

# **ELECTRODYNAMIC TETHER SATELLITE DYNAMICS MODELLING**

**by**

**KAIWEN CHEN**

A Thesis Submitted to  
The Hong Kong University of Science and Technology  
in Partial Fulfillment of the Requirements for  
the Degree of Master of Science  
in Aeronautical Engineering

May 2018, Hong Kong

## **Authorization**

I hereby declare that I am the sole author of the thesis.

I authorize the Hong Kong University of Science and Technology to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize the Hong Kong University of Science and Technology to reproduce the thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

---

KAIWEN CHEN

# **ELECTRODYNAMIC TETHER SATELLITE DYNAMICS MODELLING**

by

**KAIWEN CHEN**

This is to certify that I have examined the above M.Sc. thesis  
and have found that it is complete and satisfactory in all respects,  
and that any and all revisions required by  
the thesis examination committee have been made.

---

Prof. Xun HUANG, Thesis Supervisor

Department of Mechanical and Aerospace Engineering

29 May 2018

## ACKNOWLEDGMENTS

I would never have completed this work without the help from many people. First of all, I thank my advisor, Professor Xun Huang, for his kind mentoring, helpful advices, and encouragement. I have got generous material support, well-condition laboratory experiences and engineering-oriented practice.

I thank the members of my thesis committee, Professor Xin Zhang for his insightful comments on improving this work.

I thank my colleagues in HKUST – Huanxian Bu, Zheng Liu, Wei Feng, Jiafeng Wu, Jingwen Guo, Jiaqi Song and many others. We have finished several deadlines and projects as a team. In daily life, we have been good friends and enjoy many activities in school campus. Without them, my graduate study in HKUST would not be so colorful.

I give my special thanks to Dr.Pok Wang KWAN who spent nearly a whole year instructing me on how to do research. He gave the project a lot of remarkable ideas using his sufficient engineering background and rich industry experience. When everything goes more and more complicated, his clear mind and intuition always work quite well. I learned how to proceed an engineering research project and how to analyze problems logically and incrementally from his face to face teaching.

I thank Github user Cheedoong and wenbinf for their contribution and sharing of the HKUST Latex thesis Template. This thesis is writing in Latex based on that template-ustthesis Latex class. It saves me a lot of time to do typesetting and let me focus on the contents.

I thank my parents for their support and encouragement.

Hope this thesis will not be the end of my academia, I hope the first half sentence is not just a hope.

# TABLE OF CONTENTS

<b>Title Page</b>	i
<b>Authorization Page</b>	ii
<b>Signature Page</b>	iii
<b>Acknowledgments</b>	iv
<b>Table of Contents</b>	v
<b>List of Figures</b>	vii
<b>List of Tables</b>	viii
<b>Abstract</b>	ix
<b>Chapter 1 Motivation</b>	1
<b>Chapter 2 Literature Review</b>	6
2.1 Space tether past missions	6
2.1.1 General tether-related mission	6
2.1.2 CubeSat missions with space tether	10
2.2 Design and construction	11
2.2.1 Space tether deployment/retrieval system	11
2.2.2 Space tether dynamical modelling	15
<b>Chapter 3 Case study and MATLAB simulation implementation</b>	17
3.1 Physical Model Assumption	18
3.2 Motion governing equation and Physics model for the system	20
3.2.1 Motion governing equation	20
3.2.2 Physics model for the system	21
3.2.3 Force and Torque analysis	21
3.2.4 Tether length, tether tension and tether torque	23

3.3 Control goal of Chaser vehicle	24
3.4 MATLAB Implementation	25
3.4.1 Thrust stabilization	25
<b>Chapter 4 Experiment Design</b>	<b>28</b>
4.1 ArduSat	28
4.1.1 Arduino Data Acquisition	29
4.1.2 Xbee modules configuration	31
4.1.3 Real-time data streaming to MATLAB via USB	33
4.2 Testbed fabrication	35
<b>Chapter 5 Appendix</b>	<b>39</b>
5.1 MATLAB Thrust Stabilization-Single tether implementation	39
5.2 Arduino CSV Streaming Example	47
5.3 MATLAB Serial REALTIME PLOTING	48
<b>References</b>	<b>52</b>

# LIST OF FIGURES

1.1 Catalogued space debris over time	2
1.2 Conut evolution by object type	3
1.3 Tracked space debris in 1963 and fifty years later	4
2.1 Tether mission history	12
2.2 Summary of tether deployment and retrieval	13
3.1 Reference frame and vector definition for planar dynamics modeling	17
3.2 Definition of the Inertial frame, Chaser body frame and Target body coordinate systems	19
3.3 System model	21
3.4 Stiffness stetch relation	21
3.5 Flowchart of the calculation process	22
3.6 Positive tether target angle	25
3.7 Comparison of Target tether angle. Left: Kaiwen Right: Hovell	26
3.8 Comparison of Target angular rate. Left: Kaiwen Right: Hovell	26
3.9 Comparison of Total angular momentum. Left: Kaiwen Right: Hovell	26
3.10 Comparison of Total angular momentum. Left: Kaiwen Right: Hovell	27
3.11 Track of Chaser	27
3.12 Track of Target	27
4.1 ArduSat DemoSat	28
4.2 LSM303 Triple-axis Accelerometer plus Magnetometer (Compass) Board	30
4.3 Xbee part with Shield	31
4.4 Xbee-networks example	33
4.5 MATLAB plot real-time streaming figure	34
4.6 Air Bushing Planar Torque Free Motion simulator	35
4.7 Air Bushing Planar Torque Free Motion simulator-Three View Drawing	36
4.8 Exerted force Right View	36
4.9 Exerted force Front View	36
4.10 Solidworks Motion Analysis	37
4.11 Compensation configuration conception	38
4.12 Hanging configuration	38

## **LIST OF TABLES**

2.1	Tether Modeling Method for different problems	16
4.1	Xbee parts setting	32

# **ELECTRODYNAMIC TETHER SATELLITE DYNAMICS MODELLING**

by

**KAIWEN CHEN**

Department of Mechanical and Aerospace Engineering

The Hong Kong University of Science and Technology

## **ABSTRACT**

More than 500,000 pieces of debris, or “space junk” are tracked as they orbit the Earth. They all travel at speeds fast enough for a relatively small piece of orbital debris to damage a satellite or a spacecraft. The rising population of space debris increases the potential danger to all space vehicles, but especially to the International Space Station, space shuttles and other spacecraft with humans aboard. Tether satellite system, designed as a space debris sweeper or scavenger, will play a more and more important role in the aerospace industry.

In this thesis lots of tether missions and tether satellite dynamic modeling methods are analyzed and compared. A detail and self-contained tether satellite system dynamics model and its MATLAB implementation are given. The data communication of test sample- Arduino DemoSat integrated physical sensors and data plotting/collection using MATLAB serial communication are achieved. Also the design of air bushing planar torque free Motion testbed is fabricated and refined.

# CHAPTER 1

## MOTIVATION

Satellites in orbit around Earth are used in many areas and disciplines, including space science, Earth observation, meteorology, climate research, telecommunication, navigation and human space exploration. They offer a unique resource for collecting scientific data, commercial opportunities and various essential applications and services, which lead to unrivalled possibilities for research and exploitation.

However, in the past decades, with increasing space activities, a new and unexpected hazard has started to emerge: space debris.

The Figure 1.1 shows the debris changing trend over nearly 60 years. The y direction shows the number of objects. Each point represents the monthly number of objects in Earth Orbit. The x direction is time axis. Space debris is divided into four types, each type has its own color:

1. Fragmentation Debris
2. Spacecraft
3. Mission-related Debris
4. Rocket Bodies

The orbit increasing of Spacecraft and Rocket Bodies are quite linearly. Mission-related Debris is quite stable. The most dramatic changing type is Fragmentation debris and this type of debris is directly related to satellite explosion and artificial celestial body collision.

From another statistical result , we can see this trend much more clearly. In almost 60 years of space activities, more than 5250 launches have resulted in some 42 000 tracked objects in orbit, of which about 23000 remain in space and are regularly tracked by the US

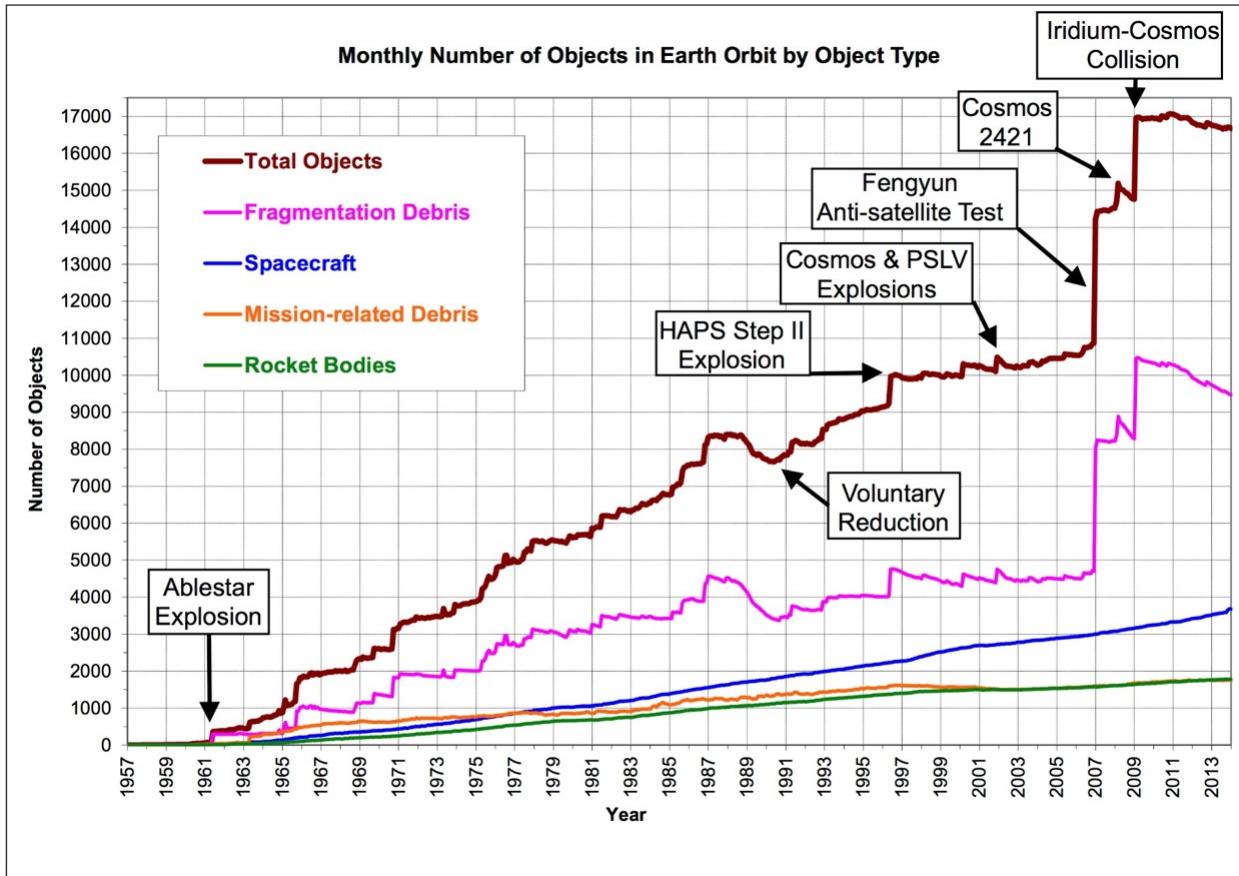


Figure 1.1. Catalogued space debris over time  
[32]

Space Surveillance Network and maintained in their catalog, which covers objects larger than about 5-10 cm in low-Earth orbit (LEO) and 30 cm to 1 m at geostationary (GEO) altitudes. Only a small fraction, about 1200, are intact, operational satellites today.

This large amount of space hardware has a total mass of more than 7500 tonnes.

### 1. Spent upper stages

About 24% of the catalogued objects are satellites (less than a third of which are operational), and about 18% are spent upper stages and mission related objects such as launch adapters and lens covers.

More than 290 in-orbit fragmentation events have been recorded since 1961. Only a few were collisions and the majority of the events were explosions of spacecraft and upper stages.

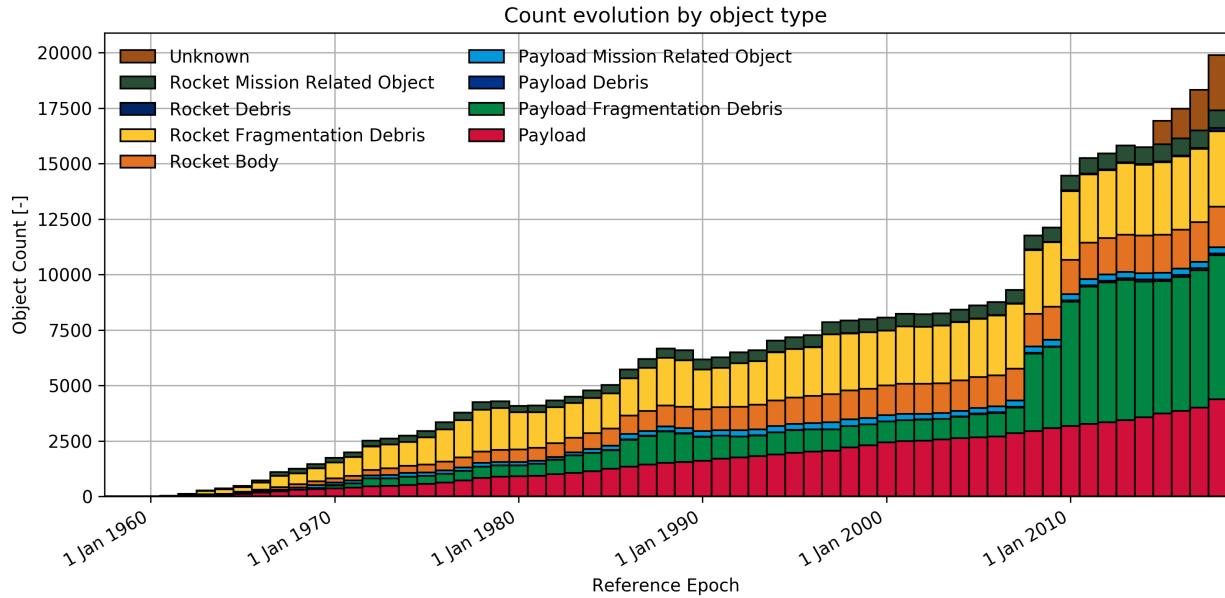


Figure 1.2. Conut evolution by object type

## 2. Explosions of satellites and rocket bodies

These fragmentation events are assumed to have generated a population of objects larger than 1 cm numbering on the order of 750000. The sporadic flux from naturally occurring meteoroids may only prevail over that from human-made debris objects near sizes of 0.1-1 mm.

The main cause of in-orbit explosions is related to residual fuel that remains in tanks or fuel lines, or other remaining energy sources, that remain on board once a rocket stage or satellite has been discarded in Earth orbit.

Over time, the harsh space environment can reduce the mechanical integrity of external and internal parts, leading to leaks and/or mixing of fuel components, which could trigger self-ignition. The resulting explosion can destroy the object and spread its mass across numerous fragments with a wide spectrum of masses and imparted velocities.

## 3. Antisatellite test

Besides such accidental break-ups, satellite interceptions by surface-launched missiles have been a major contributor in the recent past.

The Chinese FengYun-1C engagement in January 2007 alone increased the trackable

space object population by 25%.

#### 4. Other sources

Solid rocket-motor firings, the ejection of reactor cores from Buk reactors after the end of operation of Russian radar ocean reconnaissance satellites in the 1980s and the release of thin copper wires as part of a radio communication experiment during the Midas missions in the 1960s are the three most important sources.

The nightmare scenario that space debris experts contemplate is called the Kessler syndrome, after American astrophysicist Donald Kessler. In 1978, while working for NASA, he published an analysis[24] that showed frequent collisions exponentially increased the amount of space debris, leading to many more collisions, leading to much more debris until we lose the use of certain orbits because anything we put there would certainly be hit.

*"Since 2002, The growth has entered into the more feared exponential trend."*

There can be absolutely no doubt that the time to do something about space debris has arrived.

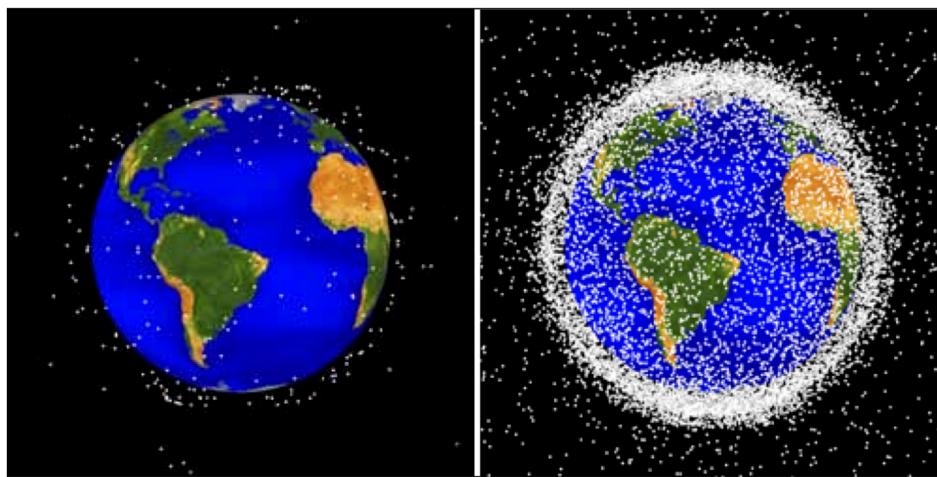


Figure 1.3. Tracked space debris in 1963 and fifty years later

The Earth is now surrounded by tens of thousands of sizeable chunks of debris orbiting at speeds upwards of 17,000 miles per hour. Add to these larger pieces an estimated

hundreds of thousands of sub-centimeter-sized artificial particles, and you have got a recipe for potential disaster-for while these small particles may not seem imposing, they can present a serious threat to functional spacecraft.

Active Debris Removal (ADR) is necessary to stabilize the growth of space debris, but even more important is that any newly launched objects comply with post-mission disposal guidelines especially orbital decay in less than 25 years. If this were not the case, most of the required ADR effort would go to compensate for the non-compliance of new objects.

Some actively removing space debris ideas or concepts has been proposed by the researchers and institutions around the world. Japan Aerospace Exploration Agency (JAXA), European Space Agency and Texas A&M University each presents its own plan.

## CHAPTER 2

### LITERATURE REVIEW

#### 2.1 Space tether past missions

This chapter is the literature survey[16, 15] on the following topics:

- Space tether past missions
- CubeSat missions with space tether
- Design and construction of:
  1. Space tether deployment/retrieval system
  2. Plasma contactor

It tried to organize the past missions regarding to the space tether and the CubeSat mission in a much clear order. In the same time some historical researches about deployment/retrieval system are listed. The main point is focus on dynamics and control method of tether satellite system.

##### 2.1.1 General tether-related mission

With the development of space technology, a series of missions[9, 31, 5, 6, 22, 7, 25] have been delivered for aerospace application using tethered satellite systems over the past forty years. Many proposals were implemented including scientific experiments in the microgravity environment, upper atmospheric research, the generation of electricity, cargo transfer between orbiting bodies, collections of planetary dust, and the expansion of the geostationary orbit resource by tethered chain satellites.

## 1. NASA

NASA has been developing tether technology for space applications since the 1960s, including electrodynamic tether propulsion, the Propulsive Small Expendable Deployer System (ProSEDS) flight experiment, "Hanging" momentum exchange tethers, rotating momentum exchange tethers and tethers supporting scientific space research.

The Gemini XI was a manned spaceflight in NASA's Gemini program, launched on September 12, 1966. One of the main objectives was related to tethers. It was synoptic terrain photography and a tethered vehicle test. The objectives were all achieved.

The Gemini XII was a manned spaceflight in NASA's Gemini program launched on November 11, 1966. One of its major objectives was tethered vehicle evaluation. The objective was achieved.

The tether length of MAIMIK was about 400m. The mission was designed to study the charging of an electron-beam emitting payload using a tethered mother-daughter payload configuration.

## 2. Tethered Payload Experiment (TPE)

TPE-1 mission was launched on January 16, 1980. Its plan was to deploy 400 metres of cable, but its deployed cable was about 38 metres. The TPE-2 mission was launched on 29 January, 1981, and its tether was deployed to a distance about 65 metres. In 1983, the TPE-3, which was also called CHARGE-1, had a length of about 500m. As the deployment system was improved, the tether deployed to its full length of 418 meters, and the tether was also found to act as a radio antenna for the electrical current through the cable. CHARGE-2 was carried out as an international program between Japan and the USA using a NASA sounding rocket at White Sands Missile Range, in December 1985, its tether deployed to a length of 426 metres.

CHARGE-2B tethered rocket mission was launched in 1992 by NASA with a Black Brant V rocket. The mission was to generate electromagnetic waves by modulating the electron beam. The tether was fully deployed over 400 meters and the experiments all worked as planned.

### 3. US Air Force Geophysics Lab- Echo-7

Designed to study the artificial electron beam propagation along magnetic field lines in space, the mission planned to study how the artificial electron beam propagates along magnetic field lines in space.

### 4. OEDIPUS

OEDIPUS-A. The conducting tether was deployed over 985 metres during the flight of a Black Brant sounding rocket in to the auroral ionosphere.

OEDIPUS-C. The following OEDIPUS mission was the OEDIPUS-C tethered payload mission, which was launched in 1995 with an 1174-metre deployed tether, and a Tether Dynamics Experiment (TDE) was also included as a part of the OEDIPUS-C.

### 5. Tethered Satellite System (TSS)

The first orbital flight experiment with a long tether was the Tethered Satellite System (TSS) mission, launched on the Space Shuttle in July 1992. The Tethered Satellite System-1 (TSS-1) was flown during STS-46, aboard the Space Shuttle Atlantis, from July 31 to August 8, 1992. The TSS-1 mission discovered a lot about the dynamics of the tethered system. Although the satellite was deployed only 260 metres, it was able to show that the tether could be deployed, controlled, and retrieved, and that the TSS was easy to control, and even more stable than predicted. The TSS was an electrodynamic tether, its deployment mechanism jammed resulting in tether sever and less than 1000 metres of deployment. The objectives of TSS-1 were

- (a) to verify the performance of the TSS equipment;
- (b) to study the electromagnetic interaction between the tether and the ambient space plasma;
- (c) to investigate the dynamical forces acting on a tethered satellite. In the first tether deployment, when the satellite was moving excessively from side to side, the deployment was aborted. The second trial of deployment was unreeled to a length of 260 metres.

In 1996, the Tethered Satellite System Reflight (TSS-1R) was carried by using US space shuttle STS-75 successfully. The primary objective of STS-75 was to carry the

Tethered Satellite System Reflight (TSS-1R) into orbit and to deploy it spacewards on a conducting tether.

#### 6. Shuttle Electrodynamic Tether System (SETS)

The Shuttle Electrodynamic Tether System (SETS) experiment formed part of the scientific experiments comprising the first flight of the NASA/ASI Tethered-Satellite System flown at an altitude of 300 km and at an orbital inclination of 28.5 degrees in July/August 1992. The SETS experiment was designed to study the electrodynamic behaviour of the Orbiter-Tether-Satellite system, as well as to provide background measurements of the ionospheric environment near the Orbiter. The SETS experiment was able to operate continuously during the mission thereby providing a large data set.

#### 7. Tether Physics and Survivability Experiment (TiPS)

The Tether Physics and Survivability Experiment (TiPS) was deployed on June 20, 1996 at an altitude of 1,022 kilometres as a project of the US Naval Research Laboratory. The satellite was a tether physics experiment consisting of two end masses connected by a 4 km nonconducting tether. This experiment was designed to increase knowledge about gravity-gradient tether dynamics and the survivability of tethers in space.

#### 8. The Advanced Tether Experiment (ATEx)

The Advanced Tether Experiment (ATEx) was launched into orbit aboard the National Reconnaissance Office (NRO) sponsored by Space Technology Experiment spacecraft (STEX) on October 3, 1998. ATEx was intended to demonstrate the deployment and survivability of a novel tether design, as well as being used for controlled libration maneuvers. On January 16, 1999, after a deployment of only 22m of tether, ATEx was jettisoned from STEX due to an out-of-limit condition sent by the experiment's tether angle sensor. The ATEx lower end mass was jettisoned from the host spacecraft and the tethered upper and lower end masses freely orbited the Earth in a demonstration of long-term tether survivability.

#### 9. The PICOSAT mission

The PICOSAT mission was launched on September 30, 2001. It was a real-time tracking satellite of the miniaturized picosatellite satellite series. The name "PICO" combined the first letters of all the four of its experiments, which were the Polymer Battery Experiment (PBEX), the Ionospheric Occultation Experiment (IOX), the Coherent Electromagnetic Radio Tomography (CERTO), and the On Orbit Mission Control (OOMC). A pair of 0.25kg MEMS picosatellites with an intersatellite communications experiment were included in this mission and were connected by a 30 metre tether.

#### 10. Multi-Application Survivable Tether

The Multi-Application Survivable Tether (MAST) experiment was launched into LEO on April 17, 2007, in which the 1 km multistrand interconnected tether (Hoytether) was intended to test and prove the long-term survivability of tethers in space, but the tether failed to deploy.

#### 11. Tether Technologies Rocket Experiment

Tether Technologies Rocket Experiment (T-REX) mission was launched August 31, 2010, on sounding rocket S-520-25, reaching a maximum altitude of 309 km. T-Rex was developed by the Kanagawa Institute of Technology/Nihon University, which led an international team to test a new type of electrodynamic tether (EDT) that may lead to a generation of propellantless propulsion systems for LEO spacecraft. The tether was 300 metres long and deployed as planned, a video of deployment was transmitted to the ground. Tether deployment was verified successfully, as was the fast ignition of a hollow cathode in the space environment.

### 2.1.2 CubeSat missions with space tether

#### 1. Micro Electro-Mechanical Systems based PicoSat Inspector

The MEPSI series (Micro Electro-Mechanical Systems based PicoSat Inspector) was a pair of tethered picosatellites, based on the CubeSat design, launched by a custom deployer aboard the STS-113 Endeavour mission on December 2, 2002. The two spacecrafts were cubic in shape, of mass 1 kg each, and were connected via a 15.2m tether in order to facilitate detection and tracking via ground-based radar.

## **2. The Space Tethered Autonomous Robotic Satellite**

On January 23, 2009, a space tether mission called "The Space Tethered Autonomous Robotic Satellite (STARS)," developed by Kagawa Satellite Development Project at Kagawa University, was launched as a secondary payload aboard H-IIA flight. The satellite was named KUKAI after launch and based on a "cubesat" platform like MAST, and consisted of two subsatellites, "Ku" and "Kai," which are linked by a 5-meter tether. The separation of the rocket and satellite and the transfer into the planned orbit were successful, but the tether only deployed to a length of several centimeters because of the launch lock trouble of the tether reel mechanism .

## **3. Tether Electrodynamic Propulsion CubeSat Experiment**

Tether Electrodynamic Propulsion CubeSat Experiment (TEPCE) mission, planned by Naval Research Laboratory, is an electrodynamic tether experiment based on a "triple cubesat" configuration. This experiment is currently planned for launch as a secondary payload in September 2013. Two nearly identical endmasses with a stacer spring between them are used in TEPCE, which separate the endmass and start deployment of a 1 km long braided-tape conducting tether. TEPCE will use a passive braking to reduce speed and hence recoil at the end of electrodynamic current in either direction. The main purpose of this mission is to raise or lower the orbit by several kilometres per day, to change libration state, to change orbit plane, and to actively maneuver.

## **2.2 Design and construction**

### **2.2.1 Space tether deployment/retrieval system**

Besides other operational phases, a TSS mission launch always involves both deployment and retrieval. Tether retrieval is the opposite of deployment and is equally important in dynamical terms. Tether deployment and retrieval are two of the most important steps in space tether applications. The following contents introduce some interesting things about the deployment and retrieval.

Mission	Year	Sponsor	Orbit	Length	Status
Gemini XI	1967	NASA	LEO	50 m	Launched
Gemini XII	1967	NASA	LEO	30 m	Launched
TPE-1	1980	NASA/ISAS	Suborbital	400 m	Launched
TPE-2	1981	NASA/ISAS	Suborbital	500 m	Launched
TPE-3 (CHARGE-1)	1983	NASA/ISAS	Suborbital	500 m	Launched
CHARGE-2	1985	NASA/ISAS	Suborbital	500 m	Launched
MAIMIK	1986	NASA/NDRE	LEO	400 m	Launched
ECHO-7	1988	USAF	Suborbital	—	Launched
OEDIPUS-A	1989	NRC/NASA/CRC/CSA	Suborbital	958 m	Launched
CHARGE-2B	1992	NASA/ISAS	Suborbital	500 m	Launched
TSS-1 (STS-46)	1992	NASA/ASI	LEO	267 m	Launched
SEDS-1	1993	NASA	LEO	20 m	Launched
PMG	1993	NASA	LEO	500 m	Launched
SEDS-2	1994	NASA	LEO	20 m	Launched
OEDIPUS-C	1995	NASA/NRC/CRC/CSA	Suborbital	1 km	Launched
TSS-1R (STS-75)	1996	NASA/ASI	LEO	19.6 km	Launched
TSS-2 (STS-75)	1996	NASA	LEO	100 m	Cancelled
TiPS	1996	NRO/NRL	LEO	4 km	Launched
YES	1997	ESA/Delta-Utec	LEO	35 m	Launched
ATEx	1999	NRO/NRL	LEO	22 m	Launched
PICOSATs	2000	Aerospace corporation	LEO	30 m	Launched
MEPSI	2002	Aerospace corporation	LEO	15.2 m	Launched
ProSEDS	2003	NASA	LEO	15 m	Cancelled
MAST	2007	NASA/TUI/Stanford	LEO	1 km	Launched
YES2	2007	ESA/Delta-Utec	LEO	31.7 m	Launched
STARS	2009	Kagawa university	LEO	5 m	Launched
T-REX	2010	ISAS/JAXA	LEO	300 m	Launched
TEPCE	2013	NRO/NRL	LEO	1 km	Planning

Figure 2.1. Tether mission history

## 1. ATEx design iterations

In 1997, Koss described the tether deployment mechanism development for the Advanced Tether Experiment (ATEx) mission with some design iterations, which included: energetic(spring) deployment, a DC brush motor-driven deployment mechanism and a stepper motor-driven deployment mechanism

## 2. Fuzzy-logic feedback deployment control

A tethered system deployment control by fuzzy-logic feedback was proposed by Licata in 1997, in which the feedback control was based on a simple fuzzy-logic rule and minimal measurements for in-plane tether deployment control problems. The fuzzy-logic law, for the proposed rate feedback control solution to the in-plane deployed tether terminal position-only problem has been associated with the tether length-angle state plane, or the physical deployment trajectory plane.

Year	Modelling	Problems
1979	Tether-connected two-body systems	General formulation of deployment dynamics considering three-dimensional librational motion, and longitudinal and transverse vibrations
1994	TSS-1	Analysis of the spectra
1995	Non-linear model for tether retrieval	Amplitude of oscillation increasing during retrieval process
1995	Exponential deployment	Dispense with tether alignment without later libration
1997	ATEx	Tether deployment mechanism development
1997	In-plane tether deployment	Fuzzy-logic rule and minimal measurements for deployment control
1998	GATE	Simulation of the retrieval and deployment characteristics
1998	Three-dimensional tether	Deployment considering both in-plane and out-of plane libration
1999	MACE	The close-loop results and insights from the on-orbit control experiments
1999, 2002	Tether-connected satellite systems	Dynamical behaviour during deployment and retrieval process
2003, 2005	Tethered satellites	Various methods for deployment control
2003, 2005	Tethered subsatellite	Optimal control strategy of deployment
2004	Tethered re-entry capsule	Adaptive neural control concept for deployment
2006	3DOF tethered subsatellite system	Optimal control of deployment and retrieval process
2006, 2008	Tethered formation with spinning	Optimal deployment and retrieval
2007	Deployment of space systems	Coupled with long and short tethers
2014	Multiple mass tether	Thrust forces and periods of thruster activation were estimated from various on-ground experiments

Figure 2.2. Summary of tether deployment and retrieval

### 3. Using Active Reel Type Deployer to retrieve and deploy tether

Carter and Greene studied the simulation of the retrieval and deployment characteristics for the Getaway Tether Experiment (GATE) in 1998. The GATE was a single tether satellite system for the study of tether dynamics and electrodynamic technology. One goal of GATE was to measure and control tether disturbances by micro meteorite impact using an active reel type deployer, which was able to retrieve and deploy the tether. A closed-loop controller for the tethered system was given, which allowed the tether to be actively reeled in (retrieval) or passively reeled out (deploy-

ment), to and from the mother subsatellite.

#### **4. Strategies for TSS retrieval-tension control in the tether**

The dynamical behaviour of tether-connected satellite systems during the deployment and retrieval process was studied by Djebli et al. in 1999 and 2002. The system consisted of a space station connected to a subsatellite by means of a tether of variable length. A simplified model was given in which the space station and the subsatellite were reduced to material points and the system mass centre moved along a circular orbit with three-dimensional transverse and longitudinal oscillations. Strategies for retrieval were obtained in order to increase the tension in the tether at the final stage of retrieval. These laws of retrieval were deduced from the laws obtained in a previous paper for the particular case of a massless tether. Retrieval of a subsatellite to a larger vehicle was examined by Djebli et al. in 2002, which concentrated on laws for retrieval and deployment, combining 'simple' linear and 'fast' laws. This would be applicable to passive momentum exchange tethers and potentially to ED tethers.

In 2006 and 2008, Williams published his works on the optimal deployment and retrieval for a tethered formation with spinning in the orbital plane, in which a tethered formation was modelled by point mass satellites and connected via inelastic tethers. The optimal deployment and retrieval trajectories using tension control were developed for different spinning conditions.

#### **5. Viscoelastic space billiard model**

In 2003 and 2005, Barkow et al. published three papers on various methods used for controlling the deployment of tethered satellites. The deployment of a tethered satellite system is one of the most critical phases in a tether mission. Based on a viscoelastic space billiard model, a targeting strategy was developed which made use of the system's chaotic nature and allowed the system to be steered into its equilibrium faster and more efficiently, when compared to conventional strategies.

#### **6. Optimal control**

In 2003, Steindl and Troger proposed their works that the optimal control of deployment of a tethered subsatellite moving on a circular orbit around the Earth, in which

an optimal control of deployment and retrieval of a tethered subsatellite from a main satellite was treated.

Jin et al. presented an optimal control of the deployment and retrieval processes of a three degree of freedom (DOF) tethered subsatellite system using truncated Cheby-shev series to approximate the state variables.

## 7. Adaptive neural control

An adaptive neural control concept for the deployment of a tethered re-entry capsule was developed by Glabell et al. in 2004, in which the indirect neural controller combined two neural networks, a controller network and a plant model network. After introducing the tether deployment scenario, assumptions and simplifications were applied to the mathematical system model. The simulation results allowed a performance comparison of the linear quadratic regulator and the neural control concept.

## 8. Tether length factor

In 2007, Mantri's research aimed to model and understand the deployment of space systems with long and short tethers, which was divided into two parts. The models are set for different length tether systems

## 9. Spool-type reel using thruster

In 2014, Iki et al. investigated an ED tether deployment from a spool-type reel using thrusters with a length of several kilometers, in which the deployment of multiple mass tether deployment was studied the key parameters of a multiple mass tether model. The key parameters of the thrust forces and periods of thruster activation were estimated from various on-ground experiments.

### 2.2.2 Space tether dynamical modelling

Different models are used for different purposes. In order to solve different problems, several dynamical models have been proposed. Here are the recent proposed models, from 2001 to 2014.[8, 35, 12, 10, 13, 14, 11, 21, 29, 28, 18, 4, 33, 26, 34, 23]

According to the problem, different level models are proposed. From simple to complex, Non-dissipative massive Tether, Electrodynamic Tether(EDT), Tether Satellite System(TSS), Advanced Safety Tether Operation and Reliability(ASTOR)and Motorized Momentum Exchange Tether(MMET) are the proposed tether types. According to the problem, researcher can choose the model like slack-sprint model, dumbbell model, flexible model, ponderable tether model, two mass point with the rigid rod model and so on.

Table 2.1. Tether Modeling Method for different problems

<b>Author</b>	<b>Year</b>	<b>Modelling Method</b>	<b>What problem to solve</b>
Cartmell, Ziegler	2001	Scale model of MMET	
Chen and Cartmell, Chen and Ismail	2001-2012	MMET series models	Dynamical modelling and control
Mazzoleni and Hoffman	2001	Tethered artificial gravity satellite	Non-planar spin-up dynamics
Ellis and Hall	2010	Two Point masses connected by rigid rod	Out-of-plane librations
Aslanov	2010	Rigid body TSS system	Plane translational motion and rotational motion
Zanutto et al.	2011	Electrodynamic tether	Perturbed classical three-body problem
Kristiansen et al.	2012	Non-dissipative massive tether	Orbital motions
Zhao et al.	2012	Dumb-bell TSS	Analytical first-order solutions of the librational angles
Jung et al.	2014	TSS with a moving mass	Libration angle with respect to the system parameters

## CHAPTER 3

### CASE STUDY AND MATLAB SIMULATION IMPLEMENTATION

In this chapter, an specific example[20] of tether satellite system is studied. In this case study, the used dynamic model and Proportional-Differentiation controller are the main analysis focuses. At the end, MATLAB simulation is also implemented.

The tether satellite system consists of two satellite and a tether. The two satellites are named as Target Vehicle and Chaser vehicle which means the task of Chaser vehicle is to capture the target vehicle and to de-rotation of the system. In the real case the target vehicle can be space debris which is moving and rotation around Earth under the gravitational force of the Earth. Space debris not only rotates around the Earth, in fact it will also rotate around its body center if any .De-orbiting can be achieved if the energy and angular momentum of space debris consumed. When we let our chaser vehicle connect

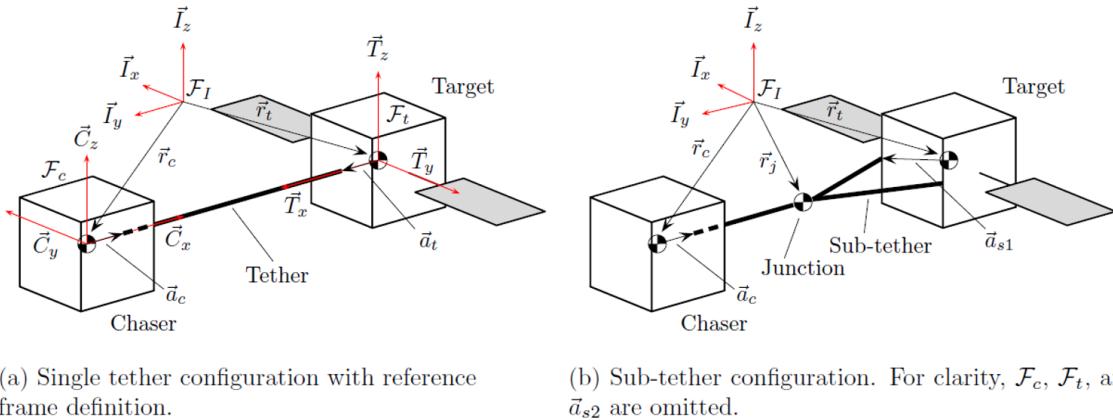


Figure 3.1. Reference frame and vector definition for planar dynamics modeling

with target vehicle using tether, the consumption of angular momentum becomes faster and the deorbiting gets quicker. So we can say that chaser vehicle plays as a role of active debris remover.

Using different kind of tether configuration, the effect of deorbiting target vehicle can be quite different. In Hovell's article, single tether configuration and sub-tether configuration are compared. The sub-tether configuration is like harpoon. The main tether links the Chaser vehicle with the junction point, which will be modeled as a point mass later. Two separate tethers link the junction and Target vehicle. The proposed sub-tether configuration can enhance the deorbiting mechanism, reducing the target tether angle and angular rate of target and finally dissipating angular momentum from the system much more quickly.

### 3.1 Physical Model Assumption

1. Massless tether, Constant end-body mass and visco-elastic tether

The tether is assumed as massless since it is much more lighter than end-body satellites. The single tether and sub-tether are modeled as nonlinear spring. The spring force are related to tether stretch and the relationship is set as visco-elastic.

2. Three degree of freedom for Chaser vehicle and Target vehicle

Only three degree of freedom is considered in this case which means the satellite can only do translational motion along X axis and Y axis and rotational motion about the Z axis. This assumption makes the dynamics modeling of satellite much easier. The two satellites, target and chaser, are at the same level height which means there is no translational motion along Z axis i.e the  $z = 0$ . Similarly, the rotational motion about X axis and the rotational motion about Y axis are not con

3. No external forces applied to the center of mass of the whole tethered satellite system

In the tether satellite system, the force of tether and the chaser's control force are force source. No external force, such as gravitational force and disturbance force applied to the center of mass of the whole system.

4. The Inertia matrix is constant

5. Body frame of each of the end-masses is aligned with the principal axes of the end-masses

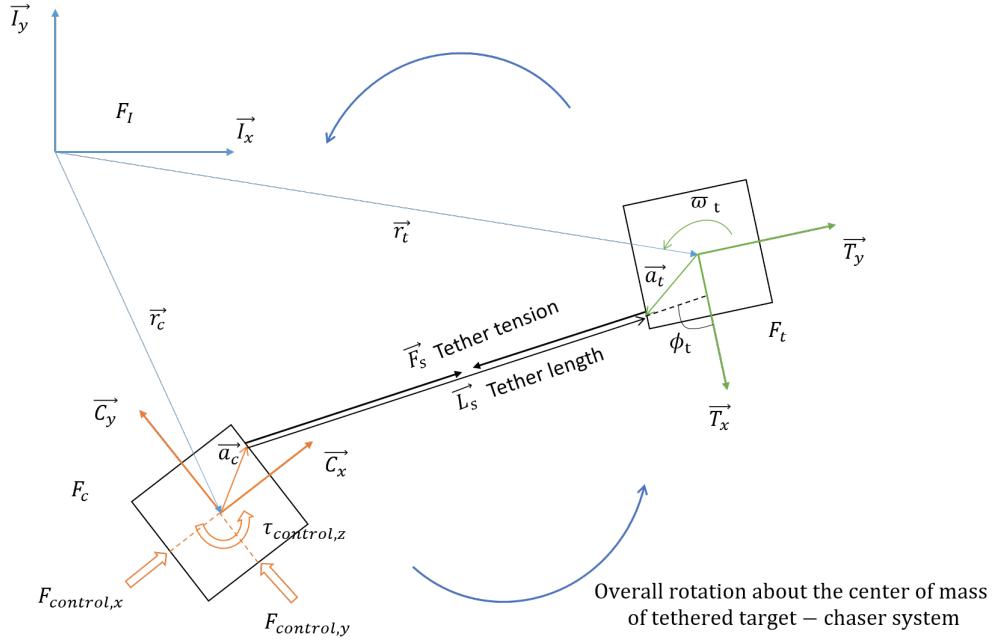


Figure 3.2. Definition of the Inertial frame, Chaser body frame and Target body coordinate systems

There are three coordinate systems defined in the article and all of them are defined as right-handed system. Between two different coordinate systems the attitude matrix  $A(\theta)$  is the transformation matrix which used to rotate the vector components from a body frame to Inertial frame.

$$A(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \quad (3.1)$$

The point where tether attached the Chaser vehicle or Target vehicle is called attached point. This point is fixed in the body frame so that we can use tether attachment vector with respect to the body center of mass to describe this attach point. The tether attachment vector is denoted as  $\alpha$ . The attitude  $\theta$  of a body frame ( $F_C$  for the chaser or  $F_T$  for the target) is the angle of rotation between the body frame relative to the Inertial frame  $F_I$  about the z-axis where the z-axis of all inertial and body frames are parallel. The transformation of vectors from body frame to Inertial frame is using the equation  $x_I = A(\theta)x_b$  which I means Inertial frame and b means body frame.

### 1. Reference inertial frame $F_I$ Coordinates

$$I = [I_x, I_y, I_z]$$

$F_I$  is fixed. The position of origin point and direction of three axes will not change with time.

### 2. Reference inertial frame $F_T$ Coordinates

$$T = [T_x, T_y, T_z]$$

$F_T$  denotes the Target body-fixed frame. The origin point of  $T$  is at the center of mass of the Target vehicle. The x axis of  $F_T$  is pointed to the chaser at the beginning.

### 3. Reference inertial frame $F_C$ Coordinates

$$C = [C_x, C_y, C_z]$$

$F_C$  denotes the Chaser body-fixed frame. The origin point of  $C$  is at the center of mass of the Target. The x axis of  $F_C$  is pointed to the target at the beginning.

## 3.2 Motion governing equation and Physics model for the system

This section talks about the details of physical model and the motion governing equation

### 3.2.1 Motion governing equation

The governing equations are Newton's second law and Euler's equation.

Using the Newton's second law we can get that the sum of force vector components acting on any body in Inertia Frame,  $F$ , equals the mass of the satellite  $m$  times the resulting acceleration vector components in Inertia Frame  $\ddot{r}$  i.e.  $F = m\ddot{r}$

Similarly, using the Euler's equation we can get that the torque of the chaser's body or target's body as a result of the force exerted by the tether and/or the chaser's control actuator equals z axis Principal moment of inertia multiplied by Angular velocity of an end-mass/satellite relative to the inertial frame (which also equals to the angular velocity relative the body frame) i.e.  $J_{zz}\dot{\omega} = \tau_z$

### 3.2.2 Physics model for the system

The whole system is modeled as a nonlinear stiffness spring-damper system. The damping coefficient of the damper is constant. The tether is made of 56% polyester and 44% rubber. The spring constant  $k$  has strong nonlinearity.

- The spring constant changes as the extension of tether change
- When tether is extended, the spring force  $F_{\text{spring}} = k \times (\text{length of stretched tether} - \text{tether natural length})$
- $F_{\text{spring}} = 0$  when tether is slacked

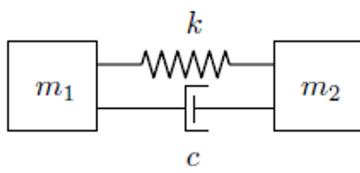


Figure 3.3. System model

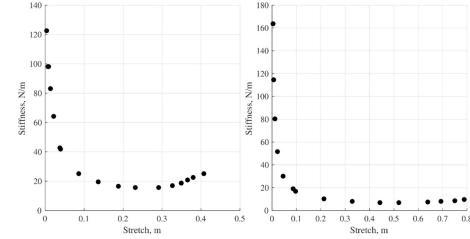


Figure 3.4. Stiffness stretch relation

### 3.2.3 Force and Torque analysis

1. For the Target vehicle, it only received the force of tether tension which denoted as  $-\vec{F}_S$ . The torque  $\tau_t$  is the torque made by tether tension about the center of mass of the target body.

$$\tau_t = a_{t,y} F_{s,x}^t - a_{t,x} F_{s,y}^t \quad (3.2)$$

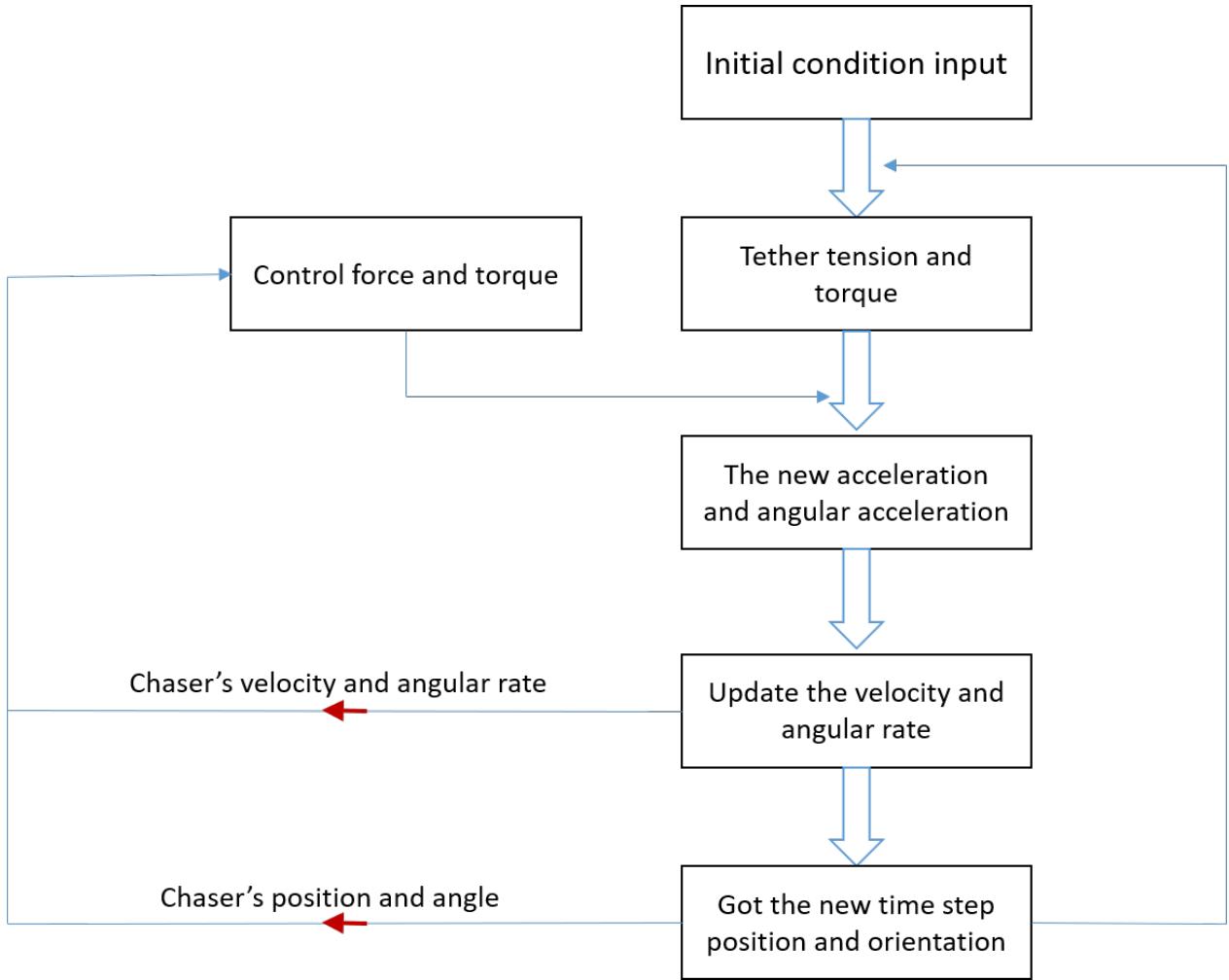


Figure 3.5. Flowchart of the calculation process

$F_s^t$  is the tether force expressed in  $F_t$  i.e.  $F_s^t = A(\theta_t)^T F_S$   
 $F_{s,x}^t, F_{s,y}^t$  are the x and y components of  $F_s^t$  in  $F_T$ , respectively.

2. For the Chaser vehicle tether force  $\vec{F}_S$  and active control force  $\vec{F}_{control}$  both exist. The torque also consists of two parts  $\tau_{control}$  and  $\tau_c$ . The  $\tau_{control}$  and  $\vec{F}_{control}$  are the active control way that research can design in order to get the better deorbiting performance.

$$\tau_c = a_{c,x} F_{s,x}^c - a_{c,y} F_{s,y}^c \quad (3.3)$$

$F_s^c$  is the tether force expressed in  $F_c$  i.e.  $F_s^c = A(\theta_c)^T F_S$   
 $F_{s,x}^c, F_{s,y}^c$  are the x and y components of  $F_s^c$  in  $F_c$ , respectively.

### 3.2.4 Tether length, tether tension and tether torque

According to Figure 3.2, changing all the vectors in the Inertial frame then the vector form of tether length can be written as follows:

$$\vec{L}_S = -(\vec{r}_c + A(\theta_c)\vec{a}_c) + (\vec{r}_t + A(\theta_t)\vec{a}_t) \quad (3.4)$$

As Figure 3.3 described before, the damping mechanism of tether results in the damping force:  $c(\Delta \vec{v} \cdot \frac{\vec{L}_S}{\|\vec{L}_S\|})$ . It results from damping factor times the velocity change in the tether's direction

The tether tension is:

$$\vec{F}_S = (k\|\vec{L}_S\| - L_{s,nature}) + c(\Delta \vec{v} \cdot \frac{\vec{L}_S}{\|\vec{L}_S\|}) \frac{\vec{L}_S}{\|\vec{L}_S\|} \quad (3.5)$$

$\frac{\vec{L}_S}{\|\vec{L}_S\|}$  is the unit vector of the tether.  $\Delta \vec{v}$  is the difference of the time derivative of two body's absolute position.

$$\Delta \vec{v} = \frac{d}{dt}(\vec{r}_t + A(\theta_t)\vec{a}_t) - \frac{d}{dt}(\vec{r}_c + A(\theta_c)\vec{a}_c) = \vec{v}_t + \frac{d}{dt}(A(\theta_t))\vec{a}_t - \vec{v}_c - \frac{d}{dt}(A(\theta_c))\vec{a}_c \quad (3.6)$$

Since the derivate is used for the angle, we can rewrite the  $\frac{d}{dt}(A(\theta_t))\vec{a}_t$  in terms of angular rate and tether attachment vector.

$$\frac{d}{dt}(A(\theta_t))\vec{a}_t = \frac{d}{dt} \begin{pmatrix} \cos(\theta_t) & -\sin(\theta_t) \\ \sin(\theta_t) & \cos(\theta_t) \end{pmatrix} \vec{a}_t = \begin{bmatrix} \cos(\theta_t) & -\sin(\theta_t) \\ \sin(\theta_t) & \cos(\theta_t) \end{bmatrix} \begin{pmatrix} -\omega_t a_{t,y} \\ \omega_t a_{t,x} \end{pmatrix} \quad (3.7)$$

which means that

$$\frac{d}{dt}(A(\theta_t))\vec{a}_t = A(\theta_t) \begin{pmatrix} -\omega_t a_{t,y} \\ \omega_t a_{t,x} \end{pmatrix} \frac{d}{dt}(A(\theta_c))\vec{a}_c = A(\theta_c) \begin{pmatrix} -\omega_c a_{c,y} \\ \omega_c a_{c,x} \end{pmatrix}$$

Tether torque can be written as the time derivative of system angular momentum. It equals the time derivative of system angular momentum with regard to the relative

moving coordinate system plus the moving frame's absolute angular velocity times the angular momentum. In this case the moving frame just has translational motion in Inertia frame. The moving frame's absolute angular velocity doesn't exist in this case.

$$M = \frac{dH}{dt} = \dot{H}_{\text{relative}} + \Omega \times H = \dot{H}_{\text{relative}} = \frac{J_{zz}\omega}{dt} = J_{zz}\dot{\omega} \quad (3.8)$$

The torque in Inertial frame equals the torque in body frame. For the rigid body  $i$  In the inertial frame, the Euler's equation gives us the relationship between the exerted torque  $\tau$  and resulted angular acceleration  $\dot{\omega}$  .i.e.

$$J_i \ddot{\omega} + \omega_i \times J_i \omega_i = \sum \tau \quad (3.9)$$

According to the assumption 3.1 we can simplified the Euler's equation as:  $J_{izz}\ddot{\omega}_{iz} = \tau_{iz}$   $i$  can be  $c$  for chaser or  $t$  for target.  $\tau = a_i \times F_s^i$   $\tau_t = a_{t,y}F_{s,x}^t - a_{t,x}F_{s,y}^t$   $\tau_c = a_{c,x}F_{s,x}^c - a_{c,y}F_{s,y}^c$

### 3.3 Control goal of Chaser vehicle

To archive the goal of deorbiting target satellite, the Chaser vehicle need to regulate and reduce the angle between tether and target so as to make the tether aligned with the target or force Chaser and target has the parallel surface. To reduce the angular rate of target vehicle, dissipate angular momentum from the system and at last make the target go to lower altitude orbit. The easiest way is to use the proportional-derivative controller for the position and altitude control of the tether. The force controller uses the difference between desired chaser radius vector and actual chaser radius vector as the feedback signal.

The force  $\overrightarrow{F_{s,\text{control}}}$  equals  $\mathbf{K}_P \left[ \overrightarrow{r_{\text{des}}(t)} - \overrightarrow{r_c(t)} \right] + \mathbf{K}_D \left[ \overrightarrow{\dot{r}_{\text{des}}(t)} - \overrightarrow{\dot{r}_c(t)} \right]$  Similarly the torque controller uses the desired chaser attitude angle and actual chaser attitude angle as the control feedback signal.

$$\tau_{\text{control}} = K_{P,\theta} [\theta_{\text{des}}(t) - \theta_c(t)] + K_{D,\theta} [\dot{\theta}_{\text{des}}(t) - \dot{\theta}_c(t)] \quad (3.10)$$

$K_P, K_{P,\theta}, K_D, K_{D,\theta}$  are the Proportional gain matrix and the Derivative gain matrix respectively. These matrix value are designed and tuned by designer for the special case.

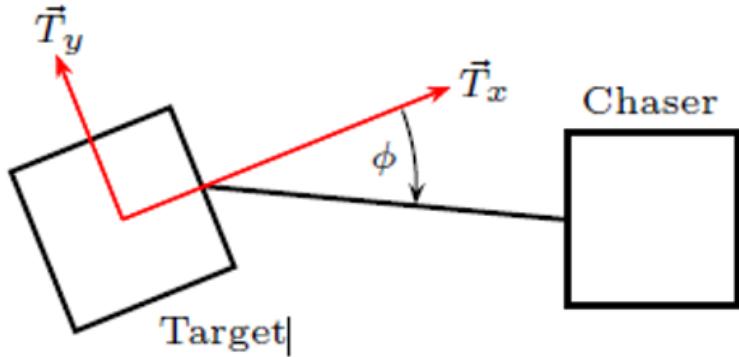


Figure 3.6. Positive tether target angle

## 3.4 MATLAB Implementation

In this section, the MATLAB simulation case is implemented. To simplified the study, the implementation case use only one controller. The thrust stabilization only uses force controller. The Tether Satellite System Spin Stabilization only use torque control, commanding the chaser to continually point its  $C_x$  axis toward the target. In the real case the controller can be the combination of these two kinds. Here we just implemented the Thrust stabilization of single tether for illustration the problem through the flowchart 3.5.

The detail MATLAB code is in the Appendix 5.1. Symbols have the same meaning and constant values are not changed, as[20].

### 3.4.1 Thrust stabilization

The chaser is designed as uniformly accelerated motion for simplicity. The motion can be replaced by any other arbitrarily defined motion.

$$\overrightarrow{r_{des}(t)} = \overrightarrow{r_0} + \overrightarrow{d} \frac{t^2}{2} \quad (3.11)$$

$d$  is the acceleration got from the desired final radius vector, the initial radius vector and assigned time for motion,  $\overrightarrow{d} = \frac{2(\overrightarrow{r}_{final} - \overrightarrow{r}_0)}{t^2}$ ,  $\overrightarrow{r}_{des}(t) = \overrightarrow{d} t$ . From the figure comparison, we know that the effect of implementation is quite well.

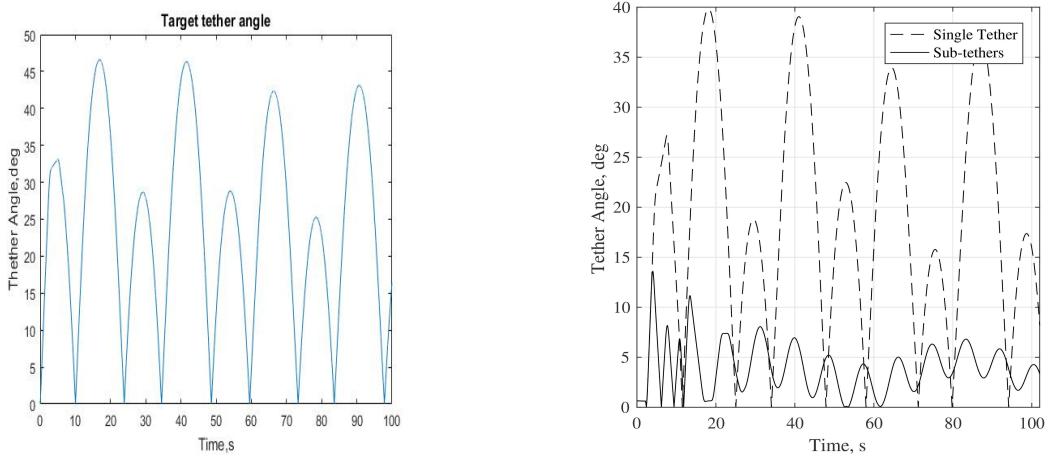


Figure 3.7. Comparison of Target tether angle. Left: Kaiwen Right: Hovell

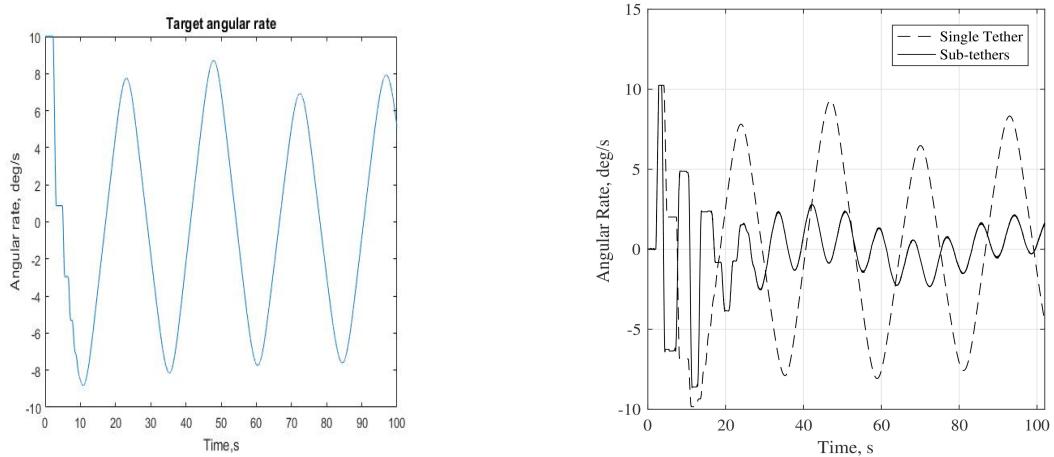


Figure 3.8. Comparison of Target angular rate. Left: Kaiwen Right: Hovell

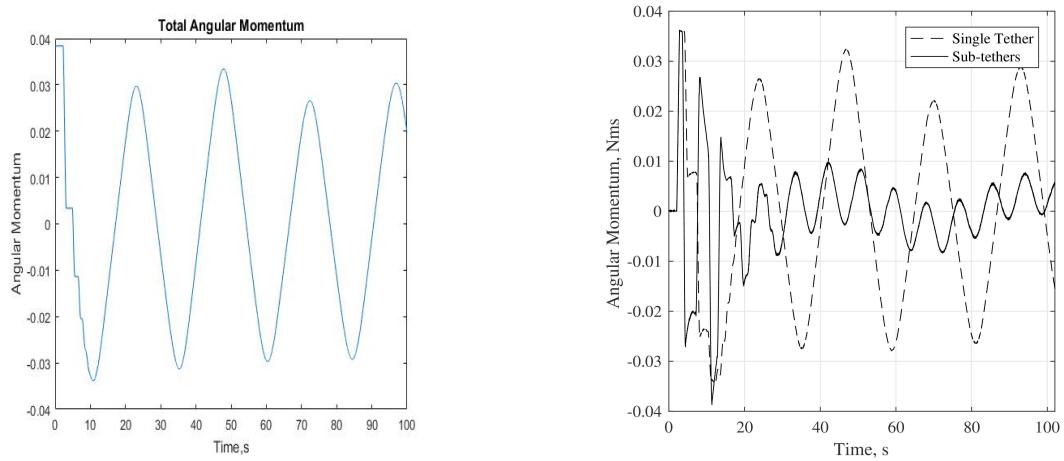


Figure 3.9. Comparison of Total angular momentum. Left: Kaiwen Right: Hovell

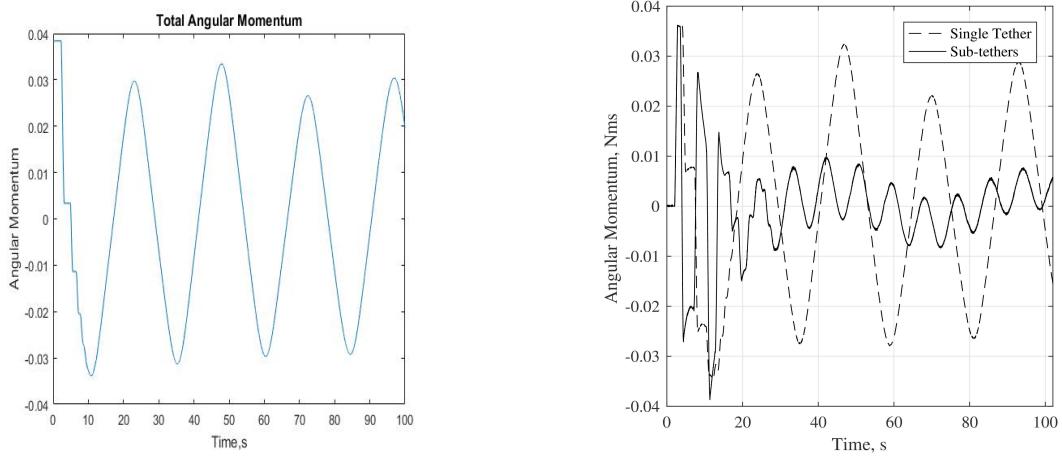


Figure 3.10. Comparison of Total angular momentum. Left: Kaiwen Right: Hovell

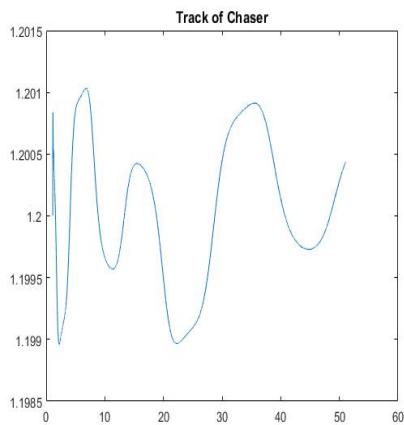


Figure 3.11. Track of Chaser

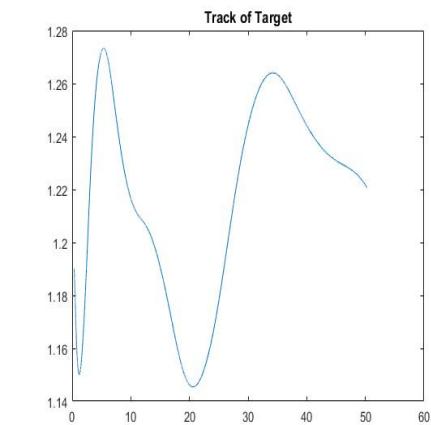


Figure 3.12. Track of Target

# CHAPTER 4

## EXPERIMENT DESIGN

This chapter talks about the Experiment set up.

### 4.1 ArduSat

The HKUST CYT Lab has a lot of ArduSat DemosSats[19, 30]. ArduSat is an Arduino based Nanosatellite, based on the CubeSat standard. It contains a set of Arduino boards and sensors. The general public will be allowed to use these Arduinos and sensors for their own creative purposes while they are in space. The DemoSat is designed to the Cube-



Figure 4.1. ArduSat DemoSat

Sat One Unit (1U) standard. The 3D printed frame comes fully assembled with standoffs and see-through acrylic platforms. Each DemoSat also includes all of the components included in the Because Learning Sensor Kit. The Because Learning Sensor Kit includes:

1. Micro controller programmed w/ Arduino
2. Because Learning 'Sensor board' sensors

- Accelerometer
- Gyroscope
- Magnetometer

Also the DemoSat includes two wireless radio frequency (RF) radios. They are Digi XBee 2.4Ghz modules. One is connected inside the DemoSat and the second is connect via USB to laptop. By doing this it enables the DemoSatellite to communicate up to 1500 meter from the computer. It can get our interested acceleration, angular rate and orientation of the Satellites at real time wirelessly.

#### **4.1.1 Arduino Data Acquisition**

Our interested physical quantities are acceleration, angular rate and Orientation. By using the ArdusatSDK[3], acceleration can be read from Sensor LSM303 triple-axis accelerometer and Angular rates can be read from Sensor L3GD20 three-axis gyroscope. The orientation i.e. roll, pitch and heading, can be derived from readings from the Acceleration and Magnetic Sensors LSM303-Triple-axis Accelerometer plus Magnetometer (Compass) Board.

For the reason of real-time plotting simplicity, the Comma-separated values(CSV) format which provides the timestamp of action, is adopted. The CSV streaming Arduino code is written at Appendix 5.2. The CSV format is: **timestamp,sensorName,values,...,checksum**.

For example, one of the result is like this(during around 100ms):

```
...
33,Acceleration,0.122,-0.022,-8.894,1217
44,Gyro,-0.177,-0.999,-0.142,416
53,Orientation,-179.737,-0.759,7.834,991
64,Acceleration,0.074,-0.062,-8.863,1217
74,Gyro,-0.181,-1.074,-0.112,416
83,Orientation,-179.598,-0.480,7.808,992
```

95,Acceleration,0.033,-0.067,-8.944,1217

105,Gyro,-0.216,-0.988,-0.177,416

115,Orientation,-179.739,-0.384,7.846,992

126,Acceleration,0.045,-0.010,-8.920,1217

137,Gyro,-0.110,-0.318,-0.061,417

...

The time-stamp is the time value from which the program was operated, in millisecond. The last number is checksum which used for check the communication. The baud rate is set as 5700 as softwareSerial does not appear to work reliably above 57600 baud.[3] When using baud rate as 57600, the update frequency of physical quantities are 32Hz, which means every second we can get 32 groups of acceleration, angular rate and orientation. The unit of baud rate is bits per second.i.e. 57600 bits per second equals 7.2KB/s[17].

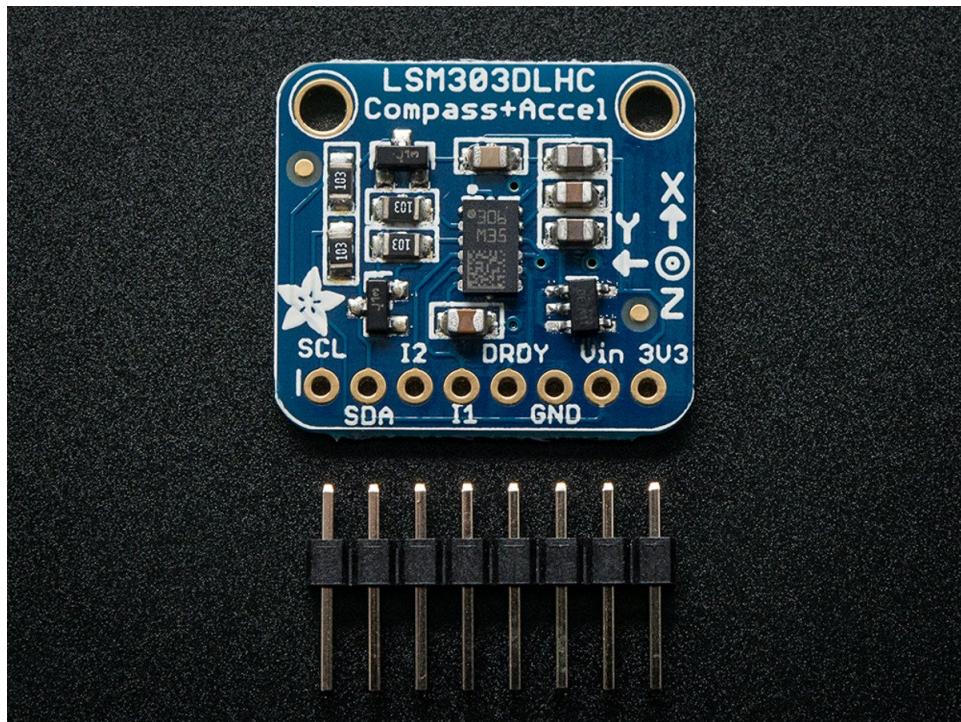


Figure 4.2. LSM303 Triple-axis Accelerometer plus Magnetometer (Compass) Board



Figure 4.3. Xbee part with Shield

#### 4.1.2 Xbee modules configuration

Digi XCTU[2] is the software used to configure the Xbee modules. In our case, two ArduSat Demosat should send the real time data to the computers separately. The two XBee parts are on ArduSat A and ArduSat B. The other two Xbee parts are plugged in shields and the shields are connected to Computer A and Computer B respectively. All the Xbee parts should be set carefully[1]. The channel controls the frequency band that Xbee communicates over. XBee's operate on the 2.4GHZ 802.15.4 band, and the channel further calibrates the operating frequency within that band. The two XBee modules having on the same network operates on the same channel, like Figure 4.1.2. The two networks are using different channels in case they interact with each other.

Personal area network ID (PAN ID) is some hexadecimal value between 0 and 0xFFFF. Similar to the channel, the two Xbee modules in the same network must have the same network ID and the different network should have different PAN ID.

Each XBee in a network should be assigned a 16-bit address (again between 0 and 0xFFFF), which is referred to as MY address, or the "source" address. Another setting, the destination address, determines which source address an XBee can send data to. For one XBee to be able to send data to another, it must have the same destination address as the

Table 4.1. Xbee parts setting

Xbee part Setting	A shield	On Ardusat A	B shield	On ArduSat B
Channel	C		17	
Personal area network ID	3332		83D5	
Destination Address High	0	0	0	13A200
Destination Address Low	35	0	1234	4163E25D
16-bit Source Address	0	35	4321	1532
Serial Number High	13A200			
Serial Number Low	4167BD1E	4163E2BA	4163E25D	4167BD2E
Interface Data Rate	57600			

other XBee's source.

Each XBee has a unique 16-bit Source Address i.e. MY address.

In our case, the task is that On Ardusat A XBee should only send data to A shield XBee and On Ardusat B XBee should only send data to B shield XBee.

There are two way to configure destination address

1. Leave Destination Address High set to 0, and set Destination Address Low to the MY address of the receiving XBee.
2. Set Destination Address High to the Serial Number High (SH) and Destination Address Low to the Serial Number Low (SL) of your destination XBee.

To assure that each network has high fidelity and the communication interference between A network and B network is as low as possible, the A network is only using method 1 and the B network is using method 2. A network's configuration obeys method 1 and disobeys method 2. B network's configuration obeys method 2 and goes against method 1.

The key is to reduce communication interference when two satellites are moving especially when they are quite close. The final configuration is showed at Table 4.1.2.

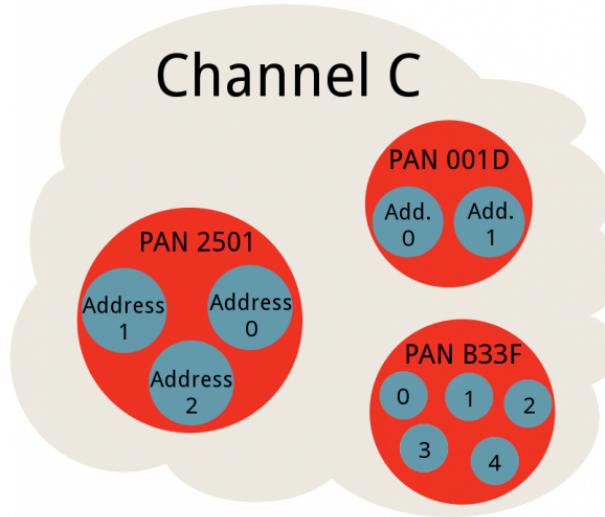


Figure 4.4. Xbee-networks example

#### 4.1.3 Real-time data streaming to MATLAB via USB

Since the Xbee modules can send data wirelessly to the shield module, the Arduino Serial Monitor is able to receive and pop out the streaming txt data. However, Arduino cannot plot the received data at real time and save the data synchronously.

To plot the real-time streaming data and record them we need to use MATLAB[27]. MATLAB serial can create the serial port object. The Xbee shield is connected to computer via USB and using serial port to do data transmission.

The MATLAB code in Appendix 5.3 gives the functionality. This MATLAB file reads the serial post, plots the three physical quantities and saves the data into record matrix.

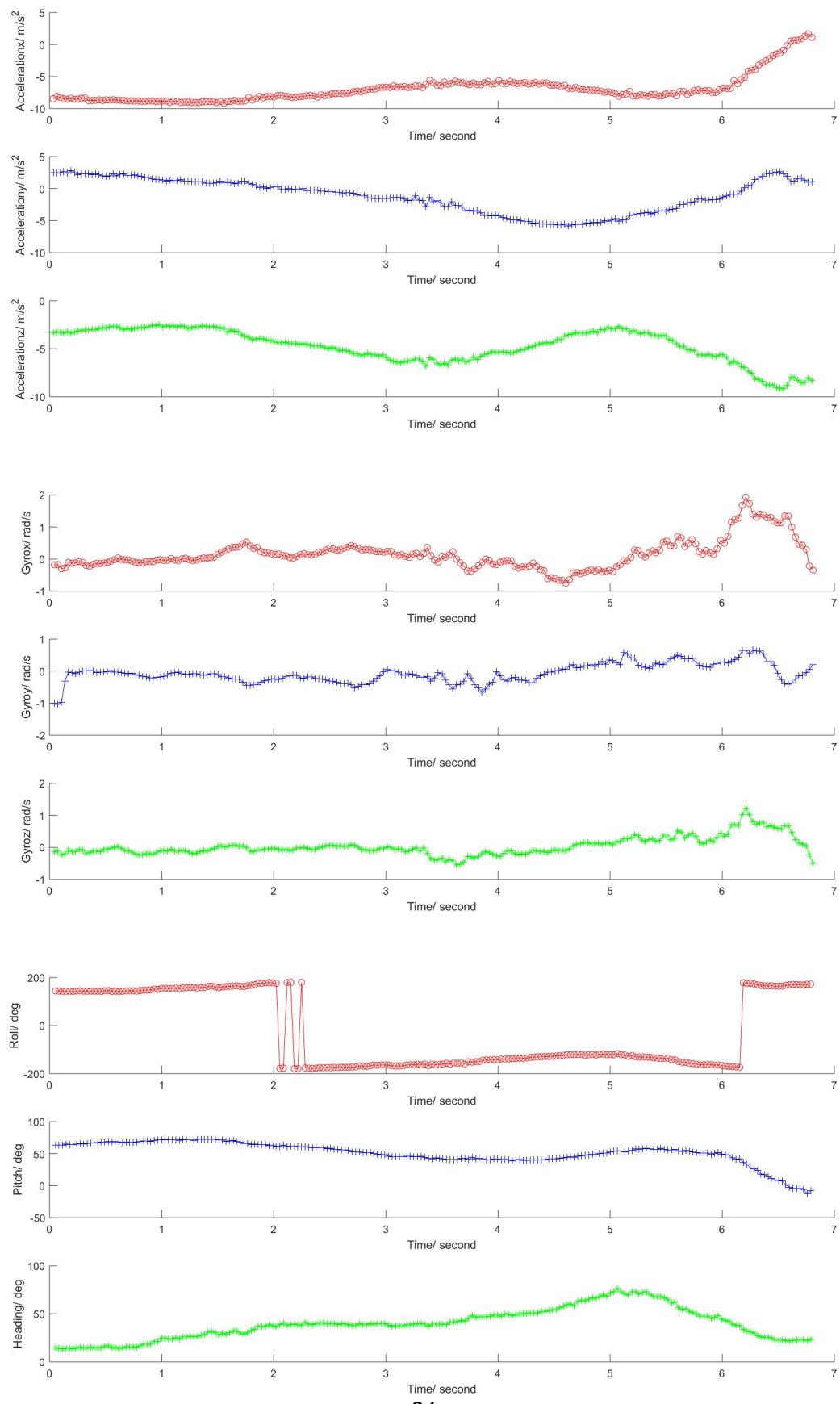


Figure 4.5. MATLAB plot real-time streaming figure

## 4.2 Testbed fabrication

Utilizing Porous Media Technology Air Bushings deliver low coefficients of friction and high motion-control resolution for less-expensive road. The ideal components for building frictionless linear motion, Air Bearings were the original standard, off-the-shelf, porous media air bearing product line.

The first version of Air Bushing Planar Torque Free Motion simulator(Hong Kong indigenous nanosatellite engineering test facilities), designed by Dr. Pok Wang KWAN is showing at Figure 4.6 The experimental has been carefully engineered to reduce the cost while it intended to provide good performance. The air bushing floating platform shall provide low friction translational motion on horizontal plane(two degrees of freedom).

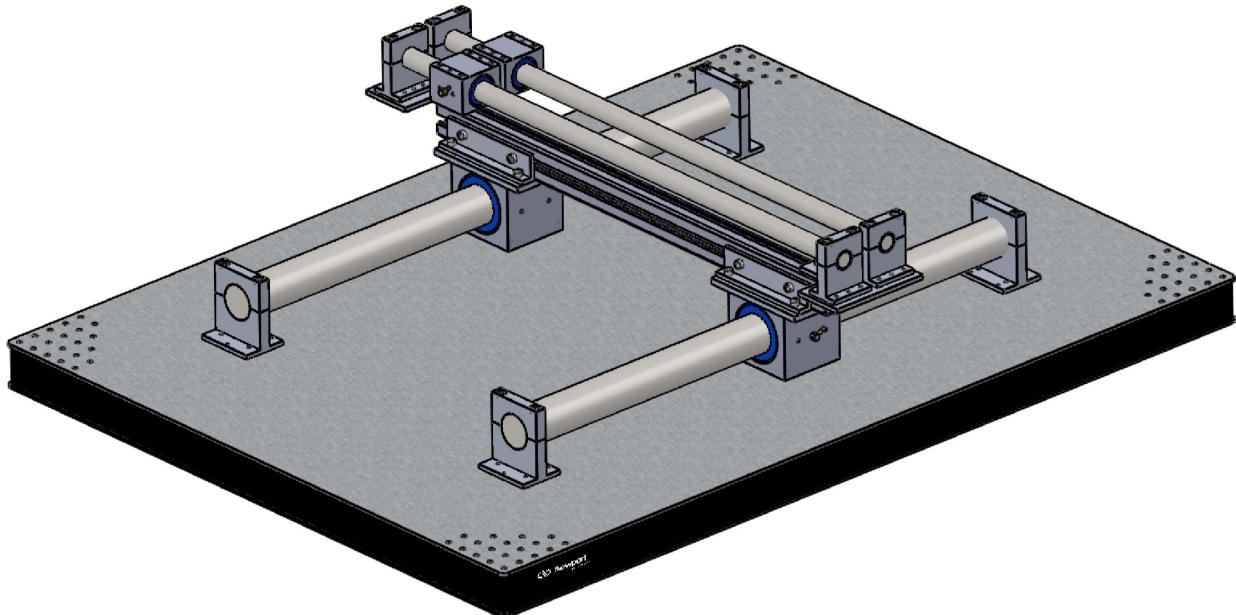


Figure 4.6. Air Bushing Planar Torque Free Motion simulator

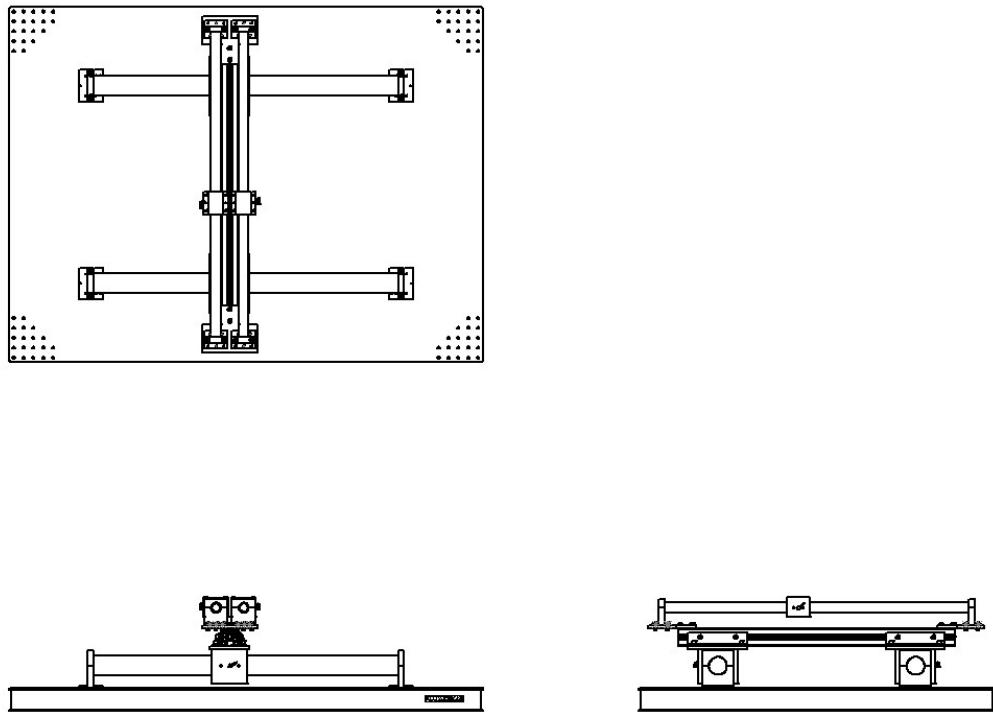


Figure 4.7. Air Bushing Planar Torque Free Motion simulator-Three View Drawing

However, the Air Bushing Torque Free Motion simulator has a fatal problem. As the weight of top floor air bushings and top floor rods are bear by the bottom floor air bushings and rods, the mass of inertia along two axes are different. If two forces with same magnitude exerted on the same point of action along two vertical axes, the top floor air bushings will generate much bigger acceleration than the bottom floor air bushings since the motion along that way is much easier.

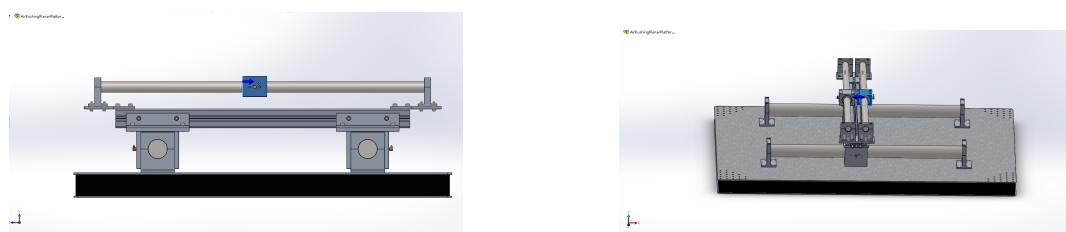


Figure 4.8. Exerted force Right View

Figure 4.9. Exerted force Front View

In Solidworks, using Motion study modules, the simulation result(Figure 4.10) of Mo-

tion Analysis also support this argument. As we can see the blue line represent the acceleration(2) of bottom floor, around  $81 \text{ mm/sec}^2$  which is much much smaller than the acceleration of top floor. The top floor air bushing has the acceleration more than  $1300 \text{ mm/sec}^2$ . The oscillation of acceleration means the air bushing of top floor hits the border and got reversal small acceleration. The exerted force have the same magnitude and vertical direction. One is along the long thick rod and the other is along the short thin rod.

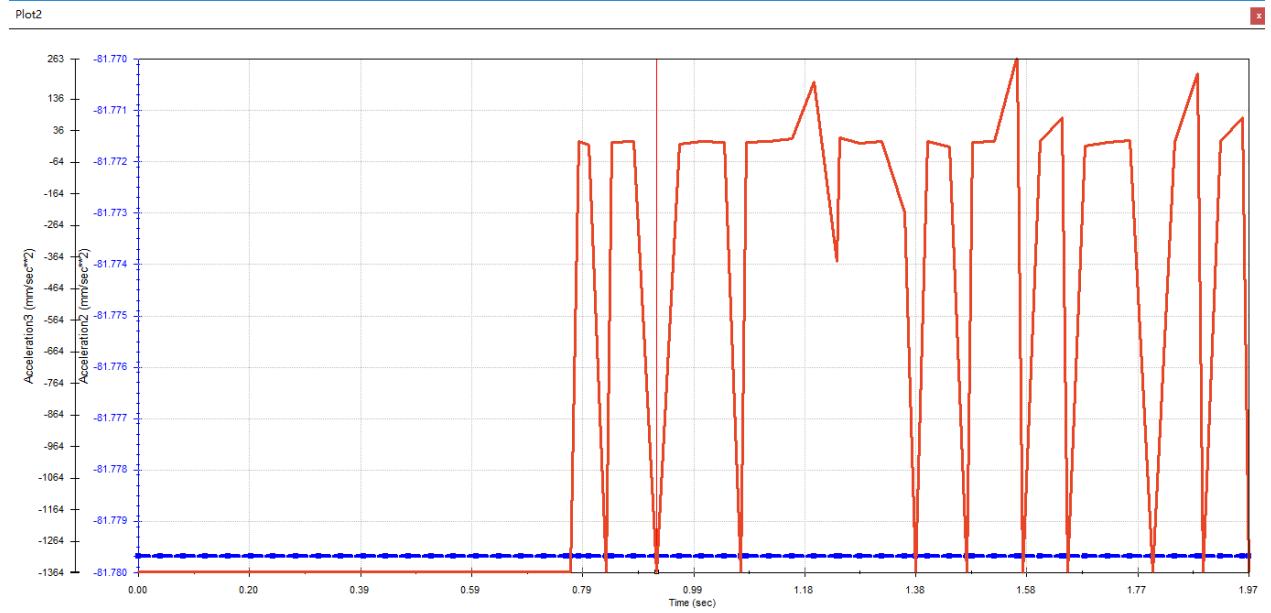


Figure 4.10. Solidworks Motion Analysis

In theory the problem can be solved by using compensation mechanism. If we add another pair of Air Bushing Torque Free Motion simulator at the top of the original one and let them be vertical, then we use some physical connection such as rod or bar to connect these two pairs of simulator. The top plane and the bottom plane are fixed. The mass of Inertia along two axes now become the same and the problem can be solved.

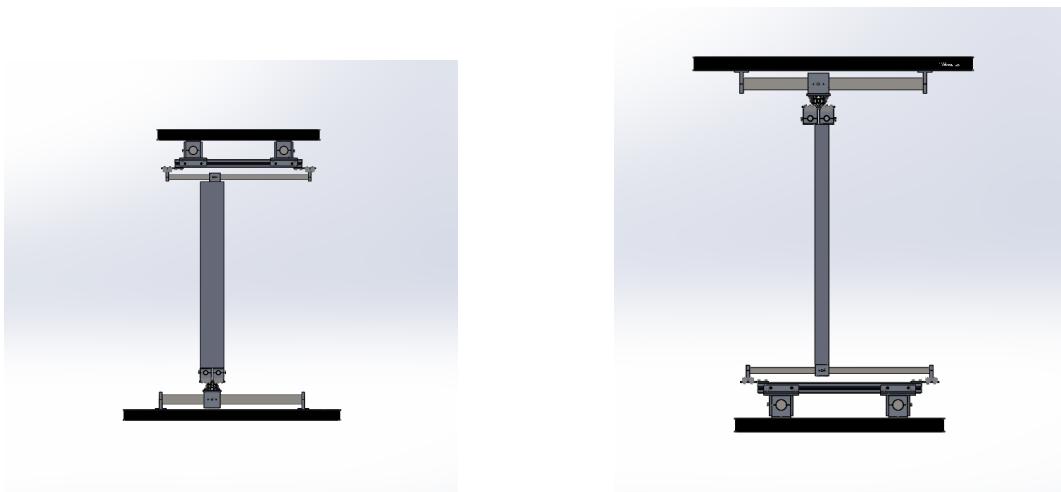


Figure 4.11. Compensation configuration conception

Due to the weight of the fixing platform is too heavy and the additional need of rotation degree of freedom, the Suspension/Hanging configuration is used. A shroud bracket is connected with the moving air bushing mounting blocks and moves together. The Cubusat is connected to shroud bracket using a string, which provide the rotational degree of freedom.

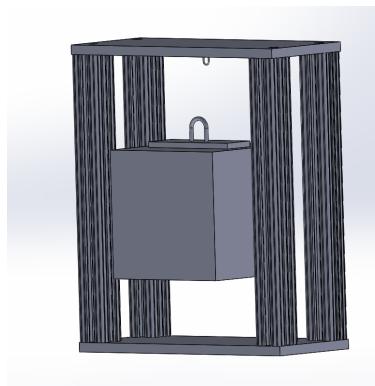


Figure 4.12. Hanging configuration

# CHAPTER 5

## APPENDIX

### 5.1 MATLAB Thrust Stabilization-Single tether implementation

```
% This code is an implementation of Thrust Stabilization  
% for single tether  
% Hovell, K, and Ulrich, S. (2017)  
% "Experimental Validation for Tethered Capture of the Spinning  
% Space Debris"  
% AIAA SciTech Forum, AIAA Guidance, Navigation, and Control  
% Conference, Grapevine, Texas. 9 ?13 January  
% Paper no AIAA 2017-1049.  
  
% =====  
% Nomenclature:  
% Jzzt = Moment of inertia , target about z-axis  
% Jzzc = Moment of inertia , chaser about z-axis  
  
% mt = Mass of the target  
% mc = Mass of the chaser  
% mj = Mass of the junction  
  
% thetac0 = Initial angle of the chaser  
% thetato = Initial angle of the target  
  
% TS = Time step  
  
% Lm0= unstretched length of the main tether  
% Ls10 = unstretched length of the sub tether 1  
% Ls20 = unstretched length of the sub tether 2  
% Ls0 = unstretched length of the single tether  
  
% ac = tether attachment point to body chaser with respect to  
% chaser's  
% center of mass  
% as1 = tether attachment point to body sub tether 1 with respect  
% to
```

```

% sub-tether1's center of mass
% as2 = tether attachment point to body sub tether 2 with respect
% to
% sub-tether2's center of mass
% at = tether attachment point to body target with respect to
% target's center of mass

% csingle = damping coefficient for single tether
% csub = damping coefficient for sub tether

% KPtheta = proportional gain matrix for the theta
% KDtheta = derivative gain matrix for the theta
% KP = proportional gain matrix for the position
% KD = derivative gain matrix for the position

% =====
% Define initial conditions and TSS parameters for all
% simulations and
% ijexperiment

%in  $\text{Kgm}^2$ 
Jzzt=0.22;
Jzzc=0.30;

%in Kilogram
mt=12.19;
mc=17.24;
mj=0.01;

%in Degree
thetac0=180;
thetat0=0;

% in Second
TS=0.004;

%in Meter
Lm0=0.28;
Ls10=0.28;
Ls20=0.28;
Ls0=0.54;
ac=[0.135;-0.009];
as1=[0.110;0.127];
as2=[0.113;-0.094];

```

```

at=[0.110;0.016];

% in Newtonsecond/meter
csingle=3.0;
csub=0.9;

KPtheta=0.33;
KDtheta=0.56;
KP=[19.1 0;0 19.1];
KD=[32.7 0;0 32.7];

% =====
% Define initial conditions for Thurst Stabilization simulations
% and experiment
rt0=[0.35;1.19];
rc0=[1.10;1.20];
vt0=[0;0];
vc0=[0;0];
wt0=10; %degree/second
rf=[3.10;1.20];

% =====
% Initialization
% Simulation at time t=2s
% Nothing change in the first two seconds
rt=rt0;
rc=rc0;
vt=vt0;
vc=vc0;
thetat=thetat0;
thetac=thetac0;
Ls=rt+A(thetat)*at-rc-A(thetac)*ac;

% fai is the target tether angle
fai=norm(atand(Ls(2)/Ls(1))-thetat);

wt=wt0;
wc=0;
t=0; % It started to simulate at T=2, which seem it as t=0;
Fs=[0;0];

% AngularMomentum
% In radius

```

```

AngularMomentum = Jzzt * wt/57.3 + Jzzc * wc/57.3 ;

accelerationc=[0;0];
accelerationt=[0;0];
angularaccelerationt=0;
taot=0;
Fcontrol=[0;0];

for i=1:25000 % i* TS = the time , i is the number of the current
time step
%record the data
recordt(i)=t;
recordrc(i,1)=rc(1);
recordrc(i,2)=rc(2);
recordrt(i,1)=rt(1);
recordrt(i,2)=rt(2);
recordthetat(i)=thetat;

recordwt(i)=wt;
recordFs(i,1)=Fs(1);
recordFs(i,2)=Fs(2);

recordLs(i,1)=Ls(1);
recordLs(i,2)=Ls(2);

recordvc(i,1)=vc(1);
recordvc(i,2)=vc(2);

recordvt(i,1)=vt(1);
recordvt(i,2)=vt(2);

recordaccelerationc(i,1)=accelerationc(1);
recordaccelerationc(i,2)=accelerationc(2);
recordaccelerationt(i,1)=accelerationt(1);
recordaccelerationt(i,2)=accelerationt(2);

recordangularaccelerationt(i)=angularaccelerationt;

recordtaot(i)=taot;

recordFcontrol(i,1)=Fcontrol(1);

```

```

recordFcontrol(i,2)=Fcontrol(2);
recordfai(i)=fai;

recordAngularMomentum(i) = AngularMomentum ;

%records stiffness(i) = stiffness(norm(Ls)-Ls0);

%Ls is the vector which is from chaser to target, pointing at the
target
Ls=rt+A(thetaT)*at-rc-A(thetaC)*ac;

% fai is target tether angle
fai=norm(-atan2(Ls(2)/Ls(1))+thetaT);

% Calculate the state of the tether
% Hovell and Ulrich (2017) eqn 19
if (norm(Ls)-Ls0)>0 % tether is tensioned
    Fs=(stiffness(norm(Ls)-Ls0)*(norm(Ls)-Ls0)+...
        dot(csingle*(vt+A(thetaT)*[-wt/57.3*at(2);wt/57.3*at(1)]-
                    vc-A(thetaC)*[-wc/57.3*ac(2);wc/57.3*ac(1)]) ,...
        Ls/norm(Ls))).*Ls/norm(Ls);
else % tether is slackness
    Fs=[0;0];
end

% Force on target due to tether tension:
% through center of mass
% referenced to the target's body frame
Fst=(A(thetaT))'*Fs;
Fsc=(A(thetaC))'*Fs;

% Torque on target due to tether tension:
% about the z-axis of body frame (which is parallel to the z-axis
% of inertial frame)
% through center of mass target
% referenced to target body frame (the torque referenced to the
% body is the same as to the inertial frame)
taot=at(2)*Fst(1)-at(1)*Fst(2); % Nm
taoc=ac(1)*Fsc(2)-ac(2)*Fsc(1); % Nm

% The control thrust of chaser vehicle
Fcontrol=KP*(rdes(t)-rc)+KD*(vdes(t)-vc);

```

```

% The control torque of chaser vehicle
taocontrol=KPtheta* (180 - thetac) + KDtheta*(0-wc) ;
accelerationt=(-Fs/mt);
accelerationc=(Fs+Fcontrol)/mc;
angularaccelerationt=taot/Jzzt;
angularaccelerationc=(taoc+taocontrol)/Jzzc;

% position and angle update
rt=rt+vt*TS;
rc=rc+vc*TS;
thetat=thetat+wt*TS;
thetac=thetac+wc*TS;

vt=vt+accelerationt*TS;
wt=wt+angularaccelerationt*TS*(180/ pi);%/degree t-1
vc=vc+accelerationc*TS;
wc=wc+angularaccelerationc*TS*(180/ pi); %/degree t-1
AngularMomentum = Jzzt * wt/57.3 + Jzzc * wc/57.3;
t=t+TS;
end
recordt=recordt';
recordthetat=recordthetat';
recordwt=recordwt';

recordrc=recordrc';
recordrt=recordrt';
recordvc=recordvc';
recordvvt=recordvvt';
recordaccelerationc=recordaccelerationc';
recordaccelerationt=recordaccelerationt';
recordangularaccelerationt=recordangularaccelerationt';
recordangularaccelerationt=recordangularaccelerationt';
recordaccelerationc=recordaccelerationc';
recordaccelerationt=recordaccelerationt';
recordtaot=recordtaot';
recordFcontrol=recordFcontrol';

n=1;
figure(n)
plot(recordt,recordwt);
title('Target angular rate')
xlabel('Time, s')
ylabel('Angular rate , deg/s')
n=n+1;

```

```

figure(n)
plot(recordt, recordthetat)
title('angle of target')
n=n+1;

figure(n)
plot(recordt, recordrc(1,:))
title('x position of the chaser')
n=n+1;

figure(n)
plot(recordt, recordrc(2,:))
title('y position of the chaser')
n=n+1;

figure(n)
plot(recordt, recordrt(1,:))
title('x position of the target')
n=n+1;

figure(n)
plot(recordt, recordrt(2,:))
title('y position of the target')
n=n+1;

figure(n)
plot(recordt,mod(recordfai,360))
title('Target tether angle')
xlabel('Time,s')
ylabel('Thether Angle,deg')
n=n+1;

figure(n)
plot(recordt,recordAngularMomentum)
title('Total Angular Momentum')
xlabel('Time,s')
ylabel('Angular Momentum')

n = n + 1;
figure(n)
plot(recordrc(1,:), recordrc(2,:))
title('Track of Chaser')

n = n+1;

```

```

figure(n)
plot(recordrt(1,:),recordrt(2,:))
title('Track of Target')

% The function gives the attitude matrix in terms of theta
% cos -sin; sin cos;

function attitude=A(sita)
attitude=[cosd(sita) -sind(sita);sind(sita) cosd(sita)];
end

% calculate the design radius vector of time t
% the goal is to reach the destination in 20s after t=2
function [rdes]=rdes(t)
rc0=[1.10;1.20];
rf=[3.10;1.20];
d=2.* (rf-rc0)./(20*20); %in 20s
[rdes]=(rc0+d.* (t*t/2));
end

% calculate the single tether stiffness(N/m) in terms of
% stretch(m)-independent variable
% x=(||Ls||-Ls,0)
function stiff=stiffness(x)
originalstiffness
=[163.500000000000,114.450000000000,80.2636363636363,
51.3857142857143,29.9096000000000,18.8842500000000,17.0724030612245,
10.1577253521127,8.06715957446808,7.09840970654628,6.99056826923077,
7.22393661971831,8.06497910662824,8.78250400534045,9.58060266159696];

originalstretch
=[0.0030000000000000,0.0060000000000001,0.0110000000000000,
0.0210000000000000,0.0450000000000000,0.0860000000000001,
0.0980000000000001,0.213000000000000,0.329000000000000,0.443000000000000,
0.520000000000000,0.639000000000000,0.694000000000000,0.749000000000000,
0.789000000000000];
stiff=spline(originalstretch,originalstiffness,x);
end

% calculate the design velocity of time t
% the goal is to reach the destination in 20s after t=2s

```

```

function vdes=vdes(t)
rc0=[1.10;1.20];
rf=[3.10;1.20];
d=2.* (rf-rc0)./(20*20); %in 20s
vdes=d*t;
end

```

## 5.2 Arduino CSV Streaming Example

```

#include <ArdusatSDK.h>
#include <Wire.h>
#include <Arduino.h>

/*
 * Setup Software Serial to allow for both RF communication and
 * USB communication
 * RX is digital pin 8 (connect to TX/DOUT of RF Device)
 * TX is digital pin 9 (connect to RX/DIN of RF Device)
 */
ArdusatSerial serialConnection(SERIAL_MODE_HARDWARE_AND_SOFTWARE,
                                8, 9);

/*
 * Constant Definitions
 */
Acceleration accel;
Gyro gyro;
Magnetic mag;
Orientation orient(accel, mag);

void setup(void) {
    serialConnection.begin(57600);

    accel.begin();
    gyro.begin();
    mag.begin();
    orient.begin();
    serialConnection.println("Start of the loop");
    serialConnection.println("The acceleration x y z unit in m/s^2");
    serialConnection.println("The angular rate x y z unit in radian per second");
}

```

```

    serialConnection.println("The orientation roll pitch heading
        unit in degree");
}

void loop(void) {
    serialConnection.println(accel.readToCSV(" Acceleration"));
    serialConnection.println(gyro.readToCSV("Gyro"));
    serialConnection.println(orient.readToCSV(" Orientation"));
}

```

## 5.3 MATLAB Serial REALTIME PLOTING

```

%% Pre-process
close all; % close all figures
clear all; % clear all workspace variables
clc; % clear the command line
fclose('all'); % close all open files
delete(instrfindall); % reset com port

%% Constatns
BAUDRATE = 57600; % Need to configure Xbee parts BD i.e.
    Interface Data Rate
INPUTBUFFER = 51200; %Unit in bytes. Definition:A location that
    holds all
% incoming information before it continues to the CPU for
    processing;
% Buffers used to store information before it processed

%% Initialize the stripchart
figure_acceleration = figure('Name',' Acceleration');
axes_accelerationx = subplot(3,1,1); % Axes Object
xlabel(axes_accelerationx,'Time/ second')
ylabel(axes_accelerationx,' Accelerationx/ m/s^2')

axes_accelerationy = subplot(3,1,2);
xlabel(axes_accelerationy,'Time/ second')
ylabel(axes_accelerationy,' Accelerationy/ m/s^2')

axes_accelerationz = subplot(3,1,3);
xlabel(axes_accelerationz,'Time/ second')
ylabel(axes_accelerationz,' Accelerationz/ m/s^2')

h_ax = animatedline(axes_accelerationx,'MaximumNumPoints',Inf,'

```

```

    Marker', 'o', 'Color', 'red'); % animated line object
h_ay = animatedline(axes_accelerationy,'MaximumNumPoints',Inf,'
    Marker', '+', 'Color', 'blue');
h_az = animatedline(axes_accelerationz,'MaximumNumPoints',Inf,'
    Marker', '*', 'Color', 'green');

figure_gyro = figure('Name','Gyro');
axes_gyrox = subplot(3,1,1);
xlabel(axes_gyrox,'Time/ second')
ylabel(axes_gyrox,'Gyrox/ rad/s')

axes_gyroy = subplot(3,1,2);
xlabel(axes_gyroy,'Time/ second')
ylabel(axes_gyroy,'Gyroy/ rad/s')

axes_gyroz = subplot(3,1,3);
xlabel(axes_gyroz,'Time/ second')
ylabel(axes_gyroz,'Gyroz/ rad/s')

h_gx = animatedline(axes_gyrox,'MaximumNumPoints',Inf,'Marker',
    'o', 'Color', 'red');
h_ty = animatedline(axes_gyroy,'MaximumNumPoints',Inf,'Marker',
    '+', 'Color', 'blue');
h_tz = animatedline(axes_gyroz,'MaximumNumPoints',Inf,'Marker',
    '*', 'Color', 'green');

figure_orientation = figure('Name','Orientation');
axes_roll = subplot(3,1,1);
xlabel(axes_roll,'Time/ second')
ylabel(axes_roll,'Roll/ deg')

axes_pitch = subplot(3,1,2);
xlabel(axes_pitch,'Time/ second')
ylabel(axes_pitch,'Pitch/ deg')

axes_heading = subplot(3,1,3);
xlabel(axes_heading,'Time/ second')
ylabel(axes_heading,'Heading/ deg')

h_roll = animatedline(axes_roll,'MaximumNumPoints',Inf,'Marker',
    'o', 'Color', 'red');
h_pitch = animatedline(axes_pitch,'MaximumNumPoints',Inf,'Marker',
    '+', 'Color', 'blue');
h_heading = animatedline(axes_heading,'MaximumNumPoints',Inf,'
    Marker', '*', 'Color', 'green');

```

```

% Index
i = 1;
j = 1;
k = 1;

% Record the data
% 10000 Row data
record_acceleration = zeros(10000,4);
record_gyro = zeros(10000,4);
record_orientation = zeros(10000,4);

%% Create a serial object
board = serial('COM4','BaudRate',BAUDRATE,'DataBits',8);
% Related with the used COM, can be different
% COM4 for wireless, COM3 for wire
% BAUDRATE unit: bits per second, e.g. 9600 means 9600 bits per
% second,
% i.e. 1200 bytes per second i.e. 1.2 KB/s or 1.17KB/s

% Set serial port buffer
set(board,'InputBufferSize',INPUTBUFFER);
% InputBufferSize: total number of bytes that can be stored in
% the input
% buffer during a read operation. The read operation is
% terminated if the
% amount of data stored in the input buffer equals the
% InputBufferSize

fopen(board);
while(1)
    a = fgetl(board);% a: character vector
    b = textscan(a,'%u32 %s %f %f %u32','Delimiter','','');
    if strcmp(char(b{2}),'Acceleration')

        time = double(b{1})/1000; % in second
        ax = b{3};
        addpoints(h_ax,time,ax);
        drawnow limitrate

        ay = b{4};
        addpoints(h_ay,time,ay);
        drawnow limitrate
    end
end

```

```

az = b{5};
    addpoints(h_az,time,az);
    drawnow limitrate

record_acceleration(i,:)=[time ax ay az];
i=i+1;

elseif strcmp(char(b{2}),'Gyro')
    time = double(b{1})/1000; % in second
    gx = b{3};
        addpoints(h_gx,time,gx);
        drawnow limitrate

    gy = b{4};
        addpoints(h_gy,time,gy);
        drawnow limitrate
    gz = b{5};
        addpoints(h_gz,time,gz);
        drawnow limitrate

record_gyro(j,:)=[time gx gy gz];
j=j+1;

elseif strcmp(char(b{2}),'Orientation')
    time = double(b{1})/1000; % in second
    roll = b{3};
        addpoints(h_roll,time,roll);
        drawnow limitrate

    pitch = b{4};
        addpoints(h_pitch,time,pitch);
        drawnow limitrate

    heading = b{5};
        addpoints(h_heading,time,heading);
        drawnow limitrate

record_orientation(k,:)= [time roll pitch heading];
k = k+1;
end

end

```

## REFERENCES

- [1] Exploring xbees and xctu. <https://learn.sparkfun.com/tutorials/exploring-xbees-and-xctu/configuring-networks>.
- [2] Xctu. <https://www.digi.com/products/xbee-rf-solutions/xctu-software/xctu>.
- [3] ArduSat. ArduSatsdk. <https://github.com/ArduSat/ArduSatSDK>.
- [4] VS Aslanov. The effect of the elasticity of an orbital tether system on the oscillations of a satellite. *Journal of Applied Mathematics and Mechanics*, 74(4):416–424, 2010.
- [5] Joseph Carroll and John Oldson. Tethers for small satellite applications. 1995.
- [6] Joseph A Carroll. Guidebook for analysis of tether applications. 1985.
- [7] Joseph A Carroll. Space transport development using orbital debris. *NIAC Phase I presentation and final report*, 2002.
- [8] M Cartmell and S Ziegler. Experimental scale model testing of a motorised momentum exchange propulsion tether. In *37th Joint Propulsion Conference and Exhibit*, page 3914, 2001.
- [9] MP Cartmell and DJ McKenzie. A review of space tether research. *Progress in Aerospace Sciences*, 44(1):1–21, 2008.
- [10] Yi Chen and M Cartmell. Dynamical modelling of the motorised momentum exchange tether incorporating axial elastic effects. 2007.
- [11] Yi Chen and Matthew Cartmell. Hybrid fuzzy sliding mode control for motorised space tether spin-up when coupled with axial and torsional oscillation. *Astrophysics and Space Science*, 326(1):105–118, 2010.
- [12] Yi Chen and Matthew P Cartmell. Multi-objective optimisation on motorised momentum exchange tether for payload orbital transfer. In *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*, pages 987–993. IEEE, 2007.

- [13] Yi Chen and Matthew P Cartmell. Hybrid fuzzy and sliding-mode control for motorised space tether spin-up when coupled with axial oscillation. In *Journal of Physics: Conference Series*, volume 181, page 012093. IOP Publishing, 2009.
- [14] Yi Chen and Matthew P Cartmell. Hybrid sliding mode control for motorised space tether spin-up when coupled with axial oscillation. 2009.
- [15] Yi Chen, Rui Huang, Liping He, Xianlin Ren, and Bin Zheng. Dynamical modelling and control of space tethers: a review of space tether research. *Nonlinear Dynamics*, 77(4):1077–1099, 2014.
- [16] Yi Chen, Rui Huang, Xianlin Ren, Liping He, and Ye He. History of the tether concept and tether missions: a review. *ISRN astronomy and astrophysics*, 2013, 2013.
- [17] Ping Dong. Device and method for on-line adaptive selection of baud rate and carrier frequency, March 22 1994. US Patent 5,297,186.
- [18] Joshua R Ellis and Christopher D Hall. Out-of-plane librations of spinning tethered satellite systems. *Celestial Mechanics and Dynamical Astronomy*, 106(1):39, 2010.
- [19] Dirk Geeroms, Sabine Bertho, Michel De Roeve, Rik Lempens, Michiel Ordies, and Jeroen Prooth. ArduSat, an arduino-based cubesat providing students with the opportunity to create their own satellite experiment and collect real-world space data. ESA Publications Division C/O ESTEC, 2015.
- [20] Kirk Hovell and Steve Ulrich. Experimental validation for tethered capture of spinning space debris. In *AIAA Guidance, Navigation, and Control Conference*, page 1049, 2017.
- [21] Norilmi Amilia Ismail. *The dynamics of a flexible motorised momentum exchange tether (MMET)*. PhD thesis, University of Glasgow, 2012.
- [22] Les Johnson, Brian Gilchrist, Robert D Estes, and Enrico Lorenzini. Overview of future nasa tether applications. *Advances in Space Research*, 24(8):1055–1063, 1999.
- [23] Wonyoung Jung, Andre P Mazzoleni, and Jintai Chung. Dynamic analysis of a tethered satellite system with a moving mass. *Nonlinear Dynamics*, 75(1-2):267–281, 2014.

- [24] Donald J Kessler and Burton G Cour-Palais. Collision frequency of artificial satellites: The creation of a debris belt. *Journal of Geophysical Research: Space Physics*, 83(A6):2637–2646, 1978.
- [25] Heiner Klinkrad. Space debris: Models and risk analysis (springer praxis books/astronautical engineering). 2006.
- [26] K Uldall Kristiansen, PL Palmer, and RM Roberts. Numerical modelling of elastic space tethers. *Celestial Mechanics and Dynamical Astronomy*, 113(2):235–254, 2012.
- [27] Mark S Manalo and Ashkan Ashrafi. Usb interfacing and real time data plotting with matlab. *San Diego University Department of Electrical and Computer Engineering Real-Time DSP and FPGA Development Lab*, 2012.
- [28] Andre P Mazzoleni and John H Hoffman. End-body dynamics of artificial-gravity-generating tethered satellite system during non-planar spin-up with elastic effects included. *Advances in the Astronautical Sciences*, 116:1–16, 2004.
- [29] AP Mazzoleni and JH Hoffman. Nonplanar spin-up dynamics of the astor tethered satellite system. In *Proceeding of the 11 th Annual AAS/AIAA Space Flight Mechanics Meeting, Santa Barbara, CA*, pages 1241–1257, 2001.
- [30] Ben Peters. ArduSat space program: Training the next generation of satellite scientists and engineers. 2016.
- [31] Michel Van Pelt. *Space tethers and space elevators*. Springer Science & Business Media, 2009.
- [32] David Wright. The current space debris situation. In *Union of Concerned Scientists (UCS), Beijing Orbital Debris Mitigation Workshop*, 2010.
- [33] Denis Zanutto, Davide Curreli, and Enrico Lorenzini. Stability of electrodynamic tethers in a three-body system. *Journal of Guidance, Control, and Dynamics*, 34(5):1441–1456, 2011.
- [34] Guowei Zhao, Liang Sun, Shuping Tan, and Hai Huang. Librational characteristics of a dumbbell modeled tethered satellite under small, continuous, constant thrust.

*Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 227(5):857–872, 2013.

- [35] Spencer W Ziegler and Matthew P Cartmell. Using motorized tethers for payload orbital transfer. *Journal of Spacecraft and Rockets*, 38(6):904–913, 2001.