to mass ratios. Although it is possible to analyse both the Sun's gravitation and radiation pressure perturbations together, it is difficult to find but limited analytical solutions. Model based on the solar pressure perturbation can only provide sufficient conditions for a spacecraft to be bound to a stable orbit [53].

The model-based classification presented at the beginning of this section illustrates the challenge of finding a stable orbit around an asteroid to enable a smooth landing. In reality, the gravitational attraction and radiation pressure from the Sun make these orbits among the most perturbed to be found in the solar system. Consequently, close proximity operations are extremely challenging. Uncontrolled trajectories are highly unstable and sensitive to small changes in initial conditions and even actively controlled trajectories are chaotic and can be become random. This implies that within a few orbits, i.e. within a time span ranging from a few hours to a few days, the spacecraft can either impact or escape [51].

2.2.3.2 Solar Radiation Perturbations Effect on Asteroids' Rotation State

Solar radiation perturbations affect bodies (and spacecrafts alike) throughout the Solar System and in particular small bodies because of their lower mass to surface area ratio. This perturbation is a radiation pressure producing forces and torques on the bodies altering their translational motion, i.e. their orbits, and their angular velocity which can accelerate or decelerate [62]. This section introduces the nature and order of magnitude of Solar radiation perturbations effects.

Yarkovsky effect As an asteroid rotates, its Sun-facing side absorbs heat from the Sun while its dark side re-emits this heat. The Yarkovsky effect is the production by this irradiated thermal flux of an impulse or infinitesimal amount of thrust either opposite or along the direction of the orbital motion of the asteroid. The net effect of the consequent deceleration or acceleration is a drift in orbital semi-major axis over millions of years, i.e. a change of orbit, which is most noticeable for smaller asteroids about 40 kilometres in diameter. As an example, the of order of magnitude of the drift in semi-major axis of a main-belt asteroids of 1 km in size is around $10^{-4}AU$ per million of years [62].

YORP effect The YORP effect is the alteration of an asteroid's angular momentum due to a net torque produced by thermal irradiation. Depending on their size, shape, composition, mass distribution and rotation state, the asteroids rotation rate can either increase or decrease. Moreover, YORP torques produce changes of obliquity angles (the angle between a perpendicular to the asteroid's orbital plane and its spin axis) which usually tend to an asymptotic value. However, the increase of obliquity angle can change the rotation state to such an extent that the asteroids transitions to a new stable rotation state via a tumbling rotation or undergoes possible YORP cycles [62]. The YORP torque affects mainly small asteroids of size < 10km but its magnitude is small and takes effect over a long period of time [14]. In [52], the author estimated that, under the effect of the YORP torque, the despin of well-studied asteroids with known shape models and known rotation poles would take between 0.1 million and 500 billion years. Therefore the YORP torque can be treated as a perturbation to the torque free rotational motion [14].

Poynting–Robertson effect The Poynting–Robertson effect applies to grain-size particles which gradually spiral and disappear into the Sun. This effect is out of scope of this PhD study [27].

2.2.4 Approach to Landing Site Choice

As seen in section 2.2.3.1, orbits around an asteroid, including stable ones, are chaotic. A spacecraft orbiting an asteroid is likely to either impact or escape over a very short period of time spanning hours or days. Therefore, for this PhD study, it is assumed that the spacecraft carrying the robot will not have the time to perform a smooth landing. Assuming the spacecraft impacts the surface of the asteroid at a small relative velocity, this impact is nevertheless likely to have an influence on the initial rotation state of the asteroid over which the robot will deploy. In order to evaluate and minimise the effect of the landing impact, the landing will be modelled in chapter 3 by considering the asteroid as a body with a time-variable mass with an instantaneous change of mass.

2.2.5 Approach to Dynamics, Perturbations and Noise Modelling

The perturbations affecting the mission design were listed in sections 2.2.3.1 and 2.2.3.2. In this section, their magnitude, frequency, time span and bandwidth are considered in order to understand how best to produce an accurate physical model of the dynamics interactions between the deploying robot and its target asteroid

which should include a perturbations and noise model.

As per [62], for a main-belt asteroids of 1 km in size, the gravitational perturbations was estimated to cause a drift in semi-major axis of 15,000 km over a million of years. Similarly, in [52], for the asteroids under consideration, the upper bound of the estimated rotational deceleration due to the YORP torque is of the order of $-2.0 \cdot 10^{-16} rad/s^2$ producing its full de-spinning effect over 140,000 years. For both the gravitational and radiation perturbations, over the scale of a mission which, between reaching the asteroid, deploying on it and maybe retrieving it, would span at best a few years, their magnitude and time scale result in a negligible effect on the angular momentum of a target asteroid. For the system of interest in this PhD study, i.e. the system composed of the asteroid and the robot, the fastest dynamic states are the angular velocity of the asteroid and the deployment velocity of the robot at its surface. Consequently, this system can be modelled as an isolated system using conservation of energy and momentum to derive torque free rotational equations of motion from first principles. The Euler equations can also be used as approximations and both the gravitational and radiation perturbations can be introduced as small perturbations [14]. Considering the respective magnitude of the gravitational and radiation perturbations and the fact that the YORP effect is caused by the infrared radiation from the Sun, both these perturbations will be aggregated and modelled as a biased finite-bandwidth noise.

The sources of noise extend to sensors and actuators noise. The sensor of interest for this study is the gyroscope which enables the robot to measure the rotational state of the combined asteroid and robot system. Actual values of space grade gyroscope noise data can be found in [33].

As for the actuators, there is a priori no power and energy requirements on the motors to be incorporated in the robot as it is one of this study's objective to attempt an estimation of these requirements. Therefore, similarly to the perturbations, it is assumed that the actuators noise can be aggregated with the sensors' and will be modelled as a gyroscope's noise.

The perturbations and noise will be formally modelled in chapter 3 section 3.5.6.

2.3 Spacecraft design basics and requirements

The space environment is one of extremes and is very challenging for robots to operate in. This imposes specific drivers on robotics technology for space. In this section, a brief introduction of spacecraft engineering concepts will be made with an emphasis on the impact and research challenges imposed on the design of robotic systems for space and specifically to the problem tackled by this thesis.

2.3.1 Spacecraft Design Constraints versus Robotics

Spacecraft are designed under stringent constraints which also apply to space robot design.

Firstly, spacecrafts are require to have a high robustness threshold to any source of stress whether mechanical, like the vibration and impact of launch and potentially landing, or radiations and extreme thermal gradients. Therefore, the structure and mechanical design of a space robot should proceed from the same standard, robustness being one of its key engineering drivers [19].

Secondly, for a spacecraft, mechanical actuation systems are usually a potential single point of failure and therefore reduced to the essential while for a robot, actuation being the mode of interaction with the environment, it is crucial to its performance. Spacecraft actuation systems encompass propulsion systems, attitude control systems or even mechanical systems for the deployment of large structures. In comparison, robotic actuation systems increase design complexity by an order of magnitude in terms of the performance required by the various interactions the robot has with its environment [19].

Thirdly, the approach to control differ significantly. For spacecrafts, control relies on dynamic models while for robots, the various environments may be completely unknown a priori. Robots controllers are intended to learn and adapt [19].

Spacecrafts' design common practice usually splits the various components into eight subsystems [19]:

1. Propulsion system;

- 2. Attitude control subsystem which control the orientation of the spacecraft to ensure that all components point in the correct direction;
- 3. Structural and Mechanical;
- 4. Power subsystem;
- 5. Thermal control subsystem;
- 6. Communications subsystem;
- 7. Onboard data handling (computer) subsystem;
- 8. Payload subsystem which provides the business end of the spacecraft.

For trade-offs and resources allocation, there are usually five main design budgets [19]:

- 1. The cost budget capping the costs of the design, development, construction, validation, and launch of a spacecraft;
- 2. The mass budget capping the total mass of the spacecraft to be launched; imposing a trade-off between weightiness and structural flexibility;
- 3. The propellant budget limiting the manoeuvring capability (function of the total mass of the spacecraft);
- 4. The power budget limiting the power and energy available to each spacecraft subsystem and the payload, imposing a trade-off between power, efficiency and computational resources;
- 5. The data budget limiting the communications capabilities and onboard storage capacity.

Additional constraints are placed upon spacecraft reliability (above 90% typically). This requires extensive testing and validation under space-like conditions, a capacity for upgrade and repair-by-replacement of modules disfavouring soft computing methods. The driver here is the limitation of mechanical complexity. For space robots, these additional constraints translate into the following requirements [19]:

- Using lightweight components to minimise launch mass but resistant enough to launch/impact loads (typically up to 20 g axial acceleration and 145 dB acoustic noise for launch);
- 2. Having a limited volume at launch;
- 3. Being capable of operating in vacuum environment using materials resistant to outgassing in vacuum and dry lubrication;
- 4. Possessing control algorithms dealing with the effects of microgravity such as no ground reaction and significant non-linear dynamics effects and restricting motion with low speeds (0.01m/s) to compensate for the lack of damping and dissipating medium.
- Being capable of sustaining extreme temperature gradients and thermal cycling between -120C to 60C usually;
- 6. Having electronic components resistant to an environment full of highly charged particle and radiations;
- 7. Being capable of efficient real-time control and navigation operations with limited onboard computational resources;
- 8. Being resistant to electrostatic charging and discharging due to the lack of grounding;
- If required, being capable of performing image processing in an environment with poor illumination;
- 10. Having a high level of autonomy under the constraint of point 7.

2.3.2 Whole-system Design Methodology

According to [45], it is recommended that robots designed for space applications use appropriate system engineering methods. The entire system, comprising the robot, supporting infrastructure, the human-in-the-loop component and their interactions must be taken into account as it is actually more important to the success and robustness of the space-robotic system than any robot-specific technology (like mobility, dexterity or intelligence).

2.3.3 Attitude Control Hardware

This section will focus on providing some background on attitude control system already in use. In current spacecrafts, the attitude control system (ACS) provides attitude stabilization and attitude manoeuvre control. It produces control torques in response to a disturbance torques measured as an error by the attitude determination system (ADS) and in response to pointing requirements. Attitude control hardware can be divided into active and passive control devices as will be seen in the subsequent sections. The passive hardware does not consume power, does not require a communication interface and is set to remove a predefined amount of energy from the spacecraft. The active hardware can adjust the amount of energy removed. Both types of hardware modify the angular momentum of the spacecraft at any given time.

2.3.4 Active Attitude Control Actuators

Reaction Wheels: Reaction wheels change the spacecraft angular momentum by changing their own angular momentum about their axis of rotation. They are used for coarse as well as fine pointing. They can store a limited amount of angular momentum depending on power availability and motor design and experience saturation [35].

Magnetorquer: The magnetorquer is a magnetic dipole generating a moment by passing through the Earth's magnetic field and dependent on the orbital position. It is used in low to medium Earth orbit for coarse pointing and performs space-craft detumbling and reaction wheels' angular momentum reduction. This actuator cannot be used outside earth orbit [35].

Thrusters: Thrusters generate thrust by expelling propellant through a nozzle. When placed at a moment arm from the centre of mass of the spacecraft, they generate a torque about the centre of mass. When used for attitude control only, they work in pairs in order to minimise translational motion. They are used for both coarse and fine pointing. However, their use is limited by the amount of propellant and power available on the spacecraft [35].

Passive Attitude Control Actuators: These actuators provide restoring torques and remove (or add sometimes) angular momentum without the need for active control. They are coarse pointing actuators only. Passive attitude actuators are not usable outside earth orbits [35].

Hysteresis Rod: The hysteresis rod is a piece of ferromagnetic material which uses the variation of the magnetic field strength with the spacecraft orbital position to generate a magnetic flux density. This process reduces angular momentum of the spacecraft over time [35].

Permanent Magnets: The permanent magnet is a permanent dipole which can only stabilize a spacecraft about two axes [35].

2.4 Attitude Control and De-spin of an Asteroid:

This section will examine what solution are currently put forward to capture and de-spin asteroids and will focus on the very capture mechanisms and task of despin and not on specific mission phases. These solutions assume that Near Earth Asteroids (NEAs) are tumbling, non-cooperative objects which will be de-spun autonomously in deep space.

In [5], an asteroid capture with a bag is proposed. Despite some station keeping prior to capture to synchronised the main spinning rate and axis with the NEA, it is expected that there would still be a residual relative angular velocity with the spacecraft which would undergo some impact. The bag would then be tightly cinched to drawn up the asteroid against the spacecraft to constrain its position and attitude so that forces and torques could be applied by the spacecraft. In order to have some order of size, the study provides an estimate of the time and mass of propellant required to de-spin the entire system: spacecraft and asteroid. It was estimated that in order to de-spin an asteroid of 12m by 6m diameter cylindrical shape with a 1100t mass rotating at 1RPM about its major axis, 33 minutes of continuous firing would be necessary consuming about 306 kg of propellant (approximately one third of the total mass of the asteroid) which could be considered as quite expensive. The study also proposes the alternative method of anchoring the spacecraft to the surface of the asteroid and winching its way to the surface to drill for regolith on the spot using the bag for retrieve.

In a similar study [47], Carlos Roithmayr proposes an almost identical approach. The asteroid is also of the order of 7m diameter and has the same agreeable cylindrical shape with known moments of inertia. Conveniently, there is no need for identification of parameters and the study confirms the order of magnitude of the propellant mass required to perform the de-spin, around 300kg for 1000t asteroid. It only optimises the moment arm length at which thruster are fired in order to maximises torques.

In [25], a similar spacecraft design is pursued but the authors abandon the idea of matching the spin rate of the spacecraft with the spin rate of the asteroid along its main axis of rotation prior to capture on the ground of excessive propellant expenses. The study briefly describes another enveloping approached with a bag which facilitates passive damping of the tumbling motion toward major-axis spin, only to dismiss the feasibility of the mechanical design. The study then describes how the latter problem can be circumvented by proposing an inflatable exoskeleton attached directly to the spacecraft bus. The bag then collapses around the asteroid with the help of actively controlled winches to achieve passive damping. The captured asteroid is modelled by a 6-DOF joint connected to the rest of the body of the spacecraft via translational and rotational spring-dampers. Simulations seems to confirm asymptotic viscoelastic energy dissipation and convergence of the combined spacecraft-asteroid system toward a flat spin. Grip and Ono nevertheless could not conclude whether this system could be physically realizable with current space-qualified materials. Finally, in [58] the authors follow in the footsteps of the previous studies. However, here the asteroid mass is assumed to be distributed in the most general possible way, meaning it can be asymmetric with three distinct principal moments of inertia. Nonetheless these moments of inertia are yet again assumed to be known. As for the previous studies model-based de-spin controller are designed to be asymptotically stable, and stay within specified thruster limits, thrusting being the main actuation option. Results are again of the same order of magnitude in terms of propellant mass used.

2.5 Active space debris capturing and removal methods

In order to complement the methods of asteroid capture and de-spin exposed in section 2.4, this section will briefly introduce a few conceptual designs of active space debris removal which is a similar problem to asteroid capture in many respects. In [56] an excellent review is provided of the current trend in the industry. This section essentially follows the structure of the paper.

2.5.1 Stiff connection capturing

Stiff connection capturing methods involve the use of tentacles, arms and multiple arms.

Tentacles embrace the space debris with a clamping mechanism either directly or as an extension of a robotic arm. Ideally the target is embraced before physical contact. That way, the chaser satellite does not bounce and the attitude control system is allowed to stand by during capturing. The clamping mechanism then locks and the new chaser-target system turns stiff.

Single arm technology has been applied in many on-orbit servicing missions but always in the case of cooperative target objects. However, space debris are uncooperative and sometimes tumbling objects. There are three main use for an arm in the context of space debris removal. It would be first used to minimise the consequences

of the unavoidable impact at contact. Different methods have been suggested and are related to the control of the direction of the relative velocity between chaser and target or the relationship between impact force and base force or the configuration between the service satellite and target. Particular methods like visual servoing are also envisaged to predict the relative motion between the target and the chaser satellite. But they require visual markers which can be very challenging to rely on in space. The second use is for de-tumbling. Tumbling rates below $3^{\circ} \cdot s^{-1}$ would be considered easy to reduce but tumbling rates above $30^{\circ} \cdot s^{-1}$ would not be considered manageable. For tumbling rate between $3^{\circ} \cdot s^{-1}$ and $30^{\circ} \cdot s^{-1}$ a release of some of the residual angular momentum of the target by soft and static contact would be considered prior to further de-tumbling. Other potential solutions involving arms include Ion-Beam shepherding by transfer of angular momentum and optimal control techniques with identification of the target unknown inertia parameters. The third and final use of a robotic arm is for an indispensable phase before capturing: the attitude synchronization which could be achieved by relative position tracking and attitude reorientation.

A multiple arms approach can also be envisaged to combine forces but also to take advantage of an improved overall flexibility through cooperation for providing a stabilizing effect for instance.

A final word on the mechanical end effector which is of prime importance as it is directly involved in the capturing motion and contacts with the target. There are several concepts of mechanical end effectors for capturing space debris. These include probes for the nozzle cone of an apogee kick motor, payload attach fitting devices, articulated hands, fingered mechanisms and universal grippers.

2.5.2 Flexible connection capturing

In section 2.5.1, the connection between the chaser satellite and the target is stiff, making the combined spacecraft-object system controllable and stable. However, this solution increases dramatically the mass to control and therefore all the different costs and in particular the energy expenditure. In order to overcome this drawback, flexible connection capturing methods in which the end effector and chaser satellite are connected by a tether, are considered as available options.

The first "flexible" option is net capturing with either a net or a gripper mechanism as end-effectors. Net capture mechanism consists of four flying weights in each corner of a net. The flying weight or bullet is shot by a spring system, the net gun. These four bullets help expand the large net to ensure that the target is wrapped up. With this method, it is not necessary to know the mass, moments of inertia and other parameter a priori for capture. Parabolic flight experiments have been performed by GMV and ESA to validate the net deployment and capturing simulations. Net capturing is held as one of the most promising capturing methods as it allows a large distance between chaser satellite and target, so that close rendezvous and docking are not mandatory. Moreover, it is flexible, light weighted and cost effective. However, more research is required in such areas like net modelling, contact influence, deployment and tumbling compatibility. Contact is also a problematic area as it is unavoidable during capturing process. The main risk is to create more and smaller debris or worse to lead to mission failure if the wrapping up is improper. Nonetheless net capturing is compatible with tumbling space debris and no attitude synchronization is needed. Close range rendezvous and removal would be less difficult. However, the acceptable tumbling range of a target is not yet understandable and a net may be twined by a high tumbling angular velocity thus rendering the spacecraft-object system uncontrollable.

The second "flexible" option is a tether–gripper mechanism. Similarly to the net capturing mechanism the end-effector in the tether–gripper mechanism is shot as a 3-finger gripper to capture a target. This 3-finger gripper is designed for a precise and stable catch of a specific part of the target precisely and stably. Requirements for tether–gripper mechanism are therefore more stringent and more complicated than net capturing. Post-capture attitude control is also problematic since the movement of the combined spacecraft-target system is unpredictable and therefore requires identification of inertia parameters to be achieved. Attitude control is a necessary condition for subsequent mission phases like de-orbiting to go ahead. The third and last "flexible" option is a harpoon mechanism shot from the spacecraft to penetrate into large space debris objects which would be pulled to de-orbit. Despite the high risk of generating new space debris, it is considered as an attractive capturing method because of its compatibility with different shaped targets, standoff distance allowed and no grappling point needed. However, it is not capable of dealing with a piece of debris with high tumbling rate. It is also favoured for anchoring a spacecraft onto an asteroid. In this latter situation, the risk of generating adverse space debris is also present. High tumbling rates would also be prohibitive.

The section 2.6 will examine some features of modular self-reconfigurable robots relevant to our proposed study.

2.6 Reconfigurable robots

This section is based on the book by Kasper Stoy [63] which provide a review and synthesis of the state state of the art in self-reconfigurable robot or SR as they will be called from this point on. It provides an introduction to SR robot leading to the description of the relevant capabilities they can offer for the problem this study sought to address.

SR robots are modular robots which are able to dynamically (i.e. while active) change shape by themselves. In general, their modules are independent units encapsulating all the sensors, actuators, processing power and communication tools required to perform their different functionalities. The shape shifting process is a sequence of modular disconnections, connections and moves along the entire structure. SR are interesting for their potential high degree of redundancy and therefore robustness to module failure, their versatility with many different shapes for one robot and possible combinations with other robots, their adaptability to various tasks and their cheapness of production SR are similar in spirit to swarm robotics with many small and relatively non-complex units [63].

In [63] is provided a useful classification of self-reconfigurable robots in terms of size and number of modules. The classification is broken down into three categories:

2.6.1 Pack Robots (Tens of Modules)

Each module is a functional unit with enough strength to lift one or a group of several other modules. In these robots, each module is essential to the overall functioning of the robot imposing strict coordination between modules. Pack robots have a limited scalability (when considering hundreds of modules) due to the fact that they are either controlled centrally or their modules are tightly coupled with local communication. Exploratory/inspection type of tasks are expected to be applications for this type of robots combined with the ability to perform gaits such as running, climbing or rolling. [63]

2.6.2 Herd Robots (Hundreds of Modules)

These robots are formed by a number of modules of the order of several hundreds of modules which, taken individually, are not encapsulated functional units. Moreover their strength is limited to the order of magnitude required to lift one module up. In order to obtain useful functionalities, the grouping of modules into functional units is required. Although herd robots architecture still requires a hierarchical control approach for both hardware and software, the number of modules allows for enough redundancy so as to relax the coordination requirement between modules without impacting the performance of the robot. It is probably the most challenging type of robot in terms of control with a core critical set of modules to maintain in tight coordination while allowing the rest of the modules to operate with less stringent coordination in a more swarm-like fashion. The increase in complexity and number of modules allows for more functionalities to be implemented by differentiation of modules design. Therefore, herd robots present the most interesting range of potential applications for our purpose in terms of interactions with heavy objects or the building or reinforcement of structures [63].

2.6.3 Swarm Robots

These robots are formed by thousands of modules with limited individual importance relative to the whole robot. Individual modules have weak functionalities despite being highly autonomous. This implies that building a functional unit requires a large number of modules. Control is decentralised and the robot develops according to local rules of interactions. As it depends on stochastic randomness, this approach is difficult to scale down to a few hundreds of modules. The main potential application for swarm reconfigurable robotics is as a type of construction materials [63].

2.6.4 Desired Properties of SR Robots

This section lists different properties a SR robot designer could seek to achieve. The following definitions are taken from [63] pages 30 to 40.

- **Versatility** is the ability to adapt or be adapted to many different functions or activities.
- Adaptability is the ability to perform tasks even if the task or the environment changes a little.
- **Robustness** is the ability to operate for many hours and to handle hardware and software failures.
- **Polymorphy** is the ability to assume many different shapes.
- Metamorphy is the ability to autonomously change among different shapes.
- **Scalability** is the maintenance of the performance of the robot with an increasing number of modules.

Responsiveness is the reaction/response time of the robot.

Functionality is the measures how functional requirements are met.

2.6.5 Types of SR robots

For SR robots, hardware and software design are coupled and linked to specific solutions. The following classes of SR robots are the most relevant for this study. Their definitions are taken form [63] page 42.

Chain-type which are chains of modules primarily design for fixed shape locomotion.



Figure 2.1: A Chain Type Robot: the Polybot [23]

Lattice-type where modules are positioned in a lattice structure like atoms in a crystalline solid. They are easy to reconfigure but perform efficient gait with difficulty.



Figure 2.2: A Lattice Type Robot: the ATRON [13]

Hybrid which could be both in lattice as well as in a tree or chain topology. They can exist in two forms: either chain-type or lattice-type but not at the same time. These robots will often use a chain-type form to achieve efficient locomotion then change into a lattice-type for self-reconfiguration.



Figure 2.3: A Hybrid Type Robot: the M-TRAN [1]

2.6.6 Computing and Communication Infrastructure

Computing and communication infrastructure is an area still very much in an exploratory phase. From [63] pages 83 to 91, the following interesting points can be made:

As far computing capabilities are concerned, it is possible to run small operating systems on modules of relatively small size and even use convenient programming abstractions such as threads. However, the development of controllers has hardly been explored so far.

Regarding the communication system, it is a critical design driver for the performance of the robot and is classified in the following categories:

- **Centralised communication system** When actuators of all modules connect to a centralised host computer.
- Localised communication system When each module can only communicate with neighbouring modules
- Global communication system when it reaches over the entire structure
- Multi-modal communication system When it combines both local and global communications
- **Stigmergic communication system** When the environment provides the medium through which communication occurs.

Finally regarding sensing capabilities, the selection and use of appropriate sensors is application-driven. The current focus on capabilities demonstration has left this area of SR robot design limited.

2.6.7 The Self-Reconfiguration problem

Self-reconfigurable robots have modules which can move around with respect to each other to change the overall shape of the robot. The self-reconfiguration problem deals with how to move these modules around to facilitate a useful change of shape. Usually software solutions are hardware and application specific. The self-reconfiguration problem can be broken down into three sub-classes of problems depending on the angle of attack to solving the problem. These can be found in [63] page 6:

- **Searches:** Searches for the sequence of module moves that will take the robot from an initial configuration to a goal configuration.
- **Control:** Individual module controllers make the individual modules move and the robot's goal configuration emerges from the initial configuration.
- **Task-driven:** Here the goal of the controllers is for the robot to perform its tasks. Reconfiguration of the robot is a process which emerges from the task completion process.

Practical solutions are mixed approaches from all three classes. The goal configuration is imposed by the task, while modules control is decentralised, distributed and combined with some search techniques.

The solutions to the self-reconfiguration problem have so far only been specific. Simplification can be made, but the difficulty essentially stems from several factors related to the motion constraints of the modules and their connectivity, the type of subconfigurations encountered, whether there are configurations with local minima or whether modules get in each other's way. [63]

2.6.8 Task-driven self-reconfiguration the way forward

A task-driven approach is by far the most appealing and the simplest from a conceptual and implementation point of view. Inferring a desired goal configuration from a task is a very difficult proposition which may prove intractable, especially if the environment is unknown beforehand, dynamic and complex. The focus placed on the completion of the task to be performed allows for the relaxation of any constraint on the self-reconfiguration process so as not to interfere with the task. This is the approach that will guide the design of the solution presented in this study [63].

2.6.9 Gaits and Manipulations: Introductions to Control of SR Robots

The task of deploying a SR robot on the surface of a randomly tumbling object can be viewed as a manipulation task in a dynamic configuration. Indeed, the robot deployment will generate disturbances which could either be mitigated or taken advantage of for achieving a better rotating state. In this perspective, it is interesting to mention how SR robots control manipulation tasks as well as robotic gaits. Some aspects of gait control are either similar or could be useful for manipulation .There is currently more research available for gaits than for manipulation. In this section will be laid out the control approaches currently in use as per chapter 9 of [63].

2.6.9.1 Gaits Control

Gait control is performed with fixed configurations. For locomotion, momentum transfer approaches which do not involve self-reconfiguration are preferred. The cyclic property of locomotion is readily exploited by control methods. The locomotion types that have already been demonstrated are caterpillaring, side-winding, walking using four or six legs, rolling, climbing and tight rope walking. Below is a list of current gait control methods.

- Gait control tables: The method is based on tables containing steps and motion to perform during steps. These tables represent a complete cycle of gait along with module synchronisation requirements and are preferably implemented with a centralised controller.
- Hormone-based control Hormones are messages passed down the chain of modules which contain information about what the emitting module is doing. This method introduces delays in the sequences of control steps. The sequence of one module is delayed with a fixed number of steps compared to a neighbouring module. This approach has two advantages: modules stay synchronised and they can be added or removed at will.

Role-based control The method is not based on discrete sets of actions sepa-

rated by coordination pauses but rather on joint positions between modules prescribed by functions of time that capture the gait as one cyclic motion.

Distributed control: The control of complex gait can be distributed in order to deal with the fact that modules perform different motions, depending on their position in the configuration. It usually works with modules selecting their function based on the local configuration and/or on their parent's function if it is required.

To date there are no methods to automate development of gaits online. A promising approach which has also potential applications to manipulation is general pattern generation (CPG). It is a bioinspired approach based on neural networks which produces an output similar to the action function in role-based control. Networks can be evolved by genetic algorithms to control the gait of a SR robot. A general pattern generator could sense changes in the terrain via the gait and optimise the gait online thanks to this information [63].

2.6.9.2 Manipulations Control

Manipulations with self-reconfigurable robots is a work in progress. The general idea is that modules connected in a chain configuration can form a serial manipulator with properties similar to those of traditional robot manipulators. However, implementing such a concept is faced with a couple of important challenges: the calculation of the inverse kinematics for a chain of modules and more importantly, the increase of the strength of individual modules and the improvement of the cooperation between them. The challenge faced by this and subsequent studies will be to find a way to make modules produce cooperative actuations that allows them to produce larger forces than those produced individually [63].

2.6.10 Designing an SR Robot Controller: the Challenges

The ability of a robot to perform a task stems from the interaction between its environment, its body and its controller. The particularity of SR robots is to have a changeable body structure which can be optimised concurrently with their controller to meet the complexity and widen the range of performable real tasks. While this approach avoids a simplicity versus versatility trade-off, there is currently a lack of methods for developing intelligent controllers for SR robots [63]. In the next section, the controller design method will be narrowed down to a behaviour based approach.

2.6.11 From Basic Functionalities to Behaviours

A good definition of behaviour based algorithms can be found in [59]. It is quoted here to highlight the rational for choosing such an approach ([59] page 309): "Behaviour-based control employs a set of distributed, interacting modules, called behaviours, that collectively achieve the desired system-level behaviour. Behaviours are patterns of the robot's activity emerging from interactions between the robot and its environment. They are control modules that cluster sets of constraints in order to achieve and maintain a goal. Each behaviour receives inputs from sensors and/or other behaviours. Thus, a behaviour-based controller is a structured network of interacting behaviours, with no centralized world representation or focus of control. Instead, individual behaviours and networks of behaviours maintain any state information and models."

This approach is particularly suited to the this study because behaviours provide a way of rendering tasks manageable rather than focusing on simplifying the tasks themselves by trying to split them into well-defined subtasks. Behaviours are lowlevel control programs with freedom of implementation which can be improved and combined into a coherent controller in order to tackle the complexity of real tasks. Each behaviour solves one aspect of a task (ex obstacle avoidance, navigation...) to achieve high-level deliberation at system level. Behaviour-based systems are good for environments with dynamic changes where excellent response time and adaptability are essential along with the possibility to do some learning and planning. Finally and most importantly, these systems scale well to multi-robot control providing robustness and fostering adaptive group behaviour [59]. **Behaviours Adaptation** Behaviours adaptation is an optimisation process of an already implemented behaviour to adapt it to aspects of a specific task that were unknown at the time of implementation. The changes made are incremental rather than radical. They are but a few studies of behaviour adaptation of SR robots in the literature and they tend to focus on adapting the controller or the body but not both at the same time, and only in the context of stereotypical tasks. One of the major challenges of behaviour adaptation is online adaptation which remains an open problem.

Behaviour Selection Behaviour selection is the continuous process of choosing which behaviour is active based on changes in the environment or the task. The process monitors sensors inputs and behaviours internal parameters to decide which behaviour should be active. It can decide to switch between behaviours while operating and ultimately change the behaviour mode of the robot.

Behaviour Mode Behaviour mode control is higher level of behaviour control whose purpose is to manage and coordinate different behaviours hierarchies to increase the versatility of a robot. In its most general form, a behaviour-based controller is a network of behaviour modes which are coupled with transition modes. These transition modes connect the behaviour modes and manage changes from one behaviour mode to another. Behaviour mode control is a unifying approach to SR robots control where real tasks can be divided into simpler subtasks and optimal behaviours designed both in terms of control and bodily structure.

In conclusion, behaviour control provides a flexible method of designing a complete solution to a task and robot controllers. It allows complete change of behaviour, combinations and adaptations to changes in the environment or to the task, thus rendering a robot adaptable, robust and versatile.

2.6.12 Low-Level Control and Interaction with the Environment

The deployment of the robot on the surface of a tumbling uncooperative object is an inertial process with competitive objectives. The aim of the deployment for each module is to travel on the robot structure and on the surface of the object towards a target surface location. This motion relative to the surface can cause the object to depart significantly and irreversibly from its initial rotational state. It is therefore paramount that the module's motion causes the least possible disturbance to the rotational motion of the object while doing so in a timely fashion with respect to the fastest dynamic state of the system, in other words, before the disturbances take effect. In this respect, the controller's goal is to provide a required output, the convergence of the deploying robot module to its target location on the surface of the object while tracking the initial or prescribed rotational state of this object. As such, the controller is required to be able to deal with multiple states, to optimise and to abide by constraints. A Model Predictive Controller lends itself quite well to meet these requirements.

2.6.12.1 Principle of Model Predictive Control

Model predictive control (MPC) covers a range of control methods which make explicit use of a dynamic model of the process to obtain the control commands by minimizing an objective function while satisfying a set of constraints. The usual structure (whose detail can be found in [8] pages 3 and 4) of these controllers is the following:

- 1. use of the model to predict the process output at future time instants called the horizon,
- 2. calculation over the horizon of a control sequence minimizing an objective function then
- implementation of a receding strategy: application of only the first control signal of the sequence calculated at each step [8].

2.6.12.2 Advantages and Drawbacks of MPC

Advantages MPC has a number of advantages over other control methods [8]:

- it can be used to control a huge diversity of systems with simple or complex dynamics alike, long time delays and even used to control unstable systems
- it deals with multivariable cases
- it compensates for dead times
- it uses feedforward control to compensate for measurable disturbances
- it is easily implemented
- it includes the treatment of constraints
- it can help track a known future reference which is one of its intended use in this study
- it is easily extendable

Drawbacks MPC drawbacks include [8]:

- its derivation can be complex
- all added extensions require extra computation at every sampling time as in the case of adaptive control and constraints
- an accurate model of the process is required for the controller's prediction process
- it can be difficult to prove stability and robustness

2.6.12.3 Comparison between Linear and Nonlinear MPC

Conceptually, the extension of MPC techniques to nonlinear processes is straightforward. However, designing a nonlinear MPC faces many challenges.

The first challenge is the availability of a nonlinear model either from first principles or from experimental data. For experimental models, identification techniques
for nonlinear processes are lacking. Fortunately for this study, an explicit model can be derived from Newton's second law of motion [8, 37, 40, 38].

The second challenge is that the NMPC control problem is a nonconvex optimization problem whose resolution is significantly more difficult than the quadratic problem and can involve tackling local optima which impacts the quality of controller and its stability. Nonlinear systems stability and robustness problems are complex and still open in most cases, although methods of proof are now available and are used in this study [8, 37, 40, 38].

The third challenge is the computation time required for the NMPC control command to be calculated. It is a critical consideration with respect to the hardware computing power which has to be implemented into a real system and it is also a critical consideration with respect to the time constant or time span of the dynamics of the system as the controller should operate sufficiently fast to tackle the problem efficiently i.e. within an acceptable time frame for the mission [8, 37, 40, 38].

The benefit of having an explicit nonlinear model to be used in a NMPC controller has to be balanced against the possibility of using a MPC controller which would be an easier option. A comparison of the respective performance of a NMPC and its linearised MPC version is carried out in this study.

2.6.12.4 Stability of Linear and Nonlinear MPC

Under general assumptions, infinite horizon optimal controllers for linear processes guarantee a stable closed-loop in the limited case when all process variables are unconstrained. Constrained optimal control problems face two main challenges. On the one hand, they have to be solved using a finite horizon and on the other hand constraints introduce non linearities which make the derivation of an explicit description of the control law but impossible and the study of stability very difficult.

In order to prove stability, the main idea is to show that the infinite horizon cost function is monotonically decreasing (i.e. that it is a Lyapunov function) under the necessary condition of feasibility of the derived control laws. Two approaches which link stability to a constraint satisfaction problem can be taken to do this. The first approach considers the cost function as being composed of two parts: one with a finite horizon and constrained, and the other with an infinite horizon and unconstrained. The second approach consists of adding a terminal state constraints and using a finite control horizon. In both cases, terminal state penalization and terminal sets are the basic techniques used to enforce the final state. However, final state constraints have to be used carefully as these are very restrictive for many control problems and can lead to infeasibility. Infeasibility can be tackled by removing the state constraints during the initial part of the infinite horizon and a linear MPC with soft constraints and state feedback can also asymptotically stabilize any stabilizable system [8, 37, 40, 38].

For non-linear MPC (NMPC), even with a perfect model, there is no guarantee of closed-loop stability in spite of the optimization algorithm finding a solution. The main approaches which successfully tackle the stability problem always start with a state space framework and a regulator analysis. A list of the main techniques can be found below:

- 1. <u>Infinite horizon</u> This consists of increasing the control and prediction horizons to infinity in which case the cost function is a Lyapunov function which provides stability. However, it is difficult to implement since it requires the computation of a large set of decision variables at each sampling time.
- 2. <u>Terminal constraint</u> This uses a finite horizon and ensures stability by adding a state terminal constraint enforcing a zero final state and control command at the end of the finite horizon. However, it introduces extra computational costs and restricts the operating region, which makes it very difficult to implement.
- 3. <u>Dual control</u> With dual control, a region is defined around the final state inside which the system is driven to the final state by a linear state feedback controller. The NMPC algorithm is used outside the region with the prediction horizon decreasing at each sampling time and once the state enters the region, the controller switches to a previously computed linear strategy.
- 4. Quasi-infinite horizon This uses the dual control technique but only for the

computation of the terminal cost. The control action is determined by solving a finite horizon problem but the linear controller is never used even inside the terminal region.

Stability can be guaranteed by solving the constrained NMPC in a finite horizon with either a terminal set which the final predicted state is constrained to reach and/or a terminal cost or cost of the terminal state. However, this can be difficult to implement because of the non-convexity of the optimisation problem rendering the linear control region hard to compute with multiple potential solutions. Nonetheless, asymptotic stability of the controller can be achieved with suboptimal solutions. In other words, stability can be guaranteed when the optimization problem is feasible but at the expense of optimality. Any feasible solution ensures feasibility, and hence constraint satisfaction. It is then sufficient to consider any feasible solution with associated strictly decreasing cost to guaranty asymptotic stability. Although suboptimality is not ideal with respect to performance and timescale, it would be good enough for this feasibility study. Another possibility to lower the computational requirement is to remove the terminal constraint, especially when the system's is unconstrained [8, 37, 40, 38].

Finally, beyond stability analyses, the robustness to modelling errors is difficult to assess, especially in view of the fact that stability results are valid for perfect models which is not often the case in reality. In the context of this feasibility study, the evaluation of robustness could only be empirical through simulations which include perturbations to be modelled in chapter 3 [8, 37, 40, 38].

2.6.12.5 MPC Conclusion

For the design of the low-level module controller, linear and nonlinear Model Predictive Control approaches offer flexible and varied techniques which are very well suited to the control problem tackled in this study [8, 37, 40, 38]. Below are summarised the main reasons why:

1. MPC techniques can draw on explicit physical models.

- 2. MPC techniques use an optimisation process to produce control commands with derivable stability properties.
- 3. Multiple objectives can be combined in a state space formulation with one cost/objective function. These objectives can be tracked as one reference trajectory only.
- 4. A set of parameters is available for the design and tuning the controller in terms of performance and stability. These are the following:
 - (a) the time horizon of the optimisation
 - (b) the cost function weights with or without terminal cost
 - (c) soft or hard constraints such as feasibility or terminal sets
- 5. There is flexibility in the analysis and synthesis of the controller with the possibility of reducing it to a regulator analysis and combining linear and nonlinear MPC techniques to play on the trade off between optimality of the control command solution and the closed-loop stability and timescale of the plant.

In the next section, the solution proposed by this study will be outlined with a focus on how it bridges the domains exposed in this chapter while addressing in a novel way some of their respective gaps.

2.7 Gap analysis and Solution Rational

The methods for de-spinning an asteroid described in sections 2.4 and 2.5 all use model-based attitude control. These dynamic models use parameters like moments of inertia to describe the rotational motion of the spacecraft and calculate the control inputs to the various actuators. However, all of these studies assume that the asteroid dynamic parameters are known a priori and are readily usable for computations which is an impractical proposition in reality. Space debris removal missions face similar problems but some take the more realistic approach of identifying the unknown dynamic parameters of the target object to be de-orbited. This identification process usually relies on the input of disturbances whose measured effects feedback into the model to tune the parameters. This process is relatively inexpensive energy-wise providing the mass ratio between the spacecraft and the object is in favour of the former or balanced.

This latter aspect of scalability represents one of the biggest challenges to the use of traditional spacecraft and actuation methods for all types and sizes of objects. Most of the works produced to date on asteroid capture and de-spin envisage a scenario where the mass and size of the asteroid and the spacecraft are of the same order of magnitude or in favour of the spacecraft. Sizing up traditional actuators like momentum wheels or thrusters seems rather impractical when considering asteroids with diameters of the order of 100m or more.

Another disadvantage of traditional asteroid capture and de-spin by spacecraft is that most of these solutions put forward different ways of enveloping the asteroid but all rely on friction to remove rotational energy from the asteroid. Enveloping, in this perspective, has implications on the mechanical design and structure of the spacecraft which has to be able to sustain collisions and in a way which does not endanger some of the key components and appendages of the spacecraft such as its solar panels for instance. It also has implications on other key aspects of the mission such as the propellant consumption which, if underestimated, could lead to failure. Finally, energy dissipation does not necessarily mean a better situation to handle. It means, as per [7], that the rotation will settle about the major axis of the object. Although this is the lowest energy and a stable rotational state, the rotation occurs about the axis with the largest moment of inertia. Since objects are never symmetrical, it makes the point of capture a critical parameter for object retrieval as it affects the maintenance of the pointing direction.

The tentacles approaches exposed in [56] for active debris removal are the inspiration behind using SR robots for capturing and building on uncooperative rotating objects, as SR robots were viewed as a mean to expand the tentacles approach to a more active and versatile solution.

44

There are no general solution to the reconfiguration problem. Solutions are usually found for specific applications where the problem definition is detailed and the engineering requirements for the SR robot are specifically suited to the problem definition [63]. For the task at hand of deploying a structure on a rotating uncooperative object, the end configuration is known and prescribed: a coiled chain of module going from the tip of the spin axis of the object round to the plane containing the centre of mass. In order to simplify the problem further, avoid local minima and disconnections and allow greater focus on the physical effect of deployment, the way the module should move around the structure is also prescribed: one-by-one deployment.

On the one hand, the problem is a reconfiguration by control, as defined in section 2.6.7, where the robot controller makes the robot modules move in such a way as to end up in the prescribed configuration. On the other hand, the interactions with the physical environment and the tracking of a reference state trajectory add an extra set of constraints that are more akin to suit a task-driven reconfiguration. The closer class to which this task belongs to is the manipulations class. In the literature, manipulations involve interactions between the environment and the robot reconfiguring as an end effector. There is not a real body of work for designing controller for manipulation tasks other than in this setting [63]. The task at hand in this study could be better defined as a manipulation through reconfiguration of the entire body via changes of mass distribution and while taking into account sensory inputs from the environment in a kind of stigmergic way. No guidelines have been found in the literature for this specific situation and no reference to the use of changes of mass distribution as a control mechanism has been found for a space application.

To all of the preceding constraints, another set of constraints has to be added which is specific to the space environment: redundancy and robustness. Only a decentralised controller can really fulfil it. This implies that, although each module can take inputs from the neighbouring modules and its environment, it makes decisions independently with no reliance on a central controller. As with any decentralised

system, the emergence of the right behaviour and the correctness of its controller are hard to predict and ensure at the design phase [63]. In order to ensure that the robot behaves correctly, a top-down approach to the controller design is more appropriate than its reverse traditional bottom up approach for decentralised systems. The rotational dynamic model of a deformable medium developed in [66] can be used to accurately model the physical interactions of the rotating object and the robot modules. This model considers the deformable medium as being composed of a rigid part which can be equated to the rotating object of this study and a deformable part or moving part which can be equated to the moving robot modules. This physical understanding of the environment and the tasks can be then bridged back with the low level controller of each module in the form of a Linear (MPC) or Non Linear (NMPC) Model Predicative Controller which ensures correctness and compliance with the disturbance minimisation constraint. Then, the higher level of control can use a behaviour-based approach whose design in simple terms and correctness assessment is rendered easier because the emergence of a behaviour complying with the physical constraints of the problem has been taken into account at the lowest level of the controller. To the knowledge of the author, this controller design approach has not been attempted for a SR robots yet. Finally, it is envisaged that tackling the physical interactions at the lowest controller level will improve design flexibility for the higher level allowing expansion of the controller ability to tackle other tasks such as identifying inertia parameters or developing more active actuation methods.

The detail of the implementation of the design principles of the robot modules controller laid out in this section will be found in chapter 3 for the physical model and in chapter 4 for the entire controller.

Chapter 3

Dynamic Model of a Rotating Ellipsoid with Relatively Rotating Additional Surface Masses and Landing Site Selection

In this chapter is laid out a dynamic model describing the interactions between a free-floating rotating object of any shape and other objects moving freely on its surface. This model assumes that this collection of objects forms an isolated system modelled as a continuum and subject to small and negligible perturbations in magnitude and timescale. The proposed model characterises the internal angular momentum exchanges of the system while it experiences changes in mass distribution in other words deformation. The chapter also highlights the way control parameters will be chosen for the robot's modules controller in the next chapter and deals with the selection of the robot's landing site as well as the set up of the system's simulation.

3.1 Introduction: the Yo-Yo de-spin mechanism towards continuum modelling

3.1.1 The Yo-Yo de-spin mechanism model and limitations

The yo-yo de-spin mechanism is as device used to de-spin symmetrical objects. It consists of a pair of masses each attached to its own cable. Both of these cables are wound around the object. When released while the symmetric object is spinning, the masses are projected outward by the centrifugal force. The resulting cable tension exert a moment on the spinning object which slows its angular velocity as the cable unwinds. [16]

In this section the object is defined as a cylinder. Because the mass distribution is symmetrical, the cylinder spins around its axis of symmetry O_z and all the forces exerted by the yo-yo de-spin mechanism are located in the perpendicular plane passing through the cylinder's centre of mass i.e. the plane of symmetry. Therefore, the analysis of the dynamics can be performed in 2D and for one mass. Figure 3.1 below shows a 2D schematic of one of the masses of a yo-yo de-spin mechanism attached to a cylinder. [16]

In [16] page 576-580, the yo-yo de-spin mechanism is modelled and its design drivers and parameters derived and sized. In the following paragraph, a short summary of the modelling process presented in the book is reproduced in order to introduce the approach that is developed in this thesis and to highlights the difference between the two.

In [16], the system is assumed to be composed of a cylinder with radius R, moment of inertia about the O_z axis \mathbf{I}_z and rotational speed about the O_z axis ω , two additional masses of respective mass $\frac{m}{2}$ and two massless cables. The mechanism controlling the de-spin, the two additional masses and cables, is a passive mechanism. The main control design parameter i.e. the length of the cables as well as all of the dynamic properties which describe the interactions between the yo-yo despin mechanism and the cylinder are system specific and derived from the system parameters described above.



Figure 3.1: Yo-Yo De-spin Mechanism Diagram.

Drawing on the extensiveness and conservation of the angular momentum and energy combined with geometric observations, a total de-spin of the cylinder is achieved by attaching the two additional masses to cables with length $l = R\sqrt{1 + \frac{\mathbf{I}_z}{mR^2}}$ which unwrap over a total angle $\phi = \sqrt{1 + \frac{\mathbf{I}_z}{mR^2}}$. The cylinder's angular velocity can be expressed as a function of time $\omega = \left(\frac{\frac{2\mathbf{I}_z}{\mathbf{I}_z}}{\frac{1}{mR^2} + \omega_0^2 t^2} - 1\right)\omega_0$ where ω_0 is the initial angular velocity of the cylinder. Moreover, between the start of the de-spinning process and its completion the cylinder travels over an angle $\theta_1 = \sqrt{1 + \frac{\mathbf{I}_z}{mR^2}} \left(\frac{\pi}{2} - 1\right)$. Finally the magnitude of the cable tension N can even be assessed as a time-dependent function $N = \frac{2\mathbf{I}_z}{R} \frac{1 + \frac{\mathbf{I}_z}{(1 + \frac{\mathbf{I}_z}{mR^2} + \omega_0^2 t^2)^2}}{R}$ [16].

All the parameters presented above give quantitative information on the state of the system but none describes how the relative motion of the additional mass affects the rate of change of angular momentum of the cylinder \dot{h} . \dot{h} is equal to the moment exerted by cable tensions of magnitude N. From a control perspective, the cable tension is the only available parameter given by the dynamic model which could constitute a control command. The problem of the deployment of multiple modules on the surface of a rotating object is akin to the deployment of mobile robots which require linear and angular velocity and acceleration as control inputs. For a deformable rotating object, four variables affect the rate of change of its angular momentum: its angular velocity and acceleration, its mass distribution or moment of inertia and its deformation rate all of which are available from first principle using a continuum model and can be used for feedback control. The modelling of the rotating object and yo-yo de-spin mechanism as a continuum will be the object of the next section.

3.1.2 The Yo-Yo de-spin mechanism from a continuum perspective

In this section, the entire system (cylinder and yo-yo de-spin mechanism) will be modelled as a continuum and seen from the perspective of angular momentum exchange between parts of this continuum and material deformation. The various components of the system are laid out in Figure 3.1.

It assumed that the system is isolated, that the cables have no inertia and that the additional masses do not rotate on themselves and can be modelled as point masses. Also, given the symmetry of the system, its dynamic properties can be modelled from the perspective of just one additional mass totalling the to additional masses.

Three frames of reference are necessary to describe the interaction between the masses and the cylinder. All of these frames are centred on O the cylinder's centre of mass. Restricting the description of the dynamics to the plane of symmetry, the first frame is the fixed inertial frame $(O, \vec{\mathbf{X}}, \vec{\mathbf{Y}})$ (not shown in Figure 3.1). The second is the body frame $(O, \vec{\mathbf{i}}_{Body}, \vec{\mathbf{j}}_{Body})$ attached to the body and rotating with angular velocity $\omega = \dot{\theta_1}$ in the inertial frame. The third frame is the polar coordinate frame $(O, \vec{\mathbf{e}}_r, \vec{\mathbf{e}}_{\theta})$ following the centre of mass of the detached mass and rotating with angular velocity $\dot{\theta}$ in the inertial frame.

Using the extensive property of the angular momentum, the system total angular momentum $\overrightarrow{\mathbf{L}}_{[Total]}$ is:

$$\overrightarrow{\mathbf{L}}_{[Total]} = \overrightarrow{\mathbf{L}}_{[Cylinder]} + \overrightarrow{\mathbf{L}}_{[Mass]}$$
(3.1)

The mass has a velocity $\overrightarrow{\mathbf{v}}$ and both the cylinder and the mass rotate about O_z . With $\overrightarrow{\mathbf{k}}$ a unit vector of O_z , from first principle we have:

$$\overrightarrow{\mathbf{L}}_{[Total]} = \left[\mathbf{I}_{Z}\right]\omega\overrightarrow{\mathbf{k}} + \overrightarrow{\mathbf{r}}\wedge m\overrightarrow{\mathbf{v}}$$
(3.2)

Expressing $\overrightarrow{\mathbf{r}}$ and $\overrightarrow{\mathbf{v}}$ in polar coordinates gives:

$$\overrightarrow{\mathbf{L}}_{[Total]} = \left[\mathbf{I}_{\mathbf{Z}}\right]\omega\overrightarrow{\mathbf{k}} + \mathbf{m}\mathbf{r}\overrightarrow{\mathbf{e}_{\mathbf{r}}} \wedge \left(\mathbf{\dot{r}}\overrightarrow{\mathbf{e}_{\mathbf{r}}} + \mathbf{r}\dot{\theta}\overrightarrow{\mathbf{e}_{\theta}}\right)$$
(3.3)

Since $\overrightarrow{\mathbf{e}_{\mathbf{r}}} \wedge \overrightarrow{\mathbf{e}_{\theta}} = \overrightarrow{\mathbf{k}}$:

$$\overrightarrow{\mathbf{L}}_{[Total]} = \left[\mathbf{I}_{\mathbf{Z}}\right] \omega \overrightarrow{\mathbf{k}} + \mathbf{m} \mathbf{r}^2 \dot{\theta} \overrightarrow{\mathbf{k}}$$
(3.4)

Hence the magnitude of the angular momentum is:

$$L_{[Total]} = ||\overrightarrow{\mathbf{L}}_{[Total]}|| = [\mathbf{I}_{\mathbf{Z}}]\omega + \mathbf{mr}^{2}\dot{\theta}$$
(3.5)

As per formulae (10.111) in [16] page 578:

$$L_{[Total]} = [\mathbf{I}_Z]\omega + mR^2[\omega + (\omega + \dot{\phi})\phi^2]$$
(3.6)

Which leads to

$$mr^2\dot{\theta} = mR^2[\omega + (\omega + \dot{\phi})\phi^2]$$
(3.7)

$$\dot{\theta} = \frac{R^2}{r^2} [\omega + (\omega + \dot{\phi})\phi^2] \tag{3.8}$$

Using the Pythagorean theorem as per Figure 3.1:

$$r^2 = R^2 (1 + \phi^2) \tag{3.9}$$

As per formulae (10.115) in [16] page 579:

$$\dot{\phi} = \omega_0 \tag{3.10}$$

Finally, the absolute angular velocity of the additional masses:

$$\dot{\theta} = \frac{[\omega + (\omega + \omega_0)\omega_0^2 t^2]}{1 + \omega_0^2 t^2}$$
(3.11)

From (3.11) the additional masses angular velocity relative to cylinder is given by:

$$\dot{\theta}_{rel} = \dot{\theta} - \omega = \frac{\omega_0^3 t^2}{1 + \omega_0^2 t^2} \tag{3.12}$$

Differentiating (3.9) gives:

$$\dot{r} = \frac{R\omega_0^2 t}{\sqrt{1 + \omega_0^2 t^2}}$$
(3.13)

From the point of view of a deformable finite material continuum, \dot{r} and θ_{rel} respectively represent its linear and angular rates of deformation. Considering the yo-yo de-spin/cylinder system from this perspective, the cylinder constitutes a rigid part while the additional masses constitute a deformable part. Moreover, these rates of deformation are relative velocities which usually constitute the control commands of mobile robots. The modules of the SR robot considered in this study can be regarded as mobile robot moving on the surface of the rotating free-floating object. It naturally follows that the SR robot and rotating free-floating object can be modelled as one deformable finite material continuum where the object constitutes the rigid part while the robot constitutes its discretised deformable part and where deformation rates and their accelerations are translated into control inputs. This idea will be formalised in 3.2.

3.2 System Modelling Using Rotational Dynamics of a Deformable Medium

In this section the system will be modelled as a continuum, a most general approach for a first principle derivation of the dynamic interactions between the rotating object and the robot. In particular, this approach exploits the extensiveness property of angular momentum to discretize the system and enables clear separation between the object and the robot. Finally, the relative motion between the object and the robot is shown and made clearer by the introduction of three different frames of reference. The original derivation of the model can be found in [66]. For the purpose of the verification of its correctness, it was rederived from a programmatic perspective to ensure a consistent writing convention for vectors and tensors used in the simulations' code. A partial version of this derivation can be found in appendix A where the focus was placed on the particle level where the main calculations occur. When not stated otherwise, a simple application of the integral operator is required to ensure validity over the continuum.

3.2.1 System definition

In the very general sense, the system was defined as a randomly tumbling deformable continuum with a rigid part and a deformable part. This definition is necessary and sufficient to derive the model that will be use throughout this study.

For the purpose of studying the interaction of a deploying self-reconfigurable robot on the surface of a rotating object, the system was further specified and should be understood as follows:

- The rigid part was defined as a randomly tumbling and uncooperative object like an asteroid for instance.
- The deformable part was assumed to be attached or in contact with the rigid part at all times allowing for exchange of momentum through contact forces.
- The deformable part was further defined as either a moving rigid mass or modules of a self-reconfigurable robot in constant contact with one another.

The system's model will be presented in the next sections on the basis of these definitions. For the purpose of illustration, figure 3.2 below shows the general definition of a system and figure 3.3 shows a robot deployment.



Figure 3.2: System Diagram.



Figure 3.3: Example of a System Deployment.

3.2.2 Physical Hypotheses

As per section 2.2.5, the system defined in 3.2.1 is subject to very small perturbations both in magnitude and timescale. Consequently, the system was assumed to be isolated and in pure rotation with respect to the inertial frame defined in 3.2.3. No external forces or moments are exerted on the system and no energy is coming in or out of it. The total energy of the system is therefore constant:

$$T_{Total} = T_{Potential} + T_{Kinetic} = constant \tag{3.14}$$

as well as the total angular momentum vector which is also conserved:

$$\overrightarrow{\mathbf{L}}_{inertial} = \overrightarrow{\mathbf{c}onstant}$$
(3.15)

It is further assumed that any moving part of the system possesses an internal energy storage containing enough but unquantified amount of potential energy to be able to move relative to the object to any target position and compensate frictions and dissipation while moving. This was equivalent to neglecting friction forces. The transfer of energy was also assumed to have no hysteresis i.e. it is instantaneous.

Moreover, in the perspective of simplifying the analysis and exploitation of the model for the design of the robot controller, this study focused on pure exchange of momentum through contact forces and its parametrization with relative angular positions velocities and accelerations. It was therefore assumed that the moments arising from stresses induced in the continuum by relative motions are neglected and the assessment of their impact left for future work.

Finally, the moving part was assumed to maintain contact with the surface of the object at all times while moving. This creates an holonomic constraint the impact of which was taken into account as will be seen in 3.2.6 and appendix D.

3.2.3 Frames of References and Relative Angular Velocity

3.2.3.1 Frames of References

The clear description of the interactions and relative motion between the rigid part and moving part of the continuum, the tumbling object and the robot is best achieved by attaching a frame of reference to each of these parts. It is however feasible and more practical to set the origins of these frame at the centre of mass of the continuum. The following frames definitions are taken directly from [66] and reproduced here to describe how the model derivation was approached.

The entire space is assumed to be a field to which is attached a Euclidean vector space and a Euclidean point space.

The continuum is formed of a rigid part and deformable moving parts. The relative rotational motion of the moving parts compared to the rigid part rather the difference between two absolute angular velocities is best seen from the point of view of an extra amount of angular velocity added to a rigid-body angular velocity that would be a base angular velocity across the continuum. This helps defined formally three frames of reference:

1) An inertial Cartesian coordinate frame of reference centred on the centre of mass of the continuum $[O] \equiv (O, \vec{\mathbf{u}}_i)$ where $\vec{\mathbf{u}}_i$ are basis vectors.

2) A rotating Cartesian frame of reference with same origin at the centre of mass of the continuum attached to the rigid part and representing the rigid-body rotation of the continuum $[O'] \equiv (O, \overrightarrow{\mathbf{u}'_i}(t))$ where $\overrightarrow{\mathbf{u}'_i}(t)$ are time-dependent basis vectors. [O'] rotates with respect to [O] with angular velocity $\overrightarrow{\Omega}(t)$

3) Another rotating Cartesian frame of reference with same origin at the centre of mass of the continuum and attached to each moving part, $[O''] \equiv (O, \vec{\mathbf{u}_i'}(x', t))$ rotating relative to [O'] with a non-rigid body rotation where $\vec{\mathbf{u}_i'}(x', t)$ is a configuration (in [O']) and time dependent vector basis. [O''] rotates with respect to [O'] with an angular velocity $\vec{\Psi}(x', t)$. Each moving particle can be regarded as possessing its own system [O''] despite having its origin in O. [O''] is responsible for a rotation different from that of [O'] relative to [O].

The three frames above have the same origin O at the centre of mass of the continuum. This choice results from the hypothesis in 3.2.2 that the continuum does not have a translational motion in the inertial frame [O] and that consequently the centre of mass remains at its initial position in [O]. Conversely, in a frame attached to another point of the continuum, the centre of mass would appear to move with any occurring deformation.

3.2.3.2 Non-Rigid Body Rotation Frame [O"]

The deformation of the continuum is best measured by a deformation rate and acceleration fields which are simply the translational velocity and acceleration of a moving particle $(\overrightarrow{\mathbf{v}''} \text{ and } \overrightarrow{\mathbf{v}''})$ expressed in and with respect to its own [O''] frame.

Angular velocities are usually expressed in body frame out of convenience [60]. Moreover, the three frames [O], [O'] and [O''] have the same origin O. In addition, as will later be see in section 3.3, if a moving part of the continuum rotates on itself about the radial axis passing through the origin of the frame, it does not contribute to the overall exchange of angular momentum. A point mass approximation for each robot module is therefore perfectly reasonable.

In this perspective, a convenient way of introducing explicit angular quantities is to define [O''] as a local spherical coordinate frame attached to the moving part of the continuum or robot module. As illustrated in Figure 3.2, the spherical coordinate base is defined as

$$\vec{\mathbf{e}_{\mathbf{r}}} = \begin{bmatrix} \cos(\phi) \sin(\theta) \\ \sin(\phi) \sin(\theta) \\ \cos(\theta) \end{bmatrix}$$
(3.16)
$$\vec{\mathbf{e}_{\theta}} = \begin{bmatrix} \cos(\phi) \cos(\theta) \\ \cos(\phi) \sin(\theta) \\ -\sin(\theta) \end{bmatrix}$$
(3.17)
$$\vec{\mathbf{e}_{\phi}} = \begin{bmatrix} -\sin(\phi) \\ \cos(\phi) \\ 0 \end{bmatrix}$$
(3.18)

In the body frame [O'], the location any moving mass on the surface of the rotating object is given by

$$\overrightarrow{\mathbf{x}}' = r \overrightarrow{\mathbf{e}_{\mathbf{r}}} \tag{3.19}$$

where r is the radial length

3.2.3.3 Relative Angular Velocity

To summarise, the total rotational velocity is a field quantity $\vec{\Omega}(x,t)$ for the total rotation of the configurations of the continuum relative to the inertial frame [O].

$$\overrightarrow{\mathbf{\Omega}}(x,t) = \overrightarrow{\mathbf{\Omega}}(t) + \overrightarrow{\mathbf{\Psi}}(x,t)$$
(3.20)

 $\vec{\Omega}(t)$ spins of [O'] relative to [O] and is the rigid body rotation of continuum. $\vec{\Psi}(x,t)$ spins of [O''] relative to [O'] expressed in inertial system [O] and is the non-rigid body rotation of the configurations of the continuum or relative angular velocity.

Using the local spherical coordinates defined in 3.2.3.2 and vector identification and differentiation, the relative rotational speed and acceleration projected in the body frame are respectively:

$$\vec{\Psi}(x,t) = \begin{bmatrix} -\sin(\phi)\dot{\theta} \\ \cos(\phi)\dot{\theta} \\ \dot{\phi} \end{bmatrix}$$
(3.21)

and

$$\dot{\vec{\Psi}}(x,t) = \begin{bmatrix} -\sin(\phi)\ddot{\theta} - \cos(\phi)\dot{\phi}\dot{\theta} \\ \cos(\phi)\ddot{\theta} - \sin(\phi)\dot{\phi}\dot{\theta} \\ \ddot{\phi} \end{bmatrix}$$
(3.22)

3.2.4 Continuous model

In this section, is laid out the general continuous model of a deformable rotating continuum. Originally derived in [66], it was rederived for the purpose of verification and consistency of writing convention across vectors and tensors used in the simulations' code. The convention followed is that left of the dyadic product \otimes , vectors are column and right of the dyadic product \otimes vectors are row. The essential part of the rederivation can be found in appendix A.

3.2.4.1 The Model

In order to avoid loading the reader with notations the notations introduced in 3.2.3.3 were simplified. In its most general version, the model includes moments exerted on the system from internal stresses and external forces although these are neglected.

- $\overrightarrow{\Omega}(t)$, the rigid body rotation constant over the continuum but dependent on time becomes $\overrightarrow{\Omega}$
- $\vec{\Psi}(\vec{\mathbf{x}}, t)$ the added angular velocity relative to the rigid body's and dependent on position within the continuum and time becomes $\vec{\Psi}$.

- [I](x,t) the continuum's moment of inertia matrix over the entire continuum dependent on position within the continuum and time becomes [I].
- $\overrightarrow{\mathbf{x}}$ is the position vector of a particle of the continuum.
- $\dot{\overline{\mathbf{x}_0''}}$ is the velocity vector of a particle of the continuum as viewed from the frame [O''] when [O''] is stationary.
- $\vec{\mathbf{x}_0''}$ is the acceleration vector of a particle of the continuum as viewed from the frame [O''] when [O''] is stationary.
- $\overrightarrow{\mathbf{M}}_{Body} = \overrightarrow{\mathbf{0}}$ the system is isolated no body force moments are exerted on it.
- $\overrightarrow{\mathbf{M}}_{Stresses} = \overrightarrow{\mathbf{0}}$ any stress produced by the interaction between a moving mass or module and the object is neglected.
- $\vec{\mathbf{M}}_{Perturbations} = \vec{\mathbf{0}}$ the moment exerted by the small perturbations introduced in section 2.2.5 can be neglected but will later be reintroduced to evaluate the robustness of the modules' low-level controller.

The continuous model is the following:

$$\begin{split} [\mathbf{I}] \cdot \overrightarrow{\widehat{\Omega}} + \overrightarrow{\Omega} \wedge [\mathbf{I}] \cdot \overrightarrow{\Omega} &= -\int_{m} 2\left[(\overrightarrow{\mathbf{x}_{0}''} \cdot \overrightarrow{\mathbf{x}}) [\mathbf{1}] - \overrightarrow{\mathbf{x}_{0}''} \otimes \overrightarrow{\mathbf{x}} \right] \cdot \overrightarrow{\Omega} \cdot \mathbf{dm} \\ &- \int_{m} \left[\left[(\overrightarrow{\mathbf{x}} \cdot \overrightarrow{\mathbf{x}}) [\mathbf{1}] - (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \right] \cdot \overrightarrow{\Psi} \right] + \left[2(\overrightarrow{\mathbf{x}} \cdot \overrightarrow{\mathbf{x}}) [\mathbf{1}] - (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) - (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \right] \cdot \overrightarrow{\Psi} \right] \cdot \mathbf{dm} \\ &- \int_{m} \left[(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}_{0}'}) - (\overrightarrow{\mathbf{x}_{0}''} \otimes \overrightarrow{\mathbf{x}}) \right] \cdot \overrightarrow{\Psi} \cdot dm \qquad (3.23) \\ &- \int_{m} \left[(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}_{0}'}) \wedge \overrightarrow{\Psi} + ((\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \wedge \overrightarrow{\Psi})^{T} \right] \cdot \overrightarrow{\Omega} \cdot dm \\ &- \int_{m} (\overrightarrow{\mathbf{x}} \wedge \overrightarrow{\mathbf{x}_{0}'}) \cdot dm \\ &+ \overrightarrow{\mathbf{M}}_{Body} + \overrightarrow{\mathbf{M}}_{Stresses} + \overrightarrow{\mathbf{M}}_{Perturbations} \end{split}$$

3.2.4.2 Derivation valid in any vector basis

A rotation is a linear application which conserves norms, angles, scalar, cross and dyadic products and transforms a vector basis into another. As it transforms vectors independently of the mass distribution, it can be applied outside and inside the integral operator indifferently in the model derivation. In addition, all the chosen frames of reference share the same origin. This implies that expressing vectors in one frame or another is achieved by applying a rotation only. This further implies that the model can be expressed in any frame of reference directly by just replacing any vector by its expression in a specific frame. The writing of the differential equation of the angular velocity's is therefore frame-independent. Any $\vec{\mathbf{x}}$ vector can be viewed as being expressed in any frame.

3.2.5 Discrete Normalised Model with Point Mass Non-Rigid Parts

For this model all parameters were normalised as per 3.5.1.1. This implies also that [I] is normalised per unit of robot mass m and becomes $[\mathbf{I_n}] = \frac{1}{\mathbf{m}}[\mathbf{I}]$.

Defining n_r as the number of robots modules:

$$\begin{split} [\mathbf{I}_{\mathbf{n}}] \cdot \vec{\Omega} + \vec{\Omega} \wedge [\mathbf{I}_{\mathbf{n}}] \cdot \vec{\Omega} &= \\ & \sum_{i=1}^{n_{\tau}} \left(-2\left[(\vec{\mathbf{x}}_{\mathbf{0}}^{'} \cdot \vec{\mathbf{x}}) [\mathbf{1}] - \vec{\mathbf{x}}_{\mathbf{0}}^{'} \otimes \vec{\mathbf{x}} \right] \cdot \vec{\Omega} \\ & - \left[\left[(\vec{\mathbf{x}} \cdot \vec{\mathbf{x}}) [\mathbf{1}] - (\vec{\mathbf{x}} \otimes \vec{\mathbf{x}}) \right] \cdot \vec{\Psi} \right] \\ & + \left[2(\vec{\mathbf{x}} \cdot \vec{\mathbf{x}}) [\mathbf{1}] - (\vec{\mathbf{x}} \otimes \vec{\mathbf{x}}) - (\vec{\mathbf{x}} \otimes \vec{\mathbf{x}}) \right] \cdot \vec{\Psi} \right] \\ & - \left[(\vec{\mathbf{x}} \otimes \vec{\mathbf{x}}_{\mathbf{0}}^{'}) - (\vec{\mathbf{x}}_{\mathbf{0}}^{'} \otimes \vec{\mathbf{x}}) \right] \cdot \vec{\Psi} \\ & - \left[(\vec{\mathbf{x}} \otimes \vec{\mathbf{x}}) \wedge \vec{\Psi} + ((\vec{\mathbf{x}} \otimes \vec{\mathbf{x}}) \wedge \vec{\Psi})^{T} \right] \cdot \vec{\Omega} \\ & - (\vec{\mathbf{x}} \wedge \vec{\mathbf{x}}_{\mathbf{0}}^{'}) \right) \\ & + \vec{\mathbf{M}}_{Body} + \vec{\mathbf{M}}_{Stresses} + \vec{\mathbf{M}}_{Perturbations} \end{split}$$

All vectors depend on the index number i.

3.24 undergoes a verification with an application of the yo-yo de-spin mechanism in appendix A.

3.2.6 Modelling the Randomly Tumbling Object as an Ellipsoid

From a dynamics stand point, the description of the rotational motion of a randomly tumbling object requires a minimum of three inertia parameters, the three moments of inertia about its principle axes. They contain all the information required to describe the object's mass distribution without any reference to its shape or volume. However, information about the relative position between the object and the robot's modules deploying on its surface is required to accurately calculate the moments they exert on each other as per model in 3.2.4 and 3.2.5. It is difficult and impractical to try to obtain the equation of the surface of an object of any shape. In order to circumvent this problem, asteroids could be modelled as ellipsoids which in many cases could be considered as a good approximation [34]. The equation of the surface of an ellipsoid in Cartesian coordinates system or spherical coordinates is easy to obtain along with its moment of inertia. The holonomic constraints between the robot and the object is also easily derivable (please refer to appendix D for its derivation) and contribute to the overall accuracy of the model. This is why the randomly tumbling object is modelled as an ellipsoid in this study.

As per [34], it is hard to find any pieces of asteroid mass distribution data other than density and sometimes estimated moments of inertia. In line with this, uniform mass distribution is assumed for the ellipsoid. As will be seen later, this hypothesis will conveniently simplify the modelling process by maintaining the centre of mass O of the ellipsoid at the centre of its volume.

Assuming that the centre of mass of the ellipsoid is the origin O of the inertial frame [O] defined in 3.2.3, the equation of its surface in Cartesian coordinates is:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1 \tag{3.25}$$

where a, b, c are positive real numbers representing half the length of the respective principal axes X, Y, Z where the points (a, 0, 0), (0, b, 0) and (0, 0, c) lie at their intersection with the surface. In this study, the convention establishes the Z axis as the main rotating or spinning axis. All shape considerations are therefore parametrised with respect to the length c.

- If (a = b) > c, the ellipsoid is an oblate spheroid (Figure 3.4).
- If (a = b) < c, the ellipsoid is a prolate spheroid (Figure 3.5).
- If a = b = c, the ellipsoid is a sphere.
- If a > b > c, the ellipsoid is a general or tri-axial ellipsoid.



Figure 3.4: Oblate Ellipsoid.



Figure 3.5: Prolate Ellipsoid.

The moments of inertia of an ellipsoid are easily obtained as follows:

Along a or X axis

$$I_x = \frac{MR}{5}(b^2 + c^2) \tag{3.26}$$

Along b or Y axis

$$I_y = \frac{MR}{5}(a^2 + c^2) \tag{3.27}$$

Along c or Z axis

$$I_z = \frac{MR}{5}(a^2 + b^2) \tag{3.28}$$

where MR is the normalised mass of the ellipsoid per unit of robot module mass. Parameters normalisation will be covered in more details in section 3.5.1.1.

The longest axis (shortest axis) has the smallest (largest) moment of inertia receptively. The major axis has the maximum moment of inertia while the minor axis has the minimum moment of inertia [60].

3.3 Non-Contributing Rotation of an Additional Mass and Point Mass Hypothesis

The section examines the possibility of modelling each robot module as a point mass. The reason for such an approximation is that these modules are small in comparison to the object and it should lead to the simplification of the model by avoiding intensive computation for calculating local time-dependent moment of inertia matrices. This comes down to neglecting the rotation of a module on its own axis.

Let's examine, in frame [O'], a simple situation where there is the body (the rigid part) and one point mass m moving on its surface (the non-rigid part). As the angular momentum is an extensive quantity the contribution of each part can be separated as follows:

$$\overrightarrow{\mathbf{L}}^{[O']} = \left[\mathbf{I}_{\mathbf{Body}}^{[O']}(\mathbf{t})\right] \overrightarrow{\Omega}_{\mathbf{t}} + \left[\mathbf{I}_{\mathbf{Mass}}^{[O']}(\mathbf{x}, \mathbf{t})\right] \left(\overrightarrow{\Omega}_{\mathbf{t}} + \overrightarrow{\Psi}(\mathbf{x}, \mathbf{t})\right)$$
(3.29)

$$\overrightarrow{\mathbf{L}}^{[O']} = \left(\left[\mathbf{I}_{\mathbf{Body}}^{[\mathbf{O}']}(\mathbf{t}) \right] + \left[\mathbf{I}_{\mathbf{Mass}}^{[\mathbf{O}']}(\mathbf{x}, \mathbf{t}) \right] \right) \overrightarrow{\Omega}_{\mathbf{t}} + \left[\mathbf{I}_{\mathbf{Mass}}^{[\mathbf{O}']}(\mathbf{x}, \mathbf{t}) \right] \overrightarrow{\Psi}(\mathbf{x}, \mathbf{t})$$
(3.30)

where $\left[\mathbf{I}_{\mathbf{Mass}}^{[\mathbf{O}']}(\mathbf{x}, \mathbf{t})\right] = \mathbf{m} \left[\mathbf{r}^{2}[\mathbf{Id}] - \mathbf{r} \overrightarrow{\mathbf{e}_{\mathbf{r}}} \otimes \mathbf{r} \overrightarrow{\mathbf{e}_{\mathbf{r}}}\right]$ in spherical coordinates $(\overrightarrow{\mathbf{x}} = r \overrightarrow{\mathbf{e}_{\mathbf{r}}})$

In spherical coordinates:

$$\begin{bmatrix} \mathbf{I}_{Mass}^{[O']}(x,t) \end{bmatrix} = m \begin{bmatrix} 1 - \cos(\phi)^2 \sin(\theta)^2 & -\sin(\phi)\cos(\phi)\sin(\theta)^2 & -\cos(\theta)\cos(\phi)\sin(\theta) \\ -\cos(\phi)\sin(\phi)\sin(\theta)^2 & 1 - \sin(\phi)^2\sin(\theta)^2 & -\cos(\theta)\sin(\phi)\sin(\theta) \\ -\cos(\phi)\sin(\theta)\cos(\theta) & -\sin(\phi)\sin(\theta)\cos(\theta) & \sin(\theta)^2 \\ & (3.31) \end{bmatrix}$$

Calculating the determinant gives:

$$det(\left[\mathbf{I}_{\mathbf{Mass}}^{[\mathbf{O}']}(\mathbf{x},\mathbf{t})\right]) = \mathbf{1} - \cos(\phi)^2 \sin(\theta)^2 - \sin(\phi)^2 \sin(\theta)^2 - \cos(\theta)^2 \qquad (3.32)$$

$$det(\left[\mathbf{I}_{\mathbf{Mass}}^{[\mathbf{O}']}(\mathbf{x},\mathbf{t})\right]) = \sin(\theta)^2 - \sin(\theta)^2(\cos(\phi)^2 + \sin(\phi)^2) = \mathbf{0}$$
(3.33)

The instantaneous eigenvalues of $[\mathbf{I}_{\mathbf{Mass}}^{[\mathbf{O}']}(\mathbf{x}, \mathbf{t})]$ are 1, 1 and 0. Its rank is therefore 2 at any given time.

Assuming that:

$$\begin{bmatrix} \mathbf{I}_{\mathbf{Mass}}(\mathbf{x}, \mathbf{t}) \end{bmatrix} = \begin{bmatrix} I_1 & I_2 & I_3 \\ I_4 & I_5 & I_6 \\ I_7 & I_8 & I_9 \end{bmatrix}$$
(3.34)

then a relative angular velocity $\overrightarrow{\Psi}(x,t)$ that would not contribute to the overall angular momentum would satisfy the equation below:

$$\overrightarrow{\Psi}(x,t) = \Psi_z \begin{bmatrix} -\begin{bmatrix} I_1 & I_2 \\ I_4 & I_5 \end{bmatrix}^{-1} \begin{bmatrix} I_3 \\ I_6 \end{bmatrix} \end{bmatrix}$$
(3.35)

where Ψ_z can be any function.

At any one time the first 2 rows combine into the third with the coefficient

$$\mu = \begin{bmatrix} -\cos(\phi) * \tan(\theta) \\ -\sin(\phi) * \tan(\theta) \end{bmatrix}$$
(3.36)

Leading to:

$$\vec{\Psi}(x,t) = \Psi_z \begin{bmatrix} \cos(\phi)\tan(\theta)\\ \sin(\phi)\tan(\theta)\\ 1 \end{bmatrix}$$
(3.37)

Eventually:

$$\vec{\Psi}(x,t) = \Psi_z \begin{bmatrix} \cos(\phi)\sin(\theta)\\\sin(\phi)\sin(\theta)\\\cos(\theta) \end{bmatrix}$$
(3.38)

It can be easily verified that $[\mathbf{I}_{\mathbf{Mass}}^{[\mathbf{O}']}(\mathbf{x}, \mathbf{t})] \overrightarrow{\Psi}(\mathbf{x}, \mathbf{t}) = \overrightarrow{\mathbf{0}}$. When the point mass rotates on itself about the radial axis, it does not contribute to the overall angular momentum. Only motions in the spherical coordinates ϕ and θ have a contribution. Anticipating on the module design in chapter 4, there is no practical application for a rotation about the radial axis as it does not cause a relative motion on the surface of the object. Moreover, the intention for the module is not to use it as a substitute for a momentum wheel. Therefore, modelling robot modules as point masses is perfectly coherent hypothesis.

3.4 Landing Site Selection

3.4.1 Modelling the Landing

Landing generates a disturbance to the rotational motion of the object and sets the initial conditions under which the robot will deploy. The optimal landing site should therefore generate minimum disturbance. A strong assumption was made to help with this optimisation process. It postulates that the robot is split in two parts or two identical robots each landing at opposite site on the surface of the object i.e. symmetrically with respect to the centre of mass. This in order to preserve the initial location of the centre of mass in the inertial frame.

The modelling or rather sizing of the landing site impact on the initial condition of the rotational state of the entire system drew on [17]. The object on which the robot is landing is modelled as a body with time-variable mass with an instantaneous change of mass. The model of mass addition used is entirely derived in [17] but adapted to the hypotheses defining the intended application of this study.

Most notably, the angular momentum for body addition was calculated at the centre of mass O of the object, origin of the previously defined frames of reference in section 3.2.3. The addition of mass being symmetric with respect to O, these calculations remain valid since O does not change location relative to the inertial frame.

3.4.2 Systems Definitions and Properties

Three systems have to be defined, the notations are taken from [17]:

- 1. S is the object prior to the addition of mass m. It possesses a mass M, a moment of inertia $[\mathbf{I}_{\mathbf{S}}]$, a linear velocity $\overrightarrow{\mathbf{v}}_{S}$, an angular velocity $\overrightarrow{\mathbf{\Omega}}_{S}$ and an angular momentum with respect to $O \overrightarrow{\mathbf{L}}_{S}$.
- 2. S_1 is the system after landing (object and robot) with mass (M + m), moment of inertia [**I**_{S1}], linear velocity $\overrightarrow{\mathbf{v}}_{S1}$, angular velocity $\overrightarrow{\mathbf{\Omega}}_{S1}$ and angular momentum with respect to $O \overrightarrow{\mathbf{L}}_{S1}$.
- 3. S_2 is the additional mass m or robot prior to landing with moment of inertia $[\mathbf{I}_{\mathbf{S2}}]$, linear velocity $\overrightarrow{\mathbf{v}}_{S2}$, angular velocity $\overrightarrow{\mathbf{\Omega}}_{S2}$ and angular momentum $\overrightarrow{\mathbf{L}}_{S2}$ with respect to its own centre of mass.
- 4. $\overrightarrow{\rho}_{S1}$ is the vector linking the centre of mass of the original object O and the centre of mass of the new object post landing S1.
- 5. $\overrightarrow{\rho}_{S2}$ is the vector linking the centre of mass of the original object O and the centre of mass of the additional mass S2.

3.4.3 Physical Assumptions

1. The original object and the additional mass constitute an isolated system prior and after the addition of this mass. There are no external forces or moments exerted on this system.

- 2. The additional mass is defined as a point mass which implies no rotation about its centre of mass i.e. $\overrightarrow{\mathbf{L}}_{S2} = \overrightarrow{\mathbf{0}}$.
- 3. Only pure angular motion is relevant for this study. Therefore, the linear velocity of the object is neglected and set to the nil vector in the inertial frame: $\overrightarrow{\mathbf{v}}_{S1} = \overrightarrow{\mathbf{0}}$.
- 4. It was assumed that the touch down is occurring under the smoothest conditions possible meaning that during the last phase of approach prior to landing, the robot's rotational motion matches the object's.
- 5. It was assumed that the object has a much larger mass than the additional mass i.e. $M \gg m$.
- 6. Finally, it was assumed that the robot was constituted of two parts landing synchronously on the surface of the object S in symmetric positions with respect to the centre of mass O of the object: $\overrightarrow{\rho}_{S1} = \overrightarrow{\mathbf{0}}$.

3.4.4 Angular Momentum Change During Landing

According to [17] page 16 formulae (2.34), the variation of the angular momentum during mass addition is:

$$\Delta \overrightarrow{\mathbf{L}}_{O} = \overrightarrow{\mathbf{L}}_{S1} - \overrightarrow{\mathbf{L}}_{S2} - \overrightarrow{\mathbf{L}}_{S} + \overrightarrow{\rho_{S2}} \wedge m \left(\overrightarrow{\mathbf{v}}_{S1} - \overrightarrow{\mathbf{v}}_{S2} \right)$$
(3.39)

From [17] page 20 quote:

"According to the principle of the angular momentum, the variation of the angular momentum (2.34) page16 in the time interval Δt is equal to the impulse $\mathbf{J}^{\mathbf{M}}$, which is the sum of the impulse of the moment $\overrightarrow{\mathbf{M}}_{0}^{Fr}$ of the resultant force for the point O and of the impulse of the resultant torque $\overrightarrow{\mathbf{M}}$, caused by active and reaction torques, i.e."

$$\Delta \overrightarrow{\mathbf{L}}_{O} = (\overrightarrow{\mathbf{M}}_{0}^{Fr} + \overrightarrow{\mathbf{M}}) \Delta t = \overrightarrow{\mathbf{J}}^{\mathbf{M}}$$
(3.40)

Again from [17] page 20 quote: "If impulses of torques $\vec{\mathbf{J}}^{\mathbf{M}}$ are quite small, due to short time Δt , system is assumed to be without action of external torques. In

that case variation of the angular momentum is $\Delta \vec{\mathbf{L}}_O = \vec{\mathbf{0}}$ ie angular momentums of the system before and after mass variation remains invariable."

From section 3.4.3, under assumption number 2 $\overrightarrow{\mathbf{L}}_{S2} = \overrightarrow{\mathbf{0}}$, under number 3 $\overrightarrow{\mathbf{v}}_{S1} = \overrightarrow{\mathbf{0}}$ and number 4 leads to $\overrightarrow{\mathbf{J}}^{\mathbf{M}} \approx \overrightarrow{\mathbf{0}}$.

Hence equation 3.39 becomes:

$$[\mathbf{I}_{\mathbf{S1}}] \overrightarrow{\mathbf{\Omega}}_{\mathbf{S1}} = [\mathbf{I}_{\mathbf{S}}] \overrightarrow{\mathbf{\Omega}}_{\mathbf{S}} + \overrightarrow{\rho_{\mathbf{S2}}} \wedge \mathbf{m} \overrightarrow{\mathbf{v}}_{\mathbf{S2}}$$
(3.41)

Since there are two robots landing symmetrically with respect to the centre of mass of the object, for the first robot $\frac{m}{2}\overrightarrow{\rho_{S2}} \wedge \overrightarrow{\mathbf{v}}_{S2}$ and for the second robot $\frac{m}{2}\overrightarrow{\rho_{S2}} \wedge -\overrightarrow{\mathbf{v}}_{S2}$. The overall velocity of the additional mass is $\overrightarrow{\mathbf{0}}$. This finally leads to:

$$\overrightarrow{\mathbf{L}}_{S1} = \overrightarrow{\mathbf{L}}_{S} \tag{3.42}$$

3.4.5 Error Vector and Sizing Method

As was seen in section 3.4.4, the addition of mass alters the angular velocity either by pure mass addition or by mass addition combined with relative angular momentum if there is a velocity difference at impact. For the purpose of this sizing exercise, an error vector was added to the original angular velocity $\vec{\varepsilon}_t$ such that $\vec{\Omega}_t^1 = \vec{\Omega}_t^0 + \vec{\varepsilon}_t$ where $\vec{\Omega}_t^0$ is the angular velocity of the object before landing and $\vec{\Omega}_t^1$ is the angular velocity of the object after landing. Then considering the additional mass as rigidly attached to the object and using the result in section 3.4.4 in body frame [O']:

$$\left(\left[\mathbf{I}_{\mathbf{Body}} \right] + \left[\mathbf{I}_{\mathbf{Mass}}(\mathbf{x}_{1}, \mathbf{t}_{1}) \right] \right) \overrightarrow{\Omega}_{\mathbf{t}}^{1} = \left[\mathbf{I}_{\mathbf{Body}} \right] \overrightarrow{\Omega}_{\mathbf{t}}^{0}$$
(3.43)

Finally

$$\overrightarrow{\varepsilon}_{t} = -\left(\left[\mathbf{I}_{\mathbf{Body}}\right] + \left[\mathbf{I}_{\mathbf{Mass}}(\mathbf{x}_{1}, \mathbf{t}_{1})\right]\right)^{-1} \left(\left[\mathbf{I}_{\mathbf{Mass}}(\mathbf{x}_{1}, \mathbf{t}_{1})\right] \overrightarrow{\Omega}_{\mathbf{t}}^{\mathbf{0}}\right)$$
(3.44)

The impact of landing is then measured by the normalised error:

$$\varepsilon = \frac{\|\overrightarrow{\varepsilon}_t\|}{\|\overrightarrow{\Omega}_t^0\|} \tag{3.45}$$

The norm of $\overrightarrow{\Omega}_{t}^{0}$ was chosen to be equal to the upper bound of the norm of an asymmetric ellipsoid's angular velocity. The reason being that, as per equation 3.56 in section 3.5.3.2, this norm explicitly depends on the parameter $\frac{h^2}{2T}$ which prescribes the magnitude of the angular momentum as a function of the level of kinetic energy and which is bounded by the lowest and largest value of the moments of inertia of the object. It is a very convenient parameter for describing explicitly the dynamic properties of the object. All the quantities were normalised as per section 3.5.1.1 and five values of $\frac{h^2}{2T}$ were chosen over the range of moments of inertia of the object. The error simulation results are presented in the next section.

3.4.6 Landing Site Location Impact on the Rotational State of the Target Asteroid

3.4.6.1 Landing Error Simulation Set Up

The landing error was calculated, as per formulae 3.44 and 3.45 in section 3.4.5, for ten ellipsoids, four mass ratios of the ellipsoid to the robot and five values of $\frac{h^2}{2T}$ chosen over the range of moments of inertia of the object. For each combination of these parameters, the value of the landing error was minimised over a wide range of landing site located by spherical coordinates angles $(\theta, \phi) \in [0, \pi] \times [0, 2\pi[$. These minimum values will now be analyse against the parameters values.

The values of the parameters are found in tables 3.1, 3.2 and 3.3 below.

The range of normalised moments of inertia of the ellipsoids in table 3.3 are calculated as per equations 3.26, 3.27 and 3.28 in section 3.2.6 and $I_{max} = max(I_x, I_y, I_z)$ and $I_{min} = min(I_x, I_y, I_z)$.

Chapter 3: Dynamic Model of a Rotating Ellipsoid with Relatively Rotating Additional Surface Masses and Landing Site Selection

Ellipsoid	1^{st} Semi-Axis Length	2^{nd} Semi-Axis Length	3^{rd} Semi-Axis Length
Sphere	1	1	1
Prolate	$\frac{1}{10}$	$\frac{1}{10}$	1
Prolate	$\frac{1}{2}$	$\frac{1}{2}$	1
Prolate	$\frac{4}{5}$	$\frac{4}{5}$	1
Oblate	$\frac{5}{4}$	$\frac{5}{4}$	1
Oblate	2	2	1
Oblate	10	10	1
Asymmetric	$\frac{1}{10}$	$\frac{4}{5}$	1
Asymmetric	$\frac{1}{3}$	$\frac{1}{2}$	1
Asymmetric	4 5	$\frac{9}{10}$	1

Table 3.1: Ellipsoids Dimensions Normalised by the Body's Z Semi-Axis Length

Mass Ratio 10 100 1000 10000

Table 3.2: Dimensionless Mass Ratios Ellipsoid to Robot

$\frac{\mathrm{h}^2}{\mathrm{2T}}$	I_{min}	$\frac{(3I_{min}+I_{max})}{4}$	$\frac{(I_{min}+I_{max})}{2}$	$\frac{(I_{min}+3I_{max})}{4}$	Imax

Table 3.3: Dimensionless $\frac{h^2}{2T}$ Parameter Values

3.4.6.2 Landing Error Simulation Results Analysis

In table 3.4 below, the spherical coordinates of the optimum landing sites are displayed for various initial rotational states of the ellipsoid which are parametrised by $\frac{\hbar^2}{2T}$.

Ellipsoid / $\frac{h^2}{2T}$	I _{min}	$\frac{(3I_{min}+I_{max})}{4}$	$\frac{(I_{min}+I_{max})}{2}$	$\frac{(I_{min}+3I_{max})}{4}$	I_{max}
Sphere $(1, 1, 1)$	$(90^{\circ}, 0^{\circ})$	$(90^{\circ}, 0^{\circ})$	$(90^{\circ}, 0^{\circ})$	$(90^{\circ}, 0^{\circ})$	$(90^{\circ}, 0^{\circ})$
Prolate $(\frac{1}{10}, \frac{1}{10}, 1)$	$(0^{\circ}, 0^{\circ})$	$(9^{\circ}, 45^{\circ})$	$(9^{\circ}, 90^{\circ})$	$(18^{\circ}, 225^{\circ})$	$(90^{\circ}, 225^{\circ})$
Prolate $(\frac{1}{2}, \frac{1}{2}, 1)$	$(0^{\circ}, 0^{\circ})$	$(27^{\circ}, 225^{\circ})$	$(45^{\circ}, 225^{\circ})$	$(54^{\circ}, 225^{\circ})$	$(90^{\circ}, 225^{\circ})$
Prolate $(\frac{4}{5}, \frac{4}{5}, 1)$	$(0^{\circ}, 0^{\circ})$	$(36^{\circ}, 45^{\circ})$	$(54^{\circ}, 45^{\circ})$	$(63^{\circ}, 225^{\circ})$	$(90^{\circ}, 225^{\circ})$
Oblate $(\frac{5}{4}, \frac{5}{4}, 1)$	$(90^{\circ}, 225^{\circ})$	$(72^{\circ}, 45^{\circ})$	$(54^{\circ}, 225^{\circ})$	$(45^{\circ}, 225^{\circ})$	$(0^\circ, 0^\circ)$
Oblate $(2,2,1)$	$(90^{\circ}, 225^{\circ})$	$(72^{\circ}, 225^{\circ})$	$(63^{\circ}, 45^{\circ})$	$(45^{\circ}, 225^{\circ})$	$(0^\circ, 0^\circ)$
Oblate $(10, 10, 1)$	$(90^{\circ}, 225^{\circ})$	$(72^{\circ}, 225^{\circ})$	$(63^{\circ}, 225^{\circ})$	$(45^{\circ}, 45^{\circ})$	$(0^\circ, 0^\circ)$
Asymmetric $\left(\frac{1}{10}, \frac{4}{5}, 1\right)$	$(0^{\circ}, 0^{\circ})$	$(45^{\circ}, 225^{\circ})$	$(54^{\circ}, 45^{\circ})$	$(72^{\circ}, 225^{\circ})$	$(90^{\circ}, 225^{\circ})$
Asymmetric $(\frac{1}{3}, \frac{1}{2}, 1)$	$(0^{\circ}, 0^{\circ})$	$(27^{\circ}, 45^{\circ})$	$(36^{\circ}, 225^{\circ})$	$(74^{\circ}, 225^{\circ})$	$(90^{\circ}, 225^{\circ})$
Asymmetric $\left(\frac{4}{5}, \frac{9}{10}, 1\right)$	$(0^{\circ}, 0^{\circ})$	$(45^{\circ}, 225^{\circ})$	$(54^{\circ}, 45^{\circ})$	$(72^{\circ}, 45^{\circ})$	$(90^{\circ}, 45^{\circ})$

Table 3.4: Optimal Landing Site in Spherical Angles (θ, ϕ) in Degrees

The parameter $\frac{h^2}{2T}$ sets the rotation state of the object and in particular the location of its instantaneous axis of rotation. For symmetric objects it will influence

the nutation angle and precession motion. In general, when $\frac{h^2}{2T} = I_{min}$, the objects spins about its minor axis whereas when $\frac{h^2}{2T} = I_{max}$, the objects spins about its major axis. For intermediary values of $\frac{h^2}{2T}$, it is difficult to locate the axis of rotation whose location also depends on the inertia properties of the object.

For spheres and symmetric ellipsoids the angle ϕ is not significant due to the symmetry of these objects but is relevant for asymmetric objects.

Spheres have an infinite number of axes of symmetry and all these axes are principal axes with identical values of moment of inertia about them. The results in table 3.4 indicate that the location of the landing is independent of the location of the axis of rotation and could be anywhere between the tip of the axis and the median plane perpendicular to it. The landing model in sections 3.4.3 and 3.4.4 assumes that the robot relative velocity and relative angular velocity with respect to the object are negligible and that angular momentum is transferred without a torque impulse. Adding to this the fact that the angular momentum of the object is collinear with the angular momentum of the robot at landing, the absence of effect at landing is coherent.

Symmetric ellipsoids are of two types: prolate and oblate. The axis of spin of prolate ellipsoids is their minor axis with moment of inertia equals to I_{min} and hence an unstable axis when the ellipsoids' kinetic energy is dissipated [60]. The axis of spin of oblate ellipsoids is their major axis with moment of inertia equals to I_{max} and hence a stable axis when the ellipsoids' kinetic energy is dissipated [60]. In this analysis, the spin axis is the Z axis against which the spherical coordinate angle θ is measured. The results in table 3.4 indicate that when the ellipsoid is in pure spin whether about its minor or major axis, it is optimal to land at the tip of this spinning axis. In table 3.4, the pure spin results for an oblate ellipsoid are the opposite of the results of a prolate ellipsoid which is to be expected given the fact that the main spin axis is the major axis for an oblate ellipsoid and the minor axis for a prolate ellipsoid and that in a symmetric ellipsoid the major and minor axes are perpendicular to each other. This validates in part the model defined in section 3.4.4. For other rotational states, it is difficult to infer the cause of the

Chapter 3: Dynamic Model of a Rotating Ellipsoid with Relatively Rotating Additional Surface Masses and Landing Site Selection

optimality of the given landing location. The current hypothesis assumes that the optimal landing site corresponds to the tip of the spinning axis under the current rotational state and that an interpolation model could be fairly accurate. However, the validation of such an hypothesis is left for further work as the choice of landing site is not the main focus this PhD study.

Asymmetric ellipsoids results follow those of the prolate symmetric case. This is consistent with the fact that their minor axis is their Z axis just like for the prolate ellipsoids. The same hypothesis about the landing location on the tip of the current instantaneous spin axis is envisaged. Again, the investigation of this hypothesis it is left for future work.

Interestingly, the spherical angle ϕ in table 3.4 often takes the value 225°. The reason for this is unclear as this value appears for symmetric and asymmetric ellipsoids alike. The landing model used is very idealised and a more accurate model is to be designed to make a meaningful investigation into this fact.

The magnitude of the angular velocity error at landing will now be analysed.



Figure 3.6: Sphere Landing Normalised Error.

The magnitude of the relative angular velocity error for spheres is extremely low

and is well below $2 \cdot 10^{-15}$ %. For each mass ratio, the error is constant which is consistent with the indifference in landing site location observed in table 3.4. From these results, only the mass ratio affects the magnitude of the angular velocity error at landing. The relative angular velocity error can be completely neglected for mass ratios equal to or higher than 1,000.



Figure 3.7: Prolate Ellipsoid Landing Normalised Error.



Figure 3.8: Prolate Ellipsoid Landing Normalised Error.



Figure 3.9: Prolate Ellipsoid Landing Normalised Error.

For prolate ellipsoids, the magnitude of the relative angular velocity error at

Chapter 3: Dynamic Model of a Rotating Ellipsoid with Relatively Rotating Additional Surface Masses and Landing Site Selection

landing is nil when the objects spins about its major or minor axis which is consistent with the optimal location of landing. The magnitude of the relative angular velocity error at landing increases with prolateness but the spread of error values increases with prolateness. For a given prolateness, the mass ratio is the main parameter affecting the error's magnitude which decreases with increasing value of the mass ratio. In the cases presented here, the error reaches a maximum of 4% which occurs when the axis of rotation is the furthest from the principal axes. This is consistent with the fact that in the symmetric case the principal axes are stable axes of rotation. Moreover, this relative error is very low and as a first approximation, the initial rotation state of the object prior to deployment could be considered the same as before landing. In any case, the relative angular velocity error can be completely neglected for mass ratios equal to or higher than 1,000.



Figure 3.10: Oblate Ellipsoid Landing Normalised Error.


Figure 3.11: Oblate Ellipsoid Landing Normalised Error.



Figure 3.12: Oblate Ellipsoid Landing Normalised Error.

For oblate ellipsoids, the magnitude of the relative angular velocity error at

landing behaves similarly to the prolate case. It is nil when the objects spins about its major or minor axis which is consistent with the optimal location of landing. The magnitude of the relative angular velocity error at landing decreases with oblateness but the spread of error values increases with oblateness. For a given oblateness, the mass ratio is the main parameter affecting the error's magnitude which decreases with increasing value of the mass ratio. In the cases presented here, the error reaches a maximum at 1.5% which occurs when the axis of rotation is the furthest from the principal axes, despite the inconsistent and unexplained outlier found in figure 3.12 at $\frac{I_{min}+I_{max}}{2}$. This is consistent with the fact that in the symmetric case the principal axes are stable axes of rotation. Moreover, this relative error is very low and as a first approximation, the initial rotation state of the object prior to deployment could be considered the same as before landing. In any case, the relative angular velocity error can be completely neglected for mass ratios equal to or higher than 1,000.



Figure 3.13: Asymmetric Ellipsoid Landing Normalised Error.



Figure 3.14: Asymmetric Ellipsoid Landing Normalised Error.



Figure 3.15: Asymmetric Ellipsoid Landing Normalised Error.

For asymmetric ellipsoids, the magnitude of the relative angular velocity error

Chapter 3: Dynamic Model of a Rotating Ellipsoid with Relatively Rotating Additional Surface Masses and Landing Site Selection

at landing is nil when the object spins about its minor axis which is consistent with the optimal location of landing and is similar to the prolate and oblate cases in this respect. The magnitude of the relative angular velocity error at landing increases with the degree of symmetry of the object and again, the mass ratio is the main parameter affecting its magnitude which decreases with increasing value of the mass ratio. In the cases presented here, the maximum relative angular velocity error varies between 0.4% for the less symmetrical object and 3.5% for the most symmetrical object and occurs when the axis of rotation is the furthest from the principal axes. This is consistent with the fact that it is easier to spin about the minor axis and that stability increases with angular velocity. Moreover, this relative error is very low and as a first approximation, the initial rotation state of the object prior to deployment could be considered the same as before landing. In any case, the relative angular velocity error can be completely neglected for mass ratios equal to or higher than 1,000.

3.4.6.3 Landing Location: Conclusion

In this conclusion, the landing location analysis will be summarised as a set of design recommendations for the mission.

- 1. For all ellipsoidal objects, the optimal landing site is located at the tip of the current rotation axis. This location is independent of the mass ratio of the object's mass to the robot's mass but depends on the initial rotation state of the object prescribed by the parameter $\frac{\hbar^2}{2T}$. If the object's axis of rotation is stable, the identification of the location of its tip should be attempted prior to landing.
- 2. The magnitude of the disturbing effect of the landing on the rotational state of the object increases with symmetry, oblateness and prolateness.
- 3. In general, the magnitude of the disturbing effect of the landing is very low, of the order of 1% to 5% at worst, and decreases with increasing values of the mass ratio between the object and the robot. For mass ratios equal to or higher

than 1,000, this effect can be completely neglected. Therefore, under this prior condition, landing at any point of the surface of the object can be considered as both feasible and acceptable from the standpoint of not disturbing the rotational state of the object.

3.5 Ellipsoid's and Robot's Simulation Set Up

3.5.1 Parametrization

3.5.1.1 Normalised parameters

Whenever possible, a dimensionless analysis was sought in order to delve into the interrelationships and relative strengths of the system's fundamental properties and avoid explicit references to units or magnitude of the quantities involved.

The problem under study was mainly concerned with the shape of the system, its mass distribution, the relative magnitude of the mass of the object and the robot and the absolute and relative angular velocities of the system and its components. The primary normalised parameters were:

• The norm of the initial angular velocity which was taken to be equal to 1 and normalised per unit of angular velocity with respect to the angular velocity of a common spherical asteroid, the sizing of which is presented in the following section.

$$\omega_0 = 1 \tag{3.46}$$

• The mass ratio between the object and the robot so that the influence of mass is measured per unit of robot mass.

$$MR = \frac{m_{Object}}{m_{Robot}} \tag{3.47}$$

• The dimensions of the ellipsoid are given by its semi-axes lengths a for the X axis, b for the Y, and c for the Z axis. In this study, the conventional choice for the main rotating or spinning axis is Z. All length units were expressed

per unit of Z semi-axis length c.

From the normalisation of these primary parameters, other key parameters could be normalised. These were:

• The moments of inertia of the ellipsoid were normalised per unit of Z semi-axis length and per unit of robot mass as per the following formulae:

$$I_x = \frac{1}{5}MR(b_{normalised}^2 + 1) \tag{3.48}$$

$$I_y = \frac{1}{5}MR(a_{normalised}^2 + 1) \tag{3.49}$$

$$I_z = \frac{1}{5}MR(a_{normalised}^2 + b_{normalised}^2)$$
(3.50)

• The rotational kinetic energy parametrising the angular velocity of asymmetric bodies was also normalised per unit of Z semi-axis length and per unit of robot mass as consequence of normalising lengths and mass as per the following formulae:

$$T = \frac{MR\omega_{normanised}^2}{5} = \frac{MR}{5} \tag{3.51}$$

• The ratio of the square of the angular momentum to the kinetic energy $\frac{L^2}{2T}$ or $\frac{\hbar^2}{2T}$ parametrising the angular velocity of asymmetric bodies was also normalised per unit of Z semi-axis length and per unit of robot mass as a consequence of normalising lengths and mass. The following dimensional analysis of the ratio shows how the normalisation is propagated:

$$\frac{h^2}{2T} \equiv \frac{[M]^2 [L]^4 [T]^{-2}}{[M] [L]^2 [T]^{-2}} = [M] [L]^2$$
(3.52)

3.5.1.2 Angular Velocity and Kinetic Energy Sizing

In order to have a reference point and obtain simulation results with realistic physical values, the sizing of the angular velocity and rotational kinetic energy of a common spherical asteroid was undertaken with all data taken from [34] pages 39 and 116:

Assuming the asteroid of interest is perfectly spherical and of the most common stony-irons type, the mid-range density of such asteroids is $\rho = 5500 \frac{kg}{m^3}$. Moreover, the majority of NEAs are less than 100m in diameter and have rotation periods of a few minutes. Assuming, for this study, that the asteroid had a rotation period of 5min and a radius r = 100m, the following angular velocity and rotational kinetic energy values were derived:

- The norm of the angular velocity vector was $\omega_0 = \frac{2\pi}{5*60} = 0.02094 rad.s^{-1}$
- The asteroid's mass was $m = \rho \frac{4}{3}\pi r^3 = 2.30e + 10kg$
- Its rotational kinetic energy was $T = \frac{1}{2}I\omega_0^2 = \frac{1}{2}\frac{2}{5}mr^2\omega_0^2 = 20.21GJ$

3.5.2 Size of Integration Steps

In this section, the simulation integration steps are sized to appropriately capture the dynamics of the system [10].

In order to do so, the Nyquist Sampling theorem, which is usually applied in signal processing, is used to provide guidelines for finding the most appropriate size of the integration of step.

In [57] page 11, Shannon's version of the Nyquist Sampling theorem states:

"THEOREM 1: If a function f(t) contains no frequencies higher than W cycles per seconds, it is completely determined by giving its ordinates at a series of points spaced 1/(2W) seconds apart."

As per the review of the robot and asteroid environment in section 2.2.5 and the model of the physical interactions between the robot and the asteroid presented in section 3.2.4.1, the fastest dynamic states are the angular velocity of the asteroid and the deployment velocity of the robot at its surface. Moreover, both these states and the perturbations and noise affecting them have a finite number of frequencies and are of bounded bandwidth.

According to the Nyquist-Shannon theorem, the maximum sampling period is the

duration of the fastest half cycle among all the system's frequencies i.e. the smallest period which can be defined as the sampling period T_s . Therefore, the sampling period T_s was chosen to be equal to half the revolution period of the asteroid as its angular velocity is a priori the fastest dynamic state of all.

Time units were then normalised by the sampling period T_s to ensure that the iteration time interval was proportional to the Nyquist-Shannon sampling rate and captured the dynamics of the system appropriately. Each time unit interval dt was normalised to become a dimensionless time interval $d\tau = \frac{dt}{T_s}$. It was found experimentally that normalising a time interval dt = 0.1s to a dimensionless time interval $d\tau = \frac{0.1}{T_s}$ was suitable to simulate the robot-asteroid system.

For later results, the time will be displayed as seconds but these seconds are normalised time units.

3.5.3 Reference Angular Velocity of a Torque-Free Rotating Ellipsoid

As will be seen in ??, the engineering objectives for the robot require the explicit knowledge of the angular velocity $\overrightarrow{\Omega}$ or state of the rotating object when no external moments are exerted on it. The purpose for it is to have its time-domain state prior to any contact or interaction and to have a reference state trajectory to for the robot controller to track.

An analytical solution for this angular velocity can be obtained by solving as set of differential equations called the Euler equations with zero-moment. The derivation of these equations can be found in [64] pages 111-113.

Two cases have to be distinguished when solving the Euler equations.

3.5.3.1 Symmetric Bodies

The first is for symmetrical bodies i.e. when two moments of inertia are equal for instance when $I_x = I_y$. The body is then in constant spin and can be subject to precession and nutation. The solution can be found in [64] pages 113-116. This solution has been normalised as per 3.5.1.1 for the purpose of this study. The normalisation implies that the angular velocity $\vec{\Omega} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$ is a unit vector with $\omega_x^2 + \omega_y^2 + \omega_z^2 = 1$ and $|\omega_z| = n \le 1$ with *n* constant.

Defining λ as:

$$\lambda = n \left(\frac{I_z - I_x}{I_x} \right) \tag{3.53}$$

The components of the angular velocity and their time derivatives are:

$$\dot{\omega_x} = -\lambda\sqrt{1 - n^2}\sin(\lambda t)$$

$$\dot{\omega_y} = \lambda\sqrt{1 - n^2}\cos(\lambda t)$$

$$\dot{\omega_z} = 0$$

(3.54)

$$\omega_x = \sqrt{1 - n^2} \cos(\lambda t)$$

$$\omega_y = \sqrt{1 - n^2} \sin(\lambda t)$$

$$\omega_z = n$$
(3.55)

3.5.3.2 Asymmetric Bodies

The second case is for asymmetrical bodies i.e. when the moments of inertia are unequal for instance when $I_x < I_y < I_z$. The body is then in a tumbling motion. Although a partial solution can be found in the literature and in particular in [64] pages 126-130, a complete solution could not be found. A complete and explicit derivation was performed for this study and is to be read in appendix E. This solution has been normalised as per 3.5.1.1 for the purpose of this study.

 $\forall (t, k^2) \in \mathbb{R} \times [0, +\infty[$, with moments of inertia $I_{min} < I_{mid} < I_{max}$, magnitude

of angular momentum h and rotational kinetic energy T:

$$\dot{\omega}_{mid} = -N\sqrt{\frac{h^2 - 2TI_{min}}{I_{mid}(I_{mid} - I_{min})}}} cn(Nt, k^2) dn(Nt, k^2)$$
$$\dot{\omega}_{max} = -N\sqrt{\frac{h^2 - 2TI_{min}}{I_{max}(I_{xmax} - I_{min})}}} sn(Nt, k^2) dn(Nt, k^2)$$
$$(3.56)$$
$$\dot{\omega}_{min} = -Nk^2 \sqrt{\frac{2TI_{max} - h^2}{I_{min}(I_{max} - I_{min})}}} sn(Nt, k^2) cn(Nt, k^2)$$

$$\omega_{mid} = -\sqrt{\frac{h^2 - 2TI_{min}}{I_{mid}(I_{mid} - I_{min})}}} sn(Nt, k^2)$$

$$\omega_{max} = \sqrt{\frac{h^2 - 2TI_{min}}{I_{max}(I_{xmax} - I_{min})}}} cn(Nt, k^2)$$

$$\omega_{min} = \sqrt{\frac{2TI_{max} - h^2}{I_{min}(I_{max} - I_{min})}} dn(Nt, k^2)$$
(3.57)

where:

$$k = \sqrt{\left(\frac{I_{max} - I_{mid}}{I_{mid} - I_{min}}\right) \left(\frac{h^2 - 2TI_{min}}{2TI_{max} - h^2}\right)}$$
(3.58)

$$N = \sqrt{\frac{(I_{mid} - I_{min})(2TI_{max} - h^2)}{I_{max}I_{mid}I_{min}}}$$
(3.59)

sn(x,m) is the sine amplitude elliptic function. cn(x,m) is the cosine amplitude elliptic function. dn(x,m) is the Jacobi elliptic dn function.

3.5.4 Initial Condition

The initial condition refers to the rotational state of the object prior to deployment of the robot and specifically to its angular velocity which will become a tracked trajectory by the robot controller. For each object shape, there is one parameter that will set the the initial angular velocity. The value of these parameters will be chosen at the onset of each simulation.

• <u>Sphere</u>: in this case it is simply the normalised angular velocity $\omega_0 = 1$ about the original Z body axis of the object.

- Symmetric Ellipsoid: it is the constant parameter n with $n = |\omega_z| \le 1$.
- <u>Asymmetric Ellipsoid</u>: it is the parameter $\frac{h^2}{2T}$.

3.5.5 ODE First Order Integration

In order to propagate the state of the system, a simple first order ode integration was used. In practice, for general non-linear system with state space representation at instant t:

$$\dot{X} = f(X, t) \tag{3.60}$$

this took, over the given fixed sampling rate dt, the discrete programmatic form:

$$X_{t+1} = X_t + dt * f(X_t, t)$$
(3.61)

3.5.6 Perturbations and Noise Modelling

The physical model presented in section 3.2 represents an idealised description of the rotation state and interactions of the robot-asteroid system which neglects two important sources of errors.

The first source of errors are the system parameters. These parameters are specific, fixed or time-dependent quantities characterising the system's properties and whose evaluation is subject to measurement errors. In the case of model 3.23, the most important model parameters are the various mass and moment of inertia quantities which affect the rotational state of the system. The rotational state is measured by the robot gyroscope and therefore the effect of these parameters errors can be picked up by the gyroscope as a measurement bias. Moreover, in section 2.2.5, it was assumed that actuators and sensors noise could be aggregated in the noise experienced by the robot gyroscope. In [33], the gyroscope's error is described as being the result of biases and random noise which can be modelled as a biased white noise and include the model parameters errors as a bias.

The second source of errors are the assumptions made to derive the model. In the case of model 3.23, the robot-asteroid system is assumed to be isolated. However,

as per section 2.2.5, the system's environment is perturbed. The perturbations magnitude and timescale are such that over the time span of the mission these perturbations can be considered as a coloured noise from the infrared spectrum.

In this section the models of the perturbations and sensor and actuator noises is presented.

3.5.6.1 Gyroscope Noise Model

The gyroscope model used in this study was taken from the classic gyroscope model found in [33]:

$$\overrightarrow{\omega}_{gyro} = \overrightarrow{\omega}_{true} + \overrightarrow{\mathbf{b}} + \overrightarrow{\mathbf{n}}$$
(3.62)

where $\vec{\omega}_{true}$ is the true angular velocity, $\vec{\mathbf{b}}$ is the gyroscope drift rate bias, driven by the angular velocity random walk process and $\vec{\mathbf{n}}$ is a white noise affecting the measurements.

In the table below are found the magnitudes of the random walks instantaneous standard deviations for a typical space grade gyroscope. These values were used to simulate actuators and sensors noise and were taken from [33] for the rate and angular random walks and from [9] for the drift rate bias.

Process	Value	Units
Angular Random Walk	$3.35 \cdot 10^{-8}$	$rad \cdot s^{\frac{-1}{2}}$
Rate Random Walk	$8.08 \cdot 10^{-13}$	$rad \cdot s^{\frac{-3}{2}}$
Drift Rate Bias	$3.23\cdot 10^{-6}$	$rad \cdot s^{-1}$

Table 3.5: Bias and Noise Magnitudes

3.5.6.2 Modelling White Noise

Model The white noise was modelled as a Wiener Process, i.e. normalised and instantaneous 0-mean normally distributed process uncorrelated with itself from one

instant to the next with each components of $\overrightarrow{\mathbf{n}}$: $\begin{cases} n_i \sim N(0, \sigma_i) \\ n_j \sim N(0, \sigma_j) \\ n_k \sim N(0, \sigma_k) \end{cases}$

As per [49], the frequency and magnitude of these random normally distributed numbers are described by two parameters: their expected value μ and their standard deviation σ . In this study, frequency and magnitude were prescribed through the definition of the standard deviations only. The value of these instantaneous standard deviations were taken from table 3.5 and normalised in time unit to use the normalised integration time step $d\tau$ defined in section 3.5.2. In order to do so, the instantaneous standard deviation of the angular random walk was divided by $T_s^{\frac{1}{2}}$ and the rate random walk by $T_s^{\frac{3}{2}}$, where T_s is the sampling rate period defined in 3.5.2.

The main drawback of white noises is the fact that it is almost surely continuous everywhere but cannot be derived. Instantaneous abrupt changes in direction and abnormally large values can occur which is not realistic [49]. However, since any linear combinations of normally distributed random variables remains a normally distributed random variable [49] and since this white noise is propagated through the measurements of the gyroscopes, by assuming that the system is completely observable by a linear observer, which is the working hypothesis for the design of the robot controller developed in 4.3.9, the white noise model can be solely applied to the observer.

Normally Distributed Number Generation Pseudo-random normally distributed numbers were generated with the Lehmer's algorithm in combination with Box and Muller's method. This approach use the property of the inversion of the cumulative distribution function and gives reproducible results as the random values are given once and for all. For further details about the method, the reader is referred to [49] pages 371-372, 375 and 378.

3.5.6.3 Radiation Perturbation Noise Model

As seen in section 2.2.5, the YORP torque provides a constant but small rotational acceleration due to the infrared radiation from the Sun. For this feasibility study, the value of this rotational acceleration was set as a constant whose value was taken from [52] as the maximum value of all the coefficient presented in table 2 page 445: $\alpha_2 = 2 \cdot 10^{-16} rad \cdot s^{-2}$. This rotational acceleration value was arbitrarily taken so that the radiation perturbation had a realistic magnitude. Finally, in order to simulate the infrared noise, a 0-mean white noise was be added to the rotational acceleration but with a limited bandwidth bounded at one standard deviation.

The model is as follows:

$$\vec{\mathbf{M}}_{Radiation} = \begin{cases} \alpha_2 + \epsilon_i(t) \\ \alpha_2 + \epsilon_j(t) \\ \alpha_2 + \epsilon_k(t) \end{cases}$$
(3.63)

where $\overrightarrow{\mathbf{M}}_{Radiation}$ is the moment created by the radiation perturbation and $\epsilon_{i,j,k}(t) \sim N(0,\sigma)$. The standard deviation was arbitrarily chosen at $\sigma = 10^{-17} rad \cdot s^{-2}$.

It is not realistic to expect the magnitude of the radiation perturbation moment to have equal components. However, as tumbling asteroids have a dynamic axis of spin, the model presented here is a coarse but good enough approximation for a feasibility study.

3.6 Conclusion

In this chapter, a physical model of the interactions between the rotating object and the SR robot deploying on its surface has been presented. It considers the object and the robot as one rotating continuum with a rigid part, the object and a moving deformable part, the robot with point mass modules. The three frames of reference necessary for its derivation were presented and all the model's parameters normalised. The object was modelled as a uniformly distributed ellipsoid whose initial angular velocity were the solutions of the Euler equations. Initial conditions and landing sites were also sized. Finally errors and disturbances were modelled both as coloured and biased white noises. In the next chapter, the model will be used for deriving the robot's modules low-level model-based controller.

Chapter 4

Robot Design and Control

In this chapter, the engineering application is defined along with all the necessary assumptions. The goal of this chapter is the design a SR robot and specifically its controller in order to simulate and evaluate the feasibility of the deployment of a modular robotic structure at the surface of an uncooperative tumbling object in space. This will encompass all the physical hypotheses, the precise definition of the task, the engineering requirements and the design of the controller to be simulated.

4.1 Engineering Application: Definition Requirements and Assumptions

The engineering application for this PhD study is formally defined as the autonomous construction and deployment of decentralised modular scaffolding structures around free-floating randomly tumbling objects in space. The study is narrowed down to the simulation of a modular decentralised SR robot controller in order to assess the possibility and feasibility of such constructions.

4.1.1 Robot Design and Assumed Properties

The SR robot is assumed to be made out of identical autonomous spherical modules capable of moving on top of one another while maintaining connection at all times. This structure was chosen to enable the entire SR robot to deploy itself as continuous structures at the surface of the object while maintaining contact at all time with this surface. To this end, the design assumptions are the following:

- 1. Each module possesses an autonomous controller capable of using the module's state, its direct neighbours' and the object's, to make decisions about what actions to undertake given its state and its immediate neighbours' and to send commands to its actuators. States information can be transmitted and received between modules.
- 2. The communication architecture is local, each module can communicate with its direct neighbours. Information can be passed down a chain of modules.
- 3. Each module possesses the necessary actuators to move on the object's surface, on top of the other modules, help other module to move, anchor itself to the surface and detach itself from the surface or other modules as the situation requires. Moreover, each module possesses actuators mounted with encoders to estimate the robot's odometry like a simple mobile robot.
- 4. Each module possesses sensor capabilities to retrieve its rotational state, the rotational state of the object and its linear velocity and acceleration at the surface at any given time. To this end, it possesses a gyroscope measuring its own absolute angular velocity and an accelerometer giving linear acceleration data.
- 5. Each module possesses independent internal energy storage but can also share its energy resource with other modules if required.
- 6. Each module is a sphere that can be considered as a point mass with respect to the object the robot is deploying on.

And the operating assumptions are:

- 1. Each module goes one after the other and one by one over the chain laid by preceding modules.
- 2. On the surface each module operates like a mobile robot.

- 3. If a module is faulty, it can be detected, released and replaced in the limit of number availability.
- 4. When a module reaches its destination, it anchors itself to the surface and retains all levels of connectivity (attachment and communication) with the preceding module.

In the spirit of this feasibility study, the above requirements were left very broad and, although the general design is feasible, the current state of the art of the hardware makes its actual realisation unrealistic for now. This is however good enough for keeping the focus on the physics of the problem and the development of an initial module controller and robot control algorithm.

The inclusion of a proper fault detection mechanism was avoided as it would have introduced a level of complexity which would detract from the main objective of this feasibility study. It was therefore left out of its scope and, although taken into account in the design of the robot, it was assumed that the system was free from faults and malfunction.

4.1.2 Task Description Goals and Objectives

The task under consideration begins after the SR robot's landing and anchoring on the surface of a free-floating randomly tumbling object. It consists of the robot autonomous deployment from an initial lattice configuration to a chain configuration coiling around the object and maintaining contact with its surface at all times. The overall task goals are the following:

- Deployment of the robot as a chain of modules circling half a revolution around the surface of the object from the tip of its main rotational axis to the median plane.
- 2. Stability maintenance and minimisation of the disturbance created by the robot deployment.
- 3. Convergence of the object's rotational state to a prescribed state.

This translates into the following specific objectives:

- 1. Deployment of a n_r -modules robot should result in a final chain configuration with each module *i* anchored at position $(\theta_i, \phi_i) = (i \frac{\pi}{2(n_r-1)}, i \frac{\pi}{n_r-1})_{i \in [0, n_r-1]}$ in spherical coordinates with respect to the body frame [O'].
- 2. Stabilisation and disturbance minimisation via optimisation of a cost function.
- 3. Tracking a time-varying reference trajectory $\overrightarrow{\mathbf{X}}_{ref}$ which contains the object's rotation state objectives as well as the target anchoring positions for each modules.

4.1.3 Stability Definition

The task defined in section 4.1.2 is underpinned by the solution to an optimisation, tracking and stabilisation problem. Stability, in particular, has to be understood in the perspective of the general handling of the object which, for instance, should be maintained in a constant pointing direction during retrieval operations. Resistance to degeneration into a tumbling state is therefore paramount as, as per [60], the more an object spins, the more stable the direction to which its spinning axis points. Hence, the deployment of the robot should leave the object's angular rotating state at least neutrally stable. Ideally, the robot deployment would render any tracked reference object's rotational state asymptotically stable and in particular pure spin.

This section will address the definition of stability which would guide the construction of a suitable module controller to perform the task defined in section 4.1.2. As mentioned, this revolves primarily around the object's angular rotational state and the classic Liapunov stability which can be used in the design of model-based controllers.

Stability: This classic definitions of stability can be found in [68] page 7. There are reproduced here for clarity:

Considering a general autonomous vector field (i.e. not time-dependent) $\forall x \in \mathbb{R}^n$

$$\dot{x} = f(x) \tag{4.1}$$

Let x(t) be any solution of 4.1.

Liapunov Stability: x(t) is said to be stable (or Liapunov stable) if, given $\epsilon > 0$ $\exists \ \delta(\epsilon) > 0$ such that, $\forall \ y(t)$ solution of 4.1 satisfying $|x(t_0) - y(t_0)| < \delta(\epsilon)$ (where |.| is a norm on \mathbb{R}^n), then $\forall t > t_0, t_0 \in \mathbb{R} |x(t) - y(t)| < \epsilon$.

<u>Asymptotic Stability</u>: x(t) is said to be asymptotically stable if it is Liapunov stable and $\forall y(t)$ solution of 4.1, $\exists b > 0$ constant such that, if $|x(t_0) - y(t_0)| < b$, then $\lim_{t \to +\infty} |x(t) - y(t)| = 0$.

A solution which is not stable is said to be unstable.

The Liapunov definition of stability is the stability criterion chosen for this study. In chapter 3, a physical model of the interaction between a rigid body and an object moving on its surface was laid out. From this model, a model based controller for the SR robot will be derived in section 4.2. From the linearisation of this model and the optimisation quadratic cost function, the Liapunov stability will be checked. But first the system under study and its plant model need to be defined along with all its physical assumptions. This is the purpose of the section 4.1.5. In the next section, the measures used to assess the SR robot performance are defined as these were fed into the system's plant modelling.

4.1.4 Measures for the Robot Task Performance

In order to assess whether the objectives listed in section 4.1.2 were met as well as have empirical stability measures should stability be difficult to prove formally, the following performance measures were further specified:

- 1. Achievement of task:
 - (a) Did a single module manage to go from point (θ, φ) = (0,0) to point (θ, φ) = (π/2, π) on the surface of the ellipsoid following a continuous trajectory: True or False
 - (b) Did the n_r -module robot achieve full deployment with each module *i* being laid continuously one after the other from point $(\theta, \phi) = (0, 0)$ to

its target anchoring point at $(\theta_i, \phi_i) = (i \frac{\pi}{2(n_r-1)}, i \frac{\pi}{n_r-1})_{i \in [0, n_r-1]}$ on the surface of the ellipsoid: True or False

- 2. Stability and controllability of the task:
 - (a) Measurement of the nutation angle $\nu = \arctan\left(\frac{\sqrt{[\mathbf{I}_X]^2 \omega_X^2 + [\mathbf{I}_Y]^2 \omega_Y^2}}{|[\mathbf{I}_Z] \omega_Z|}\right)$ according to [60] page 98. If ν increases, the rotational state of the object is less stable.
 - (b) Measurement of the error vector $\overrightarrow{\varepsilon}_t$ such that $\overrightarrow{\Omega}(t) = \overrightarrow{\Omega}_{ref}(t) + \overrightarrow{\varepsilon}_t$ as per chapter 3 section 3.4.5. $\overrightarrow{\varepsilon}_t$ should ideally converge and stay as close to $\overrightarrow{\mathbf{0}}$ as possible. Empirical measurements will include the phase space orbits of the object's rotational state. In particular, it will be examined whether:
 - i. $\overrightarrow{\varepsilon}_t$ is increasing, decreasing or periodic respectively indicating instability, asymptotic stability or neutral stability.
 - ii. The controlled system can converge to any reference state in particular whether the object can converge to a state of pure spin or despin.
 - iii. The model parameters influence the stability and controllability and if so in which proportion.
 - (c) Measurement of the rigid or body rotational kinetic energy: $\frac{1}{2} \vec{\Omega}_t^{T^T} [\mathbf{I}_{\mathbf{Object}}] \vec{\Omega}_t^{\mathbf{1}}$ to evaluate the state and stability of the object in conjunction with the nutation angle. Energy dissipation combined with increased nutation angle is indicative of nutation instability and of a state degenerating towards rotation about the major axis [60].
- 3. Duration of the task:
 - (a) Comparison between the timescale of the controller and of the object's rotational dynamics. Is the controller timescale appropriate: True or False
 - (b) Measurement of the controller timescale for each controller type to evaluate which is the fastest.

(c) Measurement of the controller timescale against the model parameters to evaluate their respective influence on this timescale

The next section will tackle all the modelling assumptions for the controller design.

4.1.5 Low-Level Module Controller Derivation Hypotheses

The low-level module controller was based on an explicit plant model derived from the discrete model 3.24 in chapter 3. This section lists the various hypotheses behind this derivation and discusses some of its limitations.

4.1.5.1 System Definition

<u>The system</u> that the controller's plant is describing is defined as the combination of a free-floating object and one or two modular SR robots of the type designed in this chapter.

4.1.5.2 Physical Hypotheses

The physical hypotheses underpinning the plant model derivation are:

- 1. All frames of reference which are used are defined in section 3.2.3.
- 2. The system is isolated and only in pure rotation with respect to the inertial frame of reference.
- 3. The system is subject to disturbances defined in chapter 3 section 3.5.6 as gyroscope noise and radiation perturbations. The gyroscope is disturbed by a drift bias and random walk as per model (3.62) and the radiation perturbations produce an extra moment as per model (3.63). These disturbances are realistic in their timescale and magnitude.
- 4. Friction and stress forces and moments arising from the relative motion of each robot module with respect to the object are assumed to be overcomable and each module is able to move freely at the surface of the object.

5. Each SR robot module is in contact with the object surface at all times creating an holonomic liaison.

4.1.5.3 Hypotheses for the Object

The specific hypotheses concerning the object are the following:

- The object is rigid. As per [67], asteroids with a diameter over 100 meters can be conglomerations of smaller pieces loosely kept together by gravitation. The rigidity hypothesis is only valid below a limit rotation rate which is not considered in this study.
- 2. The object's mass is uniformly distributed. As per [34], this is unrealistic but allows for the centre of mass of the object to remain at the same location in space as the origins of the frames of reference. This simplifies the calculations and computations required without loss of generality with respect to the capture of the dynamics of the system.
- 3. The object is assumed to be an ellipsoid. As per [34], in many cases this could be considered as a good approximation of the general aspect of the object's shape. The main advantages of such a model are:
 - (a) The explicit calculation of the holonomic liaison between the object and each robot module.
 - (b) The object is symmetric with respect to the centre of its volume which again allows for the centre of mass of the object to remain at the same location in space as the origins of the frames of reference.

In general however, the surface of an asteroid is uneven and likely to be deformable. Nonetheless, as the moments of inertia are the only parameters required to characterise the object's mass distribution, the SR robot could deploy in such a way as to form an ellipsoidal structure around the object and still fulfil the ellipsoidal hypothesis without changing the validity of the model.

4.1.5.4 Landing and Set Up Prior to Deployment

For the robot landing and set up prior to deployment, it was assumed that:

- 1. Two separate and identical SR robots land synchronously at symmetric location with respect to the object's centre of mass (the system is now composed of the object and two robots). This hypothesis is extremely difficult to fulfil in reality and was made only to preserve the symmetry of the overall system, allowing for the centre of mass of the object to remain at the same location in space as the origins of the frames of reference while avoiding the loss of generality with respect to the capture of the dynamics of the system, i.e. all of the interactions between the object and the robots could be modelled from the point of view of one robot only.
- 2. The landings occur at both tips of the object Z axis. As per 3.4.6.3, the landing site should be chosen close to the current rotation axis but can be chosen anywhere with negligible disturbing effects on the rotational state of the object. In this study, the Z axis is the reference rotation axis and the direction of the robot deployment was chosen to start from the tip of the Z axis all the way to the plane containing the object's centre of mass. The initial conditions of the simulations were chosen to assess any possible initial condition and therefore these initial conditions reflect and contain any disturbing effect from landing.

4.1.5.5 Deployment Strategy

The deployment strategy was designed on two principles:

- The prescription of a final configuration for the robot taking the form of a chain configuration starting at the tip of the Z axis, coiling around the Z axis and ending at the plane containing the object's centre of mass. This general requirement being achieved thanks to:
- 2. A decentralised optimisation based on a cost function: each module moving in turns and one after the other, first following previous modules' paths and then

tracking a reference trajectory on the surface of the object to anchor itself at its given target location.

4.1.5.6 Reference Trajectories

A reference trajectory for the system defined in section 4.1.5.1 is a desired final state of the system. It corresponds to a deployed robot rigidly attached to the object and a system in torque-free rotational motion. This will be formally defined in section 4.2.4 with the model states. The system final rotational motion state will be described by the solutions of the Euler equations which are available in sections 3.5.3.1 and 3.5.3.2.

4.1.5.7 Model Parameters

All model parameters were normalised as per section 3.5.1.1 in chapter 3.

4.1.5.8 Deployment Illustrations

In figures 4.1 and 4.2 below are two illustrations of the system in operation. In figure 4.1, the two robots have landed at the tip of the Z axis and are ready to be deployed. In figure 4.2 the two robots have finished their deployments and are coiled around the object about the Z axis.



Figure 4.1: Robots at Landing Sites (the Two Tips of the Spin Axis) in Lattice Form.



Figure 4.2: Symmetrical Deployment of the Two Robots from Both Tips of the Spin Axis.

4.2 Low-Level Module Controller's Plant State Space Modelling

In this section, model 3.24 is rewritten as a state space model for the controller design. n_r is defined as the number of robots modules which are identified by their index number *i*. The choice of commands as angular accelerations is justified by the fact that on the surface each module can be viewed as mobile robot whose ground speed and acceleration can be translated into angular velocity and acceleration given the dimension and shape of the ellipsoidal object. The frame of reference of choice is the body frame rigidly attached to the object which uses the object's principal axes as frame axes. In this frame, all vectors are expressed in spherical coordinates.

4.2.1 State Definition

The state $\overrightarrow{\mathbf{X}}$ is defined as:

$$\vec{\mathbf{X}} = \begin{bmatrix} \vec{\Omega} \\ \vec{\Theta} \\ \dot{\vec{\Theta}} \end{bmatrix}$$
(4.2)

where:

1. $\overrightarrow{\Omega}$ is the rigid angular velocity of the model which is also the angular velocity of the object.

2. $\overrightarrow{\Theta}$ groups the spherical coordinates $\overrightarrow{\Theta}_{i} = \begin{bmatrix} \theta_{i} \\ \phi_{i} \end{bmatrix}$ in body frame of all the robot's modules: $\overrightarrow{\Theta} = \begin{bmatrix} \theta_{1} \\ \phi_{1} \\ \vdots \\ \theta_{n_{r}} \end{bmatrix}$ (4.3)

3.
$$\dot{\overrightarrow{\Theta}}$$
 groups the non-rigid, relative angular velocities $\dot{\overrightarrow{\Theta}}_i = \begin{bmatrix} \dot{\theta}_i \\ \dot{\phi}_i \end{bmatrix}$ of all the robot's

modules:

$$\vec{\Theta} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\phi}_1 \\ \vdots \\ \dot{\theta}_{n_r} \\ \dot{\phi}_{n_r} \end{bmatrix}$$
(4.4)

4.2.2 Relative Angular Velocity Definition

The non-rigid or individual modules angular velocities and accelerations relative to the object are:

1.
$$\vec{\Psi}_{i} = \begin{bmatrix} -\sin(\phi_{i})\dot{\theta}_{i} \\ \cos(\phi_{i})\dot{\theta}_{i} \\ \dot{\phi}_{i} \end{bmatrix}$$
 the relative angular velocity of module *i*.
 $\dot{\phi}_{i}$
2. $\vec{\Psi}_{i} = \begin{bmatrix} -\sin(\phi_{i})\ddot{\theta}_{i} - \cos(\phi_{i})\dot{\phi}_{i}\dot{\theta}_{i} \\ \cos(\phi_{i})\ddot{\theta}_{i} - \sin(\phi_{i})\dot{\phi}_{i}\dot{\theta}_{i} \\ \ddot{\phi}_{i} \end{bmatrix}$ relative angular acceleration of module *i*.

4.2.3 Control Commands Definition

The control command vector $\overrightarrow{\mathbf{u}}$ which groups all the robot's modules control commands $\overrightarrow{\mathbf{u}}_i = \overrightarrow{\mathbf{\Theta}}_i = \begin{bmatrix} \ddot{\theta}_i \\ \ddot{\phi}_i \end{bmatrix}_{i \in [0, n_r]}$ is:

$$\vec{\mathbf{u}} = \vec{\Theta} = \begin{bmatrix} \vec{\theta}_1 \\ \vdots \\ \vec{\theta}_n \\ \vdots \\ \theta_{n_r} \\ \vdots \\ \vec{\theta}_{n_r} \end{bmatrix}$$
(4.5)

4.2.4 Reference State Trajectory and Reference Control Command

In this section the reference trajectory and corresponding control command is further defined in relation to the low-level controller's objectives.

The reference trajectory corresponds to the object-robot system in its final state, i.e. when it is in a rigid torque-free rotation with all the robot modules deployed and attached rigidly to the surface of the object at their respective target location. $\Gamma \rightarrow 1$

Let
$$\overrightarrow{\mathbf{X}}_{ref} = \begin{bmatrix} \overrightarrow{\mathbf{\Omega}}_{ref} \\ \overrightarrow{\mathbf{\Theta}}_{ref} \\ \vdots \\ \overrightarrow{\mathbf{\Theta}}_{ref} \end{bmatrix}$$
 be the reference trajectory at all time ($\forall t \ge 0$), then:

1. $\vec{\Omega}_{ref}$ follows the Euler torque-free rigid rotation equation where $[\mathbf{I_n}]$ is the final normalised moment of inertia matrix of the whole object-robot system:

$$[\mathbf{I_n}] \cdot \overrightarrow{\mathbf{\Omega}}_{ref} + (\overrightarrow{\mathbf{\Omega}}_{ref} \wedge [\mathbf{I_n}] \cdot \overrightarrow{\mathbf{\Omega}}_{ref}) = \overrightarrow{\mathbf{0}}$$
(4.6)

2. $\overrightarrow{\Theta}_{ref}$ represents all the target anchoring locations for each of the robot's modules, $\forall t \ge 0$:

$$\vec{\Theta}_{ref} = \begin{bmatrix} \frac{\pi}{2(n_r-1)} \\ \frac{\pi}{n_r-1} \\ \vdots \\ \frac{\pi}{2} \\ \pi \end{bmatrix}_{2n_r \times 1}$$
(4.7)

3. $\dot{\Theta}_{ref}$ represents all the modules' relative velocities which are nil in the final state, $\forall t \ge 0$:

$$\dot{\overrightarrow{\Theta}}_{ref} = \overrightarrow{\mathbf{0}}_{2n_r \times 1} \tag{4.8}$$

4. $\vec{\mathbf{u}}_{ref}$ represents all the module's relative acceleration or control command which are nil in the final state, as per equation 4.13, $\forall t \ge 0$:

$$\overrightarrow{\mathbf{u}}_{ref} = \overrightarrow{\mathbf{0}}_{2n_r \times 1} \tag{4.9}$$

4.2.5 Nonlinear State Space Model

In this section, a nonlinear state space plant model for the module's controller will be derived from the discrete model (3.24). States will be shifted by a reference trajectory as defined in section 4.2.4 so as to obtain a non linear state space plant model centred at the origin (0,0) while making this origin state a time-invariant equilibrium.

Since $[\mathbf{I_n}]$ is the normalised moment of inertia over the entire object and robot system, it is always invertible. This allows the normalised model (3.24) to be put in the following form:

$$\dot{\overrightarrow{\Omega}} = f(\overrightarrow{\Omega}, \overrightarrow{\Theta}, \overrightarrow{\Theta}, \dot{\overrightarrow{\Theta}})$$
(4.10)

with

$$\begin{split} f(\overrightarrow{\Omega}, \overrightarrow{\Theta}, \overrightarrow{\Theta}, \overrightarrow{\Theta}) &= -[\mathbf{I}_{\mathbf{n}}]^{-1} \cdot (\overrightarrow{\Omega} \wedge [\mathbf{I}_{\mathbf{n}}] \cdot \overrightarrow{\Omega}) \\ &- [\mathbf{I}_{\mathbf{n}}]^{-1} \cdot \sum_{\mathbf{i}=1}^{\mathbf{n}_{\mathbf{r}}} \left[2 \left[(\overrightarrow{\mathbf{x}_{\mathbf{i}0}''} \cdot \overrightarrow{\mathbf{x}}_{\mathbf{i}}) [\mathbf{1}] - \overrightarrow{\mathbf{x}_{\mathbf{i}0}''} \otimes \overrightarrow{\mathbf{x}}_{\mathbf{i}} \right] \cdot \overrightarrow{\Omega} \\ &+ \left[\left[(\overrightarrow{\mathbf{x}}_{i} \cdot \overrightarrow{\mathbf{x}}_{i}) [\mathbf{1}] - (\overrightarrow{\mathbf{x}}_{i} \otimes \overrightarrow{\mathbf{x}}_{i}) \right] \cdot \overrightarrow{\Psi}_{\mathbf{i}} \\ &+ \left[2 (\overrightarrow{\mathbf{x}}_{i} \cdot \overrightarrow{\mathbf{x}}_{i}) [\mathbf{1}] - (\overrightarrow{\mathbf{x}}_{i} \otimes \overrightarrow{\mathbf{x}}_{i}) - (\overrightarrow{\mathbf{x}}_{i} \otimes \overrightarrow{\mathbf{x}}_{i}) \right] \cdot \overrightarrow{\Psi}_{\mathbf{i}} \right] \\ &+ \left[(\overrightarrow{\mathbf{x}}_{i} \otimes \overrightarrow{\mathbf{x}_{i0}'}) - (\overrightarrow{\mathbf{x}_{i0}''} \otimes \overrightarrow{\mathbf{x}}_{i}) \right] \cdot \overrightarrow{\Psi}_{i} \\ &+ \left[(\overrightarrow{\mathbf{x}}_{i} \otimes \overrightarrow{\mathbf{x}}_{i}) \wedge \overrightarrow{\Psi}_{i} + ((\overrightarrow{\mathbf{x}}_{i} \otimes \overrightarrow{\mathbf{x}}_{i}) \wedge \overrightarrow{\Psi}_{i})^{T} \right] \cdot \overrightarrow{\Omega} \\ &+ \left(\overrightarrow{\mathbf{x}}_{i} \wedge \overrightarrow{\mathbf{x}_{i0}''} \right) \right] \end{split}$$

Leading to a first state space model:

$$\begin{bmatrix} \overrightarrow{\widehat{\Omega}} \\ \overrightarrow{\widehat{\Theta}} \\ \overrightarrow{\widehat{\Theta}} \\ \overrightarrow{\widehat{\Theta}} \end{bmatrix} = g(\overrightarrow{\Omega}, \overrightarrow{\Theta}, \overrightarrow{\widehat{\Theta}}, \overrightarrow{\overrightarrow{\Omega}}) = \begin{bmatrix} f(\overrightarrow{\Omega}, \overrightarrow{\Theta}, \overrightarrow{\Theta}, \overrightarrow{\Theta}) \\ \overrightarrow{\widehat{\Theta}} \\ \overrightarrow{\widehat{\Theta}} \\ \overrightarrow{\mathbf{u}} \end{bmatrix}$$
(4.12)

which is of a non-linear first order model of the form:

$$\dot{\overrightarrow{\mathbf{X}}} = g(\overrightarrow{\mathbf{X}}, \overrightarrow{\mathbf{u}}) \tag{4.13}$$

Let $\overrightarrow{\mathbf{X}}_{ref} = \begin{bmatrix} \overrightarrow{\mathbf{\Omega}}_{ref} \\ \overrightarrow{\mathbf{\Theta}}_{ref} \\ \vdots \\ \overrightarrow{\mathbf{\Theta}}_{ref} \end{bmatrix}$ be an admissible reference trajectory and $\overrightarrow{\mathbf{u}}_{ref}$ its associated

admissible reference command.

Without changing the description of the dynamics of the system, the model (4.12)/(4.13) can be shifted by the value of an admissible reference trajectory. This leads to a second formulation of the state space model as:

$$h(\overrightarrow{\mathbf{X}}, \overrightarrow{\mathbf{u}}) = g(\overrightarrow{\mathbf{X}}, \overrightarrow{\mathbf{u}}) - g(\overrightarrow{\mathbf{X}}_{ref}, \overrightarrow{\mathbf{u}}_{ref})$$
(4.14)

Immediately, $h(\vec{\mathbf{X}}_{ref}, \vec{\mathbf{u}}_{ref}) = \vec{\mathbf{0}}$. *h* has an equilibrium state at $(\vec{\mathbf{X}}_{ref}, \vec{\mathbf{u}}_{ref})$. This equilibrium state can be further shifted at the origin of *h* by introducing the variable change $\vec{\mathbf{Z}} = \vec{\mathbf{X}} - \vec{\mathbf{X}}_{ref}$ and $\vec{\mathbf{u}}_{\mathbf{Z}} = \vec{\mathbf{u}} - \vec{\mathbf{u}}_{ref}$.

This lead to the final definition of the state space model:

$$\dot{\overrightarrow{\mathbf{Z}}} = h(\overrightarrow{\mathbf{Z}}, \overrightarrow{\mathbf{u}_{\mathbf{Z}}}) = g(\overrightarrow{\mathbf{Z}} + \overrightarrow{\mathbf{X}}_{ref}, \overrightarrow{\mathbf{u}_{\mathbf{Z}}} + \overrightarrow{\mathbf{u}}_{ref}) - g(\overrightarrow{\mathbf{X}}_{ref}, \overrightarrow{\mathbf{u}}_{ref})$$
(4.15)

 $\dot{\vec{\mathbf{Z}}} = h(\vec{\mathbf{Z}}, \vec{\mathbf{u}_{\mathbf{Z}}})$ has a time-invariant equilibrium at $(\vec{\mathbf{0}}_{(2n_r+3)\times 1}, \vec{\mathbf{0}}_{2n_r\times 1}).$

4.2.6 Linear State Space Model

In this section, the model (4.15) is linearised both in the perspective of an analysis of the stability of the system and of a derivation of a linear control law.

Assuming a small deviation from the reference trajectory, h in model (4.15) can be linearised about its equilibrium at $(\overrightarrow{\mathbf{0}}_{(2n_r+3)\times 1}, \overrightarrow{\mathbf{0}}_{2n_r\times 1})$ which gives the linearised system for h as follows:

$$\overrightarrow{\mathbf{Z}} = [\mathbf{A}]_{(\mathbf{t})} \cdot \overrightarrow{\mathbf{Z}} + [\mathbf{B}]_{(\mathbf{t})} \cdot \overrightarrow{\mathbf{u}_{\mathbf{Z}}}$$
(4.16)

With

$$[\mathbf{A}]_{(\mathbf{t})} = \left[\frac{\partial \mathbf{h}}{\partial \vec{\mathbf{Z}}}\right](\mathbf{0}, \mathbf{0}) = \left[\frac{\partial \mathbf{g}}{\partial \vec{\mathbf{X}}}\right](\vec{\mathbf{X}}_{\mathbf{ref}}, \vec{\mathbf{u}}_{\mathbf{ref}})$$
(4.17)

$$[\mathbf{B}]_{(\mathbf{t})} = \left[\frac{\partial \mathbf{h}}{\partial \overrightarrow{\mathbf{u}_{\mathbf{Z}}}}\right](\mathbf{0}, \mathbf{0}) = \left[\frac{\partial \mathbf{g}}{\partial \overrightarrow{\mathbf{u}}}\right](\overrightarrow{\mathbf{X}}_{\mathbf{ref}}, \overrightarrow{\mathbf{u}}_{\mathbf{ref}})$$
(4.18)

where $\begin{bmatrix} \frac{\partial h}{\partial \mathbf{Z}} \end{bmatrix}$ is the time-varying Jacobian of h with respect to the state variables, $\begin{bmatrix} \frac{\partial h}{\partial \mathbf{u}\mathbf{Z}} \end{bmatrix}$ the time-varying Jacobian of h with respect to the manipulated variables, $\begin{bmatrix} \frac{\partial g}{\partial \mathbf{X}} \end{bmatrix}$ is the time-varying Jacobian matrix of g with respect to the state variables and $\begin{bmatrix} \frac{\partial g}{\partial \mathbf{u}} \end{bmatrix}$ is the time-varying Jacobian matrix of g with respect to the manipulated variables.

The derivation of the Jacobian of g, which is used both by the linear system (4.16) and the NMPC optimisation process, can be found in appendix B.

4.2.7 Explicit Linear State Space Model

In this section, the explicit expression of the matrices $[\mathbf{A}]_{(\mathbf{t})}$ and $[\mathbf{B}]_{(\mathbf{t})}$ of model 4.16 will be given and will be used for the stability and controllability analysis of the one module only case in section 4.3.8. In the rest of the section, the components of the reference angular velocity $\overrightarrow{\Omega}_{ref}$ are defined as ω_x , ω_y and ω_z .

4.2.7.1 Single Module Case

In the single module case, the state has a dimension of 7×1 and the module reaches the position $(\frac{\pi}{2}, \pi)$. In this case, the matrix $[\mathbf{A}]_{(\mathbf{t})}$ is equal to:

$$[\mathbf{A}]_{(\mathbf{t})} = \begin{bmatrix} 0 & \frac{I_y - I_z}{I_x} \omega_z & \frac{I_y - I_z}{I_x} \omega_y & a^2 \frac{I_x + I_z - I_y}{I_x (I_z + a^2)} \omega_x \omega_y & a^2 \frac{I_z - I_x - I_y}{I_x (I_y + a^2)} \omega_x \omega_z & 0 & 0 \\ \frac{I_z + a^2 - I_x}{I_y + a^2} \omega_z & 0 & \frac{I_z + a^2 - I_x}{I_y + a^2} \omega_x & a^2 \frac{\omega_z^2 - \omega_x^2}{I_y + a^2} & a^2 \frac{I_x + I_y - I_z}{I_x (I_y + a^2)} \omega_y \omega_z & 0 & \frac{a^2 \omega_x}{I_y + a^2} \\ \frac{I_x - (I_y + a^2)}{I_z + a^2} \omega_y & \frac{I_x - (I_y + a^2)}{I_z + a^2} \omega_x & 0 & a^2 \frac{I_y - I_z - I_x}{I_x (I_z + a^2)} \omega_y \omega_z & a^2 \frac{\omega_x^2 - \omega_y^2}{I_z + a^2} & \frac{a^2 \omega_x}{I_z + a^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$(4.19)$$

and the matrix $[\mathbf{B}]_{(\mathbf{t})}$ is equal to:

$$[\mathbf{B}] = \begin{bmatrix} 0 & 0 \\ \frac{a^2}{I_y + a^2} & 0 \\ 0 & \frac{-a^2}{I_z + a^2} \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$
(4.20)

Both $[\mathbf{A}]_{(\mathbf{t})}$ and $[\mathbf{B}]$ are bounded matrices with $[\mathbf{B}]$ time-invariant. In the case of pure spin, the torque-free reference angular velocity has a constant value ω_z about Z and $[\mathbf{A}]_{(\mathbf{t})}$ becomes time-invariant:

4.2.7.2 Multiple Modules Case

Again in the multiple module case, as per appendix B, all entries of all Jacobians are bounded. Hence, both $[\mathbf{A}]_{(\mathbf{t})}$ and $[\mathbf{B}]$ are bounded matrices with $[\mathbf{B}]$ timeinvariant. Again, $[\mathbf{A}]_{(\mathbf{t})}$ is time-invariant when the torque-free reference angular velocity is a pure spin.

4.3 Module Controller Design

In this section, the low-level controller of the individual robot module is designed. The approach taken focused on the minimisation of its complexity. The control problem can be best described as a combination of a tracking and set point control problems. Indeed, the objective for each module was to converge to a target position on the surface of the object in a timely fashion with respect to the timescale of the objects dynamics while ensuring that the object's angular velocity tracked a desired angular velocity. As per section 4.2, the state was defined so that the controller could be formulated as a regulator which is equivalent to tracking one reference trajectory. However, this tracking trajectory contains physically coupled objectives since by reaching its target surface location, the module necessarily changes the rotational state of the object.

From the formulation of the state space model of the individual module plant in section 4.2 and the physical model (3.24) of the interactions between the robot and

the object, different control strategies could be pursued. The analysis of the stability and controllability of the robot-object system defined in section 4.1.5.1 prompted the choice of a model predictive control strategy over other strategies such as using a LQR regulator.

Before performing this stability and controllability analysis, preliminary hypotheses will be established regarding the feasible and admissible trajectory sets in order to reduce the constraints placed on the controller for the calculations of the control commands. The controller will then be formally analysed and take two forms: a linear model predictive controller and a nonlinear model predictive controller, whose simulation and comparison is the subject of chapter 5.

4.3.1 Unicity of Each Trajectory

As per Cauchy-Lipschitz theorem found in [44] page 6 for each initial value $\vec{\mathbf{X}}_0$ and each initial time $t_0 \in \mathbb{R}$ and each locally Lebesgue integrable control function $\vec{\mathbf{u}} : \mathbb{R}^{2n_r} \to \mathbb{R}^{2n_r}$ there is a unique trajectory $\vec{\mathbf{X}}(t)$ with $\vec{\mathbf{X}}(t_0) \ \forall t \in \mathbb{R}$.

The system 4.14 is continuous and differentiable hence it is Lipschitz and all trajectories are unique.

4.3.2 Admissibility

Admissibility is a property characterising the system trajectories which are possible according to a given set of constraints. It could represent the set of all physically possible trajectories but not necessarily and other constraints could also apply.

The formal definition of Admissibility can be found in [26] page 48:

Consider a control system 4.14 and the state and control constraint sets $\mathbb{X} \subseteq X$ and $\mathbb{U}(x) \subseteq U$ with X the set of all possible states and U the set of all possible control commands.

1. The states $x \in \mathbb{X}$ are called admissible states and the control values $u \in \mathbb{U}(x)$ are called admissible control values for x. The elements of the set $\mathbb{Y} = [(x, u) \in X \times U | x \in \mathbb{X}, u \in \mathbb{U}(x)]$ are called admissible pairs.

- 2. For $N \in \mathbb{N}$ and an initial value $x_0 \in \mathbb{X}$, a control sequence $u \in U^N$ and the corresponding trajectory $x_u(k, x_0)$ are admissible for x_0 up to time N, if $(x_u(k, x_0), u(k)) \in \mathbb{Y} \ \forall k = 0, ..., N - 1 \text{ and } x_u(N, x_0) \in \mathbb{X} \text{ holds.}$ The set of admissible control sequences for x_0 up to time N is denoted by $\mathbb{U}^N(x_0)$.
- 3. A control sequence $u \in U^{\infty}$ and the corresponding trajectory $x_u(k, x_0)$ are called admissible for x_0 if they are admissible for x_0 up to every time $N \in \mathbb{N}$. The set of admissible control sequences for x_0 is denoted by $\mathbb{U}^{\infty}(x_0)$.
- 4. A (possibly time varying) feedback law $\mu : \mathbb{N}_0 \times \mathbb{X} \to \mathbb{U}$ is called admissible if $\mu(n, x) \in U^1(x)$ holds for all $x \in \mathbb{X}$ and all $n \in \mathbb{N}^0$.

This PhD study is a feasibility study. Therefore, it was decided that the controller was not to be restrained in its search for a solution how unrealistic this solution may come out to be. This meant that there were a priori no mathematical constraints on the state and control values, i.e. not even a restraint to what is physically possible. In other words, despite the fact that the system has a constant energy level, all angular velocities and acceleration are admissible and any energy expenditure is allowed as well as instant power levels. Hence, the sets of constraints are $\mathbb{X} = X$ and $\mathbb{U}(x) = U$. Defining n_r as the number of robot modules, the admissible sets are:

$$X = \mathbb{R}^3 \times ([0,\pi] \times [0,2\pi[\times\mathbb{R}^2)^{n_r}$$

$$(4.22)$$

$$U = \mathbb{R}^{2n_r} \tag{4.23}$$

This a priori definition of the admissible sets will be narrowed down later when designing a controller for the particular application problem under study. Two physical constraints will then be taken into account to design the controller appropriately and realistically:

 The angular velocity is constrained by Newton's second law of motion and described by model (3.23) in chapter 3. This will be an implicit constraint which does not change the definition of X the admissible set.
2. The amount of energy in the system is bounded: there exists a finite amount of energy E for which $\forall t \geq 0 \int_0^{+\infty} \vec{\mathbf{u}}(t)^T [\mathbf{\Lambda}_{\vec{\mathbf{u}}}] \vec{\mathbf{u}}(t) \mathbf{dt} \leq \mathbf{E}$ where $\vec{\mathbf{u}}(t)$ are the control commands over the infinite time-horizon and $[\mathbf{\Lambda}_{\vec{\mathbf{u}}}]$ is a positive semidefinite diagonal matrix with diagonal entries of equal value. This defines a new control commands admissible set:

$$U_E = \left\{ u \in U \middle| \int_0^{+\infty} \vec{\mathbf{u}}(t)^T [\mathbf{\Lambda}_{\vec{\mathbf{u}}}] \vec{\mathbf{u}}(t) d\mathbf{t} \le \mathbf{E} \right\}$$
(4.24)

4.3.3 Viability

Viability characterises whether or not a current system state to which is applied a set of admissible commands will lead to a new admissible state.

Viability is defined in [26] page 49 as an assumption with an important implication for feasibility:

$$\forall x \in X \; \exists u \in U(x) \text{ such that } f(x, u) \in X \tag{4.25}$$

Given the no constraints hypothesis made on admissibility in section 4.3.2, viability is guaranteed for system (4.14) since $\forall x \in X$ and $\forall u \in U$, $f(x, u) \in X$. Restraining U to U_E has no influence on the size of X which implies that system (4.12) is still viable in this case:

$$\forall x \in X \; \exists u \in U_E(x) \text{ such that } f(x, u) \in X$$
(4.26)

4.3.4 Feasibility

Feasibility characterises the achievability of finding a solution to the derivation of admissible control commands.

Feasibility is defined in [26] page 49:

A control problem is feasible if for an initial state value x_0 , the set over which the control commands are calculated is not empty i.e. over an infinite time horizon $\mathbb{U}^{\infty}(x_0) \neq \emptyset$. As per [26] page 49, viability implies feasibility, the control problem for this study is therefore feasible over the entire sets \mathbb{R}^{2n_r} and U_E . In particular, the later defined MPC and NMPC optimisations are both feasible.

4.3.5 Stabilisability Controllability and Choice of Control Method

In this section, the stabilisability and controllability of model (4.15) will be examined through its linearisation and for the particular case when the object is in pure spin which is the ideal final state the object and system should be in at the end of deployment. In this case, the linearisation is time-invariant. The limitations encountered lead to the choice of model predictive control as a control method for this study.

4.3.5.1 Stability of the Nonlinear Model via Linearisation

In this section, it is assumed that the object is initially in pure spin. As per section 4.2.7.1, this implies that the linearised system is time-invariant.

The stability study of the nonlinear model (4.15) was approach via the stability study of its linearisation (4.16) using the following results:

- 1. "If the linearisation of nonlinear system is time-invariant then having all eigenvalues in the open left-half plane guarantees local uniform asymptotic stability of the origin of the nonlinear system." ([50] chapter 5 pages 216-217).
- 2. "If at least one of the eigenvalues of the linearisation lies in the open right-half plane then the origin is unstable." ([50] chapter 5 pages 216-217).
- "When any one of the eigenvalues has zero real part, stability cannot be determined by linearisation." ([50] chapter 7 pages 288)

The characteristic polynomial of matrix $[\mathbf{A}]$ (formula (4.21)) is:

$$P(\lambda) = -\lambda^{5} \cdot \left[(Ix \cdot a^{2} + Ix \cdot Iy) \cdot \lambda^{2} + \omega_{z}^{2} \cdot (Iz^{2} - Iy \cdot a^{2} + Iz \cdot a^{2} + Ix \cdot Iy - Ix \cdot Iz - Iy \cdot Iz) \right]$$

$$(4.27)$$

By virtue of the principle of the excluded third, two possibilities:

- 1. $\underline{\omega_z^2} = 0$, the reference trajectory corresponds to a system at rest and the control objective is de-spinning the object. In this case, [**A**] has seven nil eigenvalues and regardless of the shape of the object. No conclusion can be drawn on the stability of the equilibrium at the origin of system (4.15).
- 2. $\underline{\omega_z^2 > 0}$, the reference trajectory corresponds to a system in a state of pure spin and the control objective is to bring the object to a state of pure spin. In this case, [**A**] has five nil eigenvalues and since $(Ix \cdot a^2 + Ix \cdot Iy) > 0$, the values of the two remaining eigenvalues depend on the sign of $Iz^2 - Iy \cdot a^2 + Iz \cdot a^2 + Ix \cdot$ $Iy - Ix \cdot Iz - Iy \cdot Iz$. Expressing the moments of inertia with the normalised semi-axes length of the ellipsoid with c = 1 and defining m = 0.2MR with MR the mass ratio of the object to the module, further calculations show that the values of the two remaining eigenvalues of [**A**] depend on the sign of:

$$[a^{2}(1+m) - m](b^{2} - 1)$$
(4.28)

Again two possibilities here:

- (a) If [a²(1+m) − m](b² − 1) ≥ 0 then all eigenvalues of [A] have a zero real part and no conclusion can be drawn on the stability of the equilibrium at the origin of system (4.15).
- (b) If [a²(1 + m) − m](b² − 1) < 0 then one eigenvalue of [A] has a strictly positive real part, one has a strictly negative real part and the equilibrium of system (4.15) is unstable.</p>

The stability study of the linearisation of system (4.15) can only predict those cases which are unstable. For all the other cases, no conclusion can be drawn on the stability of the equilibrium without a higher order analysis. However, the dimension of the centre space equalling that of the state space, no use can be made of a centre manifolds approach based on the three centre manifolds theorems which can be found in [50] chapter 7 page 310-311. Therefore, only a higher order stability analysis without the possibility of reducing the dimension of the problem can be pursued. This would entail the direct application of Lyapunov basic theorems (to be found in [50] chapter 5) or the building of a Lyapunov function. Both of these approaches require the derivation of the second order dynamics in the form of an array of seven 7×7 Hessian matrices in a third-order tensor. This was left for future work. In the absence of a criteria to establish whether or not the nonlinear equilibrium is stable, this equilibrium will be referred from now on as potentially stable.

Table 4.1 below summarises the conclusions about the system stability for each type of object's shape. The mention "N/A" corresponds to those cases which are not applicable to the ellipsoid shape in question because of the values of a and b.

As a reminder about the ellipsoid shapes:

- The oblate case corresponds to (a = b) > (c = 1).
- The prolate case corresponds to (a = b) < (c = 1).
- The sphere case corresponds to a = b = (c = 1).
- The asymmetric case corresponds to $a \neq b \neq (c = 1)$.

a	b	Oblate Ellipsoid	Prolate Ellipsoid	Asymmetric Ellipsoid
$a > \sqrt{\frac{m}{m+1}}$	b > 1	Potentially Stable	N/A	Potentially Stable
$a > \sqrt{\frac{m}{m+1}}$	b < 1	N/A	Unstable	Unstable
$a < \sqrt{\frac{m}{m+1}}$	b > 1	N/A	N/A	Unstable
$a < \sqrt{\frac{m}{m+1}}$	b < 1	N/A	Potentially Stable	Potentially Stable

Table 4.1: Influence of the Object's Shape on the Stability of the System's Spin

Although the potentially stable equilibrium could prove to be unstable after further analysis, the limit condition (4.28) may parametrise a potential bifurcation between a stable and unstable behaviour for the equilibrium. Hence, the behaviour of this limit is worth exploring.

Figures 4.3, 4.4 and 4.5 below show the evolution of the limit condition (4.28) with respect to the mass ratio between the object and the module, highlighting the relationship between spin stability of the object's shape. The limit condition is strictly increasing with the mass ratio. It reaches the value 0.9 for a mass ratio of about 20 and tends to 1 at infinity.



Figure 4.3: Stability Limit vs. Mass Ratio (Small MR).



Figure 4.4: Stability Limit vs. Mass Ratio (Medium MR).



Figure 4.5: Stability Limit vs. Mass Ratio (Large MR).

The conclusions which can be drawn for system (4.15) are the following:

- 1. For all object shape, a state of rotational rest (despin) is always potentially stable.
- 2. When the object is an oblate ellipsoid or a sphere a state of pure spin is always potentially stable.
- 3. When the object is a prolate ellipsoid, a state of pure spin is unstable when $a(=b) \in]\sqrt{\frac{m}{m+1}}, 1[$. This condition corresponds to prolate shapes close to a spherical shape. These prolate shapes for which the spin is unstable get closer to a spherical shape when the mass ratio between the object and the module increases (see figures 4.3, 4.4 and 4.5 above). Spheres are the limit shapes for which the system spin is potentially stable as per point number 1 above.
- 4. When the object is asymmetric, the shapes of the object for which the state of pure spin is unstable tend to a limit shape when the mass ratio between the object and the module increases. This limit shape corresponds to cases where the object's spin axis (in this case the Z axis) is the unstable medium principal axis i.e. (a < c < b) or $(b < c \le a)$.

As a general conclusion, it seems a better option a priori to aim for a final despun state rather than a final state of pure spin since for the despun state the shape of the object bears no influence on its potential stability. In the next section, the controllability through linearisation will be examined.

4.3.5.2 Controllability of the Linear Model

In this section, the controllability and stabilisability by a linear control law of system (4.15) is examined through its linearisation. The general definition of controllability can be found in [6] page 45:

A plant is controllable for a pair of state $(\vec{\mathbf{X}}_0, \vec{\mathbf{X}})$ if there exists a control which takes the plant from state $\vec{\mathbf{X}}_0$ to state $\vec{\mathbf{X}}$ in a finite time interval. A plant is then fully controllable if it is controllable for every possible pair of states $(\vec{\mathbf{X}}_0, \vec{\mathbf{X}})$.

The indirect method of Lyapunov provides a mean to stabilise a nonlinear model with a linear feedback control obtained via its linearisation. It is found in [50] chapter 6 pages 236-237:

- For a nonlinear system with an equilibrium at the origin, if the linearised time invariant system is completely controllable then there exits a matrix [K] such that the feedback control law [K]x locally stabilises the nonlinear control system.
- 2. Even when the linearisation is time varying, then stabilisation of the resulting time varying linear system stabilises the nonlinear time varying system.

The key to finding a stabilising feedback linear control law for system (4.15) is the complete controllability of its linearisation (4.16). As was established in section 4.3.5, for a system in a pure state of spin its linearisation is time-invariant. In this case the linearised model (4.16) is completely controllable if matrices [**A**] and [**B**] are completely controllable i.e. when the rank of the controllability matrix [**B**, **AB**, ..., **A**ⁿ⁻¹**B**] is equal to the size of the state *n* with n = 7 in this study [50].

Assuming that the system's trajectory is close to a pure spin reference trajectory, the rank of the controllability matrix was calculated for ten ellipsoidal shapes and for a set of normalised dimensionless angular velocities as well as for a set of mass ratios. The dimensionless angular velocity in this context can be understood as a measure of the amount of normalised dimensionless kinetic energy relative to the inertia of the object and therefore as a measure of stability of the system, since for a spinning object, the faster the spin the more stable the object is [60]. The parameters sets for the ellipsoids dimensions, the dimensionless spin rate and mass ratios were respectively:

	a		1	$\frac{1}{2}$	$\frac{1}{10}$	$\frac{8}{10}$	10	$\frac{10}{8}$	2	$\frac{1}{10}$	$\frac{1}{3}$	$\frac{8}{10}$
• <u>Ellipsoids' Normalised Dimensions</u> :	b	=	1	$\frac{1}{2}$	$\frac{1}{10}$	$\frac{8}{10}$	10	$\frac{10}{8}$	2	$\frac{8}{10}$	$\frac{1}{2}$	$\frac{9}{10}$
	c		1	1	1	1	1	1	1	1	1	1

г п

г

• $\omega_0 \in \{0, 1, 10, 100, 1000, 10000, 100000\}$

• $MR \in \{10, 100, 1000, 10000, 100000\}$

The results for each of the ten ellipsoid shapes are displayed below in ten tables.

ω_0 / MassRatio	10	100	1000	10000	100000
0	4	4	4	4	4
1	5	5	5	5	5
10	5	5	5	5	5
100	5	5	5	5	5
1000	5	5	5	5	5
10000	5	5	5	5	5
100000	5	5	5	5	5

Table 4.2: Rank of the Controllability Matrix for the Sphere

For spheres, the controllability matrix always has a rank of 5 unless the system does not rotate in which case its rank is equal to 4. The linear system (4.16) is not controllable and no conclusion can be drawn as to the possibility of stabilising the system (4.15) with a time-invariant linear control law. Neither the mass ratio nor the dimensionless angular velocity bears any influence on the rank of the controllability matrix.

ω_0 / MassRatio	10	100	1000	10000	100000
0	4	4	4	4	4
1	6	6	6	6	6
10	6	6	6	6	6
100	6	6	6	6	6
1000	2	2	6	6	6
10000	2	2	2	2	2
100000	2	2	2	2	2

Table 4.3: Rank of the Controllability Matrix for the Ellipsoid $a=\frac{1}{2} b=\frac{1}{2} c=1$

For the prolate ellipsoid $(\frac{1}{2}, \frac{1}{2}, 1)$, the controllability matrix has a rank of 6 for dimensionless angular velocity up to a 1000 and 2 for dimensionless angular velocity of 10000 or more. If the system does not rotate the rank is equal to 4. The linear system (4.16) is not controllable and no conclusion can be drawn as to the possibility of stabilising the system (4.15) with a time-invariant linear control law. The rank of the controllability matrix does not depend on the mass ratio but on the dimensionless angular velocity except for the limit cases when the angular velocity is equal to 1000 and the mass ratio below 100. This later situation is to be contrasted with the two other prolate ellipsoids where the mass ratio bears strictly no influence on the rank of the controllability matrix.

ω_0 / MassRatio	10	100	1000	10000	100000
0	4	4	4	4	4
1	6	6	6	6	6
10	6	6	6	6	6
100	6	6	6	6	6
1000	6	6	6	6	6
10000	2	2	2	2	2
100000	2	2	2	2	2

Table 4.4: Rank of the Controllability Matrix for the Ellipsoid $a=\frac{1}{10} b=\frac{1}{10} c=1$

For the prolate ellipsoid $(\frac{1}{10}, \frac{1}{10}, 1)$, the controllability matrix has a rank of 6 for dimensionless angular velocity up to a 1000 and 2 for dimensionless angular velocity of 10000 or more. If the system does not rotate the rank is equal to 4. The linear system (4.16) is not controllable and no conclusion can be drawn as to the possibility of stabilising the system (4.15) with a time-invariant linear control law. The rank of the controllability matrix does not depend on the mass ratio but on the dimensionless angular velocity.

ω_0 / MassRatio	10	100	1000	10000	100000
0	4	4	4	4	4
1	6	6	6	6	6
10	6	6	6	6	6
100	6	6	6	6	6
1000	6	6	6	6	6
10000	2	2	2	2	2
100000	2	2	2	2	2

Table 4.5: Rank of the Controllability Matrix for the Ellipsoid $a = \frac{8}{10} b = \frac{8}{10} c = 1$

For the prolate ellipsoid $(\frac{8}{10}, \frac{8}{10}, 1)$, the controllability matrix has a rank of 6 for dimensionless angular velocity up to a 1000 and 2 for dimensionless angular velocity of 10000 or more. If the system does not rotate the rank is equal to 4. The linear system (4.16) is not controllable and no conclusion can be drawn as to the possibility of stabilising the system (4.15) with a time-invariant linear control law. The rank of the controllability matrix does not depend on the mass ratio but on the dimensionless angular velocity.

ω_0 / MassRatio	10	100	1000	10000	100000
0	4	4	4	4	4
1	6	6	6	6	6
10	6	6	6	6	6
100	6	6	6	6	6
1000	2	2	6	6	6
10000	2	2	2	2	2
100000	2	2	2	2	2

Table 4.6: Rank of the Controllability Matrix for the Ellipsoid a=10 b=10 c=1

For the oblate ellipsoid (10, 10, 1), the controllability matrix has a rank of 6 for dimensionless angular velocity up to a 1000 and 2 for dimensionless angular velocity of 10000 or more. The shift from rank 6 to rank 2 occurs when the angular velocity is equal to 1000 and the mass ratio below 100. If the system does not rotate the rank is equal to 4. The linear system (4.16) is not controllable and no conclusion can be drawn as to the possibility of stabilising the system (4.15) with a time-invariant linear control law. The rank of the controllability matrix depends mostly on the dimensionless angular velocity and on the mass ratio near the rank shift.

ω_0 / MassRatio	10	100	1000	10000	100000
0	4	4	4	4	4
1	6	6	6	6	6
10	6	6	6	6	6
100	6	6	6	6	6
1000	2	2	6	6	6
10000	2	2	2	2	2
100000	2	2	2	2	2

Table 4.7: Rank of the Controllability Matrix for the Ellipsoid a=2 b=2 c=1

For the oblate ellipsoid (2, 2, 1), the controllability matrix has a rank of 6 for dimensionless angular velocity up to a 1000 and 2 for dimensionless angular velocity of 10000 or more. The shift from rank 6 to rank 2 occurs when the angular velocity is equal to 1000 and the mass ratio below 100. If the system does not rotate the rank is equal to 4. The linear system (4.16) is not controllable and no conclusion can be drawn as to the possibility of stabilising the system (4.15) with a time-invariant linear control law. The rank of the controllability matrix depends mostly on the dimensionless angular velocity and on the mass ratio near the rank shift.

ω_0 / MassRatio	10	100	1000	10000	100000
0	4	4	4	4	4
1	6	6	6	6	6
10	6	6	6	6	6
100	6	6	6	6	6
1000	2	6	6	6	6
10000	2	2	2	2	2
100000	2	2	2	2	2

Table 4.8: Rank of the Controllability Matrix for the Ellipsoid $a=\frac{5}{4}$ $b=\frac{5}{4}$ c=1

For the oblate ellipsoid $(\frac{5}{4}, \frac{5}{4}, 1)$, the controllability matrix has a rank of 6 for dimensionless angular velocity up to a 1000 and 2 for dimensionless angular velocity of 10000 or more. The shift from rank 6 to rank 2 occurs when the angular velocity is equal to 1000 and the mass ratio below 10. If the system does not rotate the rank is equal to 4. The linear system (4.16) is not controllable and no conclusion can be drawn as to the possibility of stabilising the system (4.15) with a time-invariant linear control law. The rank of the controllability matrix depends mostly on the dimensionless angular velocity and on the mass ratio near the rank shift.

ω_0 / MassRatio	10	100	1000	10000	100000
0	4	4	4	4	4
1	6	6	6	6	6
10	6	6	6	6	6
100	6	6	6	6	6
1000	6	6	6	6	6
10000	2	2	2	2	2
100000	2	2	2	2	2

Table 4.9: Rank of the Controllability Matrix for the Ellipsoid $a=\frac{1}{10} b=\frac{8}{10} c=1$

For the asymmetric ellipsoid $(\frac{1}{10}, \frac{8}{10}, 1)$, the controllability matrix has a rank of 6 for dimensionless angular velocity up to a 1000 and 2 for dimensionless angular velocity of 10000 or more. If the system does not rotate the rank is equal to 4. The linear system (4.16) is not controllable and no conclusion can be drawn as to the possibility of stabilising the system (4.15) with a time-invariant linear control law. The rank of the controllability matrix does not depend on the mass ratio but on the dimensionless angular velocity. Despite having a very asymmetric shape with spread dimensions, this case is close and similar to the prolate ellipsoid cases $(\frac{1}{10}, \frac{1}{10}, 1)$ and $(\frac{8}{10}, \frac{8}{10}, 1)$.

ω_0 / MassRatio	10	100	1000	10000	100000
0	4	4	4	4	4
1	6	6	6	6	6
10	6	6	6	6	6
100	6	6	6	6	6
1000	2	2	6	6	6
10000	2	2	2	2	2
100000	2	2	2	2	2

Table 4.10: Rank of the Controllability Matrix for the Ellipsoid $a=\frac{1}{3} b=\frac{1}{2} c=1$

For the asymmetric ellipsoid $(\frac{1}{3}, \frac{1}{2}, 1)$, the controllability matrix has a rank of 6 for dimensionless angular velocity up to a 1000 and 2 for dimensionless angular velocity of 10000 or more. If the system does not rotate the rank is equal to 4. The linear system (4.16) is not controllable and no conclusion can be drawn as to the possibility of stabilising the system (4.15) with a time-invariant linear control law. The rank of the controllability matrix does not depend on the mass ratio but on the dimensionless angular velocity except for the limit cases when the angular velocity

ω_0 / MassRatio	10	100	1000	10000	100000
0	4	4	4	4	4
1	6	6	6	6	6
10	6	6	6	6	6
100	6	6	6	6	6
1000	6	6	6	6	6
10000	2	2	2	2	2
100000	2	2	2	2	2

is equal to 1000 and the mass ratio below 100. By the object's dimension, this case is close and similar to the prolate ellipsoid case $(\frac{1}{2}, \frac{1}{2}, 1)$.

Table 4.11: Rank of the Controllability Matrix for the Ellipsoid $a = \frac{8}{10} b = \frac{9}{10} c = 1$

For the asymmetric ellipsoid $(\frac{8}{10}, \frac{9}{10}, 1)$, the controllability matrix has a rank of 6 for dimensionless angular velocity up to a 1000 and 2 for dimensionless angular velocity of 10000 or more. If the system does not rotate the rank is equal to 4. The linear system (4.16) is not controllable and no conclusion can be drawn as to the possibility of stabilising the system (4.15) with a time-invariant linear control law. The rank of the controllability matrix does not depend on the mass ratio but on the dimensionless angular velocity. By the object's dimension, this case is close and similar to the prolate ellipsoid case $(\frac{8}{10}, \frac{8}{10}, 1)$.

From the above analysis, the system (4.15) is underactuated in all configurations. In most real situations in terms of mass ratio and dimensionless angular velocity, only one state should remain uncontrollable except when the objective is to despin the object in which case three out of the seven states will not be controllable. In order to ensure a better controllability and therefore a better chance of successfully meeting the task objectives, it seems a priori better to have the system converge to a state of spin rather than despining it. Finally, it is interesting to observe that when the dimensionless angular velocity has a very large magnitude compared to the inertia of the system, the system becomes really underactuated with only two states being controllable. In this case, the object's kinetic energy level is such that the object's spin is extremely stable and therefore resistant to the disturbing effect of a module moving at its surface. In this case, making the system converge to a pure state of spin would be extremely difficult and energy intensive.

4.3.5.3 Choice of a Model Predictive Control Approach

The stabilisability and controllability analysis of model (4.15) was conducted through its linearisation (4.16) for the specific cases where the reference state trajectory describes the object in a pure state of spin which the desired final state for this study's intended engineering application.

In section 4.3.5.1, it was shown that the system's stability with respect to different ellipsoid shapes is dependent on the mass ratio between the object and the robot module. Two cases emerged. The system is either clearly unstable or potentially stable. In this later case, all the eigenvalues of the linearised model (4.16) have zero real parts. Consequently the origin of system (4.15) could possess very different behaviours ranging from stability with a well defined albeit small domain of attraction to plain insatiability. Complex behaviours such as having an equilibrium which is attractive for certain trajectories while repulsive for others are also possible.

In section 4.3.5.2, the controllability analysis of the linearised system (4.16) clearly showed that the system is underactuated in all configurations. The magnitude of the angular velocity or spin rate is the main parameter affecting the controllability of the system (4.15). Controllability deceases with the spin rate but only for very large magnitudes with respect to the system's inertia. As per section 2.2.2, very large spin rates are not possible in practice. Therefore, it can be expected that the rank of the controllability matrix for a real system would be equal to 6. Moreover, given the stabilising effect of increasing spin rate, it is to be expected that the difficulty to converge to a pure state of spin will increase with the initial angular velocity deviation from a pure spin reference state.

In order to explore the stability of system (4.15) further, only the basic Lyapunov theorems can be used. As mentioned above, all the eigenvalues of its linearisation about the reference trajectory have zero real parts. In the 1-dimensional case when the first derivative is zero at the equilibrium point, the second derivative positivity or negativity in the neighbourhood of the equilibrium point respectively identifies the maximum of a convex function or the minimum of a concave function. Extrapolating on this 1-dimension case, the derivation of a third order tensor composed of Hessian

126

matrices would help to determine the stability of each of the sub-state spaces. Each of these sub-state spaces could be stable or unstable. This analysis has been left for future work.

However, given that the matrices of the linear system (4.16) are either timeinvariant or periodic and bounded since the system is conservative and the potential energy contained in the battery is finite, it was assumed that, at least in some cases, a non-empty and invariant domain of attraction exists around the equilibrium or that the linearised system is stabilisable. In both cases, as per [12, 21], a Lyapunov function bounding an infinite horizon quadratic cost function can be constructed which helps prove the existence of stabilising control commands as will be seen in section 4.3.8.

In conclusion, offline infinite horizon techniques such as Linear Quadratic Regulator (LQR) providing constant linear feedback gains cannot be used. The feedback control law, if it exists is likely to be time-varying. The construction of the control commands for a system whose stability properties are unknown can only be an online exploration. A model predictive control approach offers the possibility to channel the search through an optimisation with a quadratic cost function which lends itself well to the construction of a Lyapunov function to be used for the proof of existence of the control commands. This is why this approach was taken for the low-level module controller design.

4.3.6 Model Predictive Control Formulation

As per the previous section, a Model Predictive Control (MPC) approach was chosen for the low-level module controller design. In this section, the MPC method is formally defined with a formulation suiting both the linear and nonlinear cases.

The system's state space model (4.15) is continuous. Continuous models with infinite time horizons have interesting stability properties [26]. With this in mind and drawing on the formulation of the NMPC control optimisation procedure in [24] chapter 2, the following definition has been adopted for this study: Given a continuous evolution model in time:

$$\dot{x}(t) = f(x(t), u(t), t)$$
(4.29)

with state x(t), initial condition $x(0) \in \mathbb{R}^n$, control command u(t) and a finite or infinite time horizon T > 0, the objective is to minimize a cost function with stage cost function l at time t and terminal cost S when T is finite (it disappears when $T = +\infty$):

$$J_T(x[t_0, T], u[t_0, T], t_0) = \int_{t_0}^T l(x(t), u(t), t) dt + S(x(T), T)$$
(4.30)

subject to the inequality constraints $\forall t \in [t_0, T]$:

$$u_{min} \le u(t) \le u_{max}$$

$$g(x(t), u(t), t) \le 0$$
(4.31)

In real world applications, it is impossible to use a continuous model and infinite time horizon. The model needs to go through discretisation and relies on data sampling. As per [26] page 72, this led to a new formulation of (4.30) for finite or infinite time horizon T > 0:

Given the evolution model at from instant k to k + 1:

$$x_u(k+1, x_0) = f(x_u(k, x_0), u(k))$$
(4.32)

With initial state at instant n:

$$x_u(n, x_0) = x_0 \tag{4.33}$$

Minimize:

$$J_T(x[n,T], u[n,T], n) = \sum_{k=0}^T l(x_u(k, x_0), u(k), n+k)$$
(4.34)

with respect to $u(\cdot) \in \mathbb{U}^{\infty}(x_0)$ admissible control commands

There is no need here to introduce the sampling time rate in these equations as it is simply a normalised and factorisable constant variable that would bear no influence on the optimisation process.

4.3.7 Cost Function

The choice of cost function for the model predictive control optimisation in this study was guided by the assumption made in section 4.3.5.3 that there exists a bounded domain of attraction around system's (4.15) equilibrium.

In order to prove the existence of a stabilising sequence of control commands for a nonlinear system at its equilibrium, most theorems found in [26] assume that the cost function is bounded by that two unbounded strictly increasing continuous functions. In most cases, especially when stabilisability of the linearised system can be proven (cf [26] page 137), a quadratic cost function fulfils these theorems' hypotheses. Moreover, as per [26] pages 124 and 139, a large time horizon is to be preferred to a final stabilising terminal constraints cost since the large time horizon is stabilising and avoids a huge computational cost.

In light of the above, the cost function for the optimisation problem was chosen to be a quadratic function with an infinite time horizon. Under conditions stated in section 4.3.8, this form of the cost function leads to the derivation of suboptimal asymptotically stabilising control commands.

In what follows, the states are viewed from the stand point of their original definition for clarity even though the actual system that is studied is system (4.15). In its continuous and most general form, the cost is as the follows:

$$J = J_{\infty}(\overrightarrow{\mathbf{X}}(t), \overrightarrow{\mathbf{u}}(t), t) = \int_{0}^{+\infty} (\overrightarrow{\mathbf{\Omega}}(t) - \overrightarrow{\mathbf{\Omega}}_{ref}(t))^{T} [\mathbf{\Lambda}_{\overrightarrow{\mathbf{\Omega}}}] (\overrightarrow{\mathbf{\Omega}}(t) - \overrightarrow{\mathbf{\Omega}}_{ref}(t)) dt + \int_{0}^{+\infty} (\overrightarrow{\mathbf{\Theta}}(t) - \overrightarrow{\mathbf{\Theta}}_{ref}(t))^{T} [\mathbf{\Lambda}_{\overrightarrow{\mathbf{\Theta}}}] (\overrightarrow{\mathbf{\Theta}}(t) - \overrightarrow{\mathbf{\Theta}}_{ref}(t)) dt +$$
(4.35)
$$\int_{0}^{+\infty} (\overrightarrow{\mathbf{\Theta}}(t) - \overrightarrow{\mathbf{\Theta}}_{ref}(t))^{T} [\mathbf{\Lambda}_{\overrightarrow{\mathbf{\Theta}}}] (\overrightarrow{\mathbf{\Theta}}(t) - \overrightarrow{\mathbf{\Theta}}_{ref}(t)) dt +$$
$$\int_{0}^{+\infty} (\overrightarrow{\mathbf{u}}(t) - \overrightarrow{\mathbf{u}}_{ref}(t))^{T} [\mathbf{\Lambda}_{\overrightarrow{\mathbf{u}}}] (\overrightarrow{\mathbf{u}}(t) - \overrightarrow{\mathbf{u}}_{ref}(t)) dt$$

where $[\Lambda_{\overrightarrow{\Omega}}]$, $[\Lambda_{\overrightarrow{\Theta}}]$, $[\Lambda_{\overrightarrow{\Theta}}]$ and $[\Lambda_{\overrightarrow{u}}]$ are positive semi-definite diagonal matrices with diagonal entries of equal value. $[\Lambda_{\overrightarrow{\Omega}}]$ represents the importance of the object's initial angular velocity tracking. $[\Lambda_{\overrightarrow{\Theta}}]$ represents the importance of the objective of the individual module of reaching its target destination at the surface of the object. $[\Lambda_{\overrightarrow{\Theta}}]$ represents the constraints placed on the individual module relative angular velocity with respect to the object. Finally, $[\Lambda_{\overrightarrow{u}}]$ represents the constraints placed on each individual module's energy expenditure.

During the practical optimisation process, the system is actually sampled and the cost function had to be discretised and was discretised in such a way so as to match the sampling rate. As per section 4.3.6, there is no need here to introduce the constant sampling time rate in these equations as it is simply a normalised and factorisable constant variable that would bear no influence on the optimisation process. The time-horizon p is an integer chosen to be large enough to be considered a good approximation of an infinite time-horizon (see section 4.5). The discrete cost function took the following form:

$$J = J_{p}(\overrightarrow{\mathbf{X}}(t_{j}), \overrightarrow{\mathbf{u}}(t_{j}), t_{j}) = \Lambda_{\overrightarrow{\mathbf{\Omega}}} \sum_{j=1}^{p} ||\overrightarrow{\mathbf{\Omega}}(t_{j}) - \overrightarrow{\mathbf{\Omega}}_{ref}(t_{j})||^{2} + \Lambda_{\overrightarrow{\mathbf{\Theta}}} \sum_{j=1}^{p} ||\overrightarrow{\mathbf{\Theta}}(t_{j}) - \overrightarrow{\mathbf{\Theta}}_{ref}(t_{j})||^{2} + \Lambda_{\overrightarrow{\mathbf{\Theta}}} \sum_{j=1}^{p} ||\overrightarrow{\mathbf{\Theta}}(t_{j}) - \overrightarrow{\mathbf{\Theta}}_{ref}(t_{j})||^{2} + \Lambda_{\overrightarrow{\mathbf{\Theta}}} \sum_{j=1}^{p} ||\overrightarrow{\mathbf{\Theta}}(t_{j}) - \overrightarrow{\mathbf{\Theta}}_{ref}(t_{j})||^{2} + \Lambda_{\overrightarrow{\mathbf{U}}} \sum_{j=1}^{p} ||\overrightarrow{\mathbf{u}}(t_{j}) - \overrightarrow{\mathbf{u}}_{ref}(t_{j})||^{2}$$

where $\Lambda_{\overrightarrow{\Omega}}$, $\Lambda_{\overrightarrow{\Theta}}$, $\Lambda_{\overrightarrow{\Theta}}$ and $\Lambda_{\overrightarrow{u}}$ are the positive real diagonal entries of $[\Lambda_{\overrightarrow{\Omega}}]$, $[\Lambda_{\overrightarrow{\Theta}}]$, $[\Lambda_{\overrightarrow{\Theta}}]$ and $[\Lambda_{\overrightarrow{u}}]$ and where j is the index of the discrete time instant t_j .

4.3.8 Optimal Control Commands Existence and Stability

In section 4.1.3, the stability objective constrains the robot to deploy in such a way that the new rigid angular velocity of the system tracks a reference angular velocity for the object. Under its hypotheses, theorem 4.16 in [26] pages 84-85 states the existence of a suboptimal control law which would asymptotically stabilize the closed-loop system as per the definition 2.16 given in [26] page 32. This has to be understood as the convergence over time ad infinitum of the state towards the tracked reference trajectory. Hence, the existence of a solution or possibility of meeting the aforementioned objective is established. Providing the assumption of existence of a bounded domain of attraction combined with the current proven properties of system (4.15) all hypotheses of theorem 4.16 can be fulfilled.

Below are listed the system's definitions and properties which fulfil the theorem's assumptions. The states are viewed from the stand point of their original definition for clarity even though the actual system that is studied is system (4.15).

- 1. The control problem is an infinite-horizon time-varying optimal control problem for a system of the form $\dot{\vec{\mathbf{X}}} = g(\vec{\mathbf{X}}, \vec{\mathbf{u}})$ for system (4.12) or $\dot{\vec{\mathbf{Z}}} = h(\vec{\mathbf{Z}}, \vec{\mathbf{u}_z})$ for system (4.15).
- 2. By construction, for the time-varying reference state $\overrightarrow{\mathbf{X}}_{ref}$ defined in section 4.2.4, there exists a control sequence $\overrightarrow{\mathbf{u}}_{ref}$ to track $\overrightarrow{\mathbf{X}}_{ref}$. It is $\overrightarrow{\mathbf{u}}_{ref} = \overrightarrow{\mathbf{0}} \forall t \ge 0$ i.e. do nothing. $\overrightarrow{\mathbf{X}}_{ref}$ and $\overrightarrow{\mathbf{u}}_{ref}$ are admissible and feasible.
- 3. As per the definition of the cost function J (4.36) the state cost at time t is:

$$l(\overrightarrow{\mathbf{X}}(t), \overrightarrow{\mathbf{u}}(t), t) = (\overrightarrow{\mathbf{\Omega}}(t) - \overrightarrow{\mathbf{\Omega}}_{ref}(t))^{T} [\mathbf{\Lambda}_{\overrightarrow{\mathbf{\Omega}}}] (\overrightarrow{\mathbf{\Omega}}(t) - \overrightarrow{\mathbf{\Omega}}_{ref}(t)) + (\overrightarrow{\mathbf{\Theta}}(t) - \overrightarrow{\mathbf{\Theta}}_{ref}(t))^{T} [\mathbf{\Lambda}_{\overrightarrow{\mathbf{\Theta}}}] (\overrightarrow{\mathbf{\Theta}}(t) - \overrightarrow{\mathbf{\Theta}}_{ref}(t)) + (\overrightarrow{\mathbf{\Theta}}(t) - \overrightarrow{\mathbf{\Theta}}_{ref}(t))^{T} [\mathbf{\Lambda}_{\overrightarrow{\mathbf{\Theta}}}] (\overrightarrow{\mathbf{\Theta}}(t) - \overrightarrow{\mathbf{\Theta}}_{ref}(t)) + (\overrightarrow{\mathbf{u}}(t) - \overrightarrow{\mathbf{u}}_{ref}(t))^{T} [\mathbf{\Lambda}_{\overrightarrow{\mathbf{\Theta}}}] (\overrightarrow{\mathbf{U}}(t) - \overrightarrow{\mathbf{u}}_{ref}(t)) + (\overrightarrow{\mathbf{u}}(t) - \overrightarrow{\mathbf{u}}_{ref}(t))^{T} [\mathbf{\Lambda}_{\overrightarrow{\mathbf{u}}}] (\overrightarrow{\mathbf{u}}(t) - \overrightarrow{\mathbf{u}}_{ref}(t))$$

Since l is quadratic and $[\Lambda_{\overrightarrow{\Omega}}]$, $[\Lambda_{\overrightarrow{\Theta}}]$, $[\Lambda_{\overrightarrow{\Theta}}]$ and $[\Lambda_{\overrightarrow{u}}]$ are positive semi-definite diagonal matrices with diagonal entries of equal value:

(a)
$$\forall t \ge 0, \ l(\overrightarrow{\mathbf{X}}_{ref}(t), \overrightarrow{\mathbf{u}}_{ref}(t), t) = 0.$$

(b) $\forall t \geq 0, \forall \vec{\mathbf{X}}(t) \in X$ with $\vec{\mathbf{X}}(t) \neq \vec{\mathbf{X}}_{ref}(t)$ and $\forall \vec{\mathbf{u}}(t) \in U$, $l(\vec{\mathbf{X}}(t), \vec{\mathbf{u}}(t), t) > 0.$

(a) and (b) satisfy assumption 3.8 in [26] page 53.

4. Finally there exists three continuous strictly increasing and unbounded functions α_1 , α_2 and α_3 such that $\alpha_1(0) = 0$, $\alpha_2(0) = 0$, $\alpha_3(0) = 0$ and such that the three inequalities (4.12) in [26] page 79 are satisfied.

<u>Proof</u> As per the definition of the cost function J (4.36), J is a quadratic function whose weights are positive semi-definite diagonal matrices with diagonal entries of equal value.

(a) From the definition of $[\Lambda_{\overrightarrow{\Omega}}]$, $[\Lambda_{\overrightarrow{\Theta}}]$, $[\Lambda_{\overrightarrow{\Theta}}]$ and $[\Lambda_{\overrightarrow{u}}]$ in section 4.3.7, one can define the real number $\Lambda = min(\Lambda_{\overrightarrow{\Omega}}, \Lambda_{\overrightarrow{\Theta}}, \Lambda_{\overrightarrow{\Theta}})$ and the matrix $[\Lambda] = \Lambda \cdot [\mathbf{Id}]_{\mathbf{n_r} \times \mathbf{n_r}}$.

Then by definition of l:

$$\forall t \geq 0, \forall \vec{\mathbf{X}}(t) \in X \text{ and } \forall \vec{\mathbf{u}}(t) \in U:$$

 $l(\vec{\mathbf{X}}(t), \vec{\mathbf{u}}(t), t) \geq \frac{1}{2} (\vec{\mathbf{X}}(t) - \vec{\mathbf{X}}_{ref}(t))^T [\mathbf{\Lambda}] (\vec{\mathbf{X}}(t) - \vec{\mathbf{X}}_{ref}(t))$
which leads to the construction of α_3 as $\alpha_3(x) = \frac{1}{2} \Lambda x^2$

(b) As per section 4.3.5.3, it is admitted that there exists a bounded domain of attraction around the reference trajectory in other words the origin of system (4.15). As per [12] and with the further assumption that the linearisation (4.16) is stabilisable on this local domain of attraction for the equilibrium point at the origin, the infinite horizon cost function can be bounded by a quadratic terminal cost which ensures the system ends up in a terminal region in the neighbourhood of the origin. This leads to the construction of $\alpha_2(x)$ as:

$$\alpha_2(\overrightarrow{\mathbf{X}}(t)) = \int_0^{T_f} l(\overrightarrow{\mathbf{X}}(t), \overrightarrow{\mathbf{u}}(t), t) dt + \overrightarrow{\mathbf{X}}^T(T_f)[\mathbf{P}] \overrightarrow{\mathbf{X}}(\mathbf{T_f})$$

where $[\mathbf{P}]$ is the constant matrix of the quadratic terminal cost.

(c) From the definition of $[\Lambda_{\overrightarrow{\Omega}}]$, $[\Lambda_{\overrightarrow{\Theta}}]$, $[\Lambda_{\overrightarrow{\Theta}}]$ and $[\Lambda_{\overrightarrow{u}}]$ in section 4.3.7, one can define the real number $\Lambda = min(\Lambda_{\overrightarrow{\Omega}}, \Lambda_{\overrightarrow{\Theta}}, \Lambda_{\overrightarrow{\Theta}})$ and the matrix $[\Lambda] = \Lambda \cdot [\mathbf{Id}]_{\mathbf{n}_{\mathbf{r}} \times \mathbf{n}_{\mathbf{r}}}$. Then by definition of J_{∞} :

$$\forall t \ge 0, \forall \vec{\mathbf{X}}(t) \in X \text{ and } \forall \vec{\mathbf{u}}(t) \in U:$$

$$J_{\infty}(\vec{\mathbf{X}}(t), \vec{\mathbf{u}}(t), t) > l(\vec{\mathbf{X}}(t), \vec{\mathbf{u}}(t), t).$$
Hence $J_{\infty}(\vec{\mathbf{X}}(t), \vec{\mathbf{u}}(t), t) \ge \frac{1}{2} (\vec{\mathbf{X}}(t) - \vec{\mathbf{X}}_{ref}(t))^T [\mathbf{\Lambda}] (\vec{\mathbf{X}}(t) - \vec{\mathbf{X}}_{ref}(t))$
which leads to the construction of $\alpha_1(x)$ as $\alpha_1(x) = \frac{1}{2} \Lambda x^2$

Providing the existence of a domain of attraction and the possibility of stabilising system (4.16), the above assumptions fulfil the theorem and ensure the existence of a suboptimal feedback control law which asymptotically stabilises system (4.15) at its origin and on the set $X = \mathbb{R}^3 \times ([0, \pi] \times [0, 2\pi[\times\mathbb{R}^2)^{n_r}]$. In plain terms, since system (4.15) is equivalent to system (4.12), it means that, for each robot module, a control sequence can be found for this module to track any reference trajectory i.e. to reach its deployment goal on the surface of the object while ensuring that the angular velocity of the system stays as close as possible to the reference angular velocity.

4.3.9 Observability of the State

A complete definition of observability can be found in [6] page 52. The state of a plant is observable if it can be determined using a finite observation sequence. A plant is fully observable if its complete state is observable.

The intersection of a plane and ellipsoid is an ellipse [4]. Drawing on this property and on our assumptions about the individual robot module in section 4.1.1, an estimation model combining the geometrical properties of the object with angular velocities, linear accelerations and odometry data can be derived to estimate the module's relative angular velocity and acceleration with respect to the object as well as the absolute angular velocity of the object. The details of such derivation is left for future work but a brief outline is presented here.

Assuming that the local radius of the ellipsoid is constant and equal to R given the relative size of the object and a robot module i, the linear velocities of this module $\begin{bmatrix} v_{\theta} \\ v_{\phi} \end{bmatrix}$ and its accelerations $\begin{bmatrix} a_{\theta} \\ a_{\phi} \end{bmatrix}$ expressed in the local spherical coordinate system are simply divided by the radius R to obtain the relative angular velocity and acceleration of the module i with respect to the objects as follows:

$$\begin{bmatrix} \dot{\theta}_i \\ \dot{\phi}_i \end{bmatrix} = \begin{bmatrix} \frac{v_{\theta}}{R} \\ \frac{v_{\phi}}{R} \end{bmatrix}$$
(4.38)

$$\begin{bmatrix} \ddot{\theta}_i \\ \ddot{\phi}_i \end{bmatrix} = \begin{bmatrix} \frac{a_\theta}{R} \\ \frac{a_\phi}{R} \end{bmatrix}$$
(4.39)

Once the relative angular velocity of module *i* is known as per formulae (3.37) in chapter 3 it is simply subtracted from the absolute angular velocity given by its gyroscope to obtain the absolute angular velocity of the object: $\vec{\Omega}_{ObjectMeas} = \vec{\Omega}_{AbsoluteMeas} - \vec{\Psi}_{Meas}$.

Alternatively or as a complement in a sensor fusion approach, if the sensory data of the other modules is available to the observing module, the already anchored modules gyroscope data could be used directly. As any already anchored module would not have any relative angular velocity with respect to the object, its gyroscope data would contain the rigid angular velocity i.e. the angular velocity of the object.

The state observer of each robot module i is assumed to be based on a likeness of the previous models (4.38) and (4.39) and capable of observing the rigid angular velocity of the object. For the purpose of this study, it was simply modelled as a full linear observer with added gyroscope noise as defined by model (3.62). Is follows that:

$$\vec{\mathbf{Y}}_{i} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \omega_{x} \\ \omega_{y} \\ \omega_{z} \end{bmatrix} + \vec{\mathbf{b}} + \vec{\mathbf{n}}$$
(4.40)

where $\overrightarrow{\Omega}_{(x,t)} = \begin{vmatrix} \omega_x \\ \omega_y \\ \omega_z \end{vmatrix}$ is the true angular velocity, $\overrightarrow{\mathbf{b}}$ is the gyroscope drift rate

bias, driven by the angular velocity random walk process and $\overrightarrow{\mathbf{n}}$ is a white noise affecting the measurements.

4.3.10 UKF: Unscented Kalman Filter

Kalman filters have two main applications [65]:

- they are used to optimally estimate the states of a system which can only be measured indirectly.
- 2. they are used to fuse and optimise multi-sensor measurements to best estimate the states of a system when noise is present.

Kalman filters are used in navigation and control problems and mainly for linear systems with quadratic cost functions for in this case, states estimates are provably optimal. Kalman filters are easy to implement and even work with nonlinear systems or systems with non-quadratic cost functions. In this cases, they usually take a linearised form called the Extended Kalman Filter (EKF). Compared to alternatives such as Hidden Markov Models (HMMs) and Particle Filters which are heuristics with much weaker provable performance guarantees, Kalman filters are far less computationally intensive [65].

As per section 4.3.9, the states of system (4.15) are observable but subject to a drift rate bias associated with a white noise coming from the gyroscope. Moreover the control problem is nonlinear with an associated quadratic cost function. Based on these two system features, a Kalman filter is an appropriate states estimation method.

Two choices were available for the implementation of this Kalman filter: the Extended Kalman Filter (EKF) or the Unscented Kalman Filter (UKF).

In essence, the EKF simply linearises nonlinear models so that a traditional linear Kalman filter could be used with this nonlinear model. However, it is difficult to implement and tune, and is only reliable for systems that are almost linear on the time scale of the update intervals. On the other hand, the UKF generalizes the EKF to nonlinear systems without a linearisation step and without restriction to Gaussian distributions. It provides more accurate estimation and better performance than the EKF for practically all applications [32].

In [31], the author exposes the difficulty to perform estimations for nonlinear

systems: "Estimation in nonlinear systems is extremely difficult. The optimal (Bayesian) solution to the problem requires the propagation of the description of the full probability density function (pdf). This solution is extremely general and incorporates aspects such as multimodality, asymmetries, discontinuities. However, because the form of the pdf is not restricted, it cannot, in general, be described using a finite number of parameters. Therefore, any practical estimator must use an approximation of some kind."

The limitations of the EKF come from the linearisation process which is only reliable if the error propagation is well approximated by a linear function. For this, the model's Jacobian matrix has to exist but its derivation and computation can be a very difficult and error-prone [31].

The UKF algorithm works as follows: it uses sample points of a Gaussian distribution, with a given mean and covariance, that are past through the nonlinear function of the model and then processed with a weighted linear regression to obtain the expected value and covariance of the estimated state. In doing so, it captures high-order information about the distribution with a fixed, small number of points [31]. Details of its algorithm and derivation can be found in [65] chapter 3.

As per [31], the UKF has a number of interesting properties:

- its algorithm can be use as a "black box". For any given model, it can calculate the predicted quantities for any given transformation.
- 2. its computational cost is of the same order of magnitude as the EKF.
- 3. it does not require the calculation of any derivatives since it includes the second-order "bias correction" term of the truncated second-order filter.
- 4. its algorithm can tackle discontinuous transformations.

Given that system (4.15) is highly nonlinear and for all the above reasons, the UKF was chosen for this study and was implemented in MATLAB using the *unscentedKalmanFilter* function.

4.3.11 Matlab Implementation

4.3.11.1 MPC in Matlab

Matlab uses two built-in QP (Quadratic Problem) solvers for performing the optimisation of the MPC control problem. These are:

- 1. The active-set solver uses the KWIK active-set algorithm which requires a positive definite Hessian matrix. This algorithm reaches a best solution by determining which constraints will influence the final result of the optimization and provides fast and robust performance for small-scale and medium-scale optimization problems.
- 2. The interior-point solver uses a primal-dual interior-point algorithm with Mehrotra predictor-corrector. This algorithm reaches a best solution by traversing the interior of the feasible region and provides superior performance for large-scale optimization problems, such as MPC applications that enforce constraints over large prediction and control horizons. Details can be found in [46].

Both these algorithms are used in a sequential fashion the details of their implementation can be found in [54]. The active-set solver was chosen for this study, as its empirical timescale evaluation proved fast enough.

4.3.11.2 NMPC in Matlab

By default, Matlab uses the fmincon function with the SQP algorithm. SQPs or Sequential quadratic programming are a class of algorithms used for solving reallife non-linear optimization problems because of their ability to tackle all degrees of non-linearity including in the constraints. SQPs combine two approaches: the active set method and Newton's method, the details of which can be found in [42]. SQPs require the explicit analytic calculation of model derivatives prior to iterating to a solution which constitute a further input to the optimisation problem under study whose general form is as follows:

$$\min f(x)$$

subject to $h(x) = 0$ (4.41)
and $g(x) \le 0$

where f(x), h(x), and g(x) could be non-linear functions of a vector x. [42]

In this study, the above default SQP algorithm was used to solve the NMPC problem with Matlab and the Jacobian of the model required to perform the optimisation was explicitly derived by the author in appendix B.

4.4 Robot Control Algorithm Design

Section 4.3 dealt with one of the essential part of the module's controller responsible for ensuring its deployment while tracking a reference state trajectory: the MPC/NMPC controller. However, both of these MPC controllers are used for the deployment subtask when the module is specifically deploying on the surface of the object. The modules has to tackle other tasks before and after this subtask. The design of a basic overall module's controller encompassing all the tasks it has to perform is the subject of this section.

4.4.1 Task-Driven Self-Reconfiguration Approach

As per engineering objectives in sections 4.1.1 and 4.1.2, there are no requirement on the shape of the robot at the end of its deployment. It is only required to be wound around the object about its main spin axis until it reaches the object's where its centre of mass and its other principal axes lie i.e. at spherical coordinate $\theta = \frac{\pi}{2}$. Consequently, the robot reconfiguration falls into the category of task-driven reconfigurations. The decentralised approach to the robot controller means that the design is shifted to the module's controller. Each module seeks to complete its individual deployment objective leading to the emergence of the robot shape and to completion of the robot's task. Based on the robot's task and its environment a behaviour-based controller approach was taken for the controller design.

4.4.2 Behaviours Based Algorithm Design Principles

Following a "from simple to the complex" design pattern with an emphasis on parallelism, the general guidelines under which the algorithm was designed can be found in [59] page 313:

- 1. "Use behaviours as the building block of both decision-making and action execution processes;
- 2. Use distributed parallel evaluation and concurrent control over lower-level behaviours, which take real time inputs from sensory data and send real-time commands to effectors;
- 3. Have no centralized components, each module carrying out its own responsibilities."

4.4.3 Robot Behaviours Based Algorithm

In figures 4.6, 4.7 and 4.8 below is shown the behaviour tree designed for the robot's modules controller broken into three sub-diagrams. It uses the py trees python library, the API of which can be found along with the definition and the function of each of the behaviour listed in the legend in the documentation at this URL [2].

The root node runs in parallel the three fundamental behaviours of a robot module:

- 1. Listening continuously to its own sensors data and the communication data from the other modules.
- 2. Helping other modules to move or requesting help from other modules to move itself or another module (e.g. if the module is trapped in a lattice for instance).

3. Deploying itself until it finds the optimal place to anchor itself at the surface of the object.

In section 4.1.1, the operating assumptions 3 states that if a module is faulty, it can be detected, released and replaced in the limit of number availability. Since the focus of this study was to analyse the physical effect of reconfiguration rather than the details of its implementation and failure management, it was assumed that the "Help Mode" behaviour enables each robot module to successfully perform collaborative manoeuvrers in order to:

- 1. release itself or other modules when enclosed in the starting lattice,
- 2. remove obstacles, i.e. faulty modules, by releasing them in space and
- 3. keep a clear path at all times for each module to deploy.

Moreover, catastrophic failures, where a large number of module fail for instance, were also ignored and assumed never to materialise: the robot is always able to deploy itself.



Figure 4.6: Robot Algorithm Top Level.



Figure 4.7: Robot Algorithm Initialisation.



Figure 4.8: Robot Algorithm Deployment.

4.4.4 Correctness of the Controller

In this section, the correctness of the module's controller algorithm is briefly examined alongside its termination. Correctness is defined as per [15] page 6: "for every input instance, the algorithm halts with correct output".

4.4.4.1 Initialisation Sequence

If at any point of the initialisation sequence a FAILURE status or the absence of a status is detected the initialisation node returns FAILURE and publishes its status to all the other modules if it can. If it cannot publish, the other modules that are initialised detect this absence of communication and identify the faulty module's position via an elimination process. In any case, the controller algorithm returns the correct output: FAILURE and terminates.

If the initialisation of the module is successful, the behaviour "Initialisation" returns and locks a SUCCESS status.

4.4.4.2 Principle of Deployment

The main idea behind the module deployment is a one-by-one approach:

- 1. Each module gets released from the original lattice from the top layer to the bottom layer and from the outside of the top plane to the inside of the top plane.
- Each module then travels down to the surface on the side of the lattice to reach the very near position of the first anchored module. This is possible if the first anchored module is both on the bottom plane and at least on one of its edges.
- 3. Each module then proceeds to move on top of the already deployed and anchored modules to reach the end of the chain.
- 4. Each module then uses its MPC/NMPC controller to reach its final goal location on the surface of the object and finally anchors.

4.4.4.3 Deployment: Getting out of the Lattice

In a lattice form the robot module has up to six connections, two per direction. If three connections or less are used, the module is ready to be deployed. The behaviour "Is it My Turn" returns SUCCESS when confirmation that no other module will deploy has reached the module.

The module can then proceed to the next sequence "Reach Starting Point". The moving module travels first on the top plane to reach the side where the first anchored module is. Then it follows down a path on the side plane to the first anchored module.

At every point it relies on the "Can I Deploy" behaviour to detect failed modules and achieve a viable route via collaboration. It returns a RUNNING status if a problem is encountered until this problem is solved and returns SUCCESS. Until the module reaches the first anchored module, "Deploy to Anchored Module" returns RUNNING and upon reaching it, it returns SUCCESS.

"Reach Starting Point" then returns SUCCESS. The module then moves on the next deployment sequence "Follow the Chain".

4.4.4.4 Deployment: Follow the Chain

The "Follow the Chain" behaviour uses the same "Can I Deploy" behaviour as above, to tackle encountered module failures. "Can I Deploy" is in a sequence with "Follow Chain of Anchoring Behaviour" which returns RUNNING until the module reaches the end of the chain upon which it returns SUCCESS. The module then moves on to the surface for the final phase of deployment.

4.4.4.5 Deployment: Go On the Surface and Anchor the Chain

"Get on Surface" returns RUNNING until the module is on the surface of the object. The behaviour then returns SUCCESS.

With the "MPC Controlled Deployment" behaviour, the MPC/NMPC controller is triggered for the module to reach the target deployment goal point where it will anchor. As per section 4.3.8, for both the cases of the MPC and NMPC controllers, the existence of a control sequence that ensures success is likely and assumed. Further assuming the optimisation procedure finds such a control sequence, the module reaches the goal and the behaviour returns SUCCESS.

The last behaviour "Anchoring" ensures that the deploying module connects to the previously anchored module. The module then sends various messages to all the other modules informing them of its successful deployment allowing another module to deploy. If possible, the module then broadcasts all its sensory inputs and allows power sharing before returning SUCCESS.

4.4.4.6 Conclusion

As per sections 4.4.4.3, 4.4.4 and 4.4.4.5, the sub-behaviours of the "Deployment" behaviour return SUCCESS sequentially which leads "Deployment" to return SUCCESS. This ends the deployment of a single module at its optimal target goal position. The control algorithm is correct.

4.5 Experimental Setup

In this section are detailed the actual values of the parameters used in the simulations. These parameters are:

- <u>Number of Modules</u>: 10
- Optimisation Cost Function Weights:
 - 1. $\Lambda_{\overrightarrow{\Omega}} = 1 \ \Lambda_{\overrightarrow{\Theta}} = 1 \ \Lambda_{\overrightarrow{\Theta}} = 0 \ \Lambda_{\overrightarrow{u}} = \frac{1}{2}$
 - 2. $\Lambda_{\overrightarrow{\Omega}} = 1 \ \Lambda_{\overrightarrow{\Theta}} = 1 \ \Lambda_{\overrightarrow{\Theta}} = 1 \ \Lambda_{\overrightarrow{u}} = 1$
- <u>Prediction Horizon</u> = 20 steps
- Modules' Target Location in Spherical Coordinates:

 $\begin{bmatrix} \theta \\ \phi \end{bmatrix} = \begin{bmatrix} 0.1745 & 0.3491 & 0.5236 & 0.6981 & 0.8727 & 1.0472 & 1.2217 & 1.3963 & 1.5708 \\ 0.3491 & 0.6981 & 1.0472 & 1.3963 & 1.7453 & 2.0944 & 2.4435 & 2.7925 & 3.1416 \end{bmatrix}$

• End Condition for a Module's Deployment: θ and ϕ to be within 0.01*rad* i.e.

 5.7° of their respective targets to ensure termination.

- Optimisation Constraints: None
- Normalised Process, Measurement and Disturbance Noise: σ ∈ [0; 0.05; 0.1; 0.2].
 Actual values of gyroscope bias and noise data can be found in table 3.62 and in [33].

Γ٦

Г

• Ellipsoids' Normalised Dimensions:
$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{10} & \frac{8}{10} & 10 & \frac{10}{8} & 2 & \frac{1}{10} & \frac{1}{3} & \frac{8}{10} \\ 1 & \frac{1}{2} & \frac{1}{10} & \frac{8}{10} & 10 & \frac{10}{8} & 2 & \frac{8}{10} & \frac{1}{2} & \frac{9}{10} \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

- Mass Ratio of the Object's to the Robot's: $MR \in [10, 100, 1000, 10000]$
- Initial Normalised Spin Rate for Symmetric Objects: $\omega_{Z_{init}} \in [0.5; 0.75; 1]$
- Initial Condition for Asymmetric Body Parameter: $\frac{h^2}{2T} \in [I_{min}, \frac{(I_{min}+I_{mid})}{2}, I_{mid}, \frac{(I_{mid}+I_{max})}{2}, I_{max}]$ with moments of inertia calculated as per section 3.5.1.1.
- Reference Angular velocity Value of a Common Asteroid: $\omega_0 = 0.02094 rad^{-1}$.

4.6 Conclusion

In this chapter, the engineering objectives and the basic design of the robot have been presented. The focus was placed on designing a state space model for simulating the robot's deployment on an object and an individual module controller at two levels. The first atomic level dealt with the design of a MPC/NMPC controller based on a state space model and destined to ensure that the module's deployment on the surface of the object occurs while tracking a reference angular velocity for the object. The second level dealt with the design of a behaviour-based controller aimed at ensuring the task would be performed while including some basic failure management. Finally the values of the parameters of the simulation were given. In chapter 5 will be presented the results of the simulations of the deployment of only one module.
Chapter 5

Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

5.1 Objectives and Simulations Parameters

This chapter is dedicated to the simulation, verification and validation and empirical performance evaluation of the robot's low-level module controller and, for this purpose, focuses on the deployment of one module only. The evaluation of the full robot deployment was left for future work.

5.1.1 Objectives

As per section 1.2, the low-level module controller's objective is to track a reference trajectory using a MPC controller. This reference trajectory includes the target anchoring location of the module, its surface velocity encoded in spherical coordinates and a target object's angular velocity. This controller objective can be further spilt into more precise and quantifiable objectives. These objectives are the following:

1. The deployment of the module on the surface of the object from the starting point $(\theta, \phi) = (0, 0)$ at the tip of the Z axis (in spherical coordinates) to its target location on the object at $(\theta, \phi) = (\frac{\pi}{2}, \pi)$ (in local spherical coordinates).

- 2. The deployment of the module along a surface trajectory coiling around the Z axis, i.e. this trajectory should describe a whole revolution about the Z axis. This objective is implied by the preceding one but deserves to be explicitly stated to guide a heuristic evaluation of the controller's performance. As a structure intended to be used by other devices to exploit or manipulate the object, it is important that all points of the object's surface is reachable from the structure and ideally within a reasonable average distance from it. In this study, this objective did not feature explicitly in the controller's design and left for future work. However, a heuristic evaluation of the shape of the module trajectory on the surface was performed by visual inspection in order to assess what further work should be carried out to tackle this aspect.
- 3. The tracking and ideally the convergence towards a reference object's rotational state which is either a pure state of spin or is despun. Both of these states are optimal from the point of view of the maintenance of the pointing direction of the axis of rotation which would be a requirement for future operations on the object such as its retrieval.
- 4. The delivery of the above objectives within a time frame which is compatible with the timescale of the entire system.

5.1.2 Scenarios and Simulations Parameters

This section details all the simulations parameters i.e. the simulations scenarios and all the MPC controller design parameters.

5.1.2.1 Simulations

The simulations sought to span the various situations that the robot would encounter. As such, the simulations depended on the following parameters:

1. The shape of the object the robot would deploy on which were chosen to be ellipsoids parametrised by there normalised semi-axis lengths. As per table 5.2, for all mass ratios, the shape instability parameters is $\sqrt{\frac{m}{m+1}} \ge 0.8165$. For all the ellipsoids chosen for the simulations, the system is potentially stable but close to the instability threshold in two cases. The chosen shapes and their stability status are:

Ellipsoid Type	a	b	c	System Stability Status
Prolate	$\frac{1}{10}$	$\frac{1}{10}$	1	Potentially Stable
Prolate	$\frac{1}{2}$	$\frac{1}{2}$	1	Potentially Stable
Prolate	$\frac{8}{10}$	$\frac{8}{10}$	1	Potentially Stable (Close to Unstable)
Sphere	1	1	1	Potentially Stable
Asymmetric	$\frac{1}{10}$	$\frac{8}{10}$	1	Potentially Stable
Asymmetric	$\frac{1}{3}$	$\frac{1}{2}$	1	Potentially Stable
Asymmetric	$\frac{8}{10}$	$\frac{9}{10}$	1	Potentially Stable (Close to Unstable)
Oblate	$\frac{5}{4}$	$\frac{5}{4}$	1	Potentially Stable
Oblate	2	2	1	Potentially Stable
Oblate	10	10	1	Potentially Stable

Table 5.1: Ellipsoids Normalised Semi-Axes Lengths

2. The mass ratio of the object's mass to the robot's or single module's in this case. The simulated values were:

Mass Ratio	10	100	1000	10000

Table 5.2: Mass Ratios: Object to Module

- 3. The initial level of energy of the object with respect to its inertia was fixed at a level found in the environment. Its kinetic energy and moments of inertia were normalised. The angular velocity used for normalisation is the one sized in section 3.5.1.2: $\omega_0 = 0.02094 \ rad \cdot s^{-1}$.
- 4. The initial rotational state condition (i.e. the degree of nutation or tumbling) depended on the geometry of the ellipsoid. For symmetric ellipsoids, the proportion of the total angular velocity allocated to the spin axis component was parametrised by $\omega_{Z_{init}}$ whose values were:



Table 5.3: $\omega_{Z_{init}}$

For asymmetric ellipsoids, the rotational state was parametrised by the ratio of the angular momentum to the rotational kinetic energy $\frac{h^2}{2T}$ which can only take values in the set of possible moments of inertia of the objects. Consequently, for asymmetric ellipsoids, $\frac{h^2}{2T}$ values were chosen with respect to the object's moments of inertia as follows:



Table 5.4: Asymmetric Bodies Parameters $\frac{h^2}{2T}$

5.1.2.2 Controller

The controller uses a Model Predictive Control approach implementing both the nonlinear model (4.12) and its linearised version (4.16) which can be found in chapter 4. For both the linear and nonlinear cases, the controller's design parameters are the same. These parameters are listed below:

- 1. <u>Reference Trajectory</u>: The reference trajectory to be tracked described the system in a final ideal state with an object in a pure state of spin or despun. This was parametrised by the object's angular velocity state components which were respectively $(0, 0, \omega_{Z_{Final}})^T$ or $(0, 0, 0)^T$. The other reference states were the target anchoring location parametrised in spherical coordinates $(\theta_{Final}, \phi_{Final}) = (\frac{\pi}{2}, \pi)$ and a nil final angular velocity $(\dot{\theta}_{Final}, \dot{\phi}_{Final}) = (0, 0)$.
- 2. <u>Optimisation Cost Function Weights</u>: $\Lambda_{\vec{\Omega}} = 1 \Lambda_{\vec{\Theta}} = 1 \Lambda_{\vec{\Theta}} = 1 \Lambda_{\vec{\Theta}} = 1$
- 3. <u>Prediction Horizon</u> = 20 steps. This number of steps is quite large and for a numerical application a good enough approximation of an infinite time horizon.
- 4. End Condition for a Module's Deployment: θ and ϕ to be within 0.01 rad i.e. 5.7° of their respective targets to ensure termination.
- 5. <u>Optimisation Constraints</u>: No optimisation constraints were implemented. In other words, neither an end penalty cost nor a constraint function were implemented. These were left for future work.

- 6. <u>Perturbations and Measurement Errors</u>: Perturbations and measurements and actuation errors, modelled in section 3.5.6, were not implemented. The implementation was left for future work. As a reminder and for reference only, the values to be used are:
 - (a) The YORP effect coefficient value $\alpha_2 = 2 \cdot 10^{-16} \ rad \cdot s^{-2}$ with a standard deviation of the coloured noise $\sigma = 10^{-17} \ rad \cdot s^{-2}$ (see section 3.5.6.3 for details)
 - (b) The gyroscope bias and noise values which can be found in table 3.5 with the white noise standard deviations for the gyroscope model (3.62): σ ∈ [0; 0.05; 0.1; 0.2].

5.1.3 Verification and Validation Metrics

In this chapter, the performance, verification and validation metrics defined in 4.1.4 are used for the analysis of the above objectives and to evaluate whether future work is worth being carried out. As a reminder, these performance metrics are:

- 1. Achievement of task:
 - (a) Did a single module manage to go from point $(\theta, \phi) = (0, 0)$ to point $(\theta, \phi) = (\frac{\pi}{2}, \pi)$ on the surface of the ellipsoid following a continuous trajectory: True or False
 - (b) Did the n_r-module robot achieve full deployment with each module i being laid continuously one after the other from point (θ, φ) = (0,0) to its target anchoring point at (θ_i, φ_i) = (i^π/_{2(n_r-1)}, i^π/_{n_r-1})_{i∈[0,n_r-1]} on the surface of the ellipsoid: True or False
- 2. Stability and controllability of the task:
 - (a) Measurement of the nutation angle $\nu = \arctan\left(\frac{\sqrt{[\mathbf{I}_X]^2 \omega_X^2 + [\mathbf{I}_Y]^2 \omega_Y^2}}{|[\mathbf{I}_Z] \omega_Z|}\right)$ according to [60] page 98. If ν increases, the rotational state of the object is less stable.

- (b) Measurement of the error vector $\overrightarrow{\varepsilon}_t$ such that $\overrightarrow{\Omega}(t) = \overrightarrow{\Omega}_{ref}(t) + \overrightarrow{\varepsilon}_t$ as per chapter 3 section 3.4.5. $\overrightarrow{\varepsilon}_t$ should ideally converge and stay as close to $\overrightarrow{\mathbf{0}}$ as possible. Empirical measurements will include the phase space orbits of the object's rotational state. In particular, it will be examined whether:
 - i. $\overrightarrow{\varepsilon}_t$ is increasing, decreasing or periodic respectively indicating instability, asymptotic stability or neutral stability.
 - ii. The controlled system can converge to any reference state in particular whether the object can converge to a state of pure spin or despin.
 - iii. The model parameters influence the stability and controllability and if so in which proportion.
- (c) Measurement of the rigid or body rotational kinetic energy: $\frac{1}{2} \vec{\Omega}_t^{T^T} [\mathbf{I}_{\mathbf{Object}}] \vec{\Omega}_t^{\mathbf{1}}$ to evaluate the state and stability of the object in conjunction with the nutation angle. Energy dissipation combined with increased nutation angle is indicative of nutation instability and of a state degenerating towards rotation about the major axis [60].
- 3. Duration of the task:
 - (a) Comparison between the timescale of the controller and of the object's rotational dynamics. Is the controller timescale appropriate: True or False
 - (b) Measurement of the controller timescale for each controller type to evaluate which is the fastest.
 - (c) Measurement of the controller timescale against the model parameters to evaluate their respective influence on this timescale

5.1.4 Data Processing

The data was organised into six types of graphs:

- 1. The trajectory of the module on the surface of the object.
- 2. The object's phase diagram of the Z component of the angular velocity. The phase diagrams plot the Z component of the angular acceleration of the object against its Z angular velocity component, i.e. the orbit of the Z angular velocity component. If the orbit is a closed curve then the object is stable as per definition page 7 in [68]. The shape or the orbit is also indicative of the magnitude of the deviation of the angular velocity from the target angular velocity.
- 3. The object's rotational kinetic energy vs. time. On these graphs, the variations of the rotational kinetic energy of the object normalised with respect to the angular velocity sized in 3.5.1.2 is plotted against time, highlighting the time evolution of the energy transfers between the object and the module moving at its surface.
- 4. The object's nutation angle vs. time. In these graphs, the time evolution of the nutation angle is plotted. If the rotation axis is not the Z axis the nutation angle measures a deviation around $-\frac{\pi}{2}$ or $\frac{\pi}{2}$.
- 5. The nutation angle vs rotational kinetic energy. In these graphs, the relationship between the nutation angle and object's kinetic energy is plotted to analyse in particular whether an increase of object's rotational kinetic energy leads to its rotational state to be closer to a spin and reciprocally.
- 6. Control Commands vs. time. These graphs plot the magnitude of the module's control commands in spherical coordinates normalised with respect to the angular velocity sized in 3.5.1.2. They indicate the direction of travel prescribed to the module by its MPC controller as well as the level of energy expenditure required over time from the module's internal battery. The change in the module's angular momentum should be consistent with the energy transfers to and changes of angular momentum of the object. The control commands are sized with respect to an object with an initial angular velocity of magnitude $\omega_0 = 0.02094$.

5.1.5 Organisation of Results

The verification and validation of the low-level controller's simulation data was performed using the performance measures laid out in section 4.1.4. Apart from section 5.6, the results were produced using the linear control law calculated from the linearised model (4.16). In section 5.6, the linear law and nonlinear control law are compared. Only a subset of the results are displayed owing to similarities which are exposed before drawing overall conclusions.

The results were organised in different sections as follows:

- 1. In the first section 5.2, the linear and nonlinear control laws are compared over a large period of time for mass ratio equals to 10,000 and for the following object's geometries:
 - (a) The sphere a = 1, b = 1, c = 1.
 - (b) The symmetric ellipsoid a = 2, b = 2, c = 1
 - (c) The symmetric ellipsoid $a = \frac{1}{2}, b = \frac{1}{2}, c = 1$
 - (d) The asymmetric ellipsoid $a = \frac{1}{10}, b = \frac{8}{10}, c = 1$
 - (e) The asymmetric ellipsoid $a = \frac{8}{10}, b = \frac{9}{10}, c = 1$

Although the control law and gains were not explicitly derived, the derivation being left for future work, this comparison gives an insight into the performance and pertinence of the linearised versus the nonlinear model and the observed behaviour of the system.

2. In the second section 5.3, the trajectory of the module on the ellipsoidal object is examined. In particular it is checked whether the target location is reached, whether the trajectory circles about the main axis of spin and, upon a heuristic visual inspection, whether every point of the object's surface is within a reasonable distance (defined in the section) from a point of the module's trajectory. The velocity vectors are also displayed on the module's trajectory to show the direction of travel and the velocity's magnitude over time. This analysis was performed for all object's geometries.

- 3. In the third section 5.4, the requirements to asymptotically stabilise the object's angular velocity to a pure state of spin or a despun state are verified in the worst case scenario where, for all shapes of ellipsoid and all mass ratios, the object starts in an initial tumbling or highly nutated rotational state and the module travels from the tip of the Z axis to its target location where it stops. For each of these cases, the object's spin phase diagram of angular velocity about Z, the object's kinetic energy vs. time, the object's nutation angle vs. time and the nutation angle vs kinetic energy are then analysed to verify whether the rotational state of the object converged or is converging to a stable pure state of spin or despun state.
- 4. In the fourth section 5.5 a deeper stability analysis is performed. The module is no longer constrained to stop at its target location and is left free to move on the surface of the object with the sole objective of tracking a pure state of spin or a despun state for the object. The aim is to analyse the system's behaviour beyond its specification to understand empirically its timescale and whether or not it is reasonable to assume the existence of a domain of attraction, albeit small, near the origin. The geometry chosen for this analysis were:
 - (a) The sphere a = 1, b = 1, c = 1.
 - (b) The symmetric ellipsoid a = 2, b = 2, c = 1
 - (c) The symmetric ellipsoid $a = \frac{1}{2}, b = \frac{1}{2}, c = 1$
 - (d) The asymmetric ellipsoid $a = \frac{1}{10}, b = \frac{8}{10}, c = 1$
 - (e) The asymmetric ellipsoid $a = \frac{8}{10}, b = \frac{9}{10}, c = 1$

For each of these ellipsoids and all mass ratios, the object's spin phase diagram of angular velocity about Z, the object's kinetic energy vs. time, the object's nutation angle vs. time and the nutation angle vs kinetic energy are then analysed to verify whether the rotational state of the object converged or is converging to a stable pure state of spin or despun state over time.

5. In the fifth and final section 5.6, the linear and nonlinear MPC controllers' performance are compared by analysing again the object's spin phase diagram

of angular velocity about Z, the object's kinetic energy vs. time, the object's nutation angle vs. time and the nutation angle vs kinetic energy for the following object's geometries:

- (a) The sphere a = 1, b = 1, c = 1.
- (b) The symmetric ellipsoid a = 2, b = 2, c = 1
- (c) The symmetric ellipsoid $a = \frac{1}{2}, b = \frac{1}{2}, c = 1$
- (d) The asymmetric ellipsoid $a = \frac{1}{10}, b = \frac{8}{10}, c = 1$
- (e) The asymmetric ellipsoid $a = \frac{8}{10}, b = \frac{9}{10}, c = 1$

The analysis was performed for each of these geometries with mass ratios equal to 1,000 and 10,000 and for all initial rotational state conditions. The module was not constrained to stop at its target location and was left free to move on the surface of the object with the sole objective of tracking a pure state of spin for the object.

In section 5.7, final conclusions will be drawn from the commentaries made on the presented results in all the preceding subsections.

5.2 Control Commands: A Comparison

This section deals with the analysis of the control commands ($\ddot{\theta}$ and $\ddot{\phi}$) produced by both the linear and nonlinear MPC controllers.

5.2.1 Origin of the Data

The data displayed in this section corresponds to the following specific scenario:

- 1. Both MPC controllers track a pure state of spin for the object with a target spin rate $\omega_0 \neq 0$ for the angular velocity part of the system state.
- 2. The initial rotational state of the object corresponds to the worst case scenario where symmetric objects nutate with $W_{zinit} = 0.5$ and asymmetric objects tumble with $\frac{\hbar^2}{2T} = I_{mid}$, where I_{mid} is the moment of inertia about the object's unstable axis.

- 3. The module is not constrained to stop at its target location and is left free to move on the surface of the object with the sole objective of tracking a pure state of spin for the object. The aim is to analyse the system's behaviour near the origin over a large timescale.
- 4. The mass ratio of the object's to the module is fixed at 10,000.

5.2.2 Data Description

The data is displayed for specific shapes representative of the whole spectrum of possible situations encountered by the robot. These shapes encompass a sphere, one oblate, one prolate and one asymmetric ellipsoids which are:

Ellipsoid Type	a	b	c	System Stability Status
Prolate	$\frac{1}{2}$	$\frac{1}{2}$	1	Potentially Stable
Sphere	1	1	1	Potentially Stable
Asymmetric	$\frac{1}{10}$	$\frac{8}{10}$	1	Potentially Stable
Asymmetric	$\frac{8}{10}$	$\frac{9}{10}$	1	Potentially Stable (Close to Unstable)
Oblate	2	2	1	Potentially Stable

Table 5.5: Ellipsoids Normalised Semi-Axes Lengths

No control gains or control commands function calculations were performed and this was left for future work. All the results obtained across the simulation parameters space were similar. Therefore, all descriptions and commentaries made on the data displayed below remain valid:

- 1. For all ellipsoidal shapes
- 2. For all initial conditions of the object's rotational state.
- 3. For all mass ratios MR
- 4. Whether the objective is to track a pure state of spin or a despun state.
- 5. For both the linear and nonlinear MPC controller.
- 6. For potentially stable and unstable systems.



Figure 5.2: Linear Results for $\ddot{\phi}$

Figure 5.4: Non Linear Results for ϕ

Figure 5.5: Comparison Between the Control Commands Generated by the Linear and Nonlinear Controllers

From figure 5.5, the following observations were made:

- 1. The timescale of the linear control law in figures 5.1 and 5.2 is 10 times faster than for the nonlinear control law in figures 5.3 and 5.4.
- 2. However, for each ellipsoid and after a finite amount of time, the linear control law commands diverge leading the object's angular velocity part of the system's state to diverge too as will be seen in section 5.5. The nonlinear control commands all converge to 0 over time as can be expected of a regulator.
- 3. The shape of the control command curves are identical for θ and φ for both the linear and nonlinear control laws. The magnitude of the angular acceleration produced by the linear control law is of the order of 100 rad · s⁻² which is extremely large and physically unrealistic. The magnitude of the angular acceleration produced by the nonlinear control law goes up to a maximum of 0.012 rad · s⁻² which is perfectly realistic both from a physical and engineering point of view in terms of energy storage and hardware capabilities. For both the linear and nonlinear control laws, the magnitude of φ is twice that of θ.
- 4. The smaller the moment of inertia about the minor axis compared to the other moments of inertia, the longer it takes for the angular velocity state to diverge under the input of the linear or nonlinear feedback control law.

5.2.3 Conclusion

The linear control law is much faster than the nonlinear control law but does not deliver satisfactorily on the objectives set out for the controller. This will be confirmed and expanded upon in later sections. The above data calls for the following conclusions and comments:

1. The low-level controller is designed to be a regulator tracking the equilibrium at the origin. However, the linear control law is not stabilising as instability sets in over time. A higher order analysis (2^{nd} order and higher) into the nature of the system's equilibrium would be beneficial to inform the control law design further. From the available data, using the nonlinear law only or a combination of the linear and nonlinear control law is a priori the only options for the controller.

- 2. The nonlinear control law performs better in terms of delivering a realistic set of control commands despite a longer timescale. The nonlinear controller takes two to seven days to converge the state of the system to the reference state against about 15 minutes for the linear control law. This timescale is acceptable with respect to the timescale of the system, i.e. of the robot deployment on the object. Indeed, over this timespan perturbations experienced by asteroids are negligible. Only the nonlinear model provides a sufficient level of prediction accuracy required to produce a performant control law by the controller.
- 3. From a pure empirical observation, it is not possible to know whether the control law is linear or nonlinear. However, the proportionality observed between ö and θ, the magnitude of ö is twice that of θ, suggests that, the dimension of the control space is 1. The respective set of values φ ∈ [0, 2π] and θ ∈ [0, π] are browsed at a comparable pace. This coupling is consistent with the lack of controllability of the system determined in chapter 4. As will be seen in the next section, it has a significant impact on the shape of the module trajectory on the surface of the object and in particular whether it coils about the rotation axis in a manner that would make all parts of the object's surface easily reachable from the structure. Future work should endeavour to determine whether tuning could be achieved to influence the shape of the module surface trajectory and whether the optimal control law is possibly linear and constant in time which would render its offline calculation possible.
- 4. Finally, for the feedback loop system with either the linear or nonlinear control law, the smaller the moment of inertia about the minor axis compared to the other moments of inertia, the more stable the system is. In practice, for symmetric ellipsoids, this means that stability decreases with oblateness and

increases with prolateness. For asymmetric ellipsoids, stability increases with a decreasing minor axis moment of inertia.

In the next section, the trajectory of the module on the surface of the object will be analyse.

5.3 Surface Trajectory and Reaching the Target Location

This section deals with the analysis of the trajectory of the module on the surface of the object.

5.3.1 Origin of the Data

The data displayed in this section corresponds to the following specific scenario:

- 1. The MPC controller is tracking a pure state of spin for the object with a target spin rate $\omega_0 \neq 0$.
- 2. The initial rotational state of the object corresponds to the worst case scenario where symmetric objects nutate with $W_{zinit} = 0.5$ and asymmetric objects tumble with $\frac{\hbar^2}{2T} = I_{mid}$, where I_{mid} is the moment of inertia about the unstable axis.

All the results obtained across the simulation parameters space were similar. Therefore, all descriptions and commentaries made on the data displayed below remain valid:

- 1. For all ellipsoidal shapes
- 2. For all initial conditions of the object's rotational state.
- 3. For all mass ratios
- 4. Whether the objective is to track a pure state of spin or a despun state.
- 5. For both the linear and nonlinear MPC controller.

6. For potentially stable and unstable systems.

5.3.2 Data Description

The data is displayed in two parts. The first set of graphs (figures 5.12) show all the symmetric ellipsoid cases and the second set of graphs show (figures 5.17) the asymmetric cases along with the sphere limit case.

In what follows, coverage is defined as a "visual inspection" heuristic measuring the ability of a device to reach, from the deployed robotic structure, any subsection of an object's surface. Dividing each object's surface into 18 contiguous sections separated by meridians every 20° as per graphs below, a good coverage should correspond to a situation where the module trajectory passing through two consecutive meridians is not very short. In other words, the module should move away from the tip of the Z axis as quickly as possible and then coil about the Z axis to its anchoring target.

Looking at figures 5.6 to 5.16 below, in all cases:

- 1. The controller always achieved the deployment objective. The module always reached and stopped at the target anchoring location.
- 2. The module trajectory is not dependent on the mass ratio MR. For all the cases displayed, the trajectories are identical and appear one on top of the others.
- 3. As per figure 5.14, for limit cases such as (a = 0.8, b = 0.9, c = 1) where the object has a geometry close to one for which the system is unstable, the deployment objective is achieved.
- 4. Upon visual inspection, in nearly every cases, the module trajectory covers only half of the meridians of the object's top hemisphere. The target location at $(\frac{\pi}{2}, 2\pi)$ has been chosen so as to obtain a module trajectory providing coverage of the entire surface of the object. As such, the module trajectory is expected to cover all meridians of the object's surface about the Z axis. The observed trajectories coil about the Z axis in two stage. The first stage corresponds to

the case $\phi \in [0, \pi]$, each trajectory coils very near the object's pole as if the module was rotating on the spot. The second stage corresponds to $\phi \in]\pi, 2\pi]$ where the module really starts moving from the pole to the median plane. The module controller only provides half the intended coverage.

- 5. The module trajectory depends on the shape of the object. Upon visual inspection, prolate objects have a better overall surface coverage than oblate ones. In figure 5.6, for the prolate ellipsoid $(a = \frac{1}{10}, b = \frac{1}{10}, c = 1)$, the trajectory moves away from the pole early on, enabling a coverage on all the meridians. In figure 5.11, for the oblate ellipsoid (a = 10, b = 10, c = 1), the trajectory coils tightly very near the object's pole to then descend over one single meridian, i.e. "straight down" without coiling, leaving less of half of the object's surface reachable.
- 6. As shown by the arrows on the trajectories, for the linear control law, the target anchoring location is reached with a significant residual velocity. Similarly, for the nonlinear control law, the target anchoring location is reached but with a residual velocity of four orders of magnitude less.





Figure 5.6: a=0.1 b=0.1 c=1



Module Surface Trajectory with Direction of Travel



MR=10000 MR=1000 MR=100 MR=10 1 0.5 Z (Normalised) 0 -0.5 -1 1 0.5 1 0 0.5 0 -0.5 -0.5 -1 -1 Y (Normalised) X (Normalised)

Figure 5.7: a=0.8 b=0.8 c=1

Figure 5.10: a=1.25 b=1.25 c=1



Figure 5.8: a=2 b=2 c=1

Figure 5.11: a=10 b=10 c=1

Figure 5.12: Module Surface Trajectory with Direction of Travel for Symmetric Ellipsoids



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.14: a=0.8 b=0.9 c=1

Figure 5.16: Sphere

Figure 5.17: Module Surface Trajectory with Direction of Travel for Asymmetric Ellipsoids and the Spherical Stability Limit Case

5.3.3 Conclusion

The following conclusions were drawn:

- 1. The deployment objective at the target anchoring location is always achieved and is achieved independently from the mass ratio of the object's to the module's. The angular velocity and target anchoring location objectives are coupled through the cost function only but have different timescales which is shown by the fact that there is a residual angular velocity at the target anchoring location while overtime the nonlinear control commands tend to 0. Further investigation needs to be made into the possibility of synchronising the achievement of the target anchoring position and angular velocity objectives.
- 2. Coverage, as defined in section 5.3.2, is less than required. Instead of covering all meridians of the top hemisphere of the object, the module trajectory covers only half of them. The proportional constraints observed between the control commands suggests that the module is underactuated and that the dimension of the control space is 1. This negatively impacts the coverage. The module position state φ is faster than the module position state θ which causes the module to coil about the Z axis too close to the pole. Ideally, the two states should be decoupled so as to constrain θ to be faster than φ at the onset and slower than φ afterwards in order to distribute the trajectory better on the surface. For instance, the module trajectory could coil about the Z axis more than one time. Finally, coverage increases with prolateness and decreases with oblateness.
- 3. In order to improve the coverage of the deployed structure, future work should include:
 - (a) Add a penalty cost to the cost function to force for either the position state and/or the angular velocity state to be close to its reference state.
 - (b) Add an extra constraint set to the controller such as a set of landmark surface points through which the module has to go before reaching its

target anchoring location.

- (c) Add modes to the low-level controller with the first tackling deployment near the pole of the object and the second tackling deployment away form the object's pole.
- (d) Adapt the high level controller to deploy modules in different directions,i.e. have multiple chain structures.

In the next section, systemic stability will be analysed.

5.4 Linear Model Predictive Control Law Performance and Stability: Spin Tracking Case

This section analyses the system's behaviour under the linear law over the time span it takes for the module to reach its target anchoring location and stop. As was established in the previous sections, the linear MPC law is not stabilising as its output diverges over time but the timescale of the convergence of the module to its anchoring location is much shorter than that of the angular velocity of the object. Therefore, the aim here is to examine the physical effects on and stability of the system over the time span of the module deployment, i.e. when instability has possibly not yet set in. Finally, in all cases, the nonlinear MPC law produces much better stability results than the linear control law and offer a suitable solution should the faster linear control law perform poorly.

5.4.1 Origin of the Data

The data displayed in this section corresponds to the following specific scenario:

- 1. The MPC controller is tracking a pure state of spin for the object with a target spin rate $\omega_0 \neq 0$.
- 2. The initial rotational state of the object corresponds to the worst case scenario where symmetric objects nutate with $W_{zinit} = 0.5$ and asymmetric objects

tumble with $\frac{h^2}{2T} = I_{mid}$, where I_{mid} is the moment of inertia about the unstable axis.

3. The data is collecting until the module stops at its target anchoring location.

All the results obtained across the simulation parameters space were similar for both the linear and nonlinear MPC controller and were identical to those produced when the objective was to track a despun state. Therefore, all descriptions and commentaries made on the data displayed below remain valid:

- 1. For all mass ratios
- 2. Whether the objective is to track a pure state of spin or a despun state.
- 3. For both the linear and nonlinear MPC controller.
- 4. For potentially stable and unstable systems.

In each of the following sections, the system's stability is examined with phase diagrams along with the evolutions of the nutation angle and of the object's rotational kinetic energy.

5.4.2 Symmetric Ellipsoids





Figure 5.20: Kinetic Energy vs. Time



Figure 5.19: Nutation vs. Time

Figure 5.21: Nutation vs. Kinetic Energy

Figure 5.22: Stability for the Prolate Case (a=0.1 b=0.1 c=1)

For the prolate ellipsoid case $(a = \frac{1}{10}, b = \frac{1}{10}, c = 1)$, the phase diagram indicates that, for a mass ratio equal to MR = 10, over the time span of the module deployment, the object experienced a very large deceleration and a reversal of its angular velocity to reach a magnitude nearly three orders of magnitude above its initial value. For mass ratio MR = 100, the magnitude increases by a factor of ten. For mass ratios MR = 1000 and MR = 10000, the effect of the module deployment on the rotational state of the system is negligible.

This is confirmed by the level of kinetic energy which is constant over time for mass ratios $MR = 100 \ MR = 1000$ and MR = 10000 but rises coherently and sharply for MR = 10 increasing by two orders of magnitude.

The nutation angle decreases over the module deployment from 1.57 rad to round about 0.8 rad for mass ratio MR = 10 and from 1.57 rad to round about 1.4 rad for mass ratio MR = 100. For all other mass ratios, the nutation angle remains constant although it decreases very slowly passed the anchoring location.

Lastly the nutation angle decreases with increasing rotational kinetic energy which is consistent with a convergence of the rotational state towards a pure state of spin. Indeed increasing the spin rate, i.e. the rotational kinetic energy, increases rotational stability about the spinning axis.

The lower the mass ratio of the object's to the module's, the larger the deployment effect on the rotational state of the system. For mass ratios of MR = 1000 or more this effect is negligible.





Figure 5.25: Kinetic Energy vs. Time



Figure 5.24: Nutation vs. Time

Figure 5.26: Nutation vs. Kinetic Energy

Figure 5.27: Stability for the Prolate Case (a=0.5 b=0.5 c=1)

For the prolate ellipsoid case $(a = \frac{1}{2}, b = \frac{1}{2}, c = 1)$, the phase diagram indicates that, for a mass ratio equal to MR = 10, over the time span of the module deployment, the object experienced a very large deceleration and a reversal of its angular velocity to reach a magnitude nearly three orders of magnitude above its initial value. For mass ratio MR = 100, the magnitude increases by a factor of ten. For mass ratios MR = 1000 and MR = 10000, the effect of the module deployment on the rotational state of the system is negligible.

This is confirmed by the level of kinetic energy which is constant over time for mass ratios MR = 1000 and MR = 10000 but rises coherently and sharply for MR = 10 increasing by three orders of magnitude and for MR = 100 increasing by two orders of magnitude.

The nutation angle evolution exhibits the same pattern regardless of the mass ratio but with a timescale dependent on the mass ratio. In figure 5.24, for mass ratio MR = 10, the nutation angle starts by increasing reaching 1.57 rad but then decreases over the rest of the module deployment from 1.57 rad to round about 0.3 rad. As confirmed by figure 5.24, the nutation angle reaches a local minimum of 0.3 rad. The timescale of the nutation angle evolution increases with increasing mass ratios in proportions that remain to be determined. For mass ratio MR = 10000, it is no longer observable on the graph.

Again, the nutation angle decreases with increasing rotational kinetic energy which is consistent with a convergence of the rotational state towards a pure state of spin. The lower the mass ratio of the object's to the module's, the larger the deployment effect on the rotational state of the system. For mass ratios of MR = 1000 or more this effect is negligible.





Figure 5.30: Kinetic Energy vs. Time





Figure 5.31: Nutation vs. Kinetic Energy

Figure 5.32: Stability for the Prolate Case (a=0.8 b=0.8 c=1)

For the prolate ellipsoid case $(a = \frac{4}{5}, b = \frac{4}{5}, c = 1)$, the phase diagram indicates that, for a mass ratio equal to MR = 10, over the time span of the module deployment, the object experienced a very large deceleration and a reversal of its angular velocity to reach a magnitude nearly three orders of magnitude above its initial value. For mass ratio MR = 100, the magnitude increases by a factor of ten. For mass ratios MR = 1000 and MR = 10000, the effect of the module deployment on the rotational state of the system is negligible.

This is confirmed by the level of kinetic energy which is constant over time for mass ratios MR = 1000 and MR = 10000 but rises coherently and sharply for MR = 10 increasing by three orders of magnitude and for MR = 100 increasing by two orders of magnitude.

The nutation angle evolution exhibits the same pattern regardless of the mass ratio but with a timescale dependent on the mass ratio. In figure 5.29, for mass ratio MR = 10, the nutation angle starts by increasing reaching 1.57 rad but then decreases over the rest of the module deployment from 1.57 rad to round about 0.4 rad. As confirmed by figure 5.29, the nutation angle reaches a local minimum of 0.4 rad. The timescale of the nutation angle evolution increases with increasing mass ratios in proportions that remain to be determined. For mass ratio MR = 10000, it is no longer observable on the graph. The timescale is also slightly faster than for $(a = \frac{1}{10}, b = \frac{1}{10}, c = 1)$ and $(a = \frac{1}{2}, b = \frac{1}{2}, c = 1)$. For the ellipsoid $(a = \frac{4}{5}, b = \frac{4}{5}, c =$ 1) and mass ratio MR = 10, the local minimum of the nutation angle is reached before two normalised seconds while for the other two ellipsoids, the local minimum is reached after. This is indicative that for prolate shapes closer to a sphere, the timescale of the nutation angle evolution becomes increasingly faster.

Again, the nutation angle decreases with increasing rotational kinetic energy which is consistent with a convergence of the rotational state towards a pure state of spin. The lower the mass ratio of the object's to the module's, the larger the deployment effect on the rotational state of the system. For mass ratios of MR =10000 or more this effect is negligible.



Figure 5.33: Phase Diagram

Figure 5.35: Kinetic Energy vs. Time





Figure 5.36: Nutation vs. Kinetic Energy

Figure 5.37: Stability for the Oblate Case (a=1.25 b=1.25 c=1)

For the oblate ellipsoid case $(a = \frac{5}{4}, b = \frac{5}{4}, c = 1)$, the phase diagram indicates that, for a mass ratio equal to MR = 10, over the time span of the module deployment, the object experienced a very large deceleration and a reversal of its angular velocity to reach a magnitude nearly three orders of magnitude above its initial value. For mass ratio MR = 100, the magnitude increases by a factor of ten. For mass ratios MR = 1000 and MR = 10000, the effect of the module deployment on the rotational state of the system is negligible.

This is confirmed by the level of kinetic energy which is constant over time for mass ratios MR = 1000 and MR = 10000 but rises coherently and sharply for MR = 10 increasing by four orders of magnitude and for MR = 100 increasing by three orders of magnitude. This is one order of magnitude more than for prolate ellipsoids.

The nutation angle evolution exhibits the same pattern regardless of the mass ratio but with a timescale dependent on the mass ratio. In figure 5.34, for mass ratio MR = 10, the nutation angle starts by increasing reaching 1.57 rad but then decreases over the rest of the module deployment from 1.57 rad to round about 0.4 rad. As confirmed by figure 5.34, the nutation angle reaches a local minimum of 0.4 rad. The timescale of the nutation angle evolution increases with increasing mass ratios in proportions that remain to be determined. For mass ratio MR = 10000, it is no longer observable on the graph.

Again, the nutation angle decreases with increasing rotational kinetic energy which is consistent with a convergence of the rotational state towards a pure state of spin. The lower the mass ratio of the object's to the module's, the larger the deployment effect on the rotational state of the system. For mass ratios of MR = 10000 or more this effect is negligible.





Figure 5.39: Nutation vs. Time

0.4

0.2

0

Figure 5.41: Nutation vs. Kinetic Energy

12000

14000

16000

Figure 5.42: Stability for the Oblate Case (a=2 b=2 c=1)

For the oblate ellipsoid case (a = 2, b = 2, c = 1), the phase diagram indicates that, for a mass ratio equal to MR = 10, over the time span of the module deployment, the object experienced a very large deceleration and a reversal of its angular velocity to reach a magnitude nearly three orders of magnitude above its initial value. For mass ratio MR = 100, the magnitude increases by a factor of ten. For mass ratios MR = 1000 and MR = 10000, the effect of the module deployment on the rotational state of the system is negligible.

This is confirmed by the level of kinetic energy which is constant over time for mass ratios MR = 1000 and MR = 10000 but rises coherently and sharply for MR = 10 increasing by four orders of magnitude and for MR = 100 increasing by three orders of magnitude. This is one order of magnitude more than for prolate ellipsoids.

The nutation angle evolution exhibits the same pattern regardless of the mass ratio but with a timescale dependent on the mass ratio. In figure 5.39, for mass ratio MR = 10, the nutation angle starts by increasing reaching 1.57 rad but then decreases over the rest of the module deployment from 1.57 rad to round about 0.4 rad. As confirmed by figure 5.39, the nutation angle reaches a local minimum of 0.4 rad. The timescale of the nutation angle evolution increases with increasing mass ratios in proportions that remain to be determined. For mass ratio MR = 10000, it is no longer observable on the graph.

Again, the nutation angle decreases with increasing rotational kinetic energy which is consistent with a convergence of the rotational state towards a pure state of spin. The lower the mass ratio of the object's to the module's, the larger the deployment effect on the rotational state of the system. For mass ratios of MR = 10000 or more this effect is negligible.





Figure 5.45: Kinetic Energy vs. Time





Figure 5.46: Nutation vs. Kinetic Energy

Figure 5.47: Stability for the Oblate Case (a=10 b=10 c=1)

For the oblate ellipsoid case (a = 10, b = 10, c = 1), the phase diagram indicates that, for a mass ratio equal to MR = 10, over the time span of the module deployment, the object experienced a very large deceleration and a reversal of its angular velocity to reach a magnitude nearly three orders of magnitude above its initial value. For mass ratio MR = 100, the magnitude increases by a factor of ten. For mass ratios MR = 1000 and MR = 10000, the effect of the module deployment on the rotational state of the system is negligible.

This is confirmed by the level of kinetic energy which is constant over time for mass ratios MR = 1000 and MR = 10000 but rises coherently and sharply for MR = 10 increasing by five orders of magnitude and for MR = 100 increasing by four orders of magnitude. This is one order of magnitude more than for prolate ellipsoids.

The nutation angle evolution exhibits the same pattern regardless of the mass ratio but with a timescale dependent on the mass ratio. In figure 5.44, for mass ratio MR = 10, the nutation angle starts by increasing reaching 1.57 rad but then decreases over the rest of the module deployment from 1.57 rad to round about 0.4 rad. As confirmed by figure 5.44, the nutation angle reaches a local minimum of 0.4 rad. The timescale of the nutation angle evolution increases with increasing mass ratios in proportions that remain to be determined. For mass ratio MR = 10000, it is no longer observable on the graph. The timescale is also slightly faster than for $(a = \frac{5}{4}, b = \frac{5}{4}, c = 1)$ than for the other two. For the ellipsoid (a = 10, b = 10, c = 1)and (a = 2, b = 2, c = 1) and mass ratio MR = 10, the local minimum of the nutation angle is reached after two normalised seconds while for $(a = \frac{5}{4}, b = \frac{5}{4}, c = 1)$, the local minimum is reached before. This is indicative that for oblate shapes closer to a sphere, the timescale of the nutation angle evolution becomes increasingly faster.

Again, the nutation angle decreases with increasing rotational kinetic energy which is consistent with a convergence of the rotational state towards a pure state of spin. The lower the mass ratio of the object's to the module's, the larger the deployment effect on the rotational state of the system. For mass ratios of MR =10000 or more this effect is negligible.

5.4.3 Asymmetric Ellipsoids and Limit Cases

The first limit case is the perfect sphere. As per section ??, spheres separates potentially stable and unstable shapes.



Figure 5.49: Nutation vs. Time



Figure 5.52: Stability for the Sphere Case

For the spherical case, the phase diagram indicates that, for a mass ratio equal to MR = 10, over the time span of the module deployment, the object experienced a very large deceleration and a reversal of its angular velocity to reach a magnitude nearly three orders of magnitude above its initial value. For mass ratio MR = 100, the magnitude increases by a factor of ten. For mass ratios MR = 1000 and MR =10000, the effect of the module deployment on the rotational state of the system is negligible. This is confirmed by the level of kinetic energy which rises coherently and sharply for MR = 10 increasing by four orders of magnitude and for MR = 100 increasing by three orders of magnitude. This is the same order of magnitude as for prolate ellipsoids. For mass ratios MR = 1000 and MR = 10000, the level of rotational kinetic energy is constant at the start and after two normalised seconds starts to decrease. For mass ratio MR = 10 and MR = 100, the early behaviour is similar before the steep rise. The scale of the graph does not make it visible.

The nutation angle evolution exhibits the same pattern regardless of the mass ratio but with a timescale dependent on the mass ratio. In figure 5.49, for mass ratio MR = 10, the nutation angle starts by increasing reaching 1.57 rad but then decreases over the rest of the module deployment from 1.57 rad to round about 0.4 rad. As confirmed by figure 5.51, the nutation angle reaches a local minimum of 0.4 rad. The timescale of the nutation angle evolution increases with increasing mass ratios in proportions that remain to be determined. For mass ratio MR = 10000, it is no longer observable on the graph. The nutation angle evolution shows that the system does not fully recover from the nutation caused by the module deployment. Since a sphere is symmetrical about all its principal axes, it is always in a pure state of spin. What the residual nutation means in this case is that the spinning axis direction has permanently diverged by about 20° from its original pointing direction.

Again, the nutation angle decreases with increasing rotational kinetic energy which is consistent with a convergence of the rotational state towards a pure state of spin. The lower the mass ratio of the object's to the module's, the larger the deployment effect on the rotational state of the system. For mass ratios of MR =10000 or more this effect is negligible.




Figure 5.55: Kinetic Energy vs. Time



Figure 5.54: Nutation vs. Time

Figure 5.56: Nutation vs. Kinetic Energy

Figure 5.57: Stability for the Asymmetric Case (a=0.1 b=0.8 c=1)

For the asymmetric ellipsoid case $(a = \frac{1}{10}, b = \frac{8}{10}, c = 1)$, the phase diagram indicates that, for a mass ratio equal to MR = 10, over the time span of the module deployment, the object experienced a very large deceleration and a reversal of its angular velocity to reach a magnitude nearly three orders of magnitude above its initial value. For mass ratio MR = 100, the magnitude increases by a factor of ten. For mass ratios MR = 1000 and MR = 10000, the effect of the module deployment on the rotational state of the system is negligible.

The orbit for MR = 10 and MR = 100 is closed or about to close. Although it cannot be observed on the graph, it is also the case for MR = 1000 and MR = 10000. This indicates that, over the course of the module deployment, a return to the initial angular velocity magnitude about the Z axis which is the minor principal axis.

The level of rotational kinetic energy is constant for all mass ratios except between 1.5 and 2 normalised seconds where a brief surge or drop occurs before the level of rotational kinetic energy returns to its initial level. The level of rotational kinetic energy is of the same order of magnitude as for prolate ellipsoids.

In figure 5.54, for MR = 1000 and MR = 10000, the nutation angle departs from its initial value of 0.85 rad to reach 1.57 rad to finally converge to its initial value. For MR = 100, the nutation angle departs from its initial value twice and finally converges to its initial value of 0.85 rad. For MR = 10, the nutation angle evolution exhibits the same pattern but for the fact that the nutation angle converges to 0.2 rad indicating a change of rotation axis towards the minor Z axis. For all mass ratios the timescale is similar with a slight increase for decreasing mass ratios.

Again, the nutation angle decreases with increasing rotational kinetic energy which is consistent with a convergence of the rotational state towards a pure state of spin. However, for this asymmetric case, the nutation angle does not converge to a local minimum. Extrapolating on the exhibited trend, the nutation angle could possibly converge to 0 *rad* which would correspond to a pure state of spin about the minor principal axis. In this respect, in the asymmetric object case, the linear law exhibits a far better performance than in the symmetric object case.

Again, the lower the mass ratio of the object's to the module's, the larger the

deployment effect on the rotational state of the system. For mass ratios of MR = 10000 or more this effect is negligible.





Figure 5.60: Kinetic Energy vs. Time



Figure 5.59: Nutation vs. Time

Figure 5.61: Nutation vs. Kinetic Energy

Figure 5.62: Stability for the Asymmetric Case (a=0.33 b=0.5 c=1)

Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

For the asymmetric ellipsoid case $(a = \frac{1}{3}, b = \frac{1}{2}, c = 1)$, the phase diagram indicates that, for a mass ratio equal to MR = 10, over the time span of the module deployment, the object experienced a very large deceleration and a reversal of its angular velocity to reach a magnitude nearly two orders of magnitude above its initial value. For mass ratio MR = 100, the magnitude increases by a factor of ten. For mass ratios MR = 1000 and MR = 10000, the effect of the module deployment on the rotational state of the system is negligible. The orbits are similar to the symmetric cases. The rest of the graphs share similarities with both the previous asymmetric case and the symmetric cases.

The level of rotational kinetic energy is constant for all mass ratios except between 1.5 and 2 normalised seconds where a brief surge or drop occurs before the level of rotational kinetic energy returns to its initial level, except for MR = 10 where the rotational kinetic energy settles at a new higher level. The level of rotational kinetic energy is of the same order of magnitude as for prolate ellipsoids.

In figure 5.59, for MR = 10000, the nutation angle remains constant at 1.35 rad. For MR = 1000, the nutation angle departs from its initial value of 1.35 rad to reach 1.45 rad to finally converge to 1.4 rad. For MR = 100, the nutation angle departs from its initial value of 1.35 rad to reach 1.57 rad to finally converge to 1 rad. Finally, for MR = 10, the nutation angle departs from its initial value of 1.35 rad to reach 1.57 rad to finally converge to 0.3 rad indicating a change of rotation axis towards the minor Z axis. The timescale of the nutation angle evolution increases with increasing mass ratios in proportions that remain to be determined. For mass ratio MR = 10000, it is no longer observable on the graph.

Again, the nutation angle decreases with increasing rotational kinetic energy which is consistent with a convergence of the rotational state towards a pure state of spin. For this asymmetric case, the nutation angle converges to a local minimum at 0.3 rad. What the residual nutation angle means in this case is that the rotation axis pointing fluctuates inside a 20° cone. In this respect, in the asymmetric object case, the linear law exhibits a better performance than in the symmetric object case.

Again, the lower the mass ratio of the object's to the module's, the larger the

deployment effect on the rotational state of the system. For mass ratios of MR = 10000 or more this effect is negligible.





Figure 5.65: Kinetic Energy vs. Time



Figure 5.64: Nutation vs. Time

Figure 5.66: Nutation vs. Kinetic Energy

Figure 5.67: Stability for the Asymmetric Case (a=0.8 b=0.9 c=1)

For the asymmetric ellipsoid case $(a = \frac{8}{10}, b = \frac{9}{10}, c = 1)$, the phase diagram indicates that, for a mass ratio equal to MR = 10, over the time span of the module deployment, the object experienced a very large deceleration and a reversal of its angular velocity to reach a magnitude nearly two orders of magnitude above its initial value. For mass ratio MR = 100, the magnitude increases by a factor of ten. For mass ratios MR = 1000 and MR = 10000, the effect of the module deployment on the rotational state of the system is negligible. The orbits are similar to the symmetric cases. The rest of the graphs share similarities with both the previous asymmetric case and the symmetric cases.

The level of rotational kinetic energy is constant for MR = 1000 and MR = 10000 with a slight decrease after 1.5 normalised second. It increases for MR = 100 and diverges for MR = 10. The level of rotational kinetic energy is of the same order of magnitude as for prolate ellipsoids.

The nutation angle evolution exhibits the same pattern regardless of the mass ratio but with a timescale dependent on the mass ratio. In figure 5.64, for mass ratio MR = 10, the nutation angle starts by increasing from 1.15 rad to 1.57 rad but then decreases over the rest of the module deployment from 1.57 rad to round about 0.4 rad. As confirmed by figure 5.64, the nutation angle reaches a local minimum of 0.4 rad. The timescale of the nutation angle evolution increases with increasing mass ratios in proportions that remain to be determined. For mass ratio MR = 10000, it is no longer observable on the graph. This behaviour is similar to those exhibited for the prolate ellipsoids cases.

Again, the nutation angle decreases with increasing rotational kinetic energy which is consistent with a convergence of the rotational state towards a pure state of spin. For this asymmetric case, the nutation angle converges to a local minimum at 0.4 rad. What the residual nutation angle means in this case is that the rotation axis pointing fluctuates inside a 20° cone. In this respect, in the asymmetric object case, the linear law exhibits a similar performance to the symmetric object case.

Again, the lower the mass ratio of the object's to the module's, the larger the deployment effect on the rotational state of the system. For mass ratios of MR =

10000 or more this effect is negligible.

5.4.4 Conclusion

As the objective of deploying to the target anchoring location is always fulfilled, this section examined the stability performance of the linear law from the point of view of the behaviour of the system's rotational state up to the successful deployment of one module to its target anchoring location. As mentioned in section 5.4.1, the results displayed in sections 5.4.2 and 5.4.3 are identical to those produced for a reference trajectory where the objective is to track a despun state. The results displayed in this section correspond to worst case initial conditions in terms of the object's initial rotational state. Nonetheless, for all other initial conditions the results are similar. Therefore, the conclusions drawn below are also valid for other initial conditions which will be examined in greater depth in the next section.

With respect to the deployment objectives and owing to the similarities between the linear and nonlinear control law results, the observations made in sections 5.4.2 and 5.4.3 led to the following conclusions:

- 1. The results showed that despite challenging initial rotational states and initial divergence of the nutation angle, both the linear and nonlinear controllers drove the nutation angle to a minimum value in [0.3 rad, 0.4 rad], i.e. [17°, 23°]. Indeed the trends shown by the graphs indicate that after reaching this minimum value the nutation angle would increase again. Consequently, for both the linear and nonlinear control laws, the system's objective of reaching either a pure state of spin a despun state is not achieved and a precise pointing direction for the rotation axis cannot be achieved with the current version of the controller.
- 2. For both the linear and nonlinear control laws, there is always a residual angular velocity at the module target anchoring location. The low-level module controller should behave like a regulator and the module should not have any residual angular velocity upon reaching the target anchoring location. This

suggest a timescale difference between angular velocity and anchoring location states which should be corrected with future work.

- 3. There is no provision for a dissipative energy mechanism in the design of the low-level module controller. Despite the fact that, as per section 5.2, the low-level module controller can output positive or negative control commands, this is not sufficient for dissipating the rotational kinetic energy of the object. Therefore, despinning the object is not achievable under the current controller design.
- 4. The linear law values correspond to extremely large and unrealistic energy requirements. The nonlinear law on the other hand provides far more reasonable results. In this respect and if its timescale allows it, the nonlinear law should be preferred for implementation.
- 5. Finally, the system's objective of reaching either a pure state of spin a despun state is impaired by large mass ratios. For $MR \ge 1000$, the nutation angle remains close to its initial value and is not influenced by the module deployment.
- 6. The mass ratio is the parameter which bears the greater influence on the controller's performance. All control inputs have the same order of magnitude regardless of the magnitude of the module's moments of inertia, hence the lower the mass ratio, the lower the difference between the inertia of the object and the module, the greater the energy transfer between the module and the object, all other parameters being equal. Consequently:
 - (a) The lower the mass ratio, the greater the control effects but the greater the possibility and magnitude of divergence.
 - (b) Stability increases with increasing mass ratios.
- 7. Finally, the shape of the object influences the controller's performance. This influence of shape is best captured with respect to the limit case of the sphere as follows:

- (a) Systems with asymmetric objects are more resistant to divergence than systems with a symmetric object.
- (b) The more symmetric the object is, the more the control commands tend to diverge.
- (c) Stability increases with prolateness and oblateness when departing from an initial spherical shape.
- (d) Finally, the timescale of the system increases with symmetry. It decreases with prolateness and oblateness when departing from an initial spherical shape and the limit case of the sphere has the largest timescale.

This conclusions section will end with some general comments:

- 1. It was established in chapter 4 that the system is not controllable. The system's controllability matrix has at best a rank of 6 or a rank of 4 for the despin tracking case. The results presented in this section show a correlation between increasing rotational kinetic energy and decreasing nutation angle. However, for both the pure spin or despin tracking cases, the results of the controller actions are the same which makes it unclear what the real influence of controller is. Indeed, the energy only transferred to the object by the module. There is no provision for a dissipative mechanism which would draw out rotational kinetic energy from the object. At constant rotational kinetic energy level, for an isolated system, an increase in moment of inertia leads to a decrease of the magnitude of the angular velocity. The magnitude of the influence of mass distribution should be evaluated in order to investigate the possibility of explicitly involving changes of mass distribution in the deployment strategy and controller design.
- 2. The reason for the convergence to a nutation state as opposed to a convergence to a nutationless state is unclear. It seems that the closer the rotational state to its equilibrium spinning state, the harder it is to control, especially about the minor unstable axis. Further investigations are required into this.

- 3. As per previous point, the introduction of a nutation damper should be investigated to complement the work of the controller and help the convergence process. But this is left for future work.
- 4. For equal angular velocities, oblate objects have a higher level of rotational kinetic energy than prolate objects because of the moment of inertia about their spinning axis is the largest in magnitude as opposed to prolate objects where the moment of inertia about their spinning axis is the lowest in magnitude.

In the next section, the performance of the control laws will be examined of longer timescale.

5.5 Linear Model Predictive Control Law: Stabilisation of the Spin Tracking Case

This section analyses the system's behaviour under the linear law over an unbounded time span where the module is left free to carry on moving beyond its target anchoring location. As was established in the previous sections, the linear MPC law is not stabilising as its output diverges over time and the timescale of the convergence of the module to its anchoring location is much shorter than that of the angular velocity of the object. The aim here is to further examine the physical effects on and stability of the system over an unbounded time span in order to evaluate the timescale difference between deployment and angular velocity states as well as to examine the influence of initial conditions. Again, in all cases, the nonlinear MPC law produces much better stability results than the linear control law and an explicit comparison will be made in the next section 5.6.

5.5.1 Origin of the Data

The data displayed in this section corresponds to the following specific scenario:

1. The linear MPC controller tracks a pure state of spin for the object with a target spin rate $\omega_0 \neq 0$ for the angular velocity part of the system state.

2. All initial rotational state of the object are represented: for symmetric objects $W_{zinit} \in \{0.5, 0.75, 1\}$ and for asymmetric objects

$$\frac{h^2}{2T} \in \{I_{min}, \frac{I_{min}+I_{mid}}{2}, I_{mid}, \frac{I_{max}+I_{mid}}{2}, I_{max}\}.$$

3. The module is not constrained to stop at its target location and is left free to move on the surface of the object with the sole objective of tracking a pure state of spin for the object. The aim is to analyse the system's behaviour near the origin over a large timescale.

5.5.2 Data Description

The data is displayed for specific shapes representative of a spectrum of realistic situations the robot could encounter. These shapes encompass a sphere, one oblate, one prolate and one asymmetric ellipsoids which are:

Ellipsoid Type	a	b	c	System Stability Status
Prolate	$\frac{1}{2}$	$\frac{1}{2}$	1	Potentially Stable
Sphere	1	1	1	Potentially Stable
Asymmetric	$\frac{1}{10}$	$\frac{8}{10}$	1	Potentially Stable
Asymmetric	$\frac{\frac{8}{10}}{10}$	$\frac{\overline{9}}{10}$	1	Potentially Stable (Close to Unstable)
Oblate	2	2	1	Potentially Stable

Table 5.6: Ellipsoids Normalised Semi-Axes Lengths

All the results obtained for the linear MPC controller across all simulation parameters were identical to those produced when the objective was to track a despun state. Therefore, all descriptions and commentaries made on the data displayed below remain valid:

- 1. For all ellipsoidal shapes
- 2. For all mass ratios MR
- 3. Whether the objective is to track a pure state of spin or a despun state.

In each of the following sections, the system's stability is examined with phase diagrams along with the evolutions of the nutation angle and of the object's rotational kinetic energy.

5.5.3 Sphere Case

As per section 5.4, for the sphere case, the linear control law enables the module to reach its target anchoring location within 3 normalised seconds. As per figures 5.69 and 5.70 below, the nutation angle and the object's rotational kinetic energy diverge after 4 normalised seconds. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. The frequency increases with increasing mass ratio and the amplitude decreases with increasing mass ratio. Instability sets in very shortly after the module deployed successfully. The instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use.



Figure 5.68: Phase Diagrams for the Z Axis (Sphere)

Figure 5.70: Phase Diagrams for the Z Axis (Sphere)



Figure 5.69: Phase Diagrams for the ZFigure 5.71: Phase Diagrams for the ZAxis (Sphere)Axis (Sphere)

Figure 5.72: Sphere Z Axis Stability Free Motion

5.5.4 Symmetric Case (a = 0.5, b = 0.5, c = 1)

In this subsection, the linear control law is assessed for three different initial object's rotational states. For each of these initial conditions, two sets of graphs are used. One set with all the mass ratios and the other with only the highest mass ratios to account for the differences in scale.

5.5.4.1 Initial Condition Wzinit = 0.5

As per section 5.4, for the prolate ellipsoid (a = 0.5, b = 0.5, c = 1), the linear control law enables the module to reach its target anchoring location within 2 normalised seconds. As per figures 5.73 and 5.81 below, the nutation angle and the object's rotational kinetic energy diverge after 4 normalised seconds. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in very shortly after the module deployed successfully. The instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use. Interestingly, the frequency of the nutation angle evolution is at least an order of magnitude higher for the two intermediary mass ratios MR = 100 and MR = 1,000 than for the lowest and largest mass ratios respectively MR = 10 and MR = 10,000. An order of size analysis of the ratio of rotational kinetic energy to inertia would conclude that an increase in inertia should prompt a decrease of the frequency of the nutation angle evolution. This suggests the possible existence of a resonance phenomenon parametrised by the mass ratio MR the investigation of which is left for future work.



Figure 5.74: Wzinit = 0.5

Figure 5.76: Wzinit = 0.5

Figure 5.77: Phase Diagrams for the Z Axis (a=0.5 b=0.5 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.79: Wzinit = 0.5

Figure 5.81: Wzinit = 0.5

Figure 5.82: Phase Diagrams for the Z Axis (a=0.5 b=0.5 c=1)

5.5.4.2 Initial Condition Wzinit = 0.75

As per section 5.4, for the prolate ellipsoid (a = 0.5, b = 0.5, c = 1), the linear control law enables the module to reach its target anchoring location within 2 normalised seconds. As per figures 5.83 and 5.91 below, the nutation angle and the object's rotational kinetic energy diverge after 4 normalised seconds. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in very shortly after the module deployed successfully. The instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use. Again, the frequency of the nutation angle evolution is at least an order of magnitude higher for the two intermediary mass ratios MR = 100 and MR = 1,000 than for the lowest and largest mass ratios respectively MR = 10 and MR = 10,000 and orders of magnitude larger for $W_{zinit} = 0.75$ than for $W_{zinit} = 0.5$. An order of size analysis of the ratio of rotational kinetic energy to inertia would conclude that an increase in inertia should prompt a decrease of the frequency of the nutation angle evolution. This suggests the possible existence of a resonance phenomenon parametrised by the mass ratio MR the investigation of which is left for future work.



Figure 5.84: Wzinit = 0.75

Figure 5.86: Wzinit = 0.75

Figure 5.87: Phase Diagrams for the Z Axis (a=0.5 b=0.5 c=1)

Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module





Figure 5.91: Wzinit = 0.75

Figure 5.92: Phase Diagrams for the Z Axis (a=0.5 b=0.5 c=1)

5.5.4.3 Initial Condition Wzinit = 1

As per section 5.4, for the prolate ellipsoid (a = 0.5, b = 0.5, c = 1), the linear control law enables the module to reach its target anchoring location within 2 normalised seconds. As per figures 5.93 and 5.101 below, the nutation angle and the object's rotational kinetic energy diverge after 4 normalised seconds. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Both the frequency and the amplitude increase with decreasing mass ratio. Instability sets in very shortly after the module deployed successfully. For MR = 10,000, however, non-expanding closed orbits can be observed which is indicative of stability at least over a longer period of time, 12 normalised seconds in this case, than for lower mass ratios. Nonetheless, the instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use.





Figure 5.96: Wzinit = 1

Figure 5.97: Phase Diagrams for the Z Axis (a=0.5 b=0.5 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.99: Wzinit = 1

Figure 5.101: Wzinit = 1

Figure 5.102: Phase Diagrams for the Z Axis (a=0.5 b=0.5 c=1)

5.5.5 Symmetric Case (a = 2, b = 2, c = 1)

In this subsection, the linear control law is assessed for three different initial object's rotational states. For each of these initial conditions, two sets of graphs are used. One set with all the mass ratios and the other with only the highest mass ratios to account for the differences in scale.

5.5.5.1 Initial Condition Wzinit = 0.5

As per section 5.4, for the oblate ellipsoid (a = 2, b = 2, c = 1), the linear control law enables the module to reach its target anchoring location within 2 normalised seconds. As per figures 5.103 and 5.111 below, the nutation angle and the object's rotational kinetic energy diverge after 2 normalised seconds for MR = 1,000 and immediately for all other MR. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in immediately although the module deploys successfully. The linear control law is of no practical use. Interestingly, the frequency of the nutation angle evolution is at least an order of magnitude higher for the two mass ratios MR = 100 and MR = 10,000 than for MR = 10 and MR = 1,000. An order of size analysis of the ratio of rotational kinetic energy to inertia would conclude that an increase in inertia should prompt a decrease of the frequency of the nutation angle evolution. This suggests the possible existence of a resonance phenomenon parametrised by the mass ratio MR the investigation of which is left for future work.





Figure 5.106: Wzinit = 0.5

Figure 5.107: Phase Diagrams for the Z Axis (a=2 b=2 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.109: Wzinit = 0.5

Figure 5.111: Wzinit = 0.5

Figure 5.112: Phase Diagrams for the Z Axis (a=2 b=2 c=1)

5.5.5.2 Initial Condition Wzinit = 0.75

As per section 5.4, for the oblate ellipsoid (a = 2, b = 2, c = 1), the linear control law enables the module to reach its target anchoring location within 2 normalised seconds. As per figures 5.113 and 5.121 below, the nutation angle and the object's rotational kinetic energy diverge after 2 normalised seconds for MR = 1,000 and immediately for all other MR. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in immediately although the module deploys successfully. The linear control law is of no practical use. Interestingly, the frequency of the nutation angle evolution is at least an order of magnitude higher for the two mass ratios MR = 100 and MR = 10,000 than for MR = 10 and MR = 1,000 and larger for $W_{zinit} = 0.75$ than for $W_{zinit} = 0.5$. An order of size analysis of the ratio of rotational kinetic energy to inertia would conclude that an increase in inertia should prompt a decrease of the frequency of the nutation angle evolution. This suggests the possible existence of a resonance phenomenon parametrised by the mass ratio MR the investigation of which is left for future work.



Figure 5.114: Wzinit = 0.75

Figure 5.116: Wzinit = 0.75

Figure 5.117: Phase Diagrams for the Z Axis (a=2 b=2 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.119: Wzinit = 0.75

Figure 5.121: Wzinit = 0.75

Figure 5.122: Phase Diagrams for the Z Axis (a=2 b=2 c=1)

5.5.5.3 Initial Condition Wzinit = 1

As per section 5.4, for the oblate ellipsoid (a = 2, b = 2, c = 1), the linear control law enables the module to reach its target anchoring location within 2 normalised seconds. As per figures 5.123 and 5.131 below, the nutation angle and the object's rotational kinetic energy diverge after 4 normalised seconds for MR = 1,000 and immediately for all other MR. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in immediately although the module deploys successfully. The linear control law is of no practical use. Interestingly, the frequency of the nutation angle evolution is at least an order of magnitude higher for the two mass ratios MR = 100 and MR = 10,000 than for MR = 10 and MR = 1,000. An order of size analysis of the ratio of rotational kinetic energy to inertia would conclude that an increase in inertia should prompt a decrease of the frequency of the nutation angle evolution. This suggests the possible existence of a resonance phenomenon parametrised by the mass ratio MR the investigation of which is left for future work.





Figure 5.126: Wzinit = 1

Figure 5.127: Phase Diagrams for the Z Axis (a=2 b=2 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.129: Wzinit = 1

Figure 5.131: Wzinit = 1

Figure 5.132: Phase Diagrams for the Z Axis (a=2 b=2 c=1)

5.5.6 Asymmetric Case (a = 0.1, b = 0.8, c = 1)

In this subsection, the linear control law is assessed for five different initial object's rotational states. For each of these initial conditions, two sets of graphs are used. One set with all the mass ratios and the other with only the highest mass ratios to account for the differences in scale. The shape of the asymmetric object corresponds to the case of a potentially stable system.

5.5.6.1 Initial Condition $\frac{h^2}{2T} = 130$

The initial condition $\frac{h^2}{2T} = 130$ for the asymmetric ellipsoid (a = 0.1, b = 0.8, c = 1) corresponds to a initial pure state of rotation about the unstable minor axis. As per section 5.4, the linear control law enables the module to reach its target anchoring location within 2.5 normalised seconds. As per figures 5.133 and 5.141 below, the nutation angle and the object's rotational kinetic energy diverge immediately for MR = 10 and MR = 100 and after 10 normalised seconds for MR = 1,000 and MR = 10,000. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in very shortly after the module deployed successfully. The instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use.



Figure 5.137: Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)

Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module



Figure 5.142: Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)

5.5.6.2 Initial Condition $\frac{h^2}{2T} = 166$

The initial condition $\frac{h^2}{2T} = 166$ for the asymmetric ellipsoid (a = 0.1, b = 0.8, c = 1) corresponds to a tumbling rotational state about a non-principal axis. As per section 5.4, the linear control law enables the module to reach its target anchoring location within 2.5 normalised seconds. As per figures 5.143 and 5.151 below, the nutation angle and the object's rotational kinetic energy diverge immediately for MR = 10 and MR = 100 but remain stable for MR = 1,000 and MR = 10,000 over the duration of the observations as shown by the closed, albeit expanding, orbits in figure 5.148. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in very shortly after the module deployed successfully. The instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use. Given the timescale and the magnitude of the control command as per section 5.2, even for MR = 10,000, the linear control law to be of practical use.



Figure 5.147: Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)


Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.152: Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)

5.5.6.3 Initial Condition $\frac{h^2}{2T} = 202$

The initial condition $\frac{h^2}{2T} = 202$ for the asymmetric ellipsoid (a = 0.1, b = 0.8, c = 1) corresponds to a initial pure state of rotation about the medium principal axis. As per section 5.4, the linear control law enables the module to reach its target anchoring location within 2.5 normalised seconds. As per figures 5.153 and 5.161 below, the nutation angle and the object's rotational kinetic energy diverge immediately for MR = 10 and MR = 100 and after 10 normalised seconds for MR = 1,000 and MR = 10,000. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in very shortly after the module deployed successfully. The instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use.



Figure 5.157: Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.162: Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)

5.5.6.4 Initial Condition $\frac{h^2}{2T} = 265$

The initial condition $\frac{\hbar^2}{2T} = 265$ for the asymmetric ellipsoid (a = 0.1, b = 0.8, c = 1) corresponds to a tumbling rotational state about a non-principal axis. As per section 5.4, the linear control law enables the module to reach its target anchoring location within 2.5 normalised seconds. As per figures 5.163 and 5.171 below, the nutation angle and the object's rotational kinetic energy diverge immediately for MR = 10 and MR = 100, diverge after 8 normalised seconds for MR = 1,000 and remain stable for MR = 10,000 over the duration of the observations as shown by the closed orbits in figure 5.168. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in very shortly after the module deployed successfully. The instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use. Given the timescale and the magnitude of the control command as per section 5.2, even for MR = 10,000, the linear control law to be of practical use.



Figure 5.167: Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.172: Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)

5.5.6.5 Initial Condition $\frac{h^2}{2T} = 328$

The initial condition $\frac{h^2}{2T} = 328$ for the asymmetric ellipsoid (a = 0.1, b = 0.8, c = 1) corresponds to a initial pure state of rotation about the unstable major axis. As per section 5.4, the linear control law enables the module to reach its target anchoring location within 2.5 normalised seconds. As per figures 5.173 and 5.181 below, the nutation angle and the object's rotational kinetic energy diverge immediately for MR = 10 and MR = 100 and after 10 normalised seconds for MR = 1,000 and MR = 10,000. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in very shortly after the module deployed successfully. The instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use.



Figure 5.177: Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)

3 <u>× 10</u>⁵ **Object Spin Axis Phase Diagram** Object Kinetic Energy vs. Time 10⁹ 10 MR=10000 MR=10000 9 MR=1000 MR=1000 2 8 Normalised Dimensionless Energy 1 7 6 0 5 ήC -1 4 3 -2 2 -3 0 -4 -6000 20 25 30 35 45 -4000 -2000 2000 4000 6000 8000 5 10 15 40 0 $\overrightarrow{\Omega}$ Time (Normalised Seconds) Figure 5.178: $\frac{h^2}{2T} = 328$ Figure 5.180: $\frac{h^2}{2T} = 328$ **Object Nutation Angle vs. Time** Nutation Angle vs. Kinetic Energy 1.6 1.6 MR=10000 MR=1000 MR=10000 MR-1000 1.4 1.4 1.2 1.2 1 8.0 ਯੂ 8.0 ឆ្ន 0.6 0.6 0.4 0.4 0.2 0.2 0 <mark>1</mark> 0 0 2 4 6 8 10 25 30 45 5 10 15 20 35 40 Time (Normalised Seconds) Normalised Dimensionless Energy imes10 9 Figure 5.179: $\frac{h^2}{2T} = 328$ Figure 5.181: $\frac{h^2}{2T} = 328$

Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.182: Phase Diagrams for the Z Axis (a=0.1 b=0.8 c=1)

5.5.7 Asymmetric Case (a = 0.8, b = 0.9, c = 1)

In this subsection, the linear control law is assessed for five different initial object's rotational states. For each of these initial conditions, two sets of graphs are used. One set with all the mass ratios and the other with only the highest mass ratios to account for the differences in scale. The shape of the asymmetric object corresponds to the case of a potentially stable system.

5.5.7.1 Initial Condition $\frac{h^2}{2T} = 290$

The initial condition $\frac{h^2}{2T} = 290$ for the asymmetric ellipsoid (a = 0.8, b = 0.9, c = 1) corresponds to a initial pure state of rotation about the unstable minor axis. As per section 5.4, the linear control law enables the module to reach its target anchoring location within 2.2 normalised seconds. As per figures 5.183 and 5.191 below, for MR = 10, MR = 100, the nutation angle diverge immediately and the object's rotational kinetic energy diverge after 6 normalised seconds. However, both remain stable for MR = 1,000 and MR = 10,000 over the duration of the observations as shown by the closed, albeit expanding, orbits in figure 5.188. System stability increases with increasing mass ratio. In this case, the practical use of the linear control law could be envisaged but for the fact that the timescale and the magnitude of the control command, i.e. the energy consumption as seen in section 5.2 is prohibitive.



Figure 5.187: Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.192: Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)

5.5.7.2 Initial Condition $\frac{h^2}{2T} = 309$

The initial condition $\frac{h^2}{2T} = 309$ for the asymmetric ellipsoid (a = 0.8, b = 0.9, c = 1) corresponds to a tumbling rotational state about a non-principal axis. As per section 5.4, the linear control law enables the module to reach its target anchoring location within 2.2 normalised seconds. As per figures 5.193 and 5.201 below, the nutation angle and the object's rotational kinetic energy diverge immediately for MR = 10, MR = 100 and MR = 1,000 but remain stable for MR = 10,000 over the duration of the observations as shown by the closed, albeit expanding, orbits in figure 5.198. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in very shortly after the module deployed successfully. The instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use. Given the timescale and the magnitude of the control command as per section 5.2, even for MR = 10,000, the linear control law to be of practical use.



Figure 5.197: Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.202: Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)

5.5.7.3 Initial Condition $\frac{h^2}{2T} = 328$

The initial condition $\frac{h^2}{2T} = 328$ for the asymmetric ellipsoid (a = 0.8, b = 0.9, c = 1) corresponds to a initial pure state of rotation about the medium principal axis. As per section 5.4, the linear control law enables the module to reach its target anchoring location within 2.2 normalised seconds. As per figures 5.203 and 5.211 below, the nutation angle and the object's rotational kinetic energy diverge immediately for MR = 10 and MR = 100 and after 6 normalised seconds for MR = 1,000 and MR = 10,000. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in very shortly after the module deployed successfully. The instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use.



Figure 5.207: Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.212: Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)

5.5.7.4 Initial Condition $\frac{h^2}{2T} = 345$

The initial condition $\frac{h^2}{2T} = 345$ for the asymmetric ellipsoid (a = 0.8, b = 0.9, c = 1) corresponds to a tumbling rotational state about a non-principal axis. As per section 5.4, the linear control law enables the module to reach its target anchoring location within 2.2 normalised seconds. As per figures 5.213 and 5.221 below, the nutation angle and the object's rotational kinetic energy diverge immediately for MR = 10, MR = 100 and MR = 1,000 but remain stable for MR = 10,000 over the duration of the observations as shown by the closed, albeit expanding, orbits in figure 5.218. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in very shortly after the module deployed successfully. The instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use. Given the timescale and the magnitude of the control command as per section 5.2, even for MR = 10,000, the linear control law to be of practical use.



Figure 5.217: Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.222: Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)

5.5.7.5 Initial Condition $\frac{h^2}{2T} = 362$

The initial condition $\frac{h^2}{2T} = 362$ for the asymmetric ellipsoid (a = 0.8, b = 0.9, c = 1) corresponds to a initial pure state of rotation about the unstable major axis. As per section 5.4, the linear control law enables the module to reach its target anchoring location within 2.2 normalised seconds. As per figures 5.223 and 5.231 below, the nutation angle and the object's rotational kinetic energy diverge immediately for MR = 10 and MR = 100 and after 10 normalised seconds for MR = 1,000 and MR = 10,000. Both the rotational kinetic energy and nutation angle diverge exhibiting high frequencies with large changes in magnitude which are indicative of an accelerating tumbling state. Instability sets in very shortly after the module deployed successfully. The instability and deployment timescales are nearly concurrent and therefore way too close in time for the linear control law to be of practical use.



Figure 5.227: Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.232: Phase Diagrams for the Z Axis (a=0.8 b=0.9 c=1)

5.5.8 Conclusions

The linear control law is not stabilising. Its timescale is very short with the module reaching its anchoring location within normalised seconds. Rotational instability sets in almost immediately after. The shortness of this timescale corresponds to unrealistic high magnitude control inputs, as seen in section 5.2, which lead to an increasing and unbounded amount of rotational kinetic energy being transferred to the object. The linear control law cannot be used in practice. However, the analysis of its effect on the system over time has provided some interesting insights on the parameters influencing its nature and magnitude:

- 1. The mass ratio is the first parameter which bears an influence on the system's stability. A higher mass ratio implies that the module deployment induces lower rotational kinetic energy transfers to the object, i.e less controllability of the object rotational state and therefore means more stable initial conditions. Conversely, a lower mass ratio implies a greater ability to control the object rotational state. However, the relationship between rotational kinetic energy transfer to the object and mass ratio is not monotonous, particularly for symmetric objects where resonance of the nutation angle could be observed for specific mass ratio. For all the the symmetric ellipsoids in this section, the eigen values of the system's Jacobian have zero real parts and future work should include an analysis in the frequency domain to understand further the relationship between mass ratio and nutation angle.
- 2. The second parameter influencing the system's stability is the object's shape and the following general observations made were:
 - (a) The linear control law has a better rotational stability performance for asymmetric object than for symmetric ones. The more symmetrical the object, the worse is the linear control law's rotational stability performance.
 - (b) For asymmetric object cases under the control of the linear control law, initial angular velocities about non principal axes lead to more stable

outcome than initial angular velocities about principal axes.

(c) For symmetric ellipsoids, the system behaves in opposite ways depending on whether the object is prolate or oblate. For prolate objects, the closer the initial rotational state to a pure spin, the more stable the object's rotational state generated by the linear control law. For oblate objects, the closer the initial rotational state to a pure spin, the more unstable the object's rotational state generated by the linear control law. On a prolate object the module travels closer to the spin axis than on an oblate object. As the module's contribution to the moment of inertia of the system increases with the square of its distance to the spin axis, the destabilisation of the spin due to the change of mass distribution generated by the module deployment on a prolate object is less than on an oblate object. Even more so near the tip of the spin axis where the variation of distance to the spin axis is minimum on a prolate object while it is maximum on an oblate object.

In the next section, the performance of linear and nonlinear laws will be explicitly compared.

5.6 Comparisons Between Linear and Nonlinear Model Predictive Control Laws With a Convergence to a State of Spin Objective

In this section, the linear and nonlinear control laws' performance are explicitly compared with an analysis of the system's behaviour over an unbounded time span where the module is left free to carry on moving beyond its target anchoring location. An examination of the physical effects on and stability of the system was conducted in order to compare the timescale difference and overall system's stability for all initial conditions.

5.6.1 Origin of the Data

The data displayed in this section corresponds to the following specific scenario:

- 1. Both the linear and nonlinear MPC controllers track a pure state of spin for the object with a target spin rate $\omega_0 \neq 0$ for the angular velocity part of the system state.
- 2. All initial rotational state of the object are represented: for symmetric objects $W_{zinit} \in \{0.5, 0.75, 1\}$ and for asymmetric objects

$$\tfrac{h^2}{2T} \in \{I_{min}, \tfrac{I_{min}+I_{mid}}{2}, I_{mid}, \tfrac{I_{max}+I_{mid}}{2}, I_{max}\}.$$

- 3. The module is not constrained to stop at its target location and is left free to move on the surface of the object with the sole objective of tracking a pure state of spin for the object. The aim is to analyse the system's behaviour near the origin over a large timescale.
- 4. The mass ratio of the object's to the module is fixed at 10,000.

5.6.2 Data Description

The data is displayed for specific shapes representative of a spectrum of realistic situations the robot could encounter. These shapes encompass a sphere, one oblate, one prolate and one asymmetric ellipsoids which are:

Ellipsoid Type	a	b	c	System Stability Status
Prolate	$\frac{1}{2}$	$\frac{1}{2}$	1	Potentially Stable
Sphere	1	1	1	Potentially Stable
Asymmetric	$\frac{1}{10}$	$\frac{8}{10}$	1	Potentially Stable
Asymmetric	$\frac{8}{10}$	$\frac{9}{10}$	1	Potentially Stable (Close to Unstable)
Oblate	2	2	1	Potentially Stable

Table 5.7: Ellipsoids Normalised Semi-Axes Lengths

All the results obtained across all simulation parameters were identical to those produced when the objective was to track a despun state.

In each of the following sections, the system's stability is examined with phase diagrams along with the evolutions of the nutation angle and of the object's rotational kinetic energy. Because of the difference in scale between the linear and nonlinear case, two sets of graphs were produced. One combines the linear and nonlinear control laws results. The other only displays the nonlinear control laws results.

5.6.3 Sphere Case

Figures 5.234 to 5.241 below show smooth curves for the nonlinear MPC control law with no oscillations and a nearly closed orbit for the object's angular velocity which is indicative of neutral stability over the entire observation period.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law maintains the initial rotational state close to a pure state of spin which, in the case of a sphere, corresponds to the maintenance of the direction of the spinning axis. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is akin to a typical regulator response with an overshoot at the onset of the process followed by a decrease towards the reference value. The evolution of the rotational kinetic energy is also consistent with the evolution of the nutation angle. When the rotational kinetic energy decreases, the nutation angle increases and vice versa.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to about 8 hours which is perfectly acceptable for a space application.

The nonlinear MPC control law performs according to its specifications providing reasonable stability as well as control commands with realistic and feasible magnitudes as per section 5.2.





Figure 5.236: Sphere

Figure 5.237: Stability Diagrams for the Sphere (a=1 b=1 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.239: Sphere

Figure 5.241: Sphere

Figure 5.242: Stability Diagrams for the Sphere (a=1 b=1 c=1)

5.6.4 Symmetric Case (a = 0.5, b = 0.5, c = 1)

In this section, the MPC and NMPC control laws are compared for a prolate ellipsoidal object.

5.6.4.1 Initial Condition Wzinit = 0.5

Figures 5.243 to 5.251 below show smooth curves for the nonlinear MPC control law with no oscillations and a closing orbit for the object's angular velocity which is indicative of neutral stability over the entire observation period.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law does not converge to a pure state of spin but maintains the initial rotational state, i.e. the initial nutation angle. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is akin to a typical regulator response with an overshoot at the onset of the process followed by a decrease towards the reference value. The evolution of the rotational kinetic energy is also consistent with the evolution of the nutation angle. When the rotational kinetic energy increases, the nutation angle increases at first and then converge back down to its initial value.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to about 21 hours which is perfectly acceptable for a space application.

The nonlinear MPC control law performs according to its specifications providing reasonable stability as well as control commands with realistic and feasible magnitudes as per section 5.2.



Figure 5.244: Wzinit = 0.5

Figure 5.246: Wzinit = 0.5

Figure 5.247: Phase Diagrams for the Z Axis Wzinit = 0.5 (a= 0.5 b= 0.5 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.249: Wzinit = 0.5

Figure 5.251: Wzinit = 0.5

Figure 5.252: Phase Diagrams for the Z Axis Wzinit = 0.5 (a= 0.5 b= 0.5 c=1)

5.6.4.2 Initial Condition Wzinit = 0.75

Figures 5.253 to 5.261 below show smooth curves for the nonlinear MPC control law with no significant oscillations and a closing orbit for the object's angular velocity which is indicative of neutral stability over the entire observation period.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law does not converge to a pure state of spin but maintains the initial rotational state, i.e. the initial nutation angle. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is akin to a typical regulator response with an overshoot at the onset of the process followed by a decrease towards the reference value. The evolution of the rotational kinetic energy is slightly inconsistent with the expected evolution of the nutation angle since both exhibit the same monotony as opposed to inverted monotony. Other factors, such as the change in mass distribution and its timing with respect to the object's rotational state which are not directly taken into account in this study, can influence this behaviour. The Wzinit = 0.75 initial condition for prolate ellipsoids is the most difficult to control out of the three initial conditions presented in this study.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to 1.5 days which is perfectly acceptable for a space application.

The nonlinear MPC control law performs according to its specifications providing reasonable stability as well as control commands with realistic and feasible magnitudes as per section 5.2.



Figure 5.254: Wzinit = 0.75

Figure 5.256: Wzinit = 0.75

Figure 5.257: Phase Diagrams for the Z Axis Wzinit = 0.75 (a = 0.5 b = 0.5 c = 1)


Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.259: Wzinit = 0.75

Figure 5.261: Wzinit = 0.75

Figure 5.262: Phase Diagrams for the Z Axis Wzinit = 0.75 (a= 0.5 b= 0.5 c=1)

5.6.4.3 Initial Condition Wzinit = 1

Figures 5.263 to 5.271 below show smooth curves for the nonlinear MPC control law with no oscillations and a closing orbit for the object's angular velocity which is indicative of neutral stability over the entire observation period.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law does not converge to a pure state of spin but maintains the initial rotational state, i.e. the initial nutation angle. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is akin to a typical regulator response with an overshoot at the onset of the process followed by a decrease towards the reference value. The evolution of the rotational kinetic energy is also consistent with the evolution of the nutation angle. When the rotational kinetic energy decreases, the nutation angle increases and vice versa.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to 2 days which is perfectly acceptable for a space application. The Wzinit = 1 initial condition for prolate ellipsoids is the most perturbable of the three initial conditions presented in this study and is therefore requires control over a longer period of time.



Figure 5.264: Wzinit = 1

Figure 5.266: Wzinit = 1

Figure 5.267: Phase Diagrams for the Z Axis Wzinit = 1 (a = 0.5 b = 0.5 c = 1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.269: Wzinit = 1

Figure 5.271: Wzinit = 1

Figure 5.272: Phase Diagrams for the Z Axis Wzinit = 1 (a = 0.5 b = 0.5 c = 1)

5.6.5 Symmetric Case (a = 2, b = 2, c = 1)

In this section, the MPC and NMPC control laws are compared for an oblate ellipsoidal object.

5.6.5.1 Initial Condition Wzinit = 0.5

Figures 5.273 to 5.281 below show smooth curves for the nonlinear MPC control law with no oscillations and a closing orbit for the object's angular velocity which is indicative of neutral stability over the entire observation period.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law does not converge to a pure state of spin but maintains the initial rotational state, i.e. the initial nutation angle close to its initial value here 3% higher. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is akin to a typical regulator response with an overshoot at the onset of the process followed by a decrease towards the new nutation angle value. The evolution of the rotational kinetic energy is also consistent with the evolution of the nutation angle. When the rotational kinetic energy decreases, the nutation angle increases at first and then converge back down to its initial value when the rotational kinetic energy settles.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to about 21 hours which is perfectly acceptable for a space application.



Figure 5.274: Wzinit = 0.5

Figure 5.276: Wzinit = 0.5

Figure 5.277: Phase Diagrams for the Z Axis Wzinit = 0.5 (a= 2 b= 2 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.279: Wzinit = 0.5

Figure 5.281: Wzinit = 0.5

Figure 5.282: Phase Diagrams for the Z Axis Wzinit = 0.5 (a= 2 b= 2 c=1)

5.6.5.2 Initial Condition Wzinit = 0.75

Figures 5.283 to 5.291 below show smooth curves for the nonlinear MPC control law with no significant oscillations and a closing orbit for the object's angular velocity which is indicative of neutral stability over the entire observation period.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law does not converge to a pure state of spin but maintains the initial rotational state, i.e. the initial nutation angle close to its initial values, here %0.02 lower. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is akin to a typical regulator response with an overshoot at the onset of the process followed by a decrease towards the reference value. The evolution of the rotational kinetic energy is also consistent with the evolution of the nutation angle. When the rotational kinetic energy decreases, the nutation angle increases at first and then decreases as the rotational kinetic energy increases again. The Wzinit = 0.75 initial condition for oblate ellipsoids is the most difficult to control out of the three initial conditions presented in this study.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to 1.5 days which is perfectly acceptable for a space application.



Figure 5.284: Wzinit = 0.75

Figure 5.286: Wzinit = 0.75





Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.289: Wzinit = 0.75

Figure 5.291: Wzinit = 0.75

Figure 5.292: Phase Diagrams for the Z Axis Wzinit = 0.75 (a = 2 b = 2 c = 1)

5.6.5.3 Initial Condition Wzinit = 1

Figures 5.293 to 5.301 below show smooth curves for the nonlinear MPC control law with no oscillations and a closing orbit for the object's angular velocity which is indicative of neutral stability over the entire observation period.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law does not converge to a pure state of spin but maintains the initial rotational state, i.e. the initial nutation angle close to its initial value, here 0.03% higher. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is akin to a typical regulator response with an overshoot at the onset of the process followed by a decrease towards the reference value. The evolution of the rotational kinetic energy is also consistent with the evolution of the nutation angle. When the rotational kinetic energy decreases, the nutation angle increases and vice versa.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to less than 2 days which is perfectly acceptable for a space application. The Wzinit = 1 initial condition for prolate ellipsoids is the most perturbable of the three initial conditions presented in this study and is therefore requires control over a longer period of time.



Figure 5.294: Wzinit = 1

Figure 5.296: Wzinit = 1

Figure 5.297: Phase Diagrams for the Z Axis Wzinit = 1 (a= 2 b= 2 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.299: Wzinit = 1

Figure 5.301: Wzinit = 1

Figure 5.302: Phase Diagrams for the Z Axis Wzinit = 1 (a = 2 b = 2 c = 1)

5.6.6 Symmetric Case (a = 0.1, b = 0.8, c = 1)

In this section, the MPC and NMPC control laws are compared for an asymmetric ellipsoidal object whose geometry is far from the limit case of the sphere.

5.6.6.1 Initial Condition $\frac{h^2}{2T} = 130$

Examining figures 5.303 to 5.311 below, the application of the nonlinear MPC control law results in a closing orbit for the object's angular velocity which is indicative of neutral stability over the entire observation period.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law maintains the initial object's rotational state which in this case is a spin about the minor principal axis. The final nutation angle is close to its initial value, here 0.02% higher. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations. The nutation angle ends up in a steady state with a low frequency and amplitude oscillatory motion which is acceptable given the unstable nature of the minor principal axis.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is akin to a typical regulator response with an overshoot at the onset of the process followed by a decrease towards the reference value. The evolution of the rotational kinetic energy is also consistent with the evolution of the nutation angle. When the rotational kinetic energy decreases, the nutation angle increases and vice versa.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to less than 2.5 days which is perfectly acceptable for a space application.



Figure 5.307: Phase Diagrams for the Z Axis HT = 130 (a= 0.1 b= 0.8 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.312: Phase Diagrams for the Z Axis HT = 130 (a= 0.1 b= 0.8 c=1)

5.6.6.2 Initial Condition $\frac{h^2}{2T} = 166$

Examining figures 5.313 to 5.321 below, the application of the nonlinear MPC control law results in a fully closed orbit for the object's angular velocity which is usually indicative of neutral stability.

However, this is a deceptive view. The time evolution of the nutation angle shows large oscillations with an amplitude equal to 50% of the initial nutation angle value and a period equals to 250 normalised seconds, corresponding to a 21-hour period for the object sized in chapter 3 section 3.5.1.2. The object is tumbling and the nonlinear control law has failed to stabilise its rotational state. Although the linear MPC control law leads to a divergent rotational state, it performs better and maintains the nutation angle within 5% of its initial value over the control period. Tumbling is initiated from an initial rotation state about a non principal axis by a drop in rotational kinetic energy and once this rotational kinetic energy increases back to a constant level, the object's tumbling state is steady from then on.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to less than 1.5 days.

The nonlinear MPC control law does not perform according to its stability specifications.



Figure 5.317: Phase Diagrams for the Z Axis HT = 166 (a= 0.1 b= 0.8 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.322: Phase Diagrams for the Z Axis HT = 166 (a= 0.1 b= 0.8 c=1)

5.6.6.3 Initial Condition $\frac{h^2}{2T} = 202$

Examining figures 5.323 to 5.331 below, the application of the nonlinear MPC control law results in the divergence of the nutation angle to a value equal to $\frac{\pi}{2}$. The initial object's rotational state corresponds to a pure spin about the medium unstable principal axis. The initial drop of rotational energy leads to a change of axis of rotation for one which is perpendicular to the medium principal axis, most likely the major principal axis for which a rotation is more stable. This is confirmed by the rotational kinetic energy level which reaches a steady state constant value once the transition from one axis to the other is achieved.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to less than 1 day and 18 hours which is perfectly acceptable for a space application.

The nonlinear MPC control law does not perform according to its stability specifications. However, in this case, it leads to a better more stable outcome.



Figure 5.327: Phase Diagrams for the Z Axis HT = 202 (a= 0.1 b= 0.8 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.332: Phase Diagrams for the Z Axis HT = 202 (a= 0.1 b= 0.8 c=1)

5.6.6.4 Initial Condition $\frac{h^2}{2T} = 265$

Examining figures 5.333 to 5.341 below, the application of the nonlinear MPC control law results in a fully closed orbit for the object's angular velocity which is usually indicative of neutral stability.

However, this is a deceptive view. The time evolution of the nutation angle shows large oscillations with an amplitude equal to 50% of the initial nutation angle value to reach $\frac{\pi}{2}$ and a period equals to 200 normalised seconds, corresponding to a 17-hour period for the object sized in chapter 3 section 3.5.1.2. The object is tumbling and the nonlinear control law has failed to stabilise its rotational state. Although the linear MPC control law leads to a divergent rotational state, it performs better and maintains the nutation angle within 5% of its initial value over the control period. Tumbling is initiated from an initial rotation state about a non principal axis by a drop in rotational kinetic energy and once this rotational kinetic energy increases back to a constant level, the object's tumbling state is steady from then on.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to less than 1.5 days.

The nonlinear MPC control law does not perform according to its stability specifications.



Figure 5.337: Phase Diagrams for the Z Axis HT = 265 (a= 0.1 b= 0.8 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.342: Phase Diagrams for the Z Axis HT = 265 (a= 0.1 b= 0.8 c=1)

5.6.6.5 Initial Condition $\frac{h^2}{2T} = 328$

Examining figures 5.343 to 5.351 below, the application of the nonlinear MPC control law results in a closing orbit for the object's angular velocity which is indicative of neutral stability over the entire observation period.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law maintains the initial object's rotational state which in this case is a spin about the major principal axis. The final nutation angle is close to its initial value, here less than 0.01% lower. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations. The nutation angle ends up in a steady state with a low frequency and amplitude oscillatory motion which is acceptable given that rotation about the major principal axis is more stable than about other axes.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is akin to a typical regulator response with an undershoot at the onset of the process followed by an increase towards the reference value. The evolution of the rotational kinetic energy is also consistent with the evolution of the nutation angle. When the rotational kinetic energy increases, the nutation angle decreases and vice versa.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to less than 3.5 days which is perfectly acceptable for a space application.



Figure 5.347: Phase Diagrams for the Z Axis HT = 328 (a= 0.1 b= 0.8 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.352: Phase Diagrams for the Z Axis HT = 328 (a= 0.1 b= 0.8 c=1)

5.6.7 Symmetric Case (a = 0.8, b = 0.9, c = 1)

In this section, the MPC and NMPC control laws are compared for an asymmetric ellipsoidal object whose geometry is close to the limit case of the sphere. The system's behaviour in this case is close that of a system with a prolate ellipsoid.

5.6.7.1 Initial Condition $\frac{h^2}{2T} = 290$

Examining figures 5.353 to 5.361 below, the application of the nonlinear MPC control law results in an open orbit for the object's angular velocity. From this shape no conclusion can be drawn as to the stability of the system. However the magnitude of the changes of the angular velocity is of the order of $2.5 \cdot 10^{-4} \ rad.s^{-1}$ which is negligible enough to be able to conclude that over the course of the observation period, the system is stable.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law maintains the initial object's rotational state which in this case is a spin about the minor principal axis. The final nutation angle is close to its initial value, here 0.05% higher. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is a smooth and close to a linear increase with no oscillations which is consistent with the evolution of the rotational kinetic energy which steadily decreases over the observation period.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to less than 12 hours which is perfectly acceptable for a space application.



Figure 5.357: Phase Diagrams for the Z Axis HT = 290 (a= 0.8 b= 0.9 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.362: Phase Diagrams for the Z Axis HT = 290 (a= 0.8 b= 0.9 c=1)

5.6.7.2 Initial Condition $\frac{h^2}{2T} = 309$

Examining figures 5.363 to 5.371 below, the application of the nonlinear MPC control law results in an open orbit for the object's angular velocity. From this shape no conclusion can be drawn as to the stability of the system. However the magnitude of the changes of the angular velocity is of the order of $3 \cdot 10^{-2} \ rad.s^{-1}$ which is negligible enough to be able to conclude that over the course of the observation period, the system is stable.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law maintains the initial object's rotational state which in this case is a spin about a non principal axis. The final nutation angle is close to its initial value, here 7% higher. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is a smooth and close to a linear increase with no oscillations which is consistent with the evolution of the rotational kinetic energy which is nearly constant over the observation period.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to less than 12 hours which is perfectly acceptable for a space application.



Figure 5.367: Phase Diagrams for the Z Axis HT = 309 (a= 0.8 b= 0.9 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.372: Phase Diagrams for the Z Axis HT = 309 (a= 0.8 b= 0.9 c=1)

5.6.7.3 Initial Condition $\frac{h^2}{2T} = 328$

Examining figures 5.373 to 5.381 below, the application of the nonlinear MPC control law results in an open orbit for the object's angular velocity. From this shape no conclusion can be drawn as to the stability of the system. However the magnitude of the changes of the angular velocity is of the order of $4 \cdot 10^{-2} \ rad.s^{-1}$ which is negligible enough to be able to conclude that over the course of the observation period, the system is stable.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law maintains the initial object's rotational state which in this case is a spin about the medium principal axis. The final nutation angle is close to its initial value, here 3.5% higher. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is a smooth and close to a linear increase with no oscillations which is the same for the evolution of the rotational kinetic energy which increases over the observation period. This is consistent with the fact that the initial rotational state of the system is a pure spin about the medium unstable axis. This initial state is in the process of degenerating towards a more stable state with the input of more rotational kinetic energy.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to less than 12 hours which is perfectly acceptable for a space application.



Figure 5.377: Phase Diagrams for the Z Axis HT = 328 (a= 0.8 b= 0.9 c=1)


Figure 5.382: Phase Diagrams for the Z Axis HT = 328 (a= 0.8 b= 0.9 c=1)

5.6.7.4 Initial Condition $\frac{h^2}{2T} = 345$

Examining figures 5.383 to 5.391 below, the application of the nonlinear MPC control law results in an open orbit for the object's angular velocity. From this shape no conclusion can be drawn as to the stability of the system. However the magnitude of the changes of the angular velocity is of the order of $2 \cdot 10^{-2} \ rad.s^{-1}$ which is negligible enough to be able to conclude that over the course of the observation period, the system is stable.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law maintains the initial object's rotational state which in this case is a spin about a non principal axis. The final nutation angle is close to its initial value, here 0.01% higher. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is a smooth and close to a linear increase with no oscillations which is consistent with the evolution of the rotational kinetic energy which is nearly constant over the observation period.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to less than 7 hours which is perfectly acceptable for a space application.

The nonlinear MPC control law performs according to its specifications providing reasonable stability as well as control commands with realistic and feasible magnitudes as per section 5.2.



Figure 5.387: Phase Diagrams for the Z Axis HT = 345 (a= 0.8 b= 0.9 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.392: Phase Diagrams for the Z Axis HT = 345 (a= 0.8 b= 0.9 c=1)

5.6.7.5 Initial Condition $\frac{h^2}{2T} = 362$

Examining figures 5.393 to 5.401 below, the application of the nonlinear MPC control law results in a closed orbit for the object's angular velocity which is indicative of stability. The magnitude of the changes of the angular velocity is of the order of $1.6 \cdot 10^{-3} \ rad.s^{-1}$ which is negligible enough.

Owing to the large mass ratio MR = 10,000 which renders the system less controllable, the nonlinear control law maintains the initial object's rotational state which in this case is a spin about the major principal axis. The final nutation angle is close to its initial value, here 0.01% lower. The magnitude of the nutation angle variations is orders of magnitude less than the variations obtained with the linear control law as is the magnitude of the rotational kinetic energy variations.

The pattern of the dynamic evolution of the nutation angle under the nonlinear control law is to drop at the onset before increasing back to a steady state value with no oscillations. It is consistent with the evolution of the rotational kinetic energy which mirrors that of the nutation angle over the observation period.

The timescale of the nonlinear control law is 20 times larger than the timescale of the linear control law. For the object sized in chapter 3 section 3.5.1.2, it corresponds to less than 3 days which is perfectly acceptable for a space application.

The nonlinear MPC control law performs according to its specifications providing reasonable stability as well as control commands with realistic and feasible magnitudes as per section 5.2.



Figure 5.397: Phase Diagrams for the Z Axis HT = 362 (a= 0.8 b= 0.9 c=1)



Chapter 5: Feasibility Study and Validation: Simulation of the Deployment of a Single Robot Module

Figure 5.402: Phase Diagrams for the Z Axis $\frac{h^2}{2T} = 362$ (a= 0.8 b= 0.9 c=1)

5.6.8 Conclusions

The purpose of this section was to compare for all initial conditions the performance of the linear and nonlinear MPC control laws with respect to their timescale and to the overall system's stability. The nonlinear MPC controller provides much more accurate predictions of the system's behaviour. It is clearly the best controller out of the two and is the one controller to be implemented.

The comparison between the timescale of the linear and nonlinear control laws led to the conclusion that the nonlinear MPC control law is perfectly acceptable for a space mission. Indeed:

- 1. The timescale of the nonlinear MPC control law is 20 times larger than that of the linear MPC control law.
- 2. The timescale of the nonlinear MPC control law could even be slower to account for other constraints. For a typical asteroid such as the one sized in chapter 3 section 3.5.1.2, the deployment of one module can occur within hours to days which is by far faster than the timescale of any perturbation.
- 3. Finally, the timescale of the nonlinear control law becomes longer with:
 - (a) Decreasing initial nutation angles.
 - (b) Initial rotational states of the object closer to a pure state of spin.
 - (c) Initial rotational states closer to spins about principal axes.
 - (d) Initial rotational states closer to spins about stable axes.
- 4. For symmetric ellipsoids, the timescale of the nonlinear MPC law doubles for the initial condition $W_{z_{init}} = 0.5$ to the initial condition $W_{z_{init}} = 1$

In this section, the nonlinear control law was set to track a pure state of spin for the object and was simulated with a large mass ratio MR = 10,000 which rendered the system less controllable. The control law did not lead to the convergence of the system angular velocity state to a state of pure spin. Instead it maintained the object's initial rotational state while avoiding divergence which is the main downfall of the linear control law. The stability of the rotational state is affected by its initial state and the shape of the object as follows:

- 1. The larger the initial nutation angle, the more stable this initial nutation angle is with the nonlinear control law.
- 2. An initial pure state of spin is harder to maintain than a nutated one.
- 3. After the module has successfully deployed, the nutation angle has changed its initial value by an offset value, albeit very small.
- 4. Symmetric oblate ellipsoids are more controllable when the module is away from the spin axis and symmetric prolate ellipsoids are more controllable when the module is close to the spin axis.
- 5. For asymmetric ellipsoids, it is easier for the nonlinear controller to control an initial spin about a principal axis than an initial tumbling state. An initial tumbling state leads to further tumbling.
- 6. For asymmetric ellipsoids, if the initial rotating state is close to a spin about the medium principal axis, the controller drives the rotational state to a rotation about a more stable axis most likely the major principal axis.
- 7. For asymmetric ellipsoids, The nutation angle always exhibits oscillations after the module deployment is achieved.

The nutation angle and rotational kinetic energy responses are generally smooth with a profile that would fit the response of a regulator. As per section 5.2, the nonlinear MPC control law can be considered as stabilising and as providing control commands with realistic and feasible magnitudes. The final object's rotational state is close to its initial rotational state and stable which is a favourable outcome compared to a diverging accelerated or tumbling rotational state.

However, as mentioned earlier, the nonlinear control law does not drive the object's rotational state to a pure state of spin. The controllability of the system is diminished by the high mass ratio between the object and the deploying module. Future work should endeavour to determine how to increase controllability. As such two avenues can be explored:

- 1. Perform a study similar to this section's with different mass ratios. Lowering the mass ratio should, as seen in the previous sections 5.4 and 5.5, increase the impact of the module deployment on the rotational state of the object and therefore, increase the system's controllability.
- 2. Explore further the impact of asymmetry on controllability by gauging whether controllability increases or decreases with asymmetry of the ellipsoid's shape along with determining and parametrising a set of geometries for which it is the case.

This section closes the data analysis of the feasibility and validation chapter. In the next section, a summary of the findings and contributions will be presented.

5.7 Conclusions and Recommendations

In this section, an overview of the performance of the low-level controller will be presented with respect to the main system parameters influencing its behaviour: the mass ratio between the object and the module, the object's geometry, i.e. its shape and the system's initial rotational state. Finally, conclusions will be drawn on the feasibility of the robotic structure proposed in this PhD study considering the system's controllability properties established in chapter 4.

5.7.1 Influence of the Mass Ratio

The mass ratio of the object's mass to the module's is the first parameter to consider since it bears the greatest influence on the system's stability and controllability. All other parameters being equal, the influence of the mass ratio can be summarised as follows:

1. The lower the mass ratio, the greater the rotational kinetic energy transfer between the module and the object.

- 2. Consequently as a general rule:
 - (a) The lower the mass ratio, the greater the controllability on the rotational state of the system. However, this increase in controllability has to be traded off with an increase in instability and magnitude of divergence of the system's rotational state, i.e. an increase in divergence of the nutation angle and angular velocity magnitude.
 - (b) The higher the mass ratio, the lower the controllability on the rotational state of the system which leads to a higher stability of the system's initial rotational state. The system's initial rotational state deviation from its initial value decreases by an order of magnitude for each mass ratio increase by the same order of magnitude.
- 3. The relationship between rotational kinetic energy transfer to the object and mass ratio is however not monotonous, particularly for symmetric objects where resonance of the nutation angle could be observed for specific mass ratios.

5.7.2 Influence of the Object's Shape

The object's shape, i.e. the type of ellipsoid it is, is the second most influential parameter on the system's stability and controllability. Only systems with potentially stable shapes were evaluated in this PhD study. All other parameters being equal, the influence of the object's shape can be summarised as the following sections:

5.7.2.1 Influence of the Object Shape in Relation to The Sphere Limit case

The following conclusions have to be understood from the standpoint of the object's shape departure from the limit case of the sphere. In this perspective, the influence of the shape is as follows:

1. Systems with asymmetric objects are more resistant to divergence than systems with a symmetric object.

- 2. The more symmetric the object, the worse is the control law's rotational stability performance, i.e. the more the linear control commands tend to diverge.
- 3. Stability increases with prolateness and oblateness when departing from an initial spherical shape.
- 4. Finally, the timescale of the system increases with symmetry. It decreases with prolateness and oblateness when departing from an initial spherical shape and the limit case of the sphere has the largest timescale.

5.7.2.2 Influence of the Symmetric Shapes

All other parameters being equal, oblate objects have higher levels of rotational kinetic energy than prolate objects. Moreover, on a prolate object the module travels closer to the spin axis than on an oblate object. This induces a greater destabilisation, especially near the tip of the spin axis on an oblate object than on a prolate object. Consequently:

- 1. Symmetric ellipsoid were more susceptible to nutation increase when their original state was a pure spin.
- 2. Symmetric oblate ellipsoids are more controllable when the module is away from the spin axis and symmetric prolate ellipsoids are more controllable when the module is close to the spin axis.
- 3. Stability decreases with oblateness and increases with prolateness.
- 4. Finally, coverage increases with prolateness and decreases with oblateness.

5.7.2.3 Influence of the Asymmetric Shapes

The specific characteristics that can be attributed to the shape of asymmetric objects are the following:

1. Initial angular velocities about non principal axes lead to more stable outcome than initial angular velocities about principal axes.

- 2. The nutation angle always exhibits oscillations after the module deployment is achieved.
- 3. Stability increases with a decreasing minor axis moment of inertia about this moment of inertia.

5.7.3 Influence of the Initial Conditions

Initial conditions refer here to initial rotational state of the object prior to the module's deployment. The general rule with initial conditions is that the closer the rotational state to its spinning state about a principal axis, the harder it is to control, especially about the minor unstable axis. Consequently:

- 1. An initial pure state of spin is harder to maintain than a nutated one.
- 2. The larger the initial nutation angle, the more stable this initial nutation angle is with the nonlinear control law.
- 3. For asymmetric ellipsoids, it is easier for the nonlinear controller to control an initial spin about a principal axis than an initial tumbling state. An initial tumbling state leads to further tumbling.
- 4. For asymmetric ellipsoids, if the initial rotating state is close to a spin about the medium principal axis, the controller drives the rotational state to a rotation about a more stable axis most likely the major principal axis.

For symmetric ellipsoids, the timescale of the nonlinear MPC law doubles for the initial condition $W_{z_{init}} = 0.5$ to the initial condition $W_{z_{init}} = 1$ and initial conditions have a particular impact on the timescale of the nonlinear control law which becomes longer with:

- 1. Decreasing initial nutation angles.
- 2. Initial rotational states of the object closer to a pure state of spin.
- 3. Initial rotational states closer to spins about principal axes.
- 4. Initial rotational states closer to spins about stable axes.

5.7.4 Energy Levels and Controller Validation

The assessment of how realistic are the object's levels of rotational kinetic energy offers important insights as to the validity of the controller. The linear control law produces large and diverging control commands which corresponds to unrealistic energy requirements. The nonlinear law on the other hand provides far more reasonable results. However, these evaluations are made by visual inspection for normalised values of rotational kinetic energy. An order of size analysis in energy units is required to evaluate the feasibility of the robotic structure proposed in this PhD study. The following considerations were made:

- 1. The system was assumed to be isolated and with a constant total mechanical energy. As there are only two distinct parts in the system: the object and the module, the extensive property of energy implies that the evolution of the module's energy could be known as the opposite of the object's kinetic energy, the object having no potential energy. The module's total energy comprises its potential energy stored in the batteries, its total rotational kinetic energy and the deformation energy which is the relative linear kinetic energy of its relative motion from the object. The potential energy stored in the module's batteries was assumed to be unbounded a priori for the purpose of this feasibility study.
- 2. In many figures displayed in this study such as figure 5.241 for instance, the object's rotational kinetic energy decreases and increases during the deployment process which is indicative of back and forth transfer of energy from and to the object. The relative angular velocity of the module is the time integral of the the module's relative rotational acceleration and as per section 5.2, the low-level module controller can output positive or negative control commands. Moreover, as per the system's design, energy should be transferred from the battery to the module as kinetic energy. Considering all this, the only way for the object's rotational kinetic energy to be transferred out of the object is to the module. The total system's rotational kinetic energy should therefore increase during the module deployment. In order to study this aspect as well as limit this study to realistic situations, the battery should be modelled as

part of the system design.

- 3. There is no provision for a dissipative energy mechanism for the system in the current design of the low-level module controller. A nutation damper should be considered if a strategy involving despinning the system is envisaged.
- 4. As per the above two point, despinning the object is not achievable under the current controller design.
- 5. The time evolution of the discrete normalised model 3.24 with respect to an initial value of the relative angular velocity of the module and in the absence of control commands should be simulated in order to understand how the energy transfers occur between the object and the module and to assess their magnitude.
- 6. A coarse order of size analysis using the asteroid data sized in chapter 3 section 3.5.1.2 in conjunction with the normalised rotational kinetic energy data from figure 5.241 shows that, for a mass ratio MR = 10,000, a 2300 T module travelling on the surface of a 100-m radius asteroid would cover its circumference in 1.5 minutes at 6.6 $m \cdot s^{-1}$, i.e. 23.7 $km \cdot h^{-1}$. This corresponds to the velocity of a cyclist and is a priori unrealistic for the intended space application evaluated in this study. However, this velocity is perfectly reasonable from the point of view of an exploration study with no constraints on the module's energy resource and the proposed solution cannot be dismissed on physical and engineering grounds at this stage. A priori, it would suffice to reduce the module's relative angular velocity to lower the energy expenditure as well as to lower levels of disturbance to the object's angular velocity.

5.7.5 Low-Level Controller Performance and Final Recommendations

In this final section, the observed performance of the low-level module controller is evaluated and final conclusions are drawn on the suitability of the robotic structure proposed in this PhD study.

5.7.5.1 Low-Level Controller Design Limitations

The low-level module controller has significant design limitations which are important to consider when evaluating its performance:

- 1. First, as per chapter 4, the system is not controllable. The system's controllability matrix has at best a rank of 6 or a rank of 4 for the despin tracking case and for all simulated systems, the object's shapes were chosen such that the eigen values of the system's Jacobian had zero real parts, i.e. the system were potentially stable.
- The dimension of the control space is 1. The control commands φ and θ are coupled which is consistent with the lack of controllability of the system. From a pure empirical observation, it is not possible to know whether the control law is linear or nonlinear and time constant or time variable.
- 3. The nutation angle is relevant as a stability indicator for systems with symmetric object only. Nonetheless, the nutation angle indicates the location of the axis of rotation with respect to a reference axis of rotation towards which the system should converge. In this respect, it is perfectly relevant for the asymmetric case.

5.7.5.2 Low-Level Controller Performance Assessment

Considering the performance of the low-level module controller, the following assessments were made:

- Irrespective of size, shape and initial angular velocity, each object initial angular velocity was disturbed by the motion of the module or mass at the object's surface. As expected the motion of the module generated system deformation, moments and rotational kinetic energy transfers between the object and the module. The physical modelled derived in chapter 3 is accurate enough.
- 2. The low-level module's controller objective of deploying the module to a target anchoring location on the surface of the object is always achieved and is

achieved independently from the mass ratio of the object's to the module's. However, coverage, defined as a measure of reachability of any point of the object's surface from the robotic structure, is only half of what it should be with only half of the object's top hemisphere meridians covered by the module trajectory. This is a consequence of underactuation as per section 5.7.5.1 which limits the ability of the low-level module controller to actively shape this trajectory. The low-level module's controller does not provide enough controllability.

- 3. As a general rule except for a minority of cases, increasing rotational kinetic energy is correlated to decreasing nutation angle which is very positive if the controller's objective is to converge to a final pure state of spin. However, both the linear and nonlinear control laws do not drive the object's rotational state to a pure state of spin or despun state within the deployment timespan. After the module has successfully deployed, the nutation angle's final value has approximatively the same biased value, albeit small, compared to its target value and regardless of the initial conditions. Moreover, for both the pure spin or despin tracking cases, the results are identical which, in light of the lack of controllability of the system, puts into question the ability of the controller to make the rotational state of the system converge to target final rotational state. Consequently, for both the linear and nonlinear control laws, the system's objective of reaching either a pure state of spin a despun state is not achieved within the deployment timespan and a precise pointing direction for the rotation axis cannot be achieved with the current version of the controller and the current deployment strategy.
- 4. The timescale of the nonlinear MPC control law is 20 times larger than that of the linear MPC control law. This corresponds to a two to seven days simulated time for the nonlinear controller against about 15 minutes for the linear control law. For a typical asteroid such as the one sized in chapter 3 section 3.5.1.2, this translates into a module deployment time of hours to days which is by far faster than the timescale of any perturbation. This timescale is excellent for

a space application.

- 5. Module deployment and convergence to a target rotational state have different timescale which results in the module having a residual angular velocity relative to the object upon reaching its target anchoring location. This is a failure of the controller which should ideally behave like a regulator and lead to a 0 angular velocity relative to the object upon reaching the target.
- 6. Looking at the performance of the control laws over time past the module's target anchoring point, the linear control law leads the system's rotational state to diverge with high energy inputs while the nonlinear control law is stabilising leading the relative angular velocity to 0. The nonlinear MPC control law also provides control commands with realistic and feasible magnitudes leading the nutation angle and rotational kinetic energy to respond with generally smooth profiles. The nonlinear MPC control law has much better stabilising performances than te linear control law.
- 7. Only the nonlinear model provides a sufficient level of accuracy for predicting the system's behaviour in order for the controller to produce a performant control law.
- 8. High mass ratios stiffen the system. Consequently, the system controllability decreases with increasing mass ratios between the object and the deploying module. However, increasing the mass ratio increases the stability of the system's initial rotational state which is a more favourable outcome than a diverging or tumbling rotational state.

5.7.5.3 Final Recommendations

Considering the performance of the low-level module controller in section 5.7.5.2, the following conclusions were drawn:

1. Given the performance exhibited by both the linear and nonlinear control laws, only the nonlinear model predictive controller is suitable for the robotic application proposed in this PhD study having both the ability to stabilise the system and to predict accurately the system's dynamic behaviour.

- 2. However, the system is not controllable. The model predictive controller controls the deployment state only and does not control the system's rotational state satisfactorily. The control commands are coupled with a control command space likely to be of dimension 1 instead of 2. Moreover, the timescale of the deployment is different from the timescale of the rotational state. The module deployment ends with the module having a residual relative angular velocity and a nutated final system rotational state with a risk of further divergence.
- 3. In the design proposed in this PhD study, the model predictive controller is only constrained by the optimisation of its cost function which enables the controller to make the system's rotational state converge partially to a target rotational state. Given the preceding point, the model predictive controller would be used most effectively in conjunction with a stiffening of the system with an increase of the mass ratio between the object and the module. This would allow the model predictive controller to successfully deploy the module while maintaining a stable system's initial rotational state. This later outcome would be a better than risking divergence while trying to converge to a pure state of spin or despun state, leaving this change of rotational state for a later stage potentially using other control methods.
- 4. The above observations imply that the proposed robotic system is suited for objects involving a mass ratio of at least 1000 or more. However, from an engineering standpoint with a real system in mind, a mass ratio of 10000 or above is to be preferred. This means that the proposed SR robot solution is more suited to large objects. The asteroid sized in section 3.5.1.2 provides a good example. With its mass of 2.30e + 10kg, it allows for the deployment a SR robotic structure whose mass can be up to 2300T. For a small satellite with mass of the order of a couple of tonnes, this solution is not realistic.

The limitations of the proposed model predictive controller design calls for improvements and / or a change of control strategy. As a first step, further constraints could be added to the model predictive controller:

- 1. An energy dissipation mechanism should be added to the self-reconfigurable robotic structure to allow the possibility of despinning the object.
- 2. A final penalty cost could be added to the cost function to try to enforce the convergence of the rotational state and eliminate the residual relative angular velocity as well as the timescale issue.
- 3. If stabilising the initial rotational state is the strategy of choice, the timescale could be expanded in order to try to decrease the rotational kinetic energy of the module. Further simulations would show whether the rotational kinetic energy transfer from the module to the object has a lower magnitude and disturbs less the system's initial rotational state.
- 4. A coverage constraint could be added to compensate for the restrictive effect that the control commands coupling has on the geometry of the module's deployment trajectory.
- 5. Finally, given that over time, the nonlinear control law's control commands tend to 0, it could be worth constraining the anchoring of the module to a location corresponding to a converged rotational state and observing the emerging trajectory and structure and evaluating whether they satisfy the structure continuity and coverage objectives.

The general conclusion of this thesis will be now be found in the next chapter.

Chapter 6

Conclusion

6.1 Work Undertaken, Contributions and Conclusions

The research carried out for this PhD thesis aimed at making a preliminary design and assessing the feasibility of a novel robotic structure to be deployed on or to help capturing randomly tumbling objects in space. The primary applications intended for this structure are mainly space mining, general space structures deployment, maintenance and debris removal.

The gap analysis in chapter 2 identified that the design of choice for this robotic structure should be a modular self-reconfiguring robot with a two-level decentralised module controller. The lower level of this module controller should be based on a physical model of the deployment of the robot on an uncooperative random-shaped object tumbling in a low-magnitude gravity field. The higher level of this module controller should take the form of a simple behaviour-based decentralized algorithm. The modules hardware should also be designed in such a way so as to be able to perform reconfiguration and also sense their environment. In particular, the modules should be equipped with gyroscopes to perceive the angular velocity and linear acceleration at the surface of the object.

Given the scope of this design study, it was narrowed down to a proof of concept i.e. to a physical analysis and to the individual module controller design to establish whether in principle such a robot design is feasible and could fulfil the application deployment requirements.

The research carried out was therefore narrowed down to the following specific study aims:

- 1. Model the dynamic interactions between an uncooperative random-shaped tumbling object with low-magnitude gravitation field and a modular device moving on the surface of this object in order to describe the effects of dynamic changes in mass distribution to the overall system and the resulting exchanges of angular momentum between the object and the device.
- 2. Derive from the above model a simple behaviour-based decentralized algorithm controlling the deployment of a modular self-reconfigurable robot over the surface of the object as a continuous chain of modules circling around the main spin axis. The controller resolves a constrained optimization or tracking problem in the sense that the deployment should make the object's rotational state converge to a reference rotational state.
- 3. Verify and validate the robot controller concept and correctness at the lower module level and at the higher robot level through computer simulations.

These study aims were met by the general and original contributions this thesis made. These were:

1. In chapter 3, the physical interactions between the robot and the object were modelled and the simulation designed. The physical model, originally derived in [66], was a general continuous model of a deformable rotating continuum. The system under study was defined as the isolated combination of the object and the robot at its surface. The simulation design consisted mainly in modelling the object as ellipsoids, normalising the model parameters, determining the initial conditions including the landing site, modelling perturbations as biases coloured noises and actuation and sensing errors as biased white noises. The specific original contributions of chapter 3 were:

- (a) A rederivation of the general continuous model of a deformable rotating continuum [66] with explicit vectors and tensors writing convention for the purpose of writing consistent and verifiable simulation code.
- (b) A discretisation of the general continuous model of a deformable rotating continuum explicitly separating the object as the rigid part and the robot and its modules as the discrete deformable parts along with a normalisation of all the model parameters rendering them dimensionless.
- (c) The parametrisation of the relative motion of each robot module with respect to the object with spherical coordinates to establish that only the azimuthal angle and polar angle angles were required for evaluating this relative interaction.
- (d) A derivation of the yo-yo de-spin mechanism equations from a continuum perspective to illustrate and validate the above general model on a simple two-dimensional example.
- (e) The modelling of the object as an ellipsoid parametrised by its normalised semi-axes lengths and moments of inertia.
- (f) The modelling of the radiation perturbations experienced by asteroids as a biased coloured noise.
- (g) The modelling of the model errors and gyroscope and actuator noise as an aggregate biased white noise.
- (h) An explicit and complete rederivation of the Euler equations for objects with unequal moments of inertia for simulating asymmetric torque-free rotating objects in order to provide a reference rotational state for the low-level module controller to track.
- (i) The determination of the optimal landing sites for the robot on the surface of the object at the tip of the current rotating axis by modelling the system (object and robot) as one body with time-variable mass with an instantaneous change of mass.
- (j) A sizing of the simulation integration step and of realistic initial condi-

tions for the system simulation.

- (k) The derivation of the parametrisation of the holonomic liaison between the object and each robot module.
- 2. In chapter 4, from its intended use as a space debris removal, spacecraft maintenance, asteroids capture and mining device, the study engineering application was precisely defined as the autonomous and controlled construction and deployment of decentralised modular and self-reconfigurable scaffolding structures around free-floating randomly tumbling objects with a low-magnitude gravitational field in space. The chapter first dealt with the definition of the robot task objectives and performance measurements. Broad assumptions were made for the robot engineering requirements and hardware design to allow focus on the preliminary design of a decentralised control strategy with two levels: one high-level behaviour-based component controlling the orderly deployment of the robot on the surface of the object and one low-level component controlling the module motion on the surface of the object by generating commands via a constrained optimisation using either a linear or non-linear model predictive control approach. In this perspective, the robot deployment or reconfigurations are task-driven and akin to a manipulation through reconfiguration of the entire robot body via changes of mass distribution. The combination of the decentralised behaviour-based and Model Predictive Control approaches with a manipulation through reconfiguration, angular momentum exchanges and mass distribution changes is a novel use of a SR robot. This new function aims at maintaining the entire system (object + robot) either in a state of pure spin or despun so as to keep a stable pointing direction for the main spinning axis for further processing such as an asteroid retrieval for instance. The specific original contributions of chapter 4 were:

The original contributions of this work were:

(a) The transformation of the discrete model (3.24) in chapter 3 into a nonlinear state space model (4.15) with an equilibrium point at its origin corresponding to the system being in a state of torque-free rotating motion in space.

- (b) The linearisation of the state space model (4.15) about its origin into the linear model (4.16) and the derivation of all the necessary Jacobian matrices.
- (c) A stability analysis of the non-linear model performed through the model linearisation which established that the system is at best neutrally stable. Oblate ellipsoids are potentially stable while prolate and asymmetric ellipsoids can be either potentially stable or unstable. Unstable shapes can be clearly identified by an increasing function of the mass ratio of the object's to the module's. Further calculation or empirical investigation are required to clarify the system behaviour near its equilibrium.
- (d) A controllability analysis which established that:
 - i. The system is not completely controllable.
 - ii. A state of pure spin is more controllable than a despun state.
 - iii. If the angular velocity or rotational kinetic energy becomes too large with respect to the magnitude of the moments of inertia of the object, the system becomes less controllable as the level of kinetic energy renders the object very stable.
- (e) The conclusion that, with the above level of analysis, the Model Predictive Control law's existence and stabilisability property cannot be formally proven. The existence of a small domain of attraction about the equilibrium would suffice but a higher order analysis is required to conclude. An empirical exploratory analysis is what could be attempted and a Model Predictive Control approach is the best suited for the exploration of the control space.
- (f) The design, for each module, of two decentralised controllers, a linear and nonlinear Model Predictive Controller, which control its motion on the surface of the object while tracking a reference state trajectory. This process encompassed:

- i. The proof of unicity of trajectories, admissibility, viability and feasibility of the system.
- ii. A stability and controllability analysis of the system.
- iii. The choice of a cost function and its weights.
- iv. The proof of observability of the system's state. The state is observable on an ellipsoid with gyroscopes and accelerometers providing a model of the local shape of the surface.
- (g) The design of a decentralised behaviour-based control algorithm taking the form of a behaviour tree. This controller controls the reconfiguration of the robot from a lattice configuration to a chain configuration coiling around the spinning axis of the object from the tip to the median plane, each module moving one by one. The controller also encompasses basic failure safety features.
- (h) The proof of correctness of the above behaviour-based algorithm under the hypothesis that the lower-level MPC controllers performs as per its specifications.
- (i) The design of a decentralised behaviour-based control algorithm for a selfreconfigurable robot which does not rely on emergence but uses a combination of physical modelling of the environment with the implementation of behaviours to complete a specific task and achieve a prescribed goal with proven correctness.
- 3. In chapter 5, extensive simulations of one module's deployment on the object's surface were carried out in order to evaluate the performance of the low-level linear and nonlinear model predictive controllers. Simulations were carried out for 10 object geometries under 32 initial conditions parametrised by the mass ratio of the object's to the module's and the object's initial rotational state. The performance evaluation was based on the performance measures laid out in section 4.1.4 and the observation of:
 - (a) The trajectory of the module on the surface of the object.

- (b) The object's phase diagram of the Z component of the angular velocity.
- (c) The object's rotational kinetic energy vs. time.
- (d) The object's nutation angle vs. time.
- (e) The nutation angle vs rotational kinetic energy.
- (f) The magnitude of the control commands vs. time.

The results originated from both linear and nonlinear control law data and their display was limited to the most significant cases highlighting similarities when it was relevant to do so. The results were organised in five sections respectively focusing on:

- (a) A comparison of the linear and nonlinear control commands over a large period of time for a mass ratio equals to 10,000.
- (b) An examination of the coverage of the module trajectory on the surface of the object coupled with a check that the module reached its target anchoring location.
- (c) The requirement to asymptotically stabilise the object's angular velocity to a pure state of spin or a despun state while the module travelling from the tip of the Z axis converges to its target anchoring location.
- (d) The requirement to asymptotically stabilise the object's angular velocity to a pure state of spin or a despun state beyond the timescale required by the controller's requirements in order to evaluate whether convergence of the rotational state occurs overtime despite the lack of controllability identified in chapter 4.
- (e) A final holistic comparison between the linear and nonlinear MPC controllers' performance.

The simulations highlighted the following points:

(a) Coverage, defined as a measure of reachability of any point of the object's surface from the robotic structure, is only half of what it should be with only half of the object's top hemisphere meridians covered by the module trajectory.

- (b) The model predictive controller controls the deployment state only and does not control the system's rotational state satisfactorily. The control commands are coupled with a control command space likely to be of dimension 1 instead of 2.
- (c) The nonlinear model predictive controller has nonetheless both the ability to stabilise the system with the relative angular velocity converging to 0 overtime and to predict accurately the system's dynamic behaviour.
- (d) However, for both the linear and nonlinear control laws, the system's objective of converging either to a pure state of spin or a despun state is not achieved within the deployment timespan as deployment and convergence have different timescale. The final rotational state is always nutated.
- (e) The timescale of the nonlinear control is perfectly acceptable for a space application and could be longer.
- (f) The nonlinear MPC control law provides control commands with realistic and feasible magnitudes.
- (g) High mass ratios stiffen the system. The lower the mass ratio, the greater the system controllability but the greater the possibility and magnitude of divergence. Conversely, stability increases with increasing mass ratios.

The general conclusion of this work is that the nonlinear model predictive controller should be chosen over the linear one. The simulations validated the nonlinear model predictive controller for driving the module's deployment to a target location but not for controlling the system's rotational state which never converge to the reference rotational state. However, the simulations also showed that the nonlinear model predictive controller could be used effectively for a strategy focusing on stabilising the system's initial rotational state. This strategy trades off controllability for stability by combining the controller with a stiffening of the system through an increase of the mass ratio which is shown to be the main stabilising parameter. The benefit of stability outweighs the risk of divergence undergone when trying to converge to a reference rotational state and in this case, the proposed self-reconfigurable robot is suited for objects involving a mass ratio of at least 1000 or more. From an engineering standpoint with a real system in mind, however, a mass ratio of 10000 or above is to be preferred. This means that the proposed SR robot solution is more suited to large objects. The asteroid sized in section 3.5.1.2 provides a good example. With its mass of 2.30e + 10kg, it allows for the deployment a SR robotic structure whose mass can be up to 2300T. For a small satellite with mass of the order of a couple of tonnes, this solution is not realistic.

6.2 Limitations and Future Work

The limitations of the robotic solution proposed in this PhD study not only call for further understanding and improvements of the proposed approach but also for the further exploration of different control strategies. The two types of future work is divided into two sections, respectively: further evaluation or further exploration.

6.2.1 Further Evaluations

Further evaluations and improvements of the proposed approach should:

- 1. Calculate the energy expenditure in real terms and size energy with respect to the relative inertia of the system's components to evaluate how realistic the feasibility study results are.
- 2. Run a simulation of the model with no control command input and with different initial conditions on the rotational state of a module moving on the surface of the object. Evaluate the "free" interactions between the object and a passive mass moving on its surface in order to understand the energy transfers between the object and the modules further.
- 3. Determine the mass ratios which optimise both controllability and stability.

- 4. Perform a higher order analysis $(2^{nd} \text{ order and higher})$ into the nature of the system's equilibrium to refine the control law design further.
- 5. Explore further the impact of asymmetry on controllability by gauging whether controllability increases or decreases with asymmetry of the ellipsoid's shape along with determining and parametrising a set of geometries for which it is the case.
- Vary the cost function weights in order to optimise the controller's performance.
- 7. Add a final penalty cost to the cost function to try to enforce the convergence of the rotational state and eliminate the residual relative angular velocity as well as the timescale issue. This final penalty cost take the form of further constraints on energy expenditure.
- 8. Determine an explicit form of the control commands function.
- 9. Add a coverage constraint to compensate for the restrictive effect of the control commands coupling. This could take the form of an explicit encoding of the even distribution of the trajectory about the chosen rotation axis so that all points of the object's surface are optimally reachable within an optimal average distance while minimising the energy expenditure.
- 10. Expand the timescale of the module deployment to match the timescale of the rotational state and of the nonlinear control law so that the controller acts as a regulator.
- Run a full robot simulation in order to test the behaviour-based control algorithm.
- 12. Include noise and perturbations in the simulations.

6.2.2 Further Explorations

The explorations of new control strategies should:

- 1. Add a nutation damper to the self-reconfigurable robotic structure to allow energy dissipation and the possibility of despinning the object or of the system's convergence to any reference rotational state.
- 2. Determine a method for explicitly decoupling the control commands so that the dimension of the control command space is 2.
- 3. Include the online identification of the moments of inertia of the object. Currently, in the model, these moments of inertia are assumed to be known. In reality, they would have been estimated prior to the mission and may have to be confirmed or re-evaluated online once the robot has landed on the object.
- 4. Include the moments arising from material stresses in the general model presented in chapter 3.
- 5. Devise a method for each module to accurately evaluate its odometry at the surface of the object.
- 6. Refine the robot's hardware design to work on a more realistic solution. In particular, explore retractable and flexible material solutions.
- 7. Explore the use of grown appendages such as makeshift arms to actuate and control the rotational state of the entire system.

This final section concludes this PhD study.

Appendix A

Physical Model Derivation with Consistent Vector Dimension Convention

In this appendix is laid out the most significant part of the derivation of the model presented in 3.2.4, the original of which is found in [66]. The focus is placed on a programmatic perspective to ensure a consistent writing convention for vectors and tensors used in the simulations' code. The calculation occur mainly at the particle level. When not stated otherwise, a simple application of the integral operator is required to ensure validity over the continuum.

All calculations rules are based on the dyadic identities found in appendix A of [28]. The convention followed is that left of the dyadic product \otimes , vectors are column and right of the dyadic product \otimes vectors are row. Drawing on the normalisation of the model parameters, there is also no explicit reference to mass.

The reader is referred to chapter 3 section 3.2.3 for the definitions of the frames of reference involved and to section 3.2.4 for the definitions of all the notations.

The continuous model 3.23 found in section 3.2.4 is reproduced below for mem-

ory:

$$\begin{split} [\mathbf{I}] \cdot \dot{\overrightarrow{\Omega}} + \overrightarrow{\Omega} \wedge [\mathbf{I}] \cdot \overrightarrow{\Omega} &= -\int_{m} 2\left[(\dot{\overrightarrow{\mathbf{x}_{0}'}} \cdot \overrightarrow{\mathbf{x}}) [\mathbf{1}] - \dot{\overrightarrow{\mathbf{x}_{0}'}} \otimes \overrightarrow{\mathbf{x}} \right] \cdot \overrightarrow{\Omega} \cdot \mathbf{dm} \\ &- \int_{m} \left[\left[(\overrightarrow{\mathbf{x}} \cdot \overrightarrow{\mathbf{x}}) [\mathbf{1}] - (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \right] \cdot \dot{\overrightarrow{\Psi}} \right] \\ &+ \left[2(\overrightarrow{\mathbf{x}} \cdot \dot{\overrightarrow{\mathbf{x}}}) [\mathbf{1}] - (\dot{\overrightarrow{\mathbf{x}}} \otimes \overrightarrow{\mathbf{x}}) - (\overrightarrow{\mathbf{x}} \otimes \dot{\overrightarrow{\mathbf{x}}}) \right] \cdot \overrightarrow{\Psi} \right] \cdot \mathbf{dm} \\ &- \int_{m} \left[(\overrightarrow{\mathbf{x}} \otimes \dot{\overrightarrow{\mathbf{x}_{0}'}}) - (\dot{\overrightarrow{\mathbf{x}_{0}'}} \otimes \overrightarrow{\mathbf{x}}) \right] \cdot \overrightarrow{\Psi} \cdot dm \qquad (A.1) \\ &- \int_{m} \left[(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \wedge \overrightarrow{\Psi} + ((\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \wedge \overrightarrow{\Psi})^{T} \right] \cdot \overrightarrow{\Omega} \cdot dm \\ &- \int_{m} (\overrightarrow{\mathbf{x}} \wedge \ddot{\overrightarrow{\mathbf{x}_{0}'}}) \cdot dm \\ &+ \overrightarrow{\mathbf{M}}_{Body} + \overrightarrow{\mathbf{M}}_{Stresses} + \overrightarrow{\mathbf{M}}_{Perturbations} \end{split}$$

In [66], the derivation starts from first principles using the kinematic relationships between relatively moving frames of reference to express the field angular momentum of the system as a function of the field angular velocity of the local particle and its relative velocity and acceleration. It results in the following formulae over the entire continuum:

$$\overrightarrow{\mathbf{L}}(x,t)_{Total} = \int_{m} \overrightarrow{\mathbf{x}} \wedge \left(\dot{\overrightarrow{\mathbf{\Omega}}}(x,t) \wedge \overrightarrow{\mathbf{x}} + 2\overrightarrow{\mathbf{\Omega}}(x,t) \wedge \dot{\overrightarrow{\mathbf{x}_{0}'}} + \overrightarrow{\mathbf{\Omega}}(x,t) \wedge (\overrightarrow{\mathbf{\Omega}}(x,t) \wedge \overrightarrow{\mathbf{x}}) + \dot{\overrightarrow{\mathbf{x}_{0}'}} \right)$$
(A.2)

The focus of this appendix will be on the further expansion of the four terms composing the angular momentum $\overrightarrow{\mathbf{L}}(x,t)_{Total}$ at the mass particle level. The application of the integral operator gives immediately their value over the entire continuum.

A.1 First Term

The first term is $\overrightarrow{\mathbf{x}} \wedge (\dot{\overrightarrow{\mathbf{\Omega}}}_{(x,t)} \wedge \overrightarrow{\mathbf{x}})$

Linearity gives:

$$\overrightarrow{\mathbf{x}} \wedge (\overrightarrow{\widehat{\mathbf{\Omega}}}_{(x,t)} \wedge \overrightarrow{\mathbf{x}}) = \overrightarrow{\mathbf{x}} \wedge (\overrightarrow{\widehat{\mathbf{\Omega}}} \wedge \overrightarrow{\mathbf{x}}) + \overrightarrow{\mathbf{x}} \wedge (\overrightarrow{\overline{\mathbf{\Psi}}} \wedge \overrightarrow{\mathbf{x}})$$
(A.3)

 $\overrightarrow{\mathbf{x}} \wedge (\dot{\overrightarrow{\mathbf{\Omega}}} \wedge \overrightarrow{\mathbf{x}})$ leads to $[\mathbf{I}]\dot{\overrightarrow{\mathbf{\Omega}}}$ via the integral operator. For the other term:

$$\left(\overrightarrow{\mathbf{x}}^{T}\cdot\overrightarrow{\mathbf{x}}\right)\cdot\overrightarrow{\overrightarrow{\mathbf{\Psi}}}=\left(\overrightarrow{\overrightarrow{\mathbf{\Psi}}}\otimes\overrightarrow{\mathbf{x}}\right)\cdot\overrightarrow{\mathbf{x}}$$
(A.4)

$$(\overrightarrow{\Psi} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{x}} = \frac{d[(\overrightarrow{\Psi} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{x}}]}{dt} - (\overrightarrow{\Psi} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{x}} - (\overrightarrow{\Psi} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{x}}$$
(A.5)

$$(\overrightarrow{\mathbf{\Psi}}\otimes\overrightarrow{\mathbf{x}})\cdot\overrightarrow{\mathbf{x}} = \frac{d[(\overrightarrow{\mathbf{\Psi}}\otimes\overrightarrow{\mathbf{x}})\cdot\overrightarrow{\mathbf{x}}]}{dt} - 2(\overrightarrow{\mathbf{\Psi}}\otimes\overrightarrow{\mathbf{x}})\cdot\overrightarrow{\mathbf{x}}$$
(A.6)

Since $\dot{\overrightarrow{\mathbf{x}}} = \dot{\overrightarrow{\mathbf{x}_0''}} + \overrightarrow{\Omega_t} \wedge \overrightarrow{\mathbf{x}} = \dot{\overrightarrow{\mathbf{x}_0''}} + (\overrightarrow{\Omega} + \overrightarrow{\Psi}) \wedge \overrightarrow{\mathbf{x}}$

$$(\overrightarrow{\Psi} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{x}} = [\overrightarrow{\Psi} \otimes (\overrightarrow{\mathbf{x}_0'} + (\overrightarrow{\Omega} + \overrightarrow{\Psi}) \wedge \overrightarrow{\mathbf{x}})] \cdot \overrightarrow{\mathbf{x}}$$
(A.7)

$$(\overrightarrow{\Psi} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{x}} = [\overrightarrow{\Psi} \otimes \overrightarrow{\mathbf{x}_0''}] \cdot \overrightarrow{\mathbf{x}}$$
(A.8)

Hence

$$(\vec{\overline{\Psi}} \otimes \vec{\mathbf{x}}) \cdot \vec{\mathbf{x}} = \frac{d[(\vec{\overline{\Psi}} \otimes \vec{\mathbf{x}}) \cdot \vec{\mathbf{x}}]}{dt} - 2[\vec{\overline{\Psi}} \otimes \vec{\mathbf{x}_0''}] \cdot \vec{\mathbf{x}}$$
(A.9)

Now

$$\left(\overrightarrow{\mathbf{x}}\otimes\overrightarrow{\mathbf{x}}\right)\cdot\overrightarrow{\overrightarrow{\mathbf{\Psi}}} = \frac{d\left[\left(\overrightarrow{\mathbf{x}}\otimes\overrightarrow{\mathbf{x}}\right)\cdot\overrightarrow{\mathbf{\Psi}}\right]}{dt} - \left(\overrightarrow{\mathbf{x}}\otimes\overrightarrow{\mathbf{x}}\right)\cdot\overrightarrow{\mathbf{\Psi}} - \left(\overrightarrow{\mathbf{x}}\otimes\overrightarrow{\mathbf{x}}\right)\cdot\overrightarrow{\mathbf{\Psi}} \qquad (A.10)$$

$$(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{\Psi}} = \frac{d[(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{\Psi}}]}{dt} - (\overrightarrow{\mathbf{x}_0''} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{\Psi}} - (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}_0''}) \cdot \overrightarrow{\mathbf{\Psi}} - ([\overrightarrow{\mathbf{\Omega}} \wedge \overrightarrow{\mathbf{x}} + \overrightarrow{\mathbf{\Psi}} \wedge \overrightarrow{\mathbf{x}}] \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{\Psi}} - ((\overrightarrow{\mathbf{x}} \otimes [\overrightarrow{\mathbf{\Omega}} \wedge \overrightarrow{\mathbf{x}} + \overrightarrow{\mathbf{\Psi}} \wedge \overrightarrow{\mathbf{x}}]) \cdot \overrightarrow{\mathbf{\Psi}}$$

$$(A.11)$$

Then

$$-([\overrightarrow{\Psi}\wedge\overrightarrow{\mathbf{x}}]\otimes\overrightarrow{\mathbf{x}})\cdot\overrightarrow{\Psi} = ([\overrightarrow{\mathbf{x}}\wedge\overrightarrow{\Psi}]\otimes\overrightarrow{\mathbf{x}})\cdot\overrightarrow{\Psi} = (\overrightarrow{\mathbf{x}}\otimes[\overrightarrow{\mathbf{x}}\wedge\overrightarrow{\Psi}])^{T}\cdot\overrightarrow{\Psi} = ([\overrightarrow{\mathbf{x}}\otimes\overrightarrow{\mathbf{x}}]\wedge\overrightarrow{\Psi})^{T}\cdot\overrightarrow{\Psi}$$
(A.12)

Similarly

$$-\left(\left[\overrightarrow{\Omega}\wedge\overrightarrow{\mathbf{x}}\right]\otimes\overrightarrow{\mathbf{x}}\right)\cdot\overrightarrow{\Psi}=\left(\left[\overrightarrow{\mathbf{x}}\otimes\overrightarrow{\mathbf{x}}\right]\wedge\overrightarrow{\Omega}\right)^{T}\cdot\overrightarrow{\Psi}$$
(A.13)

$$-\left(\overrightarrow{\mathbf{x}}\otimes[\overrightarrow{\Omega}\wedge\overrightarrow{\mathbf{x}}]\right)\cdot\overrightarrow{\Psi}=\left(\left[\overrightarrow{\mathbf{x}}\otimes\overrightarrow{\mathbf{x}}\right]\wedge\overrightarrow{\Omega}\right)\cdot\overrightarrow{\Psi}$$
(A.14)

$$-\left(\overrightarrow{\mathbf{x}}\otimes\left[\overrightarrow{\mathbf{\Psi}}\wedge\overrightarrow{\mathbf{x}}\right]\right)\cdot\overrightarrow{\mathbf{\Psi}}=-\overrightarrow{\mathbf{x}}\otimes\left(\left[\overrightarrow{\mathbf{\Psi}}\wedge\overrightarrow{\mathbf{x}}\right]\cdot\overrightarrow{\mathbf{\Psi}}\right)=\overrightarrow{\mathbf{0}}$$
(A.15)

Finally

$$\begin{aligned} (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{\Psi}} &= \\ & \frac{d[(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{\Psi}}]}{dt} - (\overrightarrow{\mathbf{x}_0''} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{\Psi}} - (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}_0''}) \cdot \overrightarrow{\mathbf{\Psi}} \\ &+ ([\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}] \wedge \overrightarrow{\mathbf{\Psi}})^T \cdot \overrightarrow{\mathbf{\Psi}} \\ &+ ([\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}] \wedge \overrightarrow{\mathbf{\Omega}})^T \cdot \overrightarrow{\mathbf{\Psi}} \\ &+ ([\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}] \wedge \overrightarrow{\mathbf{\Omega}}) \cdot \overrightarrow{\mathbf{\Psi}} \end{aligned}$$
(A.16)

A.2 Second Term

The second term is $2[\overrightarrow{\mathbf{x}} \wedge (\overrightarrow{\mathbf{\Omega_t}} \wedge \dot{\overrightarrow{\mathbf{x_0'}}})]$

$$2[\overrightarrow{\mathbf{x}}\wedge(\overrightarrow{\Omega_{\mathbf{t}}}\wedge\overrightarrow{\mathbf{x}_{\mathbf{0}}''})] = 2[(\overrightarrow{\mathbf{x}_{\mathbf{0}}''}\cdot\overrightarrow{\mathbf{x}})[\mathbf{1}] - (\overrightarrow{\mathbf{x}_{\mathbf{0}}''}\otimes\overrightarrow{\mathbf{x}})]\cdot\overrightarrow{\Omega} + 2[(\overrightarrow{\mathbf{x}_{\mathbf{0}}''}\cdot\overrightarrow{\mathbf{x}})[\mathbf{1}] - (\overrightarrow{\mathbf{x}_{\mathbf{0}}''}\otimes\overrightarrow{\mathbf{x}})]\cdot\overrightarrow{\Psi} \quad (A.17)$$

With $2[(\overrightarrow{\mathbf{x}_0''} \cdot \overrightarrow{\mathbf{x}})] = (\overrightarrow{\mathbf{x}_0''} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{\Omega}}$ leading to $2[\mathbf{J}] \overrightarrow{\mathbf{\Omega}}$ via the integral operator.

A.3 Third Term

The third term is $\overrightarrow{\mathbf{x}} \wedge [\overrightarrow{\Omega_{\mathbf{t}}} \wedge (\overrightarrow{\Omega_{\mathbf{t}}} \wedge \overrightarrow{\mathbf{x}})]$

$$\vec{\mathbf{x}} \wedge [\vec{\Omega}_{t} \wedge (\vec{\Omega}_{t} \wedge \vec{\mathbf{x}})] =$$

$$\vec{\mathbf{x}} \wedge [\vec{\Omega} \wedge (\vec{\Omega} \wedge \vec{\mathbf{x}})]$$

$$+ \vec{\mathbf{x}} \wedge [\vec{\Psi} \wedge (\vec{\Omega} \wedge \vec{\mathbf{x}})]$$

$$+ \vec{\mathbf{x}} \wedge [\vec{\Omega} \wedge (\vec{\Psi} \wedge \vec{\mathbf{x}})]$$

$$+ \vec{\mathbf{x}} \wedge [\vec{\Psi} \wedge (\vec{\Psi} \wedge \vec{\mathbf{x}})]$$

$$(A.18)$$

With $\overrightarrow{\mathbf{x}} \wedge [\overrightarrow{\Omega} \wedge (\overrightarrow{\Omega} \wedge \overrightarrow{\mathbf{x}})]$ leading to $\overrightarrow{\Omega} \wedge [\mathbf{I}] \overrightarrow{\Omega}$ via the integral operator.

Since

$$\overrightarrow{\mathbf{x}} \wedge [\overrightarrow{\mathbf{a}} \wedge (\overrightarrow{\mathbf{b}} \wedge \overrightarrow{\mathbf{x}})] = \overrightarrow{\mathbf{a}} \cdot (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \wedge \overrightarrow{\mathbf{b}} = -\overrightarrow{\mathbf{b}} \wedge (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{a}}$$
(A.19)

Hence

$$\overrightarrow{\mathbf{x}} \wedge [\overrightarrow{\mathbf{\Psi}} \wedge (\overrightarrow{\mathbf{\Omega}} \wedge \overrightarrow{\mathbf{x}})] = -\overrightarrow{\mathbf{\Omega}} \wedge (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{\Psi}} = [(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \wedge \overrightarrow{\mathbf{\Omega}}]^T \cdot \overrightarrow{\mathbf{\Psi}}$$
(A.20)

$$\overrightarrow{\mathbf{x}} \wedge [\overrightarrow{\mathbf{\Omega}} \wedge (\overrightarrow{\mathbf{\Psi}} \wedge \overrightarrow{\mathbf{x}})] = -\overrightarrow{\mathbf{\Psi}} \wedge (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{\Omega}} = [(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \wedge \overrightarrow{\mathbf{\Psi}}]^T \cdot \overrightarrow{\mathbf{\Omega}}$$
(A.21)

$$\overrightarrow{\mathbf{x}} \wedge [\overrightarrow{\mathbf{\Psi}} \wedge (\overrightarrow{\mathbf{\Psi}} \wedge \overrightarrow{\mathbf{x}})] = -\overrightarrow{\mathbf{\Psi}} \wedge (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \cdot \overrightarrow{\mathbf{\Psi}} = [(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \wedge \overrightarrow{\mathbf{\Psi}}]^T \cdot \overrightarrow{\mathbf{\Psi}}$$
(A.22)

From appendix A of [28]:

$$(\underline{\mathbf{C}} \wedge \mathbf{A}) \cdot \mathbf{B} = \underline{\mathbf{C}} \wedge \mathbf{A} \cdot \mathbf{B} = -(\underline{\mathbf{C}} \wedge \mathbf{B}) \cdot \mathbf{A}$$
(A.23)
leads to

$$-[(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \wedge \overrightarrow{\mathbf{\Omega}}] \cdot \overrightarrow{\mathbf{\Psi}} = [-(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \wedge \overrightarrow{\mathbf{\Omega}}] \cdot \overrightarrow{\mathbf{\Psi}}$$
$$= -[-(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \wedge \overrightarrow{\mathbf{\Psi}}] \cdot \overrightarrow{\mathbf{\Omega}}$$
$$[(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \wedge \overrightarrow{\mathbf{\Psi}}] \cdot \overrightarrow{\mathbf{\Omega}}$$

A.4 Fourth Term

The fourth term is immediately:

$$\overrightarrow{\mathbf{x}} \wedge \overrightarrow{\mathbf{x}_0''}$$
 (A.25)

A.5 Conclusion

The reader can verify that adding up all these terms and passing them through the integral operator over the entire continuum leads to model A.1

Appendix B

Jacobian of the State Function

This appendix lists the constituting elements of the Jacobian of the state space system 4.12 which is reproduced for reference in section B.1 as equation B.2. The derivation of these elements is also presented.

The calculations occur at the particle level focusing on the Jacobian with respect to the particles state variables. The discrete particles are identified by the index ior j when two particles are considered at the same time. The construction of the Jacobian follows a progressive bottom up approach, starting with the differentiation of individual vectors, then continuing with the differentiation of the moments of the state space system 4.12 and finally finishing by the aggregation of these into the main components of the overall Jacobian.

The focus is placed on dimensional consistency to ensure a consistent code writing convention for the simulations. Tensors in particular can be used with different representations depending on the application. The convention followed in this study is that all Jacobians are represented as two-dimensional matrices and not as multi-dimensional tensors. Therefore, the Jacobian of a vectorial function a with respect to the vectorial variable b is a matrix $\begin{bmatrix} \frac{\partial \mathbf{a}}{\partial \mathbf{b}} \end{bmatrix}$. The entry $\begin{bmatrix} \frac{\partial a_i}{\partial b_j} \end{bmatrix}$ at row i and column j represents the derivative of the i_{th} component of a with respect to the j_{th} component of b.

All calculations rules are based on the dyadic identities found in appendix A of [28]. The convention followed is that left of the dyadic product \otimes , vectors are column and right of the dyadic product \otimes vectors are row. Drawing on the normal-

isation of the model parameters, there is also no explicit reference to mass.

The reader is referred to chapter 3 section 3.2.3 for the definitions of the frames of reference involved and to section 3.2.4 for the definitions of all the notations.

B.1 Main Notations and Definitions

As per definition in section 4.2.1, the state vector is defined as $\vec{\mathbf{X}}$ and contains all the particles or robot modules state variables. The number of robot modules in the rest of this appendix is n_r . The dimensions of $\vec{\mathbf{X}}$ is $(3 + 4n_r) \times 1$.

$$\overrightarrow{\mathbf{X}} = \begin{bmatrix} \overrightarrow{\mathbf{\Omega}} \\ \overrightarrow{\mathbf{\Theta}} \\ \vdots \\ \overrightarrow{\mathbf{\Theta}} \end{bmatrix}$$
(B.1)

As per the definition of the system 4.12, its state space model is:

$$\begin{bmatrix} \dot{\overrightarrow{\Omega}} \\ \dot{\overrightarrow{\Theta}} \\ \vdots \\ \vec{\overrightarrow{\Theta}} \end{bmatrix} = g(\vec{\Omega}, \vec{\Theta}, \dot{\overrightarrow{\Theta}}, \vec{\overrightarrow{\Omega}}) = \begin{bmatrix} f(\vec{\Omega}, \vec{\Theta}, \dot{\overrightarrow{\Theta}}, \ddot{\overrightarrow{\Theta}}) \\ \vdots \\ \vec{\overrightarrow{\Theta}} \\ \vec{\overrightarrow{u}} \end{bmatrix}$$
(B.2)

where:

$$\begin{split} f(\overrightarrow{\Omega}, \overrightarrow{\Theta}, \overrightarrow{\Theta}, \overrightarrow{\Theta}) &= -[\mathbf{I}_{\mathbf{n}}]^{-1} \cdot (\overrightarrow{\Omega} \wedge [\mathbf{I}_{\mathbf{n}}] \cdot \overrightarrow{\Omega}) \\ &- [\mathbf{I}_{\mathbf{n}}]^{-1} \cdot \sum_{\mathbf{i}=1}^{\mathbf{n}_{\mathbf{r}}} \left[2 \left[(\overrightarrow{\mathbf{x}_{\mathbf{i}0}''} \cdot \overrightarrow{\mathbf{x}}_{\mathbf{i}}) [\mathbf{1}] - \overrightarrow{\mathbf{x}_{\mathbf{i}0}''} \otimes \overrightarrow{\mathbf{x}}_{\mathbf{i}} \right] \cdot \overrightarrow{\Omega} \\ &+ \left[\left[(\overrightarrow{\mathbf{x}}_{i} \cdot \overrightarrow{\mathbf{x}}_{i}) [\mathbf{1}] - (\overrightarrow{\mathbf{x}}_{\mathbf{i}} \otimes \overrightarrow{\mathbf{x}}_{\mathbf{i}}) \right] \cdot \overrightarrow{\Psi}_{\mathbf{i}} \\ &+ \left[2 (\overrightarrow{\mathbf{x}}_{i} \cdot \overrightarrow{\mathbf{x}}_{i}) [\mathbf{1}] - (\overrightarrow{\mathbf{x}}_{\mathbf{i}} \otimes \overrightarrow{\mathbf{x}}_{\mathbf{i}}) - (\overrightarrow{\mathbf{x}}_{\mathbf{i}} \otimes \overrightarrow{\mathbf{x}}_{\mathbf{i}}) \right] \cdot \overrightarrow{\Psi}_{\mathbf{i}} \right] \\ &+ \left[(\overrightarrow{\mathbf{x}}_{i} \otimes \overrightarrow{\mathbf{x}}_{i0}') - (\overrightarrow{\mathbf{x}_{i0}''} \otimes \overrightarrow{\mathbf{x}}_{i}) \right] \cdot \overrightarrow{\Psi}_{i} \\ &+ \left[(\overrightarrow{\mathbf{x}}_{i} \otimes \overrightarrow{\mathbf{x}}_{i}) \wedge \overrightarrow{\Psi}_{i} + ((\overrightarrow{\mathbf{x}}_{i} \otimes \overrightarrow{\mathbf{x}}_{i}) \wedge \overrightarrow{\Psi}_{i})^{T} \right] \cdot \overrightarrow{\Omega} \\ &+ (\overrightarrow{\mathbf{x}}_{i} \wedge \overrightarrow{\mathbf{x}_{i0}''}) \right] \end{split}$$

By definition, the Jacobian of g with respect to the main state variables is a

matrix of dimension $(4n_r + 3) \times (4n_r + 3)$ which is written as follows:

$$\begin{bmatrix} \frac{\partial g}{\partial \vec{\mathbf{X}}} \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial \vec{\Omega}} & \frac{\partial f}{\partial \vec{\Theta}} & \frac{\partial f}{\partial \vec{\Theta}} \\ \frac{\partial \vec{\Theta}}{\partial \vec{\Omega}} & \frac{\partial \vec{\Theta}}{\partial \vec{\Theta}} & \frac{\partial \vec{\Theta}}{\partial \vec{\Theta}} \\ \frac{\partial \vec{\Theta}}{\partial \vec{\Omega}} & \frac{\partial \vec{\Theta}}{\partial \vec{\Theta}} & \frac{\partial \vec{\Theta}}{\partial \vec{\Theta}} \end{bmatrix}$$
(B.4)

Similarly, the Jacobian of g with respect to the manipulated variables is a matrix of dimension $(4n_r + 3) \times 2n_r$ which is written as follows:

$$\begin{bmatrix} \frac{\partial g}{\partial \overrightarrow{\mathbf{u}}} \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial \overrightarrow{\mathbf{u}}} \\ \frac{\partial \overrightarrow{\mathbf{e}}}{\partial \overrightarrow{\mathbf{u}}} \\ \frac{\partial \overrightarrow{\mathbf{e}}}{\partial \overrightarrow{\mathbf{u}}} \end{bmatrix}$$
(B.5)

B.2 Jacobians of Position Vectors and Angular Velocities

This section lays out the Jacobians of the position, linear and angular velocity vectors on which the Jacobian of function g is based. All these vectors are expressed in the object's body frame of reference focusing on particle or robot module i. For each particle i, its vectors' Jacobians with respect to the state variables of particle j are null matrices of dimension 3×2 , i.e. $[0]_{3\times 2}$.

B.2.1 Jacobian of Basis Vectors

Starting with the basis vectors of the particles reference frame in spherical coordinates, the 3×2 Jacobians with respect to particle or module *i* state variables are:

$$\begin{bmatrix} \overrightarrow{\partial \overrightarrow{\mathbf{e_{ri}}}} \\ \overrightarrow{\partial \overrightarrow{\mathbf{\Theta}}_{i}} \end{bmatrix} = \begin{bmatrix} \cos(\theta_{i})\cos(\phi_{i}) & -\sin(\theta_{i})\sin(\phi_{i}) \\ \cos(\theta_{i})\sin(\phi_{i}) & \sin(\theta_{i})\cos(\phi_{i}) \\ -\sin(\theta_{i}) & 0 \end{bmatrix}$$
(B.6)

$$\begin{bmatrix} \partial \overrightarrow{\mathbf{e}}_{\theta \mathbf{i}} \\ \overline{\partial \overrightarrow{\mathbf{\Theta}}}_{i} \end{bmatrix} = \begin{bmatrix} -\sin(\theta_{i})\cos(\phi_{i}) & -\cos(\theta_{i})\sin(\phi_{i}) \\ -\sin(\theta_{i})\sin(\phi_{i}) & \cos(\theta_{i})\cos(\phi_{i}) \\ -\cos(\theta_{i}) & 0 \end{bmatrix}$$
(B.7)
$$\begin{bmatrix} \partial \sin(\theta_{\mathbf{i}})(\overrightarrow{\mathbf{e}}_{\phi \mathbf{i}}) \\ \overline{\partial \overrightarrow{\mathbf{\Theta}}}_{i} \end{bmatrix} = \begin{bmatrix} -\cos(\theta_{i})\sin(\phi_{i}) & -\sin(\theta_{i})\cos(\phi_{i}) \\ \cos(\theta_{i})\cos(\phi_{i}) & -\sin(\theta_{i})\sin(\phi_{i}) \\ 0 & 0 \end{bmatrix}$$
(B.8)

B.2.2 Jacobian of *r* the Radius of the Particle's Position

For each particle *i*, the Jacobian of its radius *r* with respect to the state variables of particle *j* are null matrices of dimension 1×2 , i.e. $[0]_{1 \times 2}$.

Defining *base* as:

$$base(\theta_i, \phi_i) = \frac{\cos(\phi_i)^2 \sin(\theta_i)^2}{a^2} + \frac{\sin(\phi_i)^2 \sin(\theta_i)^2}{b^2} + \frac{\cos(\theta_i)^2}{c^2}$$
(B.9)

For *prime*:

$$prime(\theta_i, \phi_i) = base(\theta_i, \phi_i) = \frac{1}{a^2} [\cos(\phi_i)^2 \sin(2\theta_i)\dot{\theta}_i - \sin(2\phi_i)\dot{\phi}_i \sin(\theta_i)^2] + \frac{1}{b^2} [\sin(\phi_i)^2 \sin(2\theta_i)\dot{\theta}_i + \sin(2\phi_i)\dot{\phi}_i \sin(\theta_i)^2] - \frac{1}{c^2} [\sin(2\theta_i)\dot{\theta}_i]$$
(B.10)

In spherical coordinates:

$$r(\theta_i, \phi_i) = \sqrt{\frac{1}{base(\theta_i, \phi_i)}}$$
(B.11)

The Jacobians $\left[\frac{\partial \mathbf{r}_{i}}{\partial \vec{\Theta}_{i}}\right]$, $\left[\frac{\partial \dot{\mathbf{r}}_{i}}{\partial \vec{\Theta}_{i}}\right]$, $\left[\frac{\partial \mathbf{r}_{i}}{\partial \vec{\Theta}_{i}}\right]$ and $\left[\frac{\partial \dot{\mathbf{r}}_{i}}{\partial \vec{\Theta}_{i}}\right]$ are 1×2 matrices whose entries

are derived below.

$$\left[\frac{\partial \mathbf{r}_{\mathbf{i}}}{\partial \theta_{\mathbf{i}}}\right] = \frac{-r_i^3}{2} \cdot \left[\frac{\partial \mathbf{base}}{\partial \theta_{\mathbf{i}}}\right] = \frac{-r_i^3}{2} \sin(2\theta_i) \left(\frac{\cos^2(\phi_i)}{a^2} + \frac{\sin^2(\phi_i)}{b^2} - \frac{1}{c^2}\right)$$
(B.12)

$$\left[\frac{\partial \mathbf{r_i}}{\partial \phi_{\mathbf{i}}}\right] = \frac{-r_i^3}{2} \cdot \left[\frac{\partial \mathbf{base}}{\partial \phi_{\mathbf{i}}}\right] = \frac{-r_i^3}{2} \sin^2(\theta_i) \sin(2\phi_i) \left(\frac{1}{b^2} - \frac{1}{a^2}\right)$$
(B.13)

$$\left[\frac{\partial \mathbf{r_i}}{\partial \dot{\theta_i}}\right] = 0 \tag{B.14}$$

$$\left[\frac{\partial \mathbf{r_i}}{\partial \dot{\phi_i}}\right] = 0 \tag{B.15}$$

$$\left[\frac{\partial \dot{\mathbf{r}}_{\mathbf{i}}}{\partial \theta_{\mathbf{i}}}\right] = \frac{3r_{i}^{5}}{4} \cdot prime \cdot \frac{\partial \mathbf{base}}{\partial \theta_{\mathbf{i}}} - \frac{r_{i}^{3}}{2} \cdot \frac{\partial \mathbf{prime}}{\partial \theta_{\mathbf{i}}} \tag{B.16}$$

$$\left[\frac{\partial \dot{\mathbf{r}}_{\mathbf{i}}}{\partial \phi_{\mathbf{i}}}\right] = \frac{3r_i^5}{4} \cdot prime \cdot \frac{\partial \mathbf{base}}{\partial \phi_{\mathbf{i}}} - \frac{r_i^3}{2} \cdot \frac{\partial \mathbf{prime}}{\partial \phi_{\mathbf{i}}} \tag{B.17}$$

$$\left[\frac{\partial \dot{\mathbf{r}}_{\mathbf{i}}}{\partial \dot{\theta}_{\mathbf{i}}}\right] = -\frac{r_i^3}{2} \cdot \frac{\partial \mathbf{prime}}{\partial \dot{\theta}_{\mathbf{i}}} \tag{B.18}$$

$$\left[\frac{\partial \dot{\mathbf{r}}_{\mathbf{i}}}{\partial \dot{\phi}_{\mathbf{i}}}\right] = -\frac{r_i^3}{2} \cdot \frac{\partial \mathbf{prime}}{\partial \dot{\phi}_{\mathbf{i}}} \tag{B.19}$$

$$\begin{bmatrix} \frac{\partial \mathbf{prime}}{\partial \theta_{\mathbf{i}}} \end{bmatrix} = \frac{2\cos(2\theta_{i})\dot{\theta}_{i}\cos^{2}(\phi_{i}) - \sin(2\theta_{i})\sin(2\phi_{i})\dot{\phi}_{i}}{a^{2}} + \frac{2\cos(2\theta_{i})\dot{\theta}_{i}\sin^{2}(\phi_{i}) + \sin(2\theta_{i})\sin(2\phi_{i})\dot{\phi}_{i}}{b^{2}} - \frac{2\cos(2\theta_{i})\dot{\theta}_{i}}{c^{2}}$$
(B.20)

$$\left[\frac{\partial \mathbf{prime}}{\partial \phi_{\mathbf{i}}}\right] = (\sin(2\theta_i)\dot{\theta}_i \sin(2\phi_i) + 2\sin^2(\theta_i)\cos(2\phi_i)\dot{\phi}_i)(\frac{1}{b^2} - \frac{1}{a^2})$$
(B.21)

$$\left[\frac{\partial \mathbf{prime}}{\partial \dot{\theta}_{\mathbf{i}}}\right] = \sin(2\theta_i)\left(\frac{\cos^2(\phi_i)}{a^2} + \frac{\sin^2(\phi_i)}{b^2} - \frac{1}{c^2}\right) \tag{B.22}$$

$$\left[\frac{\partial \mathbf{prime}}{\partial \dot{\phi}_{\mathbf{i}}}\right] = \sin^2(\theta_i) \sin(2\phi_i) (\frac{1}{b^2} - \frac{1}{a^2}) \tag{B.23}$$

B.2.3 Jacobian of Relative Velocities

The Jacobian of the particles' relative velocities with respect to all the state variables has an overall dimension of $3n_r \times (2n_r + 3)$. For each particle *i*, the Jacobian of its relative velocity with respect to the state variables of particle *j* are null matrices of dimension 3×2 , i.e. $[0]_{3\times 2}$ and the Jacobian of its relative velocity with respect to its own state variables is also of dimension 3×2 . When calculated with respect to the angular velocity of the object, the Jacobian of the relative velocity of particle *i* is of dimension 3×3 .

$$\begin{bmatrix} \overrightarrow{\partial \mathbf{x}_{i0}''} \\ \overrightarrow{\partial \mathbf{\Theta}_i} \end{bmatrix} = \begin{bmatrix} \overrightarrow{\partial \mathbf{\dot{r}_i}} \\ \overrightarrow{\partial \mathbf{\Theta}_i} \end{bmatrix} \otimes \overrightarrow{\mathbf{e}_{ri}} + \dot{r}_i \otimes \begin{bmatrix} \overrightarrow{\partial \mathbf{e}_{ri}} \\ \overrightarrow{\partial \mathbf{\Theta}_i} \end{bmatrix}$$
(B.24)

$$\begin{bmatrix} \overrightarrow{\partial \mathbf{x}_{i0}''} \\ \overrightarrow{\partial \mathbf{\Theta}_i} \end{bmatrix} = \begin{bmatrix} \overrightarrow{\partial \mathbf{\dot{r}}_i} \\ \overrightarrow{\partial \mathbf{\Theta}_i} \end{bmatrix} \otimes \mathbf{\overrightarrow{e}_r}_i$$
(B.25)

$$\begin{bmatrix} \partial \overrightarrow{\mathbf{x}}_i \\ \partial \overrightarrow{\mathbf{\Theta}}_i \end{bmatrix} = \begin{bmatrix} \partial \mathbf{r}_i \\ \partial \overrightarrow{\mathbf{\Theta}}_i \end{bmatrix} \otimes \overrightarrow{\mathbf{e}}_{\mathbf{r}i} + r_i \otimes \begin{bmatrix} \partial \overrightarrow{\mathbf{e}}_{\mathbf{r}i} \\ \partial \overrightarrow{\mathbf{\Theta}}_i \end{bmatrix}$$
(B.26)

$$\begin{bmatrix} \partial \overrightarrow{\mathbf{x}}_i \\ \partial \overrightarrow{\mathbf{\Theta}}_i \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$
(B.27)

$$\left[\frac{\partial \dot{\overrightarrow{\mathbf{x}}}_i}{\partial \overrightarrow{\Omega}}\right] = [\mathbf{Id}]_{\mathbf{3} \times \mathbf{3}} \wedge \mathbf{r}_i \overrightarrow{\mathbf{e}_{\mathbf{r}i}}$$
(B.28)

$$\begin{bmatrix} \partial \overrightarrow{\mathbf{x}}_{i} \\ \partial \overrightarrow{\mathbf{\Theta}}_{i} \end{bmatrix} = \begin{bmatrix} \partial \overrightarrow{\mathbf{x}_{i0}'} \\ \partial \overrightarrow{\mathbf{\Theta}}_{i} \end{bmatrix} + \dot{\phi}_{i} \begin{bmatrix} \partial \mathbf{r}_{i} \\ \partial \overrightarrow{\mathbf{\Theta}}_{i} \end{bmatrix} \otimes \sin(\theta_{i}) \overrightarrow{\mathbf{e}}_{\phi_{i}} \\ + \dot{\phi}_{i} r_{i} \otimes \begin{bmatrix} \partial \sin(\theta_{i}) \overrightarrow{\mathbf{e}}_{\phi_{i}} \\ \partial \overrightarrow{\mathbf{\Theta}}_{i} \end{bmatrix} + \begin{bmatrix} \partial \mathbf{r}_{i} \\ \partial \overrightarrow{\mathbf{\Theta}}_{i} \end{bmatrix} \otimes \dot{\theta}_{i} \overrightarrow{\mathbf{e}}_{\theta_{i}} \\ + r_{i} \dot{\theta}_{i} \otimes \begin{bmatrix} \partial \overrightarrow{\mathbf{e}}_{\theta_{i}} \\ \partial \overrightarrow{\mathbf{\Theta}}_{i} \end{bmatrix} + \overrightarrow{\mathbf{\Omega}} \wedge \begin{bmatrix} \partial \overrightarrow{\mathbf{x}}_{i} \\ \partial \overrightarrow{\mathbf{\Theta}}_{i} \end{bmatrix}$$
(B.29)

$$\begin{bmatrix} \partial \overrightarrow{\mathbf{x}}_i \\ \partial \overrightarrow{\mathbf{\Theta}}_i \end{bmatrix} = \begin{bmatrix} \partial \overrightarrow{\mathbf{x}_{i0}'} \\ \partial \overrightarrow{\mathbf{\Theta}}_i \end{bmatrix} + \begin{bmatrix} 0 & 1 \end{bmatrix} \otimes r_i \sin(\theta_i) \overrightarrow{\mathbf{e}}_{\phi_i} + \begin{bmatrix} 1 & 0 \end{bmatrix} \otimes r_i \overrightarrow{\mathbf{e}}_{\theta_i}$$
(B.30)

B.2.4 Jacobian of Relative Angular Velocities

The Jacobian of the particles' relative angular velocities with respect to all the state variables has an overall dimension of $3n_r \times (2n_r + 3)$. For each particle *i*, the Jacobian of its relative angular velocity with respect to the state variables of particle *j* is a null matrix of dimension 3×2 , i.e. $[0]_{3\times 2}$ and the Jacobian of its relative angular velocity with respect to its own state variables is also of dimension 3×2 . When calculated with respect to the angular velocity of the object, the Jacobian of the relative angular velocity of particle *i* is a null matrix of dimension 3×3 (i.e. $[0]_{3\times 3}$) and when calculated with respect to the manipulated variables, the Jacobian of the relative velocity of particle *i* is of dimension 3×2 .

$$\begin{bmatrix} \overrightarrow{\partial \Psi_i} \\ \overrightarrow{\partial \Theta_i} \end{bmatrix} = \begin{bmatrix} 0 & -\cos(\phi_i)\dot{\theta_i} \\ 0 & -\sin(\phi_i)\dot{\theta_i} \\ 0 & 0 \end{bmatrix}$$
(B.31)

$$\begin{bmatrix} \partial \vec{\Psi}_i \\ \partial \vec{\Theta}_i \end{bmatrix} = \begin{bmatrix} -\sin(\phi_i) & 0 \\ \cos(\phi_i) & 0 \\ 0 & 1 \end{bmatrix}$$
(B.32)
$$\begin{bmatrix} \partial \vec{\Psi}_i \\ \partial \vec{\Theta}_i \end{bmatrix} = \begin{bmatrix} 0 & -\cos(\phi_i)\ddot{\theta}_i + \sin(\phi_i)\dot{\phi}_i\dot{\theta}_i \\ 0 & -\sin(\phi_i)\ddot{\theta}_i - \cos(\phi_i)\dot{\phi}_i\dot{\theta}_i \\ 0 & 0 \end{bmatrix}$$
(B.33)

$$\begin{bmatrix} \frac{\partial \vec{\Psi}_i}{\partial \vec{\Theta}_i} \end{bmatrix} = \begin{bmatrix} -\cos(\phi_i)\dot{\phi}_i & -\cos(\phi_i)\dot{\theta}_i \\ -\sin(\phi_i)\dot{\phi}_i & -\sin(\phi_i)\dot{\theta}_i \\ 0 & 0 \end{bmatrix}$$
(B.34)
$$\begin{bmatrix} \frac{\partial \vec{\Psi}_i}{\partial \vec{\Pi}_i} \end{bmatrix} = \begin{bmatrix} -\sin(\phi_i) & 0 \\ \cos(\phi_i) & 0 \\ 0 & 1 \end{bmatrix}$$
(B.35)

B.3 Jacobian of g with Respect to the State Variables

Again, in this section, n_r stands for the number of robot modules or discrete particles.

B.3.1 Jacobian of the Particles State Variables

The Jacobian of g particles' state variables with respect to all state variables has an overall dimension of $4n_r \times (4n_r + 3)$.

B.3.1.1 $\vec{\Theta}$ with respect to $\vec{\Omega}$:

$$\left[\frac{\partial \overleftrightarrow{\Theta}}{\partial \overrightarrow{\Omega}}\right] = [\mathbf{0}]_{\mathbf{2n_r} \times \mathbf{3}} \tag{B.36}$$

B.3.1.2 $\vec{\Theta}$ with respect to $\vec{\Theta}$:

$$\begin{bmatrix} \partial \overrightarrow{\Theta} \\ \partial \overrightarrow{\Theta} \end{bmatrix} = [\mathbf{0}]_{\mathbf{2n_r} \times \mathbf{2n_r}}$$
(B.37)

B.3.1.3 $\stackrel{\cdot}{\Theta}$ with respect to $\stackrel{\cdot}{\Theta}$:

$$\left[\frac{\partial \overrightarrow{\Theta}}{\partial \overrightarrow{\Theta}}\right] = [\mathbf{Id}]_{\mathbf{2n_r} \times \mathbf{2n_r}} \tag{B.38}$$

At particle i level:

$$\begin{bmatrix} \partial \dot{\overrightarrow{\Theta}}_i \\ \partial \dot{\overrightarrow{\Theta}}_i \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$
(B.39)

Particle i with respect to j:

$$\begin{bmatrix} \overrightarrow{\partial} \overrightarrow{\overrightarrow{\Theta}}_i \\ \overrightarrow{\partial} \overrightarrow{\overrightarrow{\Theta}}_j \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$
(B.40)

B.3.1.4 $\ddot{\vec{\Theta}}$ with respect to $\vec{\Omega}$:

$$\begin{bmatrix} \overrightarrow{\partial} \overrightarrow{\Theta} \\ \overline{\partial} \overrightarrow{\Omega} \end{bmatrix} = [\mathbf{0}]_{\mathbf{2n_r} \times \mathbf{3}}$$
(B.41)

B.3.1.5 $\ddot{\vec{\Theta}}$ with respect to $\vec{\Theta}$: $\left[\frac{\partial \vec{\Theta}}{\partial \vec{\Theta}}\right] = [\mathbf{0}]_{\mathbf{2n_r} \times \mathbf{2n_r}} \qquad (B.42)$

B.3.1.6 $\ddot{\vec{\Theta}}$ with respect to $\dot{\vec{\Theta}}$:

$$\left[\frac{\partial \overrightarrow{\Theta}}{\partial \overrightarrow{\Theta}}\right] = [\mathbf{0}]_{\mathbf{2n_r} \times \mathbf{2n_r}} \tag{B.43}$$

B.3.2 Jacobian of f With Respect to the Rigid Angular Velocity

The Jacobian $\left[\frac{\partial f}{\partial \vec{\Omega}}\right]$ has dimension 3×3 .

$$\begin{bmatrix} \frac{\partial f}{\partial \overrightarrow{\Omega}} \end{bmatrix} = -[\mathbf{I}_{\mathbf{n}}]^{-1} \cdot \left[([\mathbf{Id}]_{\mathbf{3} \times \mathbf{3}} \wedge [\mathbf{I}_{\mathbf{n}}] \cdot \overrightarrow{\Omega}) + (\overrightarrow{\Omega} \wedge [\mathbf{I}_{\mathbf{n}}]) + \sum_{i=1}^{n_{r}} \left[2[(\overrightarrow{\mathbf{x}_{i0}'} \cdot \overrightarrow{\mathbf{x}}_{i})[\mathbf{1}] - \overrightarrow{\mathbf{x}_{i0}'} \otimes \overrightarrow{\mathbf{x}}_{i}] + [(\overrightarrow{\mathbf{x}}_{i} \otimes \overrightarrow{\mathbf{x}}_{i}) \wedge \overrightarrow{\mathbf{\Psi}}_{i} + ((\overrightarrow{\mathbf{x}}_{i} \otimes \overrightarrow{\mathbf{x}}_{i}) \wedge \overrightarrow{\mathbf{\Psi}}_{i})^{T}] \right] \right]$$
(B.44)

B.3.3 Jacobian of f With Respect to the Angular Positions

of the Particles

 $\begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}} \end{bmatrix}$ has a dimension of $3 \times 2n_r$. In this section, $\begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}} \end{bmatrix}$ is split into two components for readability:

$$\left[\frac{\partial f}{\partial \overrightarrow{\Theta}}\right] = \left[\frac{\partial f}{\partial \overrightarrow{\Theta}}\right]_{MoI} + \left[\frac{\partial f}{\partial \overrightarrow{\Theta}}\right]_{Moments}$$
(B.45)

B.3.3.1 $\left[\frac{\partial f}{\partial \vec{\Theta}}\right]_{MoI}$ $\left[\frac{\partial [\mathbf{I_n}]^{-1}}{\partial \vec{\Theta}}\right]$ is of dimension $3 \times 6n_r$, hence:

$$\begin{bmatrix} \frac{\partial f}{\partial \overrightarrow{\Theta}} \end{bmatrix}_{MoI} = -\begin{bmatrix} \frac{\partial [\mathbf{I_n}]^{-1}}{\partial \overrightarrow{\Theta}} \end{bmatrix} \cdot \left([\mathbf{Id}]_{2\mathbf{n_r} \times 2\mathbf{n_r}} \otimes \begin{bmatrix} (\overrightarrow{\Omega} \wedge [\mathbf{I_n}] \cdot \overrightarrow{\Omega}) \\ + \sum_{i=1}^{n_r} \begin{bmatrix} 2[(\overrightarrow{\mathbf{x}''_{i0}} \cdot \overrightarrow{\mathbf{x}}_i)]\mathbf{1}] - \overrightarrow{\mathbf{x}''_{i0}} \otimes \overrightarrow{\mathbf{x}}_i] \cdot \overrightarrow{\Omega} \\ + [(\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\mathbf{x}}_i)]\mathbf{1}] - (\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}}_i)] \cdot \overrightarrow{\Psi}_i \\ + [2(\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\mathbf{x}}_i)]\mathbf{1}] - (\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}}_i) - (\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}}_i)] \cdot \overrightarrow{\Psi}_i \\ + [(\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}'_{i0}}) - (\overrightarrow{\mathbf{x}''_{i0}} \otimes \overrightarrow{\mathbf{x}}_i)] \cdot \overrightarrow{\Psi}_i \\ + [(\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}'_{i0}}) - (\overrightarrow{\mathbf{x}''_{i0}} \otimes \overrightarrow{\mathbf{x}}_i)] \cdot \overrightarrow{\Psi}_i \\ + [(\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}}_i) \wedge \overrightarrow{\Psi}_i + ((\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}}_i) \wedge \overrightarrow{\Psi}_i)^T] \cdot \overrightarrow{\Omega} \\ + (\overrightarrow{\mathbf{x}}_i \wedge \overrightarrow{\mathbf{x}''_{i0}}) \end{bmatrix} \right] \end{pmatrix}$$

B.3.3.2
$$\begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}} \end{bmatrix}_{Moments}$$

 $\begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}} \end{bmatrix}_{Moments}$ is of dimension $3 \times 2n_r$. Defining $\begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}} \end{bmatrix}_{Moments} = \begin{bmatrix} \begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}_i} \end{bmatrix}_M \end{bmatrix}$ with $i \in [1, n_r]$ and $\begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}_i} \end{bmatrix}_M$ of dimension 3×2 .

$$\forall i \in [1, n_r]:$$

$$\begin{bmatrix} \left[\frac{\partial f}{\partial \overrightarrow{\Theta}_i} \right]_M \end{bmatrix} = -[\mathbf{I}_n]^{-1} \cdot \begin{bmatrix} \overrightarrow{\Omega} \times i \\ \overrightarrow{\partial \Theta}_i \end{bmatrix} \otimes \overrightarrow{\Omega} - \begin{bmatrix} \frac{\partial \overrightarrow{X}_i}{\partial \overrightarrow{\Theta}_i} \end{bmatrix} \otimes (\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\Omega}) - \overrightarrow{\mathbf{x}}_i \otimes \left(\overrightarrow{\Omega}^T \cdot \begin{bmatrix} \frac{\partial \overrightarrow{X}_i}{\partial \overrightarrow{\Theta}_i} \end{bmatrix} \right) \end{bmatrix} \\ + 2 \left(\overrightarrow{\mathbf{x}}_i^T \cdot \begin{bmatrix} \frac{\partial \overrightarrow{\mathbf{x}}_{i0}}{\partial \overrightarrow{\Theta}_i} \end{bmatrix} \right) \otimes \overrightarrow{\Omega} + 2 \left((\overrightarrow{\mathbf{x}}_{i0}^T)^T \cdot \begin{bmatrix} \frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \end{bmatrix} \right) \otimes \overrightarrow{\Omega} \\ - 2 \left[\frac{\partial \overrightarrow{\mathbf{x}}_{i0}^T}{\partial \overrightarrow{\Theta}_i} \right] \otimes (\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\Omega}) - 2 \overrightarrow{\mathbf{x}}_{i0}^T \otimes \left(\overrightarrow{\Omega}^T \cdot \begin{bmatrix} \frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \end{bmatrix} \right) \\ + \left[\frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \otimes ((\overrightarrow{\mathbf{x}}_i \wedge \overrightarrow{\Psi}_i) \cdot \overrightarrow{\Omega}) + \overrightarrow{\mathbf{x}}_i \otimes \left[\overrightarrow{\Omega}^T \cdot \left(\begin{bmatrix} \frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \wedge \overrightarrow{\Psi}_i + \overrightarrow{\mathbf{x}}_i \wedge \left[\frac{\partial \overrightarrow{\Psi}_i}{\partial \overrightarrow{\Theta}_i} \right] \right) \\ + \left(\begin{bmatrix} \frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \wedge \overrightarrow{\Psi}_i + \overrightarrow{\mathbf{x}}_i \wedge \left[\frac{\partial \overrightarrow{\Psi}_i}{\partial \overrightarrow{\Theta}_i} \right] \right) \otimes (\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\Omega}) + (\overrightarrow{\mathbf{x}}_i \wedge \overrightarrow{\Psi}_i) \otimes (\overrightarrow{\Omega}^T \cdot \left[\frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \right) \\ + \left(2 \overrightarrow{\mathbf{x}}_i^T \cdot \left[\frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \right) \otimes \overrightarrow{\Psi}_i - \left[\frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \otimes (\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\Psi}_i) - \overrightarrow{\mathbf{x}}_i \otimes \left(\overrightarrow{\Psi}_i^T \cdot \left[\frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \right) \\ + \left[(\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\mathbf{x}}_i) \left[1 - (\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}}_i) - (\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}}_i) \right] \cdot \left[\frac{\partial \overrightarrow{\Psi}_i}{\partial \overrightarrow{\Theta}_i} \right] \\ + 2 \left[\left(\overrightarrow{\mathbf{x}}_i^T \cdot \left[\frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right) \right) + \left(\overrightarrow{\mathbf{x}}_i^T \cdot \left[\frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \right) \right] \otimes \overrightarrow{\Psi}_i \\ - \left[\frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \right) + \left(\overrightarrow{\mathbf{x}}_i \cdot \left[\frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \right) \right]$$

$$(B.47)$$

B.3.4 Jacobian of f With Respect to the Angular Velocities of the Particles

$$\begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}} \end{bmatrix} \text{ is of dimension } 3 \times 2n_r. \text{ Defining } \begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}_i} \end{bmatrix} \text{ with } i \in [1, n_r] \text{ and } \begin{bmatrix} \frac{\partial f}{\partial \vec{\Theta}_i} \end{bmatrix} \text{ of dimension } 3 \times 2.$$

$$\forall i \in [1, n_r]:$$

$$\begin{bmatrix} \frac{\partial f}{\partial \overrightarrow{\Theta}_i} \end{bmatrix} = -[\mathbf{I}_n]^{-1} \cdot \left[2 \left(\overrightarrow{\mathbf{x}}_i^T \cdot \left[\frac{\partial \overrightarrow{\mathbf{x}}_{i0}^{''}}{\partial \overrightarrow{\Theta}_i} \right] \right) \otimes \overrightarrow{\Omega} - 2 \left[\frac{\partial \overrightarrow{\mathbf{x}}_{i0}^{''}}{\partial \overrightarrow{\Theta}_i} \right] \otimes (\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\Omega}) \right]$$

$$+ \overrightarrow{\mathbf{x}}_i \otimes \left((\overrightarrow{\Omega})^T \cdot \left(\overrightarrow{\mathbf{x}}_i \wedge \left[\frac{\partial \overrightarrow{\Psi}_i}{\partial \overrightarrow{\Theta}_i} \right] \right) \right) + \left(\overrightarrow{\mathbf{x}}_i \wedge \left[\frac{\partial \overrightarrow{\Psi}_i}{\partial \overrightarrow{\Theta}_i} \right] \right) \otimes (\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\Omega})$$

$$+ [(\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\mathbf{x}}_i)] \mathbf{1} - (\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}}_i)] \cdot \left[\frac{\partial \overrightarrow{\Psi}_i}{\partial \overrightarrow{\Theta}_i} \right]$$

$$+ [2(\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\mathbf{x}}_i)] \mathbf{1} - (\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}}_i) - (\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}}_i)] \cdot \left[\frac{\partial \overrightarrow{\Psi}_i}{\partial \overrightarrow{\Theta}_i} \right]$$

$$+ 2 \left(\overrightarrow{\mathbf{x}}_i^T \cdot \left[\frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \right) \otimes \overrightarrow{\Psi}_i$$

$$- \left[\frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \otimes (\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\Psi}_i) - \overrightarrow{\mathbf{x}}_i \otimes \left((\overrightarrow{\Psi}_i)^T \cdot \left[\frac{\partial \overrightarrow{\mathbf{x}}_i}{\partial \overrightarrow{\Theta}_i} \right] \right)$$

$$(B.48)$$

B.4 Jacobian of g with Respect to the Manipulated Variables

 $\begin{bmatrix} \frac{\partial f}{\partial \vec{\mathbf{u}}} \end{bmatrix} \text{ is of dimension } 3 \times 2n_r. \text{ Defining } \begin{bmatrix} \frac{\partial f}{\partial \vec{\mathbf{u}}} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \frac{\partial f}{\partial \vec{\mathbf{u}}_i} \end{bmatrix} \text{ with } i \in [1, n_r] \text{ and } \begin{bmatrix} \frac{\partial f}{\partial \vec{\mathbf{u}}_i} \end{bmatrix} \text{ of dimension } 3 \times 2.$

$$\forall i \in [1, n_r]:$$

$$\left[\frac{\partial f}{\partial \overrightarrow{\mathbf{u}}_{i}}\right] = -[\mathbf{I}_{\mathbf{n}}]^{-1} \cdot \left[[(\overrightarrow{\mathbf{x}}_{i} \cdot \overrightarrow{\mathbf{x}}_{i})[\mathbf{1}] - (\overrightarrow{\mathbf{x}}_{i} \otimes \overrightarrow{\mathbf{x}}_{i})] \cdot \left[\frac{\partial \overleftarrow{\overrightarrow{\mathbf{\psi}}}_{i}}{\partial \overrightarrow{\mathbf{u}}_{i}}\right] \right]$$
(B.49)

Moreover:

$$\begin{bmatrix} \partial \overrightarrow{\Theta} \\ \overline{\partial} \overrightarrow{u} \end{bmatrix} = [\mathbf{0}]_{\mathbf{2n_r} \times \mathbf{2n_r}}$$
(B.50)

$$\begin{bmatrix} \partial \overrightarrow{\Theta} \\ \overline{\partial \overrightarrow{u}} \end{bmatrix} = \begin{bmatrix} \partial \overrightarrow{u} \\ \overline{\partial \overrightarrow{u}} \end{bmatrix} = [\mathbf{Id}]_{\mathbf{2n_r} \times \mathbf{2n_r}}$$
(B.51)

B.5 Jacobian of the Moment of Inertia Matrices

B.5.1 Jacobian of the Moment of Inertia $[I_n]$

The moment of inertia $[\mathbf{I_n}]$ is only dependent on the state variable $\overrightarrow{\Theta}$ and has to be derived only once for equation B.45. As per section 3.2.6, I_x , I_y and I_z represent the moment of inertia of the object about its principal axes. At any given instant of time t, the moment of inertia $[\mathbf{I_n}]$ of the object-robot system with n_r robot modules is equal to:

$$[\mathbf{I_n}] = \begin{bmatrix} I_x & 0 & 0\\ 0 & I_y & 0\\ 0 & 0 & I_z \end{bmatrix} + \sum_{\mathbf{j=1}}^{\mathbf{n_r}} [(\overrightarrow{\mathbf{x}}_{\mathbf{j}} \cdot \overrightarrow{\mathbf{x}}_{\mathbf{j}})[\mathbf{1}] - (\overrightarrow{\mathbf{x}}_{\mathbf{j}} \otimes \overrightarrow{\mathbf{x}}_{\mathbf{j}})]$$
(B.52)

 $\begin{bmatrix} \underline{\partial [\mathbf{I}_{\mathbf{n}}]} \\ \overline{\partial \mathbf{\Theta}} \end{bmatrix}$ is of dimension $3 \times 6n_r$ and:

$$\left[\frac{\partial[\mathbf{I_n}]}{\partial \overrightarrow{\mathbf{\Theta}}}\right] = \left[\frac{\partial\left[\left[(\overrightarrow{\mathbf{x}}_i \cdot \overrightarrow{\mathbf{x}}_i)[\mathbf{1}] - (\overrightarrow{\mathbf{x}}_i \otimes \overrightarrow{\mathbf{x}}_i)]\right]\right]}{\partial \overrightarrow{\mathbf{\Theta}}_i}\right]_{i \in [1, n_r]}$$
(B.53)

$$\forall i \in [1, n_r], \begin{bmatrix} \underline{\partial}[\mathbf{I}_n] \\ \overline{\partial} \overline{\mathbf{\Theta}}_i \end{bmatrix} \text{ is a } 3 \times 6 \text{ matrix and:}$$
$$\begin{bmatrix} \underline{\partial}[\mathbf{I}_n] \\ \overline{\partial} \overline{\mathbf{\Theta}}_i \end{bmatrix} = \left(2(\overrightarrow{\mathbf{x}}_i^T) \cdot \begin{bmatrix} \underline{\partial} \overrightarrow{\mathbf{x}}_i \\ \overline{\partial} \overline{\mathbf{\Theta}}_i \end{bmatrix} \right) \otimes [\mathbf{1}]_{\mathbf{3} \times \mathbf{3}} - \begin{bmatrix} \underline{\partial} \overrightarrow{\mathbf{x}}_i \\ \overline{\partial} \overline{\mathbf{\Theta}}_i \end{bmatrix} \otimes \overrightarrow{\mathbf{x}}_i - \overrightarrow{\mathbf{x}}_i \otimes \begin{bmatrix} \underline{\partial} \overrightarrow{\mathbf{x}}_i \\ \overline{\partial} \overline{\mathbf{\Theta}}_i \end{bmatrix}$$
(B.54)

Hence

$$\begin{bmatrix} \overline{\partial} [\mathbf{I}_{\mathbf{n}}] \overrightarrow{\Omega} \\ \overline{\partial} \overrightarrow{\Theta}_{i} \end{bmatrix} = \left(2(\overrightarrow{\mathbf{x}}_{i}^{T}) \cdot \begin{bmatrix} \overline{\partial} \overrightarrow{\mathbf{x}}_{i} \\ \overline{\partial} \overrightarrow{\Theta}_{i} \end{bmatrix} \right) \otimes \overrightarrow{\Omega} - \begin{bmatrix} \overline{\partial} \overrightarrow{\mathbf{x}}_{i} \\ \overline{\partial} \overrightarrow{\Theta}_{i} \end{bmatrix} \otimes (\overrightarrow{\mathbf{x}}_{i} \cdot \overrightarrow{\Omega}) - \overrightarrow{\mathbf{x}}_{i} \otimes \left(\overrightarrow{\Omega}^{T} \cdot \begin{bmatrix} \overline{\partial} \overrightarrow{\mathbf{x}}_{i} \\ \overline{\partial} \overrightarrow{\Theta}_{i} \end{bmatrix} \right)$$
(B.55)

$$\left[\frac{\partial [\mathbf{I_n}]}{\partial \overrightarrow{\mathbf{\Omega}}}\right] = [\mathbf{0}]_{\mathbf{3} \times \mathbf{9}} \tag{B.56}$$

$$\left[\frac{\partial [\mathbf{I_n}]}{\partial \overleftarrow{\Theta}}\right] = [\mathbf{0}]_{\mathbf{3} \times \mathbf{6n_r}} \tag{B.57}$$

B.5.2 Jacobian of the Inverse of the Moment of Inertia $[I_n]^{-1}$

The moment of inertia matrix $[\mathbf{I}_n]$ is always invertible and by definition of the matrix inverse:

$$[\mathbf{I}_{\mathbf{n}}]^{-1} \cdot [\mathbf{I}_{\mathbf{n}}] = [\mathbf{Id}]_{\mathbf{3} \times \mathbf{3}} \tag{B.58}$$

Since $\left[\frac{\partial [\mathbf{I_n}]^{-1}}{\partial \overrightarrow{\Theta}}\right]$ is a $3 \times 6n_r$ matrix: $\left[\frac{\partial [\mathbf{I_n}]^{-1}}{\partial \overrightarrow{\Theta}}\right] \cdot \left([\mathbf{Id}]_{\mathbf{2n_r} \times \mathbf{2n_r}} \otimes [\mathbf{I_n}]\right) + [\mathbf{I_n}]^{-1} \left[\frac{\partial [\mathbf{I_n}]}{\partial \overrightarrow{\Theta}}\right] = [\mathbf{0}]_{\mathbf{3} \times \mathbf{6n_r}}$ (B.59)

Therefore

$$\left[\frac{\partial [\mathbf{I_n}]^{-1}}{\partial \overrightarrow{\Theta}}\right] = -[\mathbf{I_n}]^{-1} \cdot \left[\left[\frac{\partial [\mathbf{I_n}]}{\partial \overrightarrow{\Theta}}\right] \cdot \left[[\mathbf{Id}]_{\mathbf{2n_r} \times \mathbf{2n_r}} \otimes [\mathbf{I_n}]^{-1} \right] \right]$$
(B.60)

B.6 Jacobian of h at the Origin (0,0)

The Jacobians of the linearised model h at the origin (0,0) are the Jacobians of g at $(\overrightarrow{\mathbf{X}}_{ref}, \overrightarrow{\mathbf{u}}_{ref})$ a general reference trajectory with $\forall t \ge 0 \ \overrightarrow{\mathbf{X}}_{ref} = \begin{bmatrix} \overrightarrow{\mathbf{\Omega}}_{ref} \\ \overrightarrow{\mathbf{\Theta}}_{ref} \\ \vdots \\ \overrightarrow{\mathbf{\Theta}}_{ref} \end{bmatrix}$:

In this study, the reference trajectory is such that $\overrightarrow{\Omega}_{ref}$ is a rigid body torquefree angular velocity where $[\mathbf{I_n}]$ is the time-invariant normalised moment of inertia matrix of the whole object-robot system. $\overrightarrow{\Omega}_{ref}$ follows Euler equation:

$$[\mathbf{I}_{\mathbf{n}}] \cdot \overrightarrow{\mathbf{\Omega}}_{\mathbf{ref}} + (\overrightarrow{\mathbf{\Omega}}_{\mathbf{ref}} \wedge [\mathbf{I}_{\mathbf{n}}] \cdot \overrightarrow{\mathbf{\Omega}}_{\mathbf{ref}}) = \overrightarrow{\mathbf{0}}$$
(B.61)

As per section 4.2.4, $\overrightarrow{\Theta}_{ref}$ represents all the target anchoring location for each

of the robot's module which are immobile:

$$\vec{\Theta}_{ref} = \begin{bmatrix} \frac{\pi}{2(n_r-1)} \\ \frac{\pi}{n_r-1} \\ \vdots \\ \frac{\pi}{2} \\ \pi \end{bmatrix}_{2n_r \times 1}$$
(B.62)

Hence

$$\dot{\overrightarrow{\Theta}}_{ref} = \overrightarrow{\mathbf{0}}_{2n_r \times 1} \tag{B.63}$$

$$\overrightarrow{\mathbf{u}}_{ref} = \overrightarrow{\mathbf{0}}_{2n_r \times 1} \tag{B.64}$$

The Jacobian matrices at the reference trajectory will use the subscript notation ref, i.e. $\begin{bmatrix} \frac{\partial g}{\partial \vec{\mathbf{X}}} \end{bmatrix} (\vec{\mathbf{X}}_{ref}, \vec{\mathbf{u}}_{ref}) = \begin{bmatrix} \frac{\partial g}{\partial \vec{\mathbf{X}}} \end{bmatrix}_{ref}$ and $\begin{bmatrix} \frac{\partial g}{\partial \vec{\mathbf{u}}} \end{bmatrix} (\vec{\mathbf{X}}_{ref}, \vec{\mathbf{u}}_{ref}) = \begin{bmatrix} \frac{\partial g}{\partial \vec{\mathbf{u}}} \end{bmatrix}_{ref}$

The entries of g which are depend on the reference trajectory are the following:

$$\left[\frac{\partial f}{\partial \overrightarrow{\Omega}}\right]_{ref} = -[\mathbf{I}_{\mathbf{n}}]_{ref}^{-1} \cdot \left[[\mathbf{Id}]_{\mathbf{3}\times\mathbf{3}} \wedge [\mathbf{I}_{\mathbf{n}}]_{ref} \cdot \overrightarrow{\Omega}_{ref} + \overrightarrow{\Omega}_{ref} \wedge [\mathbf{I}_{\mathbf{n}}]_{ref} \right]$$
(B.65)

 $\forall i \in [1, n_r]:$

$$\begin{bmatrix} \frac{\partial f}{\partial \overrightarrow{\Theta}_{i}} \end{bmatrix}_{ref} = -\begin{bmatrix} \frac{\partial [\mathbf{I_{n}}]^{-1}}{\partial \overrightarrow{\Theta}_{i}} \end{bmatrix}_{ref} \cdot \left([\mathbf{Id}]_{\mathbf{2} \times \mathbf{2}} \otimes \left[(\overrightarrow{\Omega}_{ref} \wedge [\mathbf{I_{n}}]_{ref} \cdot \overrightarrow{\Omega}_{ref}) \right] \right) \\ - [\mathbf{I_{n}}]_{ref}^{-1} \cdot \left[\overrightarrow{\Omega}_{ref} \wedge \left[\left(\mathbf{2} (\overrightarrow{\mathbf{x}}_{i}^{T})_{ref} \cdot \left[\frac{\partial \overrightarrow{\mathbf{x}}_{i}}{\partial \overrightarrow{\Theta}_{i}} \right]_{ref} \right) \otimes \overrightarrow{\Omega}_{ref} \\ - \begin{bmatrix} \frac{\partial \overrightarrow{\mathbf{x}}_{i}}{\partial \overrightarrow{\Theta}_{i}} \end{bmatrix}_{ref} \otimes ((\overrightarrow{\mathbf{x}}_{i})_{ref} \cdot \overrightarrow{\Omega}_{ref}) \\ - (\overrightarrow{\mathbf{x}}_{i})_{ref} \otimes \left(\overrightarrow{\Omega}_{ref}^{T} \cdot \left[\frac{\partial \overrightarrow{\mathbf{x}}_{i}}{\partial \overrightarrow{\Theta}_{i}} \right]_{ref} \right) \end{bmatrix} \end{bmatrix}$$
(B.66)

$\forall i \in [1, n_r]:$

$$\begin{split} \left[\frac{\partial f}{\partial \overrightarrow{\Theta}_{i}}\right]_{ref} &= -[\mathbf{I}_{n}]_{ref}^{-1} \cdot \left[2\left((\overrightarrow{\mathbf{x}}_{i})_{ref}^{\mathbf{T}} \cdot \left[\frac{\partial \overrightarrow{\mathbf{x}}_{i0}^{\prime\prime}}{\partial \overrightarrow{\Theta}_{i}}\right]_{ref}\right) \otimes \overrightarrow{\Omega}_{ref} - 2\left[\frac{\partial \overrightarrow{\mathbf{x}}_{i0}^{\prime\prime}}{\partial \overrightarrow{\Theta}_{i}}\right]_{ref} \otimes \left((\overrightarrow{\mathbf{x}}_{i})_{ref} \cdot \overrightarrow{\Omega}_{ref}\right) \\ &+ (\overrightarrow{\mathbf{x}}_{i})_{ref} \otimes \left((\overrightarrow{\Omega}_{ref})^{T} \cdot \left((\overrightarrow{\mathbf{x}}_{i})_{ref} \wedge \left[\frac{\partial \overrightarrow{\Psi}_{i}}{\partial \overrightarrow{\Theta}_{i}}\right]_{ref}\right)\right) \\ &+ \left((\overrightarrow{\mathbf{x}}_{i})_{ref} \wedge \left[\frac{\partial \overrightarrow{\Psi}_{i}}{\partial \overrightarrow{\Theta}_{i}}\right]_{ref}\right) \otimes \left((\overrightarrow{\mathbf{x}}_{i})_{ref} \cdot \overrightarrow{\Omega}_{ref}\right) \\ &+ \left[2((\overrightarrow{\mathbf{x}}_{i})_{ref} \cdot (\overrightarrow{\mathbf{x}}_{i})_{ref})[\mathbf{1}] - ((\overrightarrow{\mathbf{x}}_{i})_{ref} \otimes (\overrightarrow{\mathbf{x}}_{i})_{ref}\right) - ((\overrightarrow{\mathbf{x}}_{i})_{ref} \otimes (\overrightarrow{\mathbf{x}}_{i})_{ref})\right] \cdot \left[\frac{\partial \overrightarrow{\Psi}_{i}}{\partial \overrightarrow{\Theta}_{i}}\right]_{ref} \end{split}$$

$$(B.67)$$

$$\forall i \in [1, n_r]:$$

$$\begin{bmatrix} \frac{\partial f}{\partial \overrightarrow{\mathbf{u}}_{i}} \end{bmatrix}_{ref} = -[\mathbf{I}_{\mathbf{n}}]_{ref}^{-1} \cdot \left[[((\overrightarrow{\mathbf{x}}_{i})_{ref} \cdot (\overrightarrow{\mathbf{x}}_{i})_{ref})[\mathbf{1}] - ((\overrightarrow{\mathbf{x}}_{i})_{ref} \otimes (\overrightarrow{\mathbf{x}}_{i})_{ref})] \cdot \left[\frac{\partial \overrightarrow{\overline{\mathbf{\psi}}}_{i}}{\partial \overrightarrow{\mathbf{u}}_{i}} \right]_{ref} \right]$$
(B.68)

Appendix C

Yo-yo de-spin mechanism model verification

In this appendix the discrete model presented in 3.2.5 is used to derive the model of the yo-yo de-spin mechanism presented in 3.1.2 as a sanity check of its correctness and relevance. For memory, model 3.24 is reproduced below:

$$\begin{split} [\mathbf{I}_{\mathbf{n}}] \cdot \overrightarrow{\widehat{\Omega}} + \overrightarrow{\widehat{\Omega}} \wedge [\mathbf{I}_{\mathbf{n}}] \cdot \overrightarrow{\widehat{\Omega}} &= \\ & \sum_{i=1}^{n_{r}} \left(-2\left[(\overrightarrow{\mathbf{x}_{0}''} \cdot \overrightarrow{\mathbf{x}})[\mathbf{1}] - \overrightarrow{\mathbf{x}_{0}''} \otimes \overrightarrow{\mathbf{x}}\right] \cdot \overrightarrow{\widehat{\Omega}} \\ & -\left[\left[(\overrightarrow{\mathbf{x}} \cdot \overrightarrow{\mathbf{x}})[\mathbf{1}] - (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}})\right] \cdot \overrightarrow{\widehat{\Psi}} \\ & + \left[2(\overrightarrow{\mathbf{x}} \cdot \overrightarrow{\mathbf{x}})[\mathbf{1}] - (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) - (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}})\right] \cdot \overrightarrow{\widehat{\Psi}} \right] \quad (C.1) \\ & -\left[(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}_{0}''}) - (\overrightarrow{\mathbf{x}_{0}''} \otimes \overrightarrow{\mathbf{x}})\right] \cdot \overrightarrow{\Psi} \\ & -\left[(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}_{0}''}) \wedge \overrightarrow{\Psi} + ((\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \wedge \overrightarrow{\Psi})^{T}\right] \cdot \overrightarrow{\Omega} \\ & -(\overrightarrow{\mathbf{x}} \wedge \overrightarrow{\mathbf{x}_{0}''}) \right) \end{split}$$

Figure 3.1 is reproduced below to illustrate the various parameters which are going to be defined.

The three frames of reference are:

• $[O] \equiv (O, \vec{\mathbf{x}}, \vec{\mathbf{y}})$ the inertial frame centred on O the centre of mass of the

cylinder.

- $[O'] \equiv (O, \overrightarrow{\mathbf{i}}, \overrightarrow{\mathbf{j}})$ the body frame attached rigidly to the cylinder.
- [O"] ≡ (O, e_r, e_θ) a polar coordinate frame attached to the mass but centred on O as per the hypotheses of the model.



Figure C.1: Yo-Yo De-spin Mechanism Diagram.

Let's define the unit vector of the rotating axis O_Z as:

$$\overrightarrow{\mathbf{k}} = \overrightarrow{\mathbf{x}} \wedge \overrightarrow{\mathbf{y}} = \overrightarrow{\mathbf{i}} \wedge \overrightarrow{\mathbf{j}} = \overrightarrow{\mathbf{e}_{\mathbf{r}}} \wedge \overrightarrow{\mathbf{e}_{\theta}}$$
(C.2)

The rigid body angular velocity is given by:

$$\overrightarrow{\mathbf{\Omega}} = \dot{\theta}_1 \overrightarrow{\mathbf{k}} = \omega \overrightarrow{\mathbf{k}}$$
(C.3)

The absolute angular velocity of the mass is given by:

$$\vec{\Omega}_t = \vec{\Omega} + \vec{\Psi} = \dot{\theta} \vec{k}$$
(C.4)

$$\dot{\theta} = \omega + \Psi$$
 (C.5)

Hence

$$\overrightarrow{\Psi} = (\dot{\theta} - \dot{\theta}_1) \overrightarrow{\mathbf{k}} = \Psi \overrightarrow{\mathbf{k}}$$
(C.6)

As shown in section 3.2.4.2, the model can be expressed in any vector basis and from now on, for all instant in time, all vectors will be expressed in the extended frame of reference $(O, \overrightarrow{\mathbf{e_r}}, \overrightarrow{\mathbf{e_{\theta}}}, \overrightarrow{\mathbf{k}})$. Therefore:

$$\vec{\mathbf{x}} = \vec{\mathbf{r}} = r\vec{\mathbf{e}_{\mathbf{r}}}$$
$$\dot{\vec{\mathbf{x}}} = \dot{\vec{\mathbf{r}}} = \dot{r}\vec{\mathbf{e}_{\mathbf{r}}} + r\dot{\theta}\vec{\mathbf{e}_{\theta}}$$
$$\dot{\vec{\mathbf{x}}_{0}'} = \dot{r}\vec{\mathbf{e}_{\mathbf{r}}}$$
$$(C.7)$$
$$\vec{\mathbf{x}}_{0}'' = \ddot{r}\vec{\mathbf{e}_{\mathbf{r}}}$$

Given the rotational symmetry of the cylinder about the O_Z axis, the total normalised moment of inertia is:

$$\begin{bmatrix} \mathbf{I}_{\mathbf{n}} \end{bmatrix} = \begin{bmatrix} I_r & 0 & 0 \\ 0 & I_{\theta} & 0 \\ 0 & 0 & I_k \end{bmatrix} - \begin{bmatrix} (\overrightarrow{\mathbf{r}} \cdot \overrightarrow{\mathbf{r}}) [\mathbf{1}] - (\overrightarrow{\mathbf{r}} \otimes \overrightarrow{\mathbf{r}}) \end{bmatrix}$$
(C.8)

$$\begin{bmatrix} \mathbf{I_n} \end{bmatrix} = \begin{bmatrix} I_{e_r} & 0 & 0 \\ 0 & I_{\theta} & 0 \\ 0 & 0 & I_k \end{bmatrix} + \begin{bmatrix} r^2 & 0 & 0 \\ 0 & r^2 & 0 \\ 0 & 0 & r^2 \end{bmatrix} - \begin{bmatrix} r \\ 0 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} r & 0 & 0 \end{bmatrix}$$
(C.9)

$$[\mathbf{I_n}] = \begin{bmatrix} I_{e_r} & 0 & 0 \\ 0 & I_{\theta} + r^2 & 0 \\ 0 & 0 & I_k + r^2 \end{bmatrix}$$
(C.10)

$$[\mathbf{I_n}] \cdot \dot{\overrightarrow{\Omega}} = \begin{bmatrix} I_{e_r} & 0 & 0 \\ 0 & I_{\theta} + r^2 & 0 \\ 0 & 0 & I_k + r^2 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ \dot{\omega} \end{bmatrix}$$
(C.11)

$$\vec{\boldsymbol{\Omega}} \wedge [\mathbf{I}_{\mathbf{n}}] \cdot \vec{\boldsymbol{\Omega}} = \begin{bmatrix} 0\\0\\\omega \end{bmatrix} \wedge \begin{bmatrix} I_{e_{r}} & 0 & 0\\0 & I_{\theta} + r^{2} & 0\\0 & 0 & I_{k} + r^{2} \end{bmatrix} \cdot \begin{bmatrix} 0\\0\\\omega \end{bmatrix} = \vec{\mathbf{0}}$$
(C.12)

$$\left[(\overrightarrow{\mathbf{x}_{0}''}\cdot\overrightarrow{\mathbf{x}})[\mathbf{1}]-\overrightarrow{\mathbf{x}_{0}''}\otimes\overrightarrow{\mathbf{x}}\right]\cdot\overrightarrow{\mathbf{\Omega}}=\left[(\overrightarrow{\mathbf{r}}\overrightarrow{\mathbf{e}_{\mathbf{r}}}\cdot\overrightarrow{\mathbf{r}}\overrightarrow{\mathbf{e}_{\mathbf{r}}})[\mathbf{1}]-\overrightarrow{\mathbf{r}}\overrightarrow{\mathbf{e}_{\mathbf{r}}}\otimes\overrightarrow{\mathbf{r}}\overrightarrow{\mathbf{e}_{\mathbf{r}}}\right]\cdot\overrightarrow{\mathbf{\Omega}}$$
(C.13)

$$\begin{bmatrix} (\vec{\mathbf{x}_0''} \cdot \vec{\mathbf{x}}) [\mathbf{1}] - \vec{\mathbf{x}_0''} \otimes \vec{\mathbf{x}} \end{bmatrix} = \begin{bmatrix} \dot{r}r & 0 & 0 \\ 0 & \dot{r}r & 0 \\ 0 & 0 & \dot{r}r \end{bmatrix} - \begin{bmatrix} \dot{r}r & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$
(C.14)
$$\begin{bmatrix} (\vec{\mathbf{x}_0''} \cdot \vec{\mathbf{x}}) [\mathbf{1}] - \dot{\vec{\mathbf{x}_0''}} \otimes \vec{\mathbf{x}} \end{bmatrix} \cdot \vec{\Omega} = \begin{bmatrix} 0 \\ 0 \\ \dot{r}r\omega \end{bmatrix}$$
(C.15)

The same kind of calculations lead to:

$$\left[(\overrightarrow{\mathbf{x}} \cdot \overrightarrow{\mathbf{x}}) [\mathbf{1}] - (\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}}) \right] \cdot \dot{\overrightarrow{\mathbf{\Psi}}} = \mathbf{r}^{\mathbf{2}} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ \dot{\Psi} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ r^{2} \dot{\Psi} \end{bmatrix}$$
(C.16)

$$\begin{bmatrix} 2(\vec{\mathbf{x}} \cdot \dot{\vec{\mathbf{x}}})[\mathbf{1}] - (\dot{\vec{\mathbf{x}}} \otimes \vec{\mathbf{x}}) - (\vec{\mathbf{x}} \otimes \dot{\vec{\mathbf{x}}}) \end{bmatrix} \cdot \vec{\mathbf{\Psi}} = \begin{bmatrix} 0\\0\\2r\dot{r}\Psi \end{bmatrix}$$
(C.17)

$$\left[(\overrightarrow{\mathbf{x}} \otimes \overrightarrow{\mathbf{x}_0''}) - (\overrightarrow{\mathbf{x}_0''} \otimes \overrightarrow{\mathbf{x}}) \right] \cdot \overrightarrow{\mathbf{\Psi}} = \left[r\dot{r}(\overrightarrow{\mathbf{e_r}} \otimes \overrightarrow{\mathbf{e_r}}) - \dot{r}r(\overrightarrow{\mathbf{e_r}} \otimes \overrightarrow{\mathbf{e_r}}) \right] \cdot \overrightarrow{\mathbf{\Psi}} = \overrightarrow{\mathbf{0}} \qquad (C.18)$$

By property of the dyadic product (see appendix A of [28]):

$$(\vec{\mathbf{x}} \otimes \vec{\mathbf{x}}) \wedge \vec{\mathbf{\Psi}} = \vec{\mathbf{x}} \otimes (\vec{\mathbf{x}} \wedge \vec{\mathbf{\Psi}}) = \begin{bmatrix} r \\ 0 \\ 0 \end{bmatrix} \otimes \left(\begin{bmatrix} r \\ 0 \\ 0 \end{bmatrix} \wedge \begin{bmatrix} 0 \\ 0 \\ \Psi \end{bmatrix} \right) = \begin{bmatrix} r \\ 0 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 & -r\Psi & 0 \end{bmatrix}$$
$$(\vec{\mathbf{x}} \otimes \vec{\mathbf{x}}) \wedge \vec{\mathbf{\Psi}} = \vec{\mathbf{x}} \otimes (\vec{\mathbf{x}} \wedge \vec{\mathbf{\Psi}}) = \begin{bmatrix} 0 & -r^2\Psi & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Which leads to:

$$\begin{bmatrix} (\vec{\mathbf{x}} \otimes \vec{\mathbf{x}}) \wedge \vec{\Psi} + ((\vec{\mathbf{x}} \otimes \vec{\mathbf{x}}) \wedge \vec{\Psi})^T \end{bmatrix} \cdot \vec{\Omega} = \begin{bmatrix} 0 & -r^2 \Psi & 0 \\ -r^2 \Psi & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} = \vec{\mathbf{0}} \quad (C.19)$$

$$\overrightarrow{\mathbf{x}} \wedge \overrightarrow{\overline{\mathbf{x}_0''}} = \ddot{r}r \overrightarrow{\mathbf{e_r}} \wedge \overrightarrow{\mathbf{e_r}} = \overrightarrow{\mathbf{0}}$$
(C.20)

Adding up C.11 C.12, C.15 C.16 C.17 C.18 C.19 and C.20 leads to the following differential equation in the k coordinate:

$$(I_k + r^2)\dot{\omega} + 2\dot{r}r\omega + r^2\dot{\Psi} + 2\dot{r}r\Psi = 0$$
(C.21)

Remembering equation C.5 and that equation C.21 is normalised per unit mass of the yo-yo's mass m, C.21 can be rewritten as:

$$(I_k + mr^2)\dot{\omega} + 2m\dot{r}r\omega + mr^2\dot{\Psi} + 2m\dot{r}r\Psi = 0$$
 (C.22)

Then:

$$(I_k + mr^2)\dot{\omega} + 2m\dot{r}r\omega + mr^2(\ddot{\theta} - \dot{\omega}) + 2m\dot{r}r(\dot{\theta} - \omega) = 0 \qquad (C.23)$$

Finally:

$$I_k \dot{\omega} + mr^2 \ddot{\theta} + 2m\dot{r}r\dot{\theta} = 0 \tag{C.24}$$

which is the derivative of the total angular momentum of the yo-yo de-spin mechanism described by equation 3.4 in chapter 3. Therefore, from the model C.1 the yo-yo de-spin mechanism's model can be derived.

Appendix D

Holonomic Constraint for Ellipsoidal Objects with Consistent Vector Dimension Convention

The moving part of the system be it a mass or robot module is subject to an holonomic constraint by virtue of the fact that it remains in contact to the surface of the object at all times.

The choice of spherical coordinates to describe the motion of a moving point mass relative to an ellipsoidal object can be exploited to eliminate the radial variables r, \dot{r} and \ddot{r} by expressing them as a function of the spherical angles and their respective velocities and accelerations namely ϕ , $\dot{\phi}$, $\ddot{\phi}$ and θ , $\dot{\theta}$.

In this appendix, the full derivation of these radial quantities is performed and shown form a programmatic perspective i.e. as they were implemented in the simulation. The adopted vector convention is that all vectors are represented as column vectors. All data length are normalised. The position of moving part relative to the object is in spherical coordinates:

$$\overrightarrow{\mathbf{r}} = r \overrightarrow{\mathbf{e}_r} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = r \begin{bmatrix} \cos(\phi) \sin(\theta) \\ \sin(\phi) \sin(\theta) \\ \cos(\theta) \end{bmatrix}$$
(D.1)

The Cartesian equation for an ellipsoid is:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1 \tag{D.2}$$

where the strictly positive parameters a, b and c are the half lengths of the principal axes.

In spherical coordinates:

$$r^{2} \left(\frac{\cos(\phi)^{2} \sin(\theta)^{2}}{a^{2}} + \frac{\sin(\phi)^{2} \sin(\theta)^{2}}{b^{2}} + \frac{\cos(\theta)^{2}}{c^{2}} \right) = 1$$
(D.3)

Defining *base* as:

$$base = \frac{\cos(\phi)^2 \sin(\theta)^2}{a^2} + \frac{\sin(\phi)^2 \sin(\theta)^2}{b^2} + \frac{\cos(\theta)^2}{c^2}$$
(D.4)

then:

$$r^2 = \frac{1}{base} \tag{D.5}$$

$$r = \sqrt{\frac{1}{base}} \tag{D.6}$$

Deriving D.5 with respect to time gives:

$$2r\dot{r} = -\frac{\dot{base}}{base^2} \tag{D.7}$$

Defining *prime* as:

$$prime = \frac{dbase}{dt} = base \tag{D.8}$$

Finally

$$\dot{r} = -\frac{1}{2}r^3 prime \tag{D.9}$$

Deriving D.9 with respect to time gives:

$$\ddot{r} = -\frac{1}{2}[3r^2\dot{r}prime + r^3prime]$$
(D.10)

Using D.9 and defining:

$$second = \frac{dprime}{dt} = prime$$
 (D.11)

Finally

$$\ddot{r} = \frac{3}{4}r^5 prime^2 - \frac{1}{2}r^3 second \tag{D.12}$$

The derivation of *prime* and *second* relies on using the following trigonometric identities:

$$sin(2\alpha) = 2sin(\alpha)cos(\alpha)$$
 (D.13)

$$\cos(2\alpha) = 2\cos^2(\alpha) - 1 \tag{D.14}$$

For *prime*:

$$prime = \frac{1}{a^2} [\cos(\phi)^2 \sin(2\theta)\dot{\theta} - \sin(2\phi)\dot{\phi}\sin(\theta)^2] + \frac{1}{b^2} [\sin(\phi)^2 \sin(2\theta)\dot{\theta} + \sin(2\phi)\dot{\phi}\sin(\theta)^2] - \frac{1}{c^2} [\sin(2\theta)\dot{\theta}]$$
(D.15)

For *second*, in order to make its derivation more readable, *prime* can be broken in 3 parts corresponding to each axis of the ellipsoid

$$second = prime_a + prime_b + prime_c$$
 (D.16)

With

$$pri\dot{m}e_{a} = \frac{1}{a^{2}}[\cos(\phi)^{2}\cos(2\theta)2\dot{\theta}^{2} + \cos(\phi)^{2}\sin(2\theta)\ddot{\theta} - 2\sin(2\phi)\dot{\phi}\sin(2\theta)\dot{\theta} - 2\cos(2\phi)\dot{\phi}^{2}\sin(\theta)^{2} - \sin(2\phi)\ddot{\phi}\sin(\theta)^{2}]$$
(D.17)

$$pri\dot{m}e_{b} = \frac{1}{b^{2}}[\sin(\phi)^{2}2\cos(2\theta)\dot{\theta}^{2} + \sin(\phi)^{2}\sin(2\theta)\ddot{\theta} + 2\sin(2\phi)\dot{\phi}\sin(2\theta)\dot{\theta} + 2\cos(2\phi)\dot{\phi}^{2}\sin(\theta)^{2} + \sin(2\phi)\ddot{\phi}\sin(\theta)^{2}]$$
(D.18)

$$prime_c = -\frac{1}{c^2} [2\cos(2\theta)\dot{\theta}^2 + \sin(2\theta)\ddot{\theta}]$$
(D.19)

Appendix E

Explicit Solution of the Euler equation for Asymetric Body with Zero External Moment

As explained in [64], page 126-130, $\frac{h^2}{2T}$ is a constant parameter characterising the rotational state of the asymmetric object. The existence of a solution to the Euler equations for an asymmetric body with unequal principal moments of inertia $(I_{min}, I_{mid}, I_{max})$ requires that the ratio $\frac{h^2}{2T} \in [I_{min}, I_{max}]$ where the moment of inertia $I_{min} < I_{mid} < I_{max}$ are interchangeably and without loss of generality taken along the respective X, Y, and Z axes of the body. The derivation of a solution of the Euler equation for asymmetric bodies proposed in the literature is outlined and restricted to the agreeable case where $\frac{h^2}{2T} < I_{mid}$. Each component of the angular velocity can be explicitly expressed with Jacobi's elliptic integral functions and in particular the sinus amplitudinis function defined below:

$$u = F(\phi, m) = \int_0^\phi \frac{d\theta}{\sqrt{1 - m\sin(\theta)^2}}$$
(E.1)

The restriction mentioned above corresponds to the case where the integral is well defined $\forall \phi \in \mathbb{R}$ when $m \in [0, 1]$. In other words for all instants in time [29].

In this appendix, a full and explicit re-derivation is proposed and extended to the cases where m > 1 to cover all possible solution over the allowed range $\frac{h^2}{2T} \in$ $[I_{min}, I_{max}].$

E.1 General solution of the Euler equations for asymmetric bodies

For a body with unequal principal moments of inertia $I_{min} < I_{mid} < I_{max}$ (hypothesis (H.1)) taken along the principal axes designated as min, mid, and max respectively and with angular velocity

$$\overrightarrow{\mathbf{\Omega}} = \begin{bmatrix} \omega_{max} \\ \omega_{mid} \\ \omega_{min} \end{bmatrix}$$
 the torque free Euler equations are:

$$I_{max}\dot{\omega}_{max} + (I_{min} - I_{mid})\omega_{mid}\omega_{min} = 0$$
 (E.2)

$$I_{mid}\dot{\omega}_{mid} + (I_{max} - I_{min})\omega_{max}\omega_{min} = 0$$
 (E.3)

$$I_{min}\dot{\omega}_{min} + (I_{mid} - I_{max})\omega_{max}\omega_{mid} = 0$$
 (E.4)

In a torque free rotational motion, the angular momentum $\overrightarrow{\mathbf{h}}$ is constant about the centre of mass:

$$\overrightarrow{\mathbf{h}} = \overrightarrow{\mathbf{constant}}$$
 (E.5)

Hence its norm is constant as well:

$$h^{2} = \parallel \overrightarrow{\mathbf{h}} \parallel^{2} = I_{max}^{2} \omega_{max}^{2} + I_{mid}^{2} \omega_{mid}^{2} + I_{min}^{2} \omega_{min}^{2} = constant$$
(E.6)

The rotational kinetic energy T in the absence of external work done to the object is also constant:

$$T = constant$$
 (E.7)

with

$$2T = \overrightarrow{\mathbf{h}} \cdot \overrightarrow{\mathbf{\Omega}} = I_{max}\omega_{max}^2 + I_{mid}\omega_{mid}^2 + I_{min}\omega_{min}^2$$
(E.8)

From the Euler equations (E.2), (E.3) and (E.4), the following equations can be derived:

$$h^{2} - 2TI_{max} = I_{mid}(I_{mid} - I_{max})\omega_{mid}^{2} + I_{min}(I_{min} - I_{max})\omega_{min}^{2}$$
(E.9)

$$h^{2} - 2TI_{mid} = I_{max}(I_{max} - I_{mid})\omega_{max}^{2} + I_{min}(I_{min} - I_{mid})\omega_{min}^{2}$$
(E.10)

$$h^{2} - 2TI_{min} = I_{max}(I_{max} - I_{min})\omega_{max}^{2} + I_{mid}(I_{mid} - I_{min})\omega_{mid}^{2}$$
(E.11)

As per our above hypothesis (H.1), the equation (E.9) is always negative and (E.11) always positive. (E.10) can either be positive or negative.

For the limit cases $h^2 = 2TI_{max}$ or $h^2 = 2TI_{min}$ the solutions are straightforwardly $\omega_{max} = \left(\frac{2T}{I_{max}}\right)^{\frac{1}{2}}$ and $\omega_{min} = \left(\frac{2T}{I_{min}}\right)^{\frac{1}{2}}$ respectively while the other two angular velocities are nil.

For the cases $h^2 \in]2TI_{min}, 2TI_{max}[$, the derivation is as follows: ω_{max} and ω_{min} can be conveniently expressed as function of ω_{mid} :

In (E.11):

$$\omega_{max}^2 = \frac{h^2 - 2TI_{min}}{I_{max}(I_{max} - I_{min})} \left(1 - \frac{I_{mid}(I_{mid} - I_{min})}{h^2 - 2TI_{min}} \omega_{mid}^2 \right)$$
(E.12)

In (E.9):

$$\omega_{min}^2 = \frac{2TI_{max} - h^2}{I_{min}(I_{max} - I_{min})} \times \left(1 - \left(\frac{I_{max} - I_{min}}{I_{mid} - I_{min}}\right) \left(\frac{h^2 - 2TI_{min}}{2TI_{max} - h^2}\right) \left(\frac{I_{mid}(I_{mid} - I_{min})}{h^2 - 2TI_{min}}\right) \omega_{mid}^2\right)$$
(E.13)

Then changing variable by defining

$$v = \sqrt{\frac{I_{mid}(I_{mid} - I_{min})}{h^2 - 2TI_{min}}}\omega_{mid}$$
(E.14)

leads to

$$\omega_{mid} = \sqrt{\frac{h^2 - 2TI_{min}}{I_{mid}(I_{mid} - I_{min})}}v \tag{E.15}$$

and

$$\dot{\omega}_{mid} = \sqrt{\frac{h^2 - 2TI_{min}}{I_{mid}(I_{mid} - I_{min})}}\dot{v}$$
(E.16)

while by defining $\forall h^2 \in]2TI_{min}, 2TI_{max}[$:

$$k = \sqrt{\left(\frac{I_{max} - I_{mid}}{I_{mid} - I_{min}}\right) \left(\frac{h^2 - 2TI_{min}}{2TI_{max} - h^2}\right)}$$
(E.17)

(E.12) and (E.13) become

$$\omega_{max}^2 = \frac{h^2 - 2TI_{min}}{I_{max}(I_{max} - I_{min})} \left(1 - v^2\right)$$
(E.18)

$$\omega_{min}^2 = \frac{2TI_{max} - h^2}{I_{min}(I_{max} - I_{min})} \left(1 - k^2 v^2\right)$$
(E.19)

(E.18) and (E.19) require $1-v^2 \ge 0$ and $1-k^2v^2 \ge 0$ to be physically meaningful. Assuming so, (E.4) can be transformed into

$$I_{mid} \sqrt{\frac{h^2 - 2TI_{min}}{I_{mid}(I_{mid} - I_{min})}} \dot{v} = (I_{min} - I_{max}) \times$$

$$\sqrt{\frac{h^2 - 2TI_{min}}{I_{max}(I_{max} - I_{min})}} \left(1 - v^2\right)} \sqrt{\frac{2TI_{max} - h^2}{I_{min}(I_{max} - I_{min})}} \left(1 - k^2 v^2\right)}$$
(E.20)

and

$$\frac{\dot{v}}{\sqrt{(1-v^2)(1-k^2v^2)}} = -\sqrt{\frac{(I_{mid} - I_{min})(2TI_{max} - h^2)}{I_{max}I_{mid}I_{min}}}$$
(E.21)

Now defining the constant N as:

$$N = \sqrt{\frac{(I_{mid} - I_{min})(2TI_{max} - h^2)}{I_{max}I_{mid}I_{min}}}$$
(E.22)

And using the function G defined as:

$$G(\alpha) = \int_0^\alpha \frac{dv}{\sqrt{(1-v^2)(1-k^2v^2)}}$$
(E.23)

whose time derivative is:

$$\frac{G(\alpha)}{dt} = \frac{G(\alpha)}{d\alpha}\frac{d\alpha}{dt} = \frac{\dot{\alpha}}{\sqrt{(1-\alpha^2)(1-k^2\alpha^2)}}$$
(E.24)

Integrating (E.21) with respect to time gives:

$$\int_0^t \frac{\dot{v} \cdot dt}{\sqrt{(1 - v^2)(1 - k^2 v^2)}} = -Nt \tag{E.25}$$

and

$$G(\alpha) = \int_0^\alpha \frac{dv}{\sqrt{(1 - v^2)(1 - k^2 v^2)}} = -Nt$$
 (E.26)

Changing variable again by defining:

$$v = \sin(\theta) \tag{E.27}$$

With $v \in [0, \alpha]$ and $\theta \in [0, \sin^{-1}(\alpha)]$, $dv = \cos(\theta)d\theta$ and by defining $\phi = \sin^{-1}(\alpha)$, (E.26) becomes:

$$\int_{0}^{\phi} \frac{\cos(\theta)d\theta}{\sqrt{(1-\sin(\theta)^{2})(1-k^{2}\sin(\theta)^{2})}} = \int_{0}^{\phi} \frac{d\theta}{\sqrt{(1-k^{2}\sin(\theta)^{2})}} = -Nt$$
(E.28)

which is the sinus amplitudinis function:

$$F(\phi, k^2) = -Nt \tag{E.29}$$

E.2 Mapping k with the values of $\frac{h^2}{2T}$

In [64] the derivation of the Euler equation is restricted to the case where $\frac{h^2}{2T} < I_{mid}$. In this section it will proven that $\frac{h^2}{2T} < I_{mid}$ is equivalent to $0 < k \leq 1$ which is in turn equivalent to the assertion $\frac{h^2}{2T} > I_{mid}$ is equivalent to k > 1. Solving the Euler equation for all possible values of $\frac{h^2}{2T}$ can then be split into 2 cases: $0 < k \leq 1$ where the sinus amplitudinis function can be used directly and k > 1 where the sinus amplitudinis function requires a transformation which will be the subject of the next section.

Since k is positive, one can reason by equivalence between the following equations:

$$k^2 \le 1 \tag{E.30}$$

$$\frac{I_{max} - I_{mid}}{I_{mid} - I_{min}} \frac{h^2 - 2TI_{min}}{2TI_{max} - h^2} \le 1$$
(E.31)

$$\frac{h^2 - 2TI_{min}}{2TI_{max} - h^2} \le \frac{I_{mid} - I_{min}}{I_{max} - I_{mid}}$$
(E.32)

$$\frac{h^2 - 2TI_{mid} + 2TI_{mid} - 2TI_{min}}{2TI_{max} - 2TI_{mid} + 2TI_{mid} - h^2} \le \frac{I_{mid} - I_{min}}{I_{max} - I_{mid}}$$
(E.33)

$$\frac{h^2 - 2TI_{mid} + 2T(I_{mid} - I_{min}) \leq}{(I_{mid} - I_{min})(2T(I_{max} - I_{mid}) - (h^2 - 2TI_{mid}))}_{I_{max} - I_{mid}}$$
(E.34)

since $I_{max} - I_{mid} \ge 0$

$$(I_{max} - I_{mid})(h^2 - 2TI_{mid}) + 2T(I_{max} - I_{mid})(I_{mid} - I_{min})$$

$$\leq 2T(I_{max} - I_{mid})(I_{mid} - I_{min}) - (h^2 - 2TI_{mid})(I_{mid} - I_{min})$$
(E.35)

leading to

$$(I_{max} - I_{min})(h^2 - 2TI_{mid}) \le 0$$
 (E.36)

Finally since $I_{max} - I_{min} \ge 0$, we have directly

$$\frac{h^2}{2T} \le I_{mid} \tag{E.37}$$

E.3 Euler equation solution

1st CASE $0 < k \leq 1$: Under this condition, (E.29) is well defined $\forall \phi \in \mathbb{R}$.

Noting that the sine amplitude elliptic function is the inverse of the incomplete integral of the first kind i.e. of the function G:

$$G[sn(-Nt,k^2)] = -Nt \tag{E.38}$$

finally leads to

$$v = sn(-Nt, k^2) \tag{E.39}$$

The sine amplitude elliptic function sn(x,m) is an odd function with respect to x. Therefore:

$$v = -sn(Nt, k^2) \tag{E.40}$$

Consequently:

$$\omega_{mid} = -\sqrt{\frac{h^2 - 2TI_{min}}{I_{mid}(I_{mid} - I_{min})}} sn(Nt, k^2)$$
(E.41)

The cosine amplitude elliptic function cn(x,m) and the delta amplitude elliptic function dn(x,m) are an even functions with respect to x. Moreover, $\forall (x,m) \in \mathbb{C} \times [0,1]$, $sn^2(x,m) + cn^2(x,m) = 1$ and $m \cdot sn^2(x,m) + dn^2(x,m) = 1$. Therefore (E.18) and (E.19) finally become:

$$\omega_{max} = \sqrt{\frac{h^2 - 2TI_{min}}{I_{max}(I_{xmax} - I_{min})}} cn(Nt, k^2)$$
(E.42)

$$\omega_{min} = \sqrt{\frac{2TI_{max} - h^2}{I_{min}(I_{max} - I_{min})}} dn(Nt, k^2)$$
(E.43)

(E.41), (E.42) and (E.43) are the solutions found in the literature $\forall (t, k^2) \in \mathbb{R} \times [0, 1].$

<u>2nd CASE k > 1</u>: Changing variable by defining x = kv, giving with $dv = \frac{dx}{k}$

and $\dot{v} = \frac{\dot{x}}{k}$, (E.21) changes to:

$$\frac{\dot{x}}{\sqrt{(1-\frac{x^2}{k^2})(1-x^2)}} = -kN \tag{E.44}$$

Changing (E.23) into:

$$G(\alpha) = \int_0^\alpha \frac{dx}{\sqrt{(1 - \frac{x^2}{k^2})(1 - x^2)}} = -kNt$$
(E.45)

In this case, since $\frac{1}{k^2} < 1$, the first case result can be used:

$$G[sn(-kNt, \frac{1}{k^2})] = -kNt \tag{E.46}$$

Leading to:

$$x = -sn(kNt, \frac{1}{k^2}) \tag{E.47}$$

Hence

$$v = -\frac{1}{k}sn(kNt, \frac{1}{k^2}) \tag{E.48}$$

As per Abramowitz-Stegun [3]:

$$\frac{1}{k}sn(kNt, \frac{1}{k^2}) = sn(Nt, \frac{1}{k^2})$$
(E.49)

The solution of the Euler equations is then straightforwardly the same as for the first case and valid $\forall (t, k^2) \in \mathbb{R} \times [1, +\infty[$. Combining the two cases the final solution is $\forall (t, k^2) \in \mathbb{R} \times [0, +\infty[$:

$$\omega_{mid} = -\sqrt{\frac{h^2 - 2TI_{min}}{I_{mid}(I_{mid} - I_{min})}} sn(Nt, k^2)$$
(E.50)

$$\omega_{max} = \sqrt{\frac{h^2 - 2TI_{min}}{I_{max}(I_{xmax} - I_{min})}} cn(Nt, k^2)$$
(E.51)

$$\omega_{min} = \sqrt{\frac{2TI_{max} - h^2}{I_{min}(I_{max} - I_{min})}} dn(Nt, k^2)$$
(E.52)

E.4 Proof of the Abramowitz-Stegun formulae

The proof of the formulae is based on a change of variable found in [29] page 47-48 and developed in Arkadiusz Jadczyk's blog [30] (accessed January 2020). The following proof is a variation of the above and clarifies the final step of the derivation of the Euler equations.

Going back to (E.1), $\forall k > 1, \exists \phi_c \in [0, 2\pi[$ such that $k \cdot \sin(\phi_c) = 1$ or $\phi_c = \sin^{-1}(\frac{1}{k})$.

It was already seen in the first case that for $v \in [0, \sin(\phi)]$ with $\phi \leq \phi_c$:

$$u = F(\phi, k^2) = \int_0^{\sin(\phi)} \frac{dv}{\sqrt{(1 - v^2)(1 - k^2 v^2)}}$$
(E.53)

Since $\sin(\phi) = sn(u, k^2)$, (E.53) can be rewritten as:

$$u = \int_0^{sn(u,k^2)} \frac{dv}{\sqrt{(1-v^2)(1-k^2v^2)}}$$
(E.54)

Multiplying by k

$$k \cdot u = \int_0^{sn(u,k^2)} \frac{kdv}{\sqrt{(1-v^2)(1-k^2v^2)}}$$
(E.55)

Then using the change of variable $\psi = kv$ under the constraint $\psi \in [0, k \sin(\phi)]$ with $\phi \leq \phi_c$, (E.55) becomes:

$$k \cdot u = \int_0^{ksn(u,k^2)} \frac{d\psi}{\sqrt{(1 - \frac{\psi^2}{k^2})(1 - \psi^2)}}$$
(E.56)

As per the definition of ϕ_c , $\forall \phi \leq \phi_c, \exists \beta(\phi)$ such that the change of variable $\sin(\beta(\phi)) = k \sin(\phi)$ is possible. (E.56) becomes:

$$k \cdot u = \int_0^{\sin(\beta)} \frac{d\psi}{\sqrt{(1 - \frac{\psi^2}{k^2})(1 - \psi^2)}} = F(\beta, \frac{1}{k^2})$$
(E.57)

By definition, $sin(\beta) = sn(ku, \frac{1}{k^2})$ hence we have finally:

$$sn(u,k^2) = \frac{1}{k}sn(ku,\frac{1}{k^2})$$
 (E.58)
Bibliography

- M-tran by aist and tokyo tech (https://erc-bpgc.github.io/blog/blog/mssr/), 2020.
- [2] Py trees documentation: https://py-trees.readthedocs.io/en/devel/index.html, 2022.
- [3] M. Abramowitz and I. A. Stegun. Handbook of mathematical functions : with formulas, graphs and mathematical tables. New York, N.Y. : Dover, 1965.
- [4] S. Bektas. Intersection of an Ellipsoid and a Plane. International Journal of Research in Engineering and Applied Sciences, 6:273–283, 2016.
- [5] J. R. Brophy. Asteroid Retrieval Feasibility Study 2. (April):1–51, 2012.
- [6] Z. Bubnicki. Modern Control Theory. Springer: Berlin., 2005.
- [7] S. V. Burns Joseph. Asteroid Nutation Angles. Royal Astronomical Society, 1973.
- [8] E. F. Camacho. Model predictive control E.F. Camacho and C. Bordons. Advanced textbooks in control and signal processing. Springer, London, second edition. edition, 2007.
- [9] R. Cechowicz. Bias drift estimation for mems gyroscope used in inertial navigation. Acta Mechanica et Automatica, 11:104–110, 06 2017.
- [10] F. E. Cellier. Continuous System Simulation by François E. Cellier, Ernesto Kofman. Springer US, Boston, MA, 1st ed. 2006. edition, 2006.
- [11] U. N. M. I. Center. Platinum-group metals statistics and information, 2012.

- [12] H. CHEN and F. ALLGöWER. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica (Oxford)*, 34(10):1205–1217, 1998.
- [13] D. Christensen. Elements of Autonomous Self-Reconfigurable Robots. PhD thesis, 01 2008.
- [14] S. Cicalò and D. J. Scheeres. Averaged rotational dynamics of an asteroid in tumbling rotation under the yorp torque. *Celestial mechanics and dynamical astronomy*, 106(4):301–337, 2010.
- [15] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. Numerical Optimization. MIT Press: Cambridge, Mass. 3rd. ed., 2009.
- [16] H. D. Curtis. Orbital Mechanics for Engineering Students. Elsevier, 2010.
- [17] L. Cveticanin. Dynamics of Bodies with Time-Variable Mass. Springer: Switzerland., 2016.
- [18] R. Edgar and P. Artymowicz. Pumping of a planetesimal disc by a rapidly migrating planet. Monthly notices of the Royal Astronomical Society, 354(3):769– 772, 2004.
- [19] A. Ellery. P a c e. pages 118–121, 2003.
- [20] M. Elvis. How many ore-bearing asteroids? Planetary and Space Science, 91:20-26, 2014.
- [21] P. Falcone, F. Borrelli, H. E. Tseng, J. Asgari, and D. Hrovat. Linear timevarying model predictive control and its application to active steering systems: Stability analysis and experimental validation. *International journal of robust* and nonlinear control, 18(8):862–875, 2008.
- [22] S. Foundation. Space foundation 2015 annual report, 2015.
- [23] M. Fromherz and W. Jackson. Predictable motion of hyper-redundant manipulators using constrained optimization control. 07 2000.

- [24] A. Grancharova and T. A. Johansen. Explicit Nonlinear Model Predicitve Control. Springer-Verlag, Berlin, 2012.
- [25] H. F. Grip and M. Ono. Modeling and Simulation of Asteroid Retrieval Using a Flexible Capture Mechanism. 2013.
- [26] L. Grüne and J. Pannek. Nonlinear Model Predictive Control Theory and Algorithms. Springer: Switzerland. 2nd ed., 2017.
- [27] A. W. Guess. Poynting-Robertson effect for a spherical source of radiation. apj, 135:855–866, May 1962.
- [28] G. W. Hanson and A. B. Yakovlev. Operator Theory for Electromagnetics. Springer: New-York., 2002.
- [29] R. Hoppe. Elliptische Integrale und Funktionen nach Jacobi. 2015.
- [30] A. Jadczyk. Open system ark's blog physics without frontiers, 2017.
- [31] S. Julier and J. Uhlmann. Unscented filtering and nonlinear estimation. Proceedings of the IEEE, 92(3):401–422, 2004.
- [32] S. Julier, J. Uhlmann, and H. Durrant-Whyte. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE transactions on automatic control*, 45(3):477–482, 2000.
- [33] Q. M. Lam, N. Stamatakos, C. Woodruff, and S. Ashton. Gyro modeling and estimation of its random noise sources. AIAA Guidance, Navigation, and Control Conference and Exhibit, 2003.
- [34] J. S. Lewis. Asteroid Mining 101. Deep Space Industries Inc., 2015.
- [35] P. G. Lukaszynski. Attitude Control Hardware and Software. 2013.
- [36] A. Mainzer, T. Grav, J. Bauer, J. Masiero, R. S. McMillan, R. M. Cutri, R. Walker, E. Wright, P. Eisenhardt, D. J. Tholen, T. Spahr, R. Jedicke, L. Denneau, E. DeBaun, D. Elsbury, T. Gautier, S. Gomillion, E. Hand, W. Mo, J. Watkins, A. Wilkins, G. L. Bryngelson, A. Del Pino Molina, S. Desai, M. G.

Camus, S. L. Hidalgo, I. Konstantopoulos, J. A. Larsen, C. Maleszewski, M. A. Malkan, J.-C. Mauduit, B. L. Mullan, E. W. Olszewski, J. Pforr, A. Saro, J. V. Scotti, and L. H. Wasserman. Neowise Observations of Near-Earth Objects: Preliminary Results. *The Astrophysical Journal*, 743(2):156, 2011.

- [37] D. Mayne and H. Michalska. Receding horizon control of nonlinear systems. *IEEE transactions on automatic control*, 35(7):814–824, 1990.
- [38] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica (Oxford)*, 36(6):789–814, 2000.
- [39] K. McCarthy. H.r.2262 u.s. commercial space launch competitiveness act, 2015.
- [40] H. Michalska and D. Mayne. Robust receding horizon control of constrained nonlinear systems. *IEEE transactions on automatic control*, 38(11):1623–1633, 1993.
- [41] A. Morbidelli, W. Jr, and P. Michel. Origin and evolution of near-earth objects. Asteroids III, 3, 01 2002.
- [42] J. Nocedal and S. Wright. Numerical Optimization. Springer 2nd. ed., 2006.
- [43] D. P. O'Brien, A. Morbidelli, and W. F. Bottke. The primordial excitation and clearing of the asteroid belt—revisited. *Icarus (New York, N.Y. 1962)*, 191(2):434–452, 2007.
- [44] J.-L. Pac. Systemes Dynamiques. Dunod: Paris., 2012.
- [45] L. Pedersen, D. Kortenkamp, D. Wettergreen, and I. Nourbakhsh. A Survey of Space Robotics. 7th International Symposium on Artificial Intelligent, Robotics and Automation in Space (i-SAIRAS-03, page 8 pp., 2003.
- [46] F. A. Potra and S. J. Wright. Interior-point methods. Journal of Computational and Applied Mathematics, 124(1):281–302, 2000. Numerical Analysis 2000. Vol. IV: Optimization and Nonlinear Equations.

- [47] C. M. Roithmayr, H. Shen, M. Jesick, and D. M. Cornelius. Catching a Rolling Stone : Dynamics and Control of a Spacecraft and an Asteroid. 2013.
- [48] J. P. Sanchez and C. R. McInnes. Assessment on the feasibility of future shepherding of asteroid resources. Acta Astronautica, 73:49–66, 2012.
- [49] G. Saporta. Probabilités analyse des données et statistique. Paris : Technip, 2011.
- [50] S. Sastry. Nonlinear systems: analysis, stability and control, volume 10 of Interdisciplinary applied mathematics. Springer, 2013.
- [51] D. Scheeres. Dynamics about uniformly rotating triaxial ellipsoids: Applications to asteroids. *Icarus*, 110(2):225–238, 1994.
- [52] D. Scheeres. The dynamical evolution of uniformly rotating asteroids subject to yorp. *Icarus (New York, N.Y. 1962)*, 188(2):430–450, 2007.
- [53] D. Scheeres. Orbital mechanics about small bodies. Acta Astronautica, 72:1–14, 2012.
- [54] C. Schmid and L. Biegler. Quadratic programming methods for reduced hessian sqp. Computers and Chemical Engineering, 18(9):817–832, 1994. An International Journal of Computer Applications in Chemical Engineering.
- [55] E. Scott. Constraints on jupiter's age and formation mechanism and the nebula lifetime from chondrites and asteroids. 03 2006.
- [56] M. Shan, J. Guo, and E. Gill. Progress in Aerospace Sciences Review and comparison of active space debris capturing and removal methods. *Progress in Aerospace Sciences*, 80:18–32, 2016.
- [57] C. E. Shannon. Communication in the Presence of Noise. Proceedings of the IEEE, 72:1192–1201, 1984.
- [58] H. Shen and C. M. Roithmayr. Co-spin with Symmetry Axis Stabilization , and De-spin for Asteroid Capture. pages 1599–1604, 2014.

- [59] B. Siciliano and O. Kahtib. Springer Handbook of Robotics. Springer: Berlin. 2nd ed., 2016.
- [60] M. J. Sidi. Spacecraft Dynamics and Control. Cambridge University Press, 2000.
- [61] A. Sommariva. Rationale, Strategies, and Economics for Exploration and Mining of Asteroids. Astropolitics, 13(1):25–42, 2015.
- [62] J. J. Souchay. Dynamics of Small Solar System Bodies and Exoplanets edited by Jean J. Souchay, Rudolf Dvorak. LNP (Series). Springer Berlin Heidelberg, Berlin, Heidelberg, 1st ed. 2010. edition, 2010.
- [63] K. Stoy, D. Brandt, and D. Christensen. Self-Reconfigurable Robot, An Introduction. The MIT Press, 2010.
- [64] W. T. Thomson. Introduction to Space dynamics. Dover Publications Inc., 1986.
- [65] S. Thrun, W. Burgard, and D. Fox. Probabilistic Robotics. MIT Press: Cambridge, Mass., 2006.
- [66] J. Tokis and V. Geroyannis. Rotational Dynamics of a Deformable Medium. Astrophysics and Space Science, 51:409–427, 1977.
- [67] B. D. Warner, A. W. Harris, and P. Pravec. The asteroid lightcurve database. *Icarus*, 202(1):134–146, July 2009.
- [68] S. Wiggins. Introduction to Applied Nonlinear Dynamical Systems and Chaos. Springer: London., 2003.