Aalto University
School of Electrical Engineering

LULEÅ UNIVERSITY OF TECHNOLOGY

Bagus Adiwiluhung Riwanto

CubeSat Attitude System Calibration and Testing

School of Electrical Engineering
Department of Electrical Engineering and Automation

Thesis submitted in partial fulfillment of the requirements for the degree of
Master of Science in Technology

Espoo, 17.08.2015

Instructor: Tuomas Tikka, M.Sc.
Aalto University
School of Electrical Engineering

Supervisors: Jaan Praks, D. Sc.
Aalto University
School of Electrical Engineering

Dr. Thomas Kuhn
Luleå University of Technology

AALTO UNIVERSITY
SCHOOL OF ELECTRICAL ENGINEERING

ABSTRACT OF THE
MASTER'S THESIS

Author: Bagus Adiwiluhung Riwanto

Title: CubeSat Attitude System Calibration and Testing

Date: 17.08.2015 Language: English Number of pages: 13+95

Department of Electrical Engineering and Automation

Professorship: Space Technology (S-92)

Aalto Supervisor: Jaan Praks, D. Sc.

Luleå Supervisor: Dr. Thomas Kuhn

Instructor: Tuomas Tikka, M.Sc.

This thesis concentrates on the development of Aalto-2 CubeSat attitude system calibration and testing methods. The work covers the design and testing phase of the calibration algorithms to the analysis of experimental data in order to verify the performance of the attitude instruments. The instruments under test are two-axis digital Sun sensor, three-axis magnetometer, three-axis gyroscope, and three-axis magnetorquer. These devices are all commercial off-the-shelf components which are selected for their cost-to-performance efficiency.

The Sun sensor and gyroscope were calibrated with linear batch least squares method and the results showed that only minor corrections were required for the Sun angle and angular velocity readings, while the brightness readings from the Sun sensor required more corrections. For magnetometer calibration, a specific particle swarm optimization algorithm was developed with novel approach to estimate the full calibration parameters, without having to simplify the sensor model. The calibration results were evaluated with simulation data with satisfying results, while the results from experimental data itself showed heading error improvement from 5.24°–13.24° to 1.9°–7.3° for unfiltered data. Besides the magnetometer calibration parameters estimation, the magnetic properties of the spacecraft were also analyzed using inverse multiple magnetic dipole modeling approach, where multiple magnetic dipoles positions and moments are estimated using particle swarm optimization from the magnetic field strength readings around the spacecraft. The estimated total residual magnetic moment of the spacecraft is 58.5 mA m^2, lower than the maximum magnetorquer moment which is 0.2 A m^2 in each axis. The magnetorquer was tested for verifying the validity of magnetic moment generated by the magnetorquer. The result shows that the magnetorquer moment is nonlinear, in contrast to the linear theoretical model.

Keywords: CubeSat, attitude system, calibration, multi-objective particle swarm optimization, multiple magnetic dipole modeling

Acknowledgement

I would like to give my thanks to Jaan Praks as my supervisor for providing guidance in writing this thesis, to Tuomas Tikka as my instructor for providing the hardware necessary for the thesis, and everyone in Aalto-1 and Aalto-2 project team for the opportunity of working together on this ambitious project. Finally, many thanks to Round 9 SpaceMaster students for the great time this past two years.

Otaniemi, 17.08.2015

Bagus Adiwiluhung Riwanto

Contents

List of Figures

List of Tables

List of Algorithms

Symbols and Abbreviations

Symbols

$\mathbf{1}_n$	$n \times n$ identity matrix
$\boldsymbol{I}$	moment of inertia
$\boldsymbol{\omega}$	angular velocity
$\boldsymbol{\tau}_{\text{ctrl}}$, $\boldsymbol{\tau}_{\text{dist}}$	control and disturbance torque, respectively
$\boldsymbol{v}_{ss}$	Sun direction vector
α, β	Sun angle from sensor normal direction in xz- and yz-plane, respectively
N_α, N_β	Sun angle (α, β) in bit count
$\mathbf{FoV}$, $\mathbf{FoV}_0$	Sun sensor total field-of-view range and the smallest angle in Sun sensor field-of-view, respectively
$\boldsymbol{S}_{ss}$, $\mathbf{off}_{ss}$	Sun sensor scale error matrix and offset vector, respectively
$\boldsymbol{S}_g$, $\mathbf{off}_g$	gyroscope scale error matrix and offset vector, respectively
$\boldsymbol{\eta}_{\text{ARW}}$, $\boldsymbol{\eta}_{\text{RRW}}$	gyroscope angular random walk and rate random walk, respectively
$\boldsymbol{b}$	magnetic field strength/magnetic flux density
$\boldsymbol{S}_m$, $\boldsymbol{N}$, $\boldsymbol{C}_{\text{si}}$	magnetometer scale factor, nonorthogonality matrix, and soft iron error coefficients, respectively
$\boldsymbol{b}_{\text{hi}}$, $\mathbf{off}_m$	magnetometer hard iron and offset bias, respectively
$\boldsymbol{\eta}_m$	magnetometer random noise
$\boldsymbol{K}_m$, $\boldsymbol{k}_m$	magnetometer combined calibration parameters
μ_0	magnetic permeability of free space $= 4\pi \times 10^{-7}\,[\mathrm{T\,m\,A^{-1}}]$
$d = 1 \ldots n_d$	magnetic dipoles index
$s = 1 \ldots n_s$	in general context: measurement data index
$s = 1 \ldots n_s$	in MDM context: sensors/measurement points index
$\boldsymbol{p}_s$, $\boldsymbol{r}_{ds}$	position of sensor s and its relative position from dipole d, respectively
$\boldsymbol{m}_d$, $\boldsymbol{p}_d$	magnetic moment and position of dipole d, respectively
$\boldsymbol{\tau}_{\text{mtq}}$, $\boldsymbol{m}_{\text{mtq}}$	magnetorquer torque and magnetic moment, respectively
$\boldsymbol{C}_{\text{mtq}}$, $\boldsymbol{i}_{\text{mtq}}$	combined magnetorquer constants and driver current, respectively
$i = 1 \ldots n_p$	particles index in a swarm
$j = 1 \ldots n^c$	particle components index in a swarm
$j = 1 \ldots n_c$	index for unknown parameters to be estimated $\equiv n^c$
$\boldsymbol{p}_{\text{pso},i}$, $\boldsymbol{v}_{\text{pso},i}$	position and velocity vector of particle i in a swarm
$p_{ij}(k)$, $v_{ij}(k)$	value of component j from particle i in the swarm at k-th iteration and its corresponding velocity
w, c_1, c_2	PSO parameters: inertia weight, cognitive rate, and social rate, respectively
w_v, c_v	iteration length for varying w and varying c_1 and c_2, respectively
w_s, c_{1s}, c_{2s}	starting value of w, c_1, c_2, respectively
w_f, c_{1f}, c_{2f}	final value of w, c_1, c_2 in the end of varying stage, respectively
w_e, c_{1e}, c_{2e}	end value of w, c_1, c_2 after the varying stage, respectively
$\boldsymbol{pbest}_i$	best position of particle i in a swarm

$\boldsymbol{gbest}$	global best position in the entire swarm
F_i, F_{pi}, F_g	particle i fitness value, its best fitness value in its history, and the global best fitness value, respectively
n_l	number of measurement loci with index l
$\mathring{\boldsymbol{n}}_{il}$, $\mathring{\boldsymbol{\omega}}_l$	normal direction of the plane containing locus l from measurement data calibrated with the solution of particle i, and the direction of rotation axis which corresponds to locus l, respectively
c_{f_1}, c_{f_2}	fixed weights for the PSO objectives aggregation of magnetometer calibration parameters estimation
$\boldsymbol{pmax}_j$	vector defining the minimum and maximum value of j-the component position boundary for swarm initialization
$vmax_j$	absolute maximum value of j-th component velocity for velocity limit condition
v_{lim}	velocity limit modifier which defines the ratio of $\boldsymbol{pmax}_j$ to $vmax_j$
$p_{0,j}$	search space constant for j-th component which defines the $\boldsymbol{pmax}_j$ from the best solution of previous PSO run in refinement procedure

Operators

$\boldsymbol{a}$	boldface lowercase indicates vector
$\boldsymbol{A}$	boldface uppercase indicates matrix
$\boldsymbol{A}\{a, b\}$	component of matrix (or vector) $\boldsymbol{A}$ in row a column b
$\mathring{\boldsymbol{a}}$	normalized/unit vector
$\lvert\boldsymbol{A}\rvert$	determinant of matrix $\boldsymbol{A}$
$\lvert\boldsymbol{a}\rvert$	norm/magnitude of vector $\boldsymbol{a}$
$\lvert a\rvert$	absolute value of scalar a
$\dot{a}$	time derivative of a
$\breve{a}$	measured value of a
$\hat{a}$	computed/estimated value of a
$\tilde{a}$	estimation error of a
$\bar{a}$	mean value of a
$\boldsymbol{S}_n(\boldsymbol{a})$	$n \times n$ skew matrix of vector $\boldsymbol{a}$
$\boldsymbol{A}^{-1}$	inverse of matrix $\boldsymbol{A}$
$\boldsymbol{A}^{\top}$	transpose of matrix (or vector) $\boldsymbol{A}$
$\boldsymbol{a} \cdot \boldsymbol{b}$	dot product of vector $\boldsymbol{a}$ and $\boldsymbol{b}$
$\boldsymbol{a} \times \boldsymbol{b}$	cross product of vector $\boldsymbol{a}$ and $\boldsymbol{b}$
$\text{rand}(a, b)$	random number in the range of $[a, b] \in \mathbb{R}$
$\text{randi}(a, b)$	random integer in the range of $[a, b] \in \mathbb{Z}$
$\text{diag}_n(\boldsymbol{A})$	returns the n-th diagonal element of $\boldsymbol{A}$
$\text{sign}(a)$	returns the sign of a
$\sum_{i=1}^{n}$	sum over index $i = 1 \ldots n$
$\arg\min_a \{f(a)\}$	returns the value of a which minimizes $f(a)$

Abbreviations

ADCS	attitude determination and control system
AMR	anisotropic magnetoresistive
ARW	angular random walk
COTS	commercial off-the-shelf
CMG	control moment gyroscope
CVG	coriolis vibratory gyro
DSS	digital sun sensor
EM	engineering model
EKF	extended Kalman filter
FOV	field of view
HIL	hardware-in-the-loop
IGRF	international geomagnetic reference field
KF	Kalman filter
LEO	low Earth orbit
LEOP	launch and early orbit phase
LSB	least significant bit
MCP	magnetic cleanliness program
MDM	magnetic dipole model
MEMS	microelectromechanical systems
OBC	on-board computer
PSD	power spectral density
PSO	particle swarm optimization
RIST	range of interest selection technique
RMM	residual magnetic moment
RRW	rate random walk
SVD	singular value decomposition
UKF	unscented Kalman filter

1 Introduction

1.1 Background

Nanosatellites, which typically weigh 1 to 10 kg, have become a popular platform for developing low-cost space missions using commercial off-the-shelf (COTS) components and simplified design process. The most popular nanosatellite form factor so far is CubeSat, a $10 \times 10 \times 10\,\mathrm{cm}^3$ satellite platform with typical weight of 1 kg for one unit (1U-CubeSat), which can be stacked together to form a multiple-units CubeSat [11]. Initially, CubeSat was developed for educational purposes, although recent developments show that more CubeSats are launched with actual science and practical missions for their primary objectives as the platform matures [68]. Consequently, the demand for performance reliability also increases in order to fulfill more advanced mission requirements. The focus of this thesis is the development of calibration methods for attitude determination and control system (ADCS) sensors and actuators typically used in nanosatellites. The performance of ADCS could greatly affect the overall performance of the satellite, especially for instruments that depend on the accuracy of the spacecraft attitude (e. g., pointing a payload, solar panels, or antenna to a target).

The system under study is mainly based on the ADCS of Aalto-2, a 3U-CubeSat which is currently being developed in Aalto University. However, the calibration procedure was tested on Aalto-1 engineering model (EM), which is developed in parallel with Aalto-2. The instrument configuration of the two spacecraft is similar, and the specific Aalto-1 instruments that were tested will demonstrate the results of the calibration procedure equally well. The sensors are two-axis Sun angle sensors, three-axis magnetometers, and three-axis gyros; and the actuator is a three-axis magnetorquer, all of which are COTS components with some in-house developed parts.

Calibration of sensors and actuators, in the context of estimation of model parameters, can be performed in two ways: preflight calibration, which is performed in a lab environment in its design and integration phase, and on-orbit calibration, which is performed when the spacecraft is in its operational phase in space. With the demonstrated capabilities of various computational calibration algorithm, Springmann et al. argues that on-orbit calibration can (eventually) replace the need for preflight calibration [80], and consequently, the requirement for highly precise testing facility can be minimized: current trend is to replace the requirements for high-precision testing facilities and rigorous procedure in performing calibration and system simulation with more autonomous computational approach and software simulation [8, 51, 56, 83, 84]. However, it is important to note, that for a 'perfect' calibration algorithm to be developed, an exact knowledge of the ideal mathematical model for the corresponding instrument is required, and ground-based calibration process is still needed to verify it to a certain extent. Thus, it can be argued that preflight and on-orbit (real-time) calibration procedure is complementary by nature, rather than one fully substituting the other.

Another importance in developing both preflight and on-orbit calibration methods

is that it can reduce the risk factors in space missions. Studies by Hoffman et al. have shown, based on several long term space mission data, that faults and anomalies in space mission operations are dominated by software-related sources. Their studies also indicated that the majority of them occurred in Launch and Early Orbit Phase (LEOP) and critical mission phases, correlated to the increase in activity and execution of new procedures [31, 32]. Therefore, this redundancy in procedures can help reduce faults in development process and provide options in the case of fault isolation and failure analysis.

This thesis focuses on the development of preflight calibration procedure for Aalto-1 and Aalto-2 EM. However, some of the algorithms is developed with possible on-orbit calibration capabilities. Different advantages and limitations associated with preflight calibration is discussed later in this thesis. After the calibration process, evaluating the results requires a verification method: this includes software-based simulation and hardware-based simulation, and can be implemented in the instrument- or system-level of the spacecraft.

1.2 Structure Overview

The goal of this thesis is to develop calibration procedures that are suitable for preflight calibration of CubeSat ADCS. For that, this thesis is structured as follows: First, section 1 introduces the global overview of the thesis content. Next, section 2 explains an overview on satellite ADCS and the role of calibration process in affecting the performance of the instruments in the system. Then, section 3 explains the basic theory of attitude system instruments calibration. This includes the definition of different terms used for categorizing different calibration approaches and algorithms. It also explains the definition of the mathematical models of the ADCS instruments, including their underlying assumptions/simplifications and an overview on the respective state-of-the-art calibration methods from literature. Section 4 introduces the common estimation concepts and algorithms developed in this thesis, covering their basic characteristics and some insights on the improvements of the algorithms applied in this thesis. Section 5 explains the mission requirements of the first generation Aalto-series nanosatellites, which are used as the groundwork for this thesis, as they affect the simulation and experiment setups covered in sections 6 and 7. Specifically, more specific version of optimization algorithm developed for magnetic instruments calibration procedure is explained in section 6, while the actual tests setups and procedures as well as their results are reported and analyzed in section 7. Finally, key findings in the development process are summarized: the conclusions as well as suggestions for future work are presented in section 8.

2 Attitude Determination and Control System

The attitude of a spacecraft is defined as its orientation in space relative to a reference frame [88: 1]. To perform its mission, some requirements are imposed on its attitude: e. g. stabilizing the spacecraft from a tumbling condition, orienting the solar panels towards the Sun, pointing its payloads towards certain target, or periodically changing its orientation relative to the Sun for thermal control. There are many possibilities for these ADCS operation modes, and they are derived from the analysis and design of each mission. Achieving these requirements requires the ability in estimating the current attitude of the spacecraft and controlling it in order to stabilize and point the spacecraft to a certain direction, which is the role of the spacecraft ADCS. This section gives an overview of the concepts and mechanism of a spacecraft ADCS in order to understand the role of its instruments calibration process in affecting its performance.

2.1 Coordinate System

An important concept in spacecraft ADCS is the coordinate system which will become the basis for every attitude representation in the system. This covers an overview of the reference frames commonly used to define spacecraft attitude and the way the attitude information is represented mathematically. Several reference frames, usually denoted as right-handed triads in three-dimensional space, are commonly used in spacecraft ADCS and some of them are especially useful to understand the calibration process of different ADCS instruments. A brief overview of those reference frames and attitude representations are presented; more exhaustive lists and explanations on those concepts are available in literature. [52, 88]

Spacecraft-specific reference frame. A reference frame defined by its origin fixed at a point relative to the spacecraft body. Also called spacecraft-fixed [88: 26] or spacecraft body [52: 31] frame, this reference frame is used as the reference for aligning various spacecraft components. However, it is also useful for the purpose of ADCS instruments calibration by deriving this concept for each individual component itself: every instrument, where necessary, has its own reference frame. This is necessary, because some calibration parameters such as misalignment factor can be represented as the relative orientation between the instrument reference frame with the spacecraft body reference frame.

Inertial reference frame. A reference frame fixed in the inertial space, where Newton laws are valid, and can be extended to any frame that is moving at constant velocity without rotation with respect to an inertial frame [52: 31–32]. Because of this characteristic, the basic form of spacecraft dynamic equations are defined in the inertial frame. Best approximation for a true inertial frame is the celestial reference frame, where the frame is defined relative to some distant 'fixed' stars, although other less-accurate approximations such as the geocentric inertial frame is accurate enough for attitude analysis [52: 31–32, 88: 26–28].

Earth-fixed reference frame. Also called the Earth-Centered/Earth-Fixed Frame (ECEF), this is a reference frame centered at the Earth center of mass with its axes aligned with the mean North pole and Earth prime meridian, forming a right-handed system [52: 32]. This reference frame rotates along with Earth, and is useful as the base reference frame in defining earthbound phenomenon. An example is for mapping the geomagnetic field, which is a useful information for attitude estimation using magnetometer data.

Orbit-specific reference frame. A reference frame defined in relation to the spacecraft orbit, centered in the spacecraft frame. Commonly known as the Local-Vertical/Local-Horizontal (LVLH) frame or the roll-pitch-yaw system [52: 28–29, 88: 36–37]. The frame axes are aligned with the nadir vector (pointing directly to the center of the Earth) and along the spacecraft orbital velocity, forming a right-handed system.

2.2 Attitude Representations

The relative orientation between different reference frames can be represented with different mathematical representations: common representations described in literature are *rotation matrix/direction cosine matrix*, *euler angle*, *euler axis/angle*, and *quaternion* [52: 41–59, 88: 410–420]. Rotation matrix is the most general form of attitude representation, because it is simply an orthogonal rotation matrix which can be operated directly with matrix algebra. Euler angle is a representation of attitude transformation using three sequences of rotation around a defined triad axes. Euler axis/angle is a representation of attitude transformation with a single rotation around one particular axis. Quaternion is a 4-elements-based representation of attitude transformation which has its own algebra [43]. Other forms of attitude representation also exist in literature, such as Gibbs vectors/Rodrigues parameters [52: 48–52, 88: 416].

Each representation has its own advantages and shortcomings: Rotation matrix is convenient because it works directly with matrix operation, but contains redundancy and requires more computation since it contains nine elements (3×3 matrix), and computation inaccuracies may introduce singularity and loss of orthogonality. Both euler angle and euler axis/angle have clear and direct physical representations, making them useful in representing the input-output of the attitude system in an intuitive form for the users to read. However, they need to be converted into other form for computation and prone to singularity problems and non-unique solutions. Quaternion has its own class of mathematical operations which share some properties with matrix algebra while reducing redundancy (only 4 elements compared to rotation matrix 9 elements), making it more favorable for mathematical computations. However, interpreting a quaternion form into a physically meaningful information is not intuitive—it has to be converted into other representation form first. [52: 41–59, 88: 410–420]

For this thesis, relative attitude between the reference frame of the ADCS instruments and the spacecraft body is represented with rotation matrix. This is

because the direct compatibility with matrix algebra simplifies the implementation of calibration algorithm.

2.3 ADCS Mechanism

The task of a spacecraft ADCS is to estimate the current attitude of the spacecraft by combining the information gathered from its sensors, and control the attitude of the spacecraft by giving commands its actuators. This mechanism forms a closed-loop control system summarized in fig. 1.

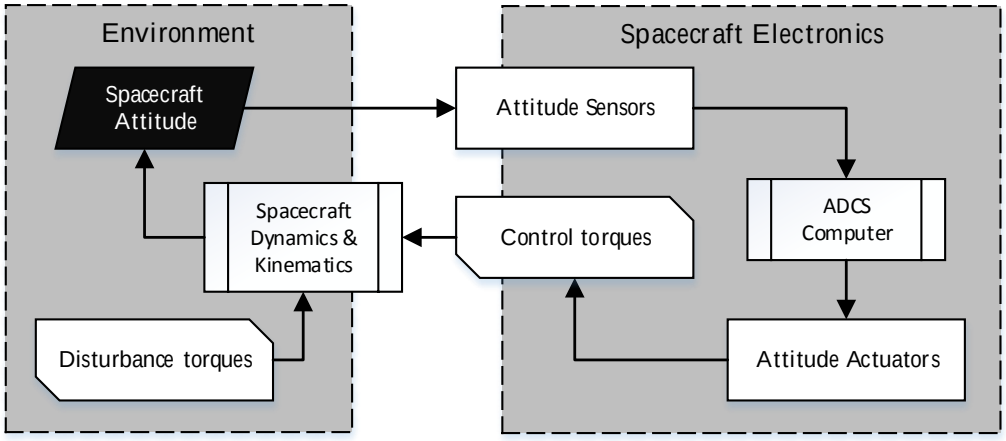


Figure 1: Summary of ADCS Mechanism.

2.3.1 Spacecraft Dynamics & Kinematics

For a rigid-body spacecraft, its dynamics can be expressed with Euler's equation [88: 521–523]:

$$\boldsymbol{I}\dot{\boldsymbol{\omega}} = \boldsymbol{S}_3(\boldsymbol{\omega})\boldsymbol{I}\boldsymbol{\omega} + \boldsymbol{\tau}_{\text{ctrl}} + \boldsymbol{\tau}_{\text{dist}}, \tag{1a}$$

where $\boldsymbol{I}$ is the spacecraft moment of inertia matrix, $\boldsymbol{S}_3(\boldsymbol{\omega})$ is the 3×3 skew matrix of the spacecraft angular rate $\boldsymbol{\omega}$

$$\boldsymbol{S}_3(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_w & \omega_v \\ \omega_w & 0 & -\omega_u \\ -\omega_v & \omega_u & 0 \end{bmatrix} \text{ with } \boldsymbol{\omega} = \begin{bmatrix} \omega_u \\ \omega_v \\ \omega_w \end{bmatrix}, \tag{1b}$$

$\boldsymbol{\tau}_{\text{ctrl}}$ is the control torque generated by the actuators, and $\boldsymbol{\tau}_{\text{dist}}$ is the disturbance torque from various sources. Then, the spacecraft angular velocity $\boldsymbol{\omega}$ is propagated into the spacecraft kinematics with (written in simplified, small-angle approximated quaternion form) [88: 511–512]

$$\boldsymbol{q}(t + \Delta t) \approx \left[\mathbf{1}_4 + \frac{1}{2}\boldsymbol{S}_4(\boldsymbol{\omega})\Delta t\right] \boldsymbol{q}(t), \tag{2a}$$

where $\boldsymbol{q}$ is the quaternion form of the attitude information, $\mathbf{1}_4$ is a 4×4 identity matrix, and $\boldsymbol{S}_4(\boldsymbol{\omega})$ is the 4×4 skew matrix of $\boldsymbol{\omega}$

$$\boldsymbol{S}_4(\boldsymbol{\omega}) = \begin{bmatrix} 0 & \omega_w & -\omega_v & \omega_u \\ -\omega_w & 0 & \omega_u & \omega_v \\ \omega_v & -\omega_u & 0 & \omega_w \\ -\omega_u & -\omega_v & -\omega_w & 0 \end{bmatrix}. \tag{2b}$$

Note that the angular rate vector $\boldsymbol{\omega}$ is defined in the inertial reference frame, denoted with the triad $[u, v, w]^\top$.

Disturbance torques. As shown in fig. 1, disturbance torques will affect the spacecraft dynamics as $\boldsymbol{\tau}_{\text{dist}}$ in eq. (1a). The dominant sources of disturbance torques in a space environment vary depending on the orbit of the spacecraft. However, for Earth orbit in general, the dominant disturbance sources are gravity gradient, geomagnetic field, solar radiation pressure, and aerodynamic drag [88: 566–576].

2.3.2 Attitude Instruments

In fig. 1, the spacecraft electronics responsible for ADCS operation consists of different instruments, i. e. attitude sensors, attitude actuators, and the ADCS computer, each with its specific function. The sensors retrieve attitude information from various reference targets in the environment, which in turn will be processed by the ADCS computer, producing an estimate of the spacecraft attitude. Then, the ADCS computer will compute the attitude error of the spacecraft relative to the target attitude and send commands to the actuators, controlling it based on a control law in order to correct the attitude error.

Attitude sensors. Attitude sensors are instruments that are capable of retrieving the attitude information of the spacecraft, with two main categories: *reference sensors* and *inertial sensors*. Reference sensors detect the direction of a certain reference target (e. g., Sun, stars, or ambient magnetic field) in the sensor frame. Inertial sensors detect the angular velocity/linear acceleration experienced by the sensor: gyroscopes detect angular velocity, and accelerometers detect linear acceleration. In spacecraft ADCS application, gyroscopes are more commonly used since the gravity acceleration is negligible in orbit. [28: 310–321]

In spacecraft ADCS, reference and inertial sensors are usually used together to obtain a complete attitude information because some reference sensors cannot give full attitude information on its own [28: 310]. For example, a Sun sensor cannot give information on the rotation around the Sun direction vector (this is true for any line-of-sight reference sensor that detects a single point target, e. g. Sun sensors and magnetometers), while a gyroscope requires information on initial attitude and is prone to drift error.

Attitude actuators. Attitude actuators are instruments that can exert torque on the spacecraft, either as *active actuators* or *passive actuators*. Active actuators

require power from the spacecraft in order to continuously make decisions and operate the actuators, e. g. thrusters, magnetorquers, control moment gyroscopes (CMGs) and momentum/reaction wheels. Passive actuators, in the other hand, do not require power from the spacecraft since they utilize environmental torques, whose effect are usually permanent, e. g. permanent magnets, solar sails, aerodynamic flaps and gravity-gradient. These torques affect the spacecraft attitude as governed by the dynamics and kinematics given in eqs. (1) and (2). [88: 502–509]

Besides from its active/passive criteria, attitude actuators can be differentiated by the source of the generated torques, either as *external torque actuators* or *internal torque actuators*. External torques are generated by the means of interaction with external entities, resulting in a change of net momentum of the spacecraft. Examples of external torque actuators are thrusters (ejecting propellant mass in one direction, resulting in a change of the spacecraft momentum in the opposite direction), magnetorquers (torques generated by the interaction of a magnetic dipole fixed to the spacecraft body with the local magnetic field), and solar sails (torques exerted directly from solar wind pressure). Internal torques are generated by the means of momentum exchange between components of the spacecraft, resulting in no net change in spacecraft momentum. This also means that the spacecraft is not a single rigid body anymore when the dynamics is broken down to each component, although the total dynamics of the spacecraft can be simplified into a rigid body depending on the actuator mechanism. Additionally, since the actuators generate torque by storing and releasing angular momentum of the spacecraft through a spinning mass in the spacecraft frame, the actuators may saturate (when the mass spinning speed is limited physically) as disturbance torques and energy loss from frictions in the spinning mass mechanism build up over time. Releasing this built up momentum requires *momentum dumping* using external torques. Examples of internal torque actuators are reaction wheel and CMGs. [52: 99–111]

ADCS computer. ADCS computer handles the attitude information from the attitude sensors and uses it for estimating the full attitude of the spacecraft and controlling the actuator in order to correct the spacecraft attitude to a given direction. Its functions can be handled by a dedicated processor (with redundancy if needed) or directly by the on-board computer (OBC) in order to reduce system complexity, mass, volume, power consumption, and costs. However, less redundancy and centralized tasking will also reduce the system reliability in the event of failure.

Calibrating ADCS instruments. The closed-loop control system in fig. 1 implies that the accuracy of the ADCS in estimating and controlling the spacecraft attitude relies on the attitude sensors and actuators in perceiving the actual state of spacecraft dynamics from the environment to the ADCS computer. In reality, sensors and actuators contain errors: sensors might provide inaccurate information on the spacecraft attitude, while actuators might not behave exactly as it is commanded by the ADCS computer. All of these errors propagate in the control loop, resulting in errors of the spacecraft desired attitude. Correcting these error sources requires

calibration of the ADCS sensors and actuators, which is explained in the next sections of this thesis.

3 Attitude System Instruments Model and Calibration Methods

3.1 Definitions and Nomenclature of Calibration Methods

Calibration, in the context of this thesis, is a process to produce the best accuracy possible from an instrument by comparing the measurements of the instrument against a reference value and formulating a model which will fit those measurements into the reference value. Different methods and algorithms are required for different type of instruments because of the different technology implemented in the instruments. As the technology advances, development of novel methods and improvement in existing methods to perform calibration is needed. A global workflow of calibration process in general is described in fig. 2.

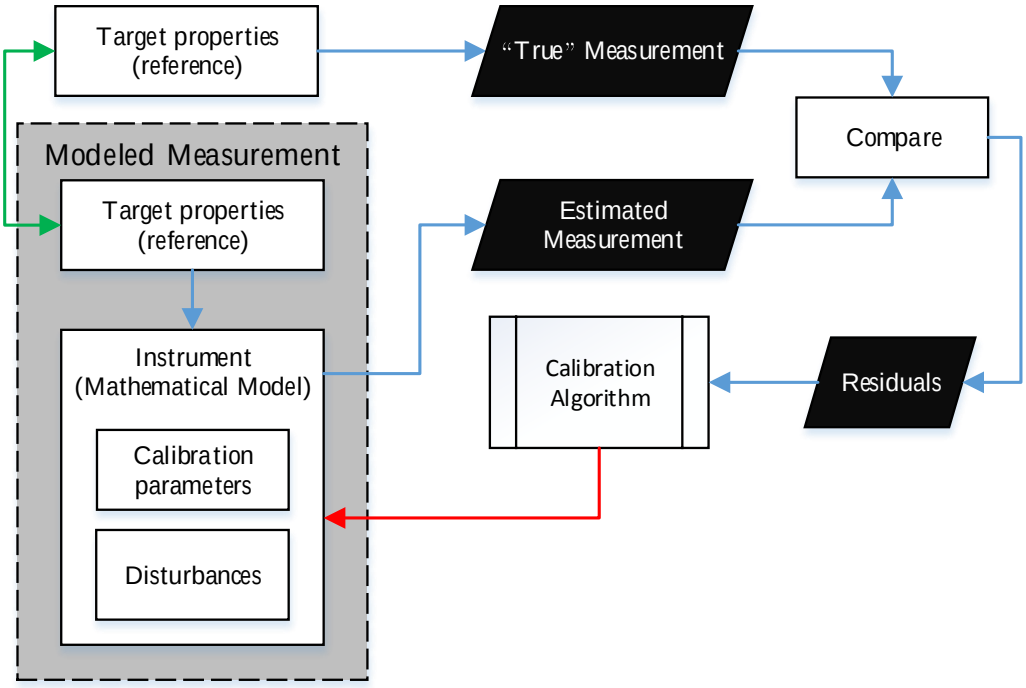


Figure 2: Diagram of general calibration process. The task of calibration algorithm is updating the model of the instrument and the disturbances involved in order to minimize the residuals/errors between the estimated and the 'true' reference measurements.

This calibration concept also applies in spacecraft ADCS technology. However, space application sector has a characteristic which sets it apart: that the system will be operating in space environment. In most cases, this means that a system which is already on orbit has to be physically self-sustainable until the end of its lifetime—in the context of calibration process, this condition prevents one from performing calibration in a lab-controlled environment once the system is on orbit, while in contrast, calibration performed in a ground-based setup with lab-controlled

environment cannot fully simulate the actual condition on orbit. Because of this condition, this thesis uses three different terms to categorize different calibration approaches: preflight calibration, on-orbit calibration, and real-time calibration. These terms are widely used in literature [20, 29, 34, 76, 77, 79, 81]; however, they are not formally defined and some authors might use these terms in a slightly different context from the other. Thus, for this thesis, the definitions of those terms shall be clarified here.

Preflight calibration: Calibration which is performed on ground, thus providing some options on the calibration approach: *i*) before and/or after integration; *ii*) by determining calibration parameters and/or by physically modifying the instrument; and *iii*) if necessary, in a controlled lab environment. These options can be useful for distinguishing different calibration parameters: the ones inherent to the instrument and the ones affected by the environment. The downside of a preflight calibration is that it cannot fully emulate the behavior of the system in space environment, where some characteristics of the instrument may change.

On-orbit calibration: Calibration which is performed when the spacecraft is on orbit. Assuming that physical access to the spacecraft is no longer possible, this limits the calibration to be performed only after integration outside of a fully controlled environment by making adjustment to the calibration parameters. The advantage of on-orbit calibration is that by using on-orbit measurement data, it will take into account the changes in the instrument characteristics once it is on orbit.

Real-time calibration: A term used to differentiate a class of calibration process that is performed simultaneously as measurements are taken in real time. Therefore, it can be performed both during preflight test and on orbit—obviously, the goal is on-orbit implementation, although ground-based implementation is useful to verify the results. In practice, it is performed with estimation filters to directly estimate the state of the system, particularly filters with recursive algorithms (as opposed to batch algorithms) and low computation cost, thus the ADCS computer can compute it in real time. Note that eliminating time-varying errors which are modeled as random noise process is only possible by using estimation filters, and it normally requires a priori knowledge on the random noise process characteristics, i. e. the mean and covariance, which can be obtained from preliminary calibration steps.

The calibration algorithms developed in this thesis are focused on preflight calibration procedures, while real-time calibration algorithms are only discussed in basic theory since the estimation filter for Aalto-2 ADCS is under development outside of this thesis work. However, some of the calibration algorithms developed in this thesis are also designed to qualify for on-orbit application. Nonetheless, it is useful to predefine these terms for the purpose of systematic analysis of different methods discussed in this thesis. Table 1 summarizes the relationship between these terms.

Table 1: Relationship between the terms widely used to categorize different calibration methods. Note that the categories can overlap each other.

PREFLIGHT *Performed before the spacecraft is launched*	ON-ORBIT *Performed after the spacecraft is launched*	REAL-TIME *Performed simultaneously as measurement data are taken*
Can be performed before or after integration; on ground.	Can only be performed after integration; in space.	Works in both; main purpose for on-orbit application.
More options and physical access to isolate errors of different sources.	No physical access to modify the hardware.	Can be either.
Analysis of ground-based data; errors in space environment only simulated.	Analysis of flight data; errors from space environment is observed.	Can analyze both; requires some knowledge on error characteristics.

Besides the different terms for different calibration approaches, this thesis uses some commonly used terms for classifying different types of estimation algorithms, namely *batch* and *recursive* estimation algorithm. [7, 88: 447–470]

Recursive estimation algorithm: A class of algorithm which process information recursively every time a new information is taken. In the context of calibration algorithm, this class of algorithm will estimate the unknown calibration parameters every time a new measurement is taken by combining it with the previous knowledge in the last recursion. Because of its capability of estimating parameters simultaneously as the measurement is taken since the first measurement (and the accuracy increases as the estimated value converges to the right value), this algorithm class is more compatible for real-time calibration approach.

Batch estimation algorithm: A class of algorithm which process information in batch. In the context of calibration algorithm, this class of algorithm estimates the unknown calibration parameters using a set of measurements from the measurement history at once, in contrast to the recursive estimation algorithm which only requires the last a priori estimate and current measurement. Its accuracy increases as the number of measurements included in that batch is increased, and only requires to be executed once to obtain the final estimate.

Note that technically, batch estimator may contain iteration in its algorithm, while the recursion process in recursive estimator can also be implemented iteratively. Thus, the differentiating feature between the two algorithms is the way the algorithm handles the information into and out of the estimator. Naturally, executing batch estimation algorithm is slower than the recursive one since it process more information (the whole measurement history) every time the algorithm is executed, thus making it less suitable for real-time calibration approach.

3.2 Sun Sensor

Sun sensors are attitude sensors which detect the relative position of the Sun in the sensor reference frame by detecting the light radiated from the Sun. There are several basic classes of Sun sensor (i. e., analog sensors, Sun presence detectors, and digital sensors [88: 156]) with different architecture and technology, and the way they are used depends on the nature of the mission (e. g., presence of albedo, distance from the Sun, eclipse period). The Sun position information can be used for many purposes (e. g., positioning the solar cell arrays, protecting sunlight-sensitive equipments such as Star tracker, information for thermal control), thus for Earth orbiting missions where the angular radius of the Sun is relatively constant along the orbit, Sun sensors are the most widely used attitude sensor [88: 155–156].

Since the position of the Sun is retrieved by detecting sunlight, individual Sun sensor can only detect the Sun position in a limited field of view (FOV). Analog and digital Sun sensors provide the Sun angle in analog or digital signal, respectively, while Sun presence detectors provide a constant output signal when the Sun is in their FOV. Sun sensors also come in different resolutions from several degrees to the order of less than arc-second [44: 373, 88: 156]. Very coarse Sun sensor can be implemented by processing the current measurements from the solar panels on the spacecraft and is capable of achieving an accuracy of 7° as a standalone sensor, although it requires extensive filtering process [61]. Similar study on solar cell-based coarse Sun sensor combined with magnetometer reading can achieve much better accuracy [4]. A more accurate miniaturized fine Sun sensor with an improved mathematical model proposed by Chang et al. can achieve an accuracy of 0.054° [16]. In the other hand, a state-of-the-art micro-digital Sun sensor with CMOS-based integrated system can achieve an accuracy of 0.01° with a very low power consumption [10, 93].

3.2.1 Sun Sensor Model

The mathematical model of Sun sensor depends on the type of technology and architecture implemented in the Sun sensor. The Sun sensor used in Aalto-2 ADCS is E910.86 two-axis digital Sun sensor (DSS) from ELMOS [25], integrated with ATMega328 microcontroller for managing the data communication. It provides the sunlight elevation angle on xz- and yz-plane (α and β, respectively) in the sensor coordinate frame (sensor surface normal is in $+z$ direction) as well as the light intensity and the sensor temperature as digital signal [1, 39]. The convention for Sun sensor reference frame is depicted in fig. 3.

The transformation of the Sun angle measurement α and β into Sun vector $\boldsymbol{v}_{ss}$ in the sensor coordinate, which can be translated later into spacecraft attitude information, is governed by [88: 227]

$$\boldsymbol{v}_{ss} = \begin{bmatrix} X_{ss} \\ Y_{ss} \\ Z_{ss} \end{bmatrix} = \frac{1}{(\tan^2\alpha + \tan^2\beta + 1)^{1/2}} \begin{bmatrix} \tan\beta \\ \tan\alpha \\ 1 \end{bmatrix}, \tag{3}$$

where X_{ss}, Y_{ss}, and Z_{ss} are the x-, y-, and z-axis components of the Sun vector. The sensor raw data represents the angle α and β given in fig. 3 in bit counts as

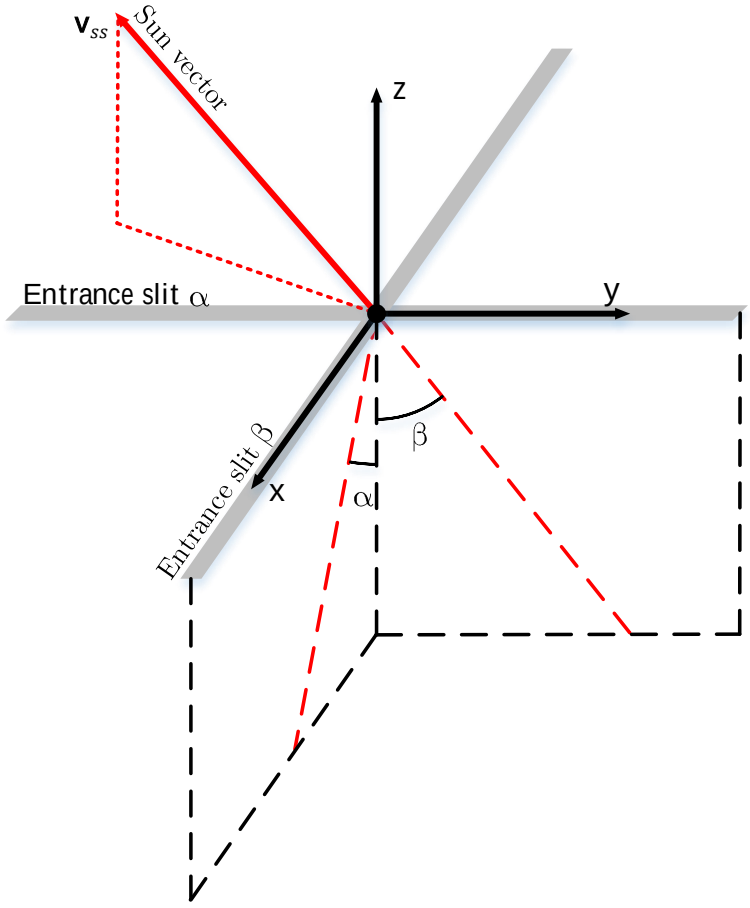


Figure 3: Definition of Sun sensor reference frame. The Sun vector points to the direction of the Sun, and the sunlight will enter through the entrance slits which will project the light to fall on different axes of the sensor detector.

N_α and N_β, respectively, and the relationship is given as

$$\begin{bmatrix}\alpha\\ \beta\end{bmatrix} = \frac{\mathbf{FoV}}{2^m}\begin{bmatrix}N_\alpha\\ N_\beta\end{bmatrix} + \mathbf{FoV}_0, \tag{4}$$

where m is the bit size of the measurement, while $\mathbf{FoV}$ and $\mathbf{FoV}_0$ are the total FOV and smallest angle in the FOV for each axis in degrees, respectively. Therefore, eq. (4) shows that the the size of measurement data defines the sensor resolution in degree per least significant bit (LSB). Equation (4) can be expanded by including error in the form of scale factor, bias, and nonorthogonality constant, resulting in

$$\begin{bmatrix}\alpha\\ \beta\end{bmatrix} = \frac{\mathbf{FoV}}{2^m}\boldsymbol{S}_{ss}\begin{bmatrix}N_\alpha\\ N_\beta\end{bmatrix} + \mathbf{FoV}_0 + \mathbf{off}_{ss}, \tag{5}$$

where $\mathbf{off}_{ss}$ is the offset vector for each axis and $\boldsymbol{S}_{ss}$ is the scale factor and nonorthogonality error matrix. Note that the E910.86 DSS Sun angle reading is represented with 6-bit data ($m = 6$) for each axis including data header, resulting in a total of

$2^m - \text{header} = 56$ data points per axis in the FOV range of $-75°$ to $75°$ [1, 39]. More comprehensive descriptions of different Sun sensor models with different architectures are available in literature [88: 218–230].

3.2.2 Sun Sensor State-of-the-art Calibration Methods

Fitting the measurement data into the mathematical model from eq. (5) in order to obtain the appropriate calibration parameters can be achieved with different methods. Many studies have been conducted and each method depends on the architecture of the Sun sensor. To name a few, Sedlak et al. devised a procedure to estimate each calibration parameters for FSS [67]; Appel combined coarse Sun angle from solar cell-based coarse Sun sensor with magnetometer data using extended Kalman filter (EKF) in order to calibrate the Sun angle [4]; Maqsood et al. used a simple calibration table to assign true sensor heading for a low-cost quadphotodiode-based Sun sensor [51]; and Strietzel used multi-line and spline interpolation method encased in a Fuzzy logic for searching the actual sensor heading of a more accurate quadphotodiode architecture [84].

3.3 Gyroscope

Gyroscope is an inertial sensor that measures the angular velocity experienced by the instrument, which can be classified into different types based on its physical mechanism [28: 319–321, 52: 140–143]: *i*) *mechanical gyros*, which use the tendency of a spinning mass angular momentum to remain fixed in inertial space; *ii*) *optical gyros*, which operate by calculating the phase difference of two reference laser light traveling in opposite direction in a rotating closed path; and *iii*) *Coriolis vibratory gyros* (CVGs), which detect movement of standing waves on a rotated vibrating surface due to the Coriolis force.

Mechanical gyros are the first type of gyro developed and the only type existed before 1980s, although many moving parts involved in them are subject to failure [52: 140–143]. This is a problem especially for spacecraft application because of the difficulty of maintenance. In the other hand, optical gyros, which are developed later, generally have the best performance and reliability but relatively cost higher. Numerous research have been conducted to miniaturize different gyroscope technologies using microelectromechanical systems (MEMS) manufacturing technology, started in 1980s with the development of micromachined vibrating gyroscopes [45]. So far, MEMS devices are popular for smaller spacecraft because of their small package, low cost, and low power consumption level.

3.3.1 Gyroscope Model

The gyroscope used in Aalto-2 ADCS is MPU-3300, a MEMS CVG from InvenSense [35], and its mathematical model is described as [5, 83, 87]

$$\check{\boldsymbol{\omega}} = \boldsymbol{S}_g \boldsymbol{\omega} + \mathbf{off}_g + \boldsymbol{\eta}_{\mathrm{ARW}}, \tag{6a}$$

where $\breve{\boldsymbol{\omega}}$ is the gyroscope measured angular rate; $\boldsymbol{S}_g$ is the gyroscope total scale error (3×3 matrix containing scale factor and misalignment errors); $\boldsymbol{\omega}$ is the true angular rate; $\mathbf{off}_g$ is the angular rate bias with drifting rate $\boldsymbol{\eta}_{\text{RRW}}$

$$\dot{\mathbf{off}}_g = \boldsymbol{\eta}_{\text{RRW}}, \tag{6b}$$

which is also known as the rate random walk (RRW); and the measurement noise $\boldsymbol{\eta}_{\text{ARW}}$, also known as the angular random walk (ARW). Scale error and rate bias of the gyroscope are usually considered to be constant. $\boldsymbol{\eta}_{\text{RRW}}$ and $\boldsymbol{\eta}_{\text{ARW}}$ are commonly assumed to be a band-limited white noise, usually zero-mean Gaussian [5, 83, 87].

3.3.2 Gyroscope State-of-the-art Calibration Methods

Ground-based calibration of MEMS gyroscope consists of estimating the calibration parameters and their time-varying counterparts statistical characteristics from eq. (6). Different approaches can be used for estimating these different parameters. Time-invariant part such as initial bias $\mathbf{off}_g$ and scale factor $\boldsymbol{S}_g$ can be solved by rotating a spin table with certain angular rate as the true angular rate reference $\boldsymbol{\omega}$ and fitting them into eq. (6) with least squares method as in [83]. However, characterizing the time-varying components (ARW and RRW) separately is not straightforward.

In order to characterize the random-drift error part separately, a method called Allan variance, where the root means square of these errors is represented as a function of averaging time, is commonly used because of its simplicity and its ability to relate the noise characteristic in power spectral density (PSD) form—which is a Fourier transform pair of the noise covariance. Allan variance plot is usually provided in the instrument datasheet. El-Sheimy et al. explained its mathematical derivation and showed some of its application to characterize different error parameters in typical inertial sensors (i. e., gyroscopes and accelerometers) by fitting the Allan variance log-log plot [69]. Vaccaro proposed an improved method to estimate the noise PSD from the Allan variance plot with best linear unbiased estimator algorithm [87].

Another approach to improve accuracy of gyroscopes is by using data fusion from gyroscopes array to form a virtual gyroscope as proposed by Bayard et al. [8]. Chang et al. and Xue et al. used noise correlation of identical gyroscopes to estimate the virtual gyroscope noise characteristics [15, 94]. In [53], Martin et al. proposed several options of configurations to improve gyroscopes array accuracy: *i*) by using data from identical gyroscopes with opposing orientation to minimize the various errors of gyroscopes; *ii*) by mounting identical gyroscopes perpendicular to each other, thus exploiting the performance differences between in-plane sensor and the out-of-plane sensors; and *iii*) combining output from gyroscopes with different operating range and accuracy, thus expanding the operating range of gyroscopes array.

3.4 Magnetometer

Magnetometer is a sensor which retrieves the direction and strength of local magnetic field and represents it as a vector. For attitude estimation, this local magnetic field

vector information is then compared with the local magnetic field model, which, for Earth orbit, can be calculated from the International Geomagnetic Reference Field (IGRF) model—IGRF model is the *de facto* standard where the model is updated every 5 years [26]. This process is performed with the ADCS computer so that the attitude information of the spacecraft can be calculated in real-time.

Magnetometers have been used in a broad range of applications by utilizing different technologies e. g. Hall Effect sensors, search coils, flux-gate sensors, and magnetoresistive/magnetoimpedance technologies with wide range of sensitivity [88: 180–184]. Those in the higher sensitivity region are usually utilized in space navigation system by using known magnetic field as reference (e. g., magnetic field of Earth), although recent studies have enabled more relaxed sensor requirements in order to lower the costs while maintaining acceptable performance [29, 47, 49, 56, 80, 90, 95]. Another common application of magnetometers in space missions is as the scientific payload for mapping magnetosphere or plasma characteristics in plasma physics. More recent, state-of-the-art hyper-sensitive magnetometer technology such as atomic magnetometer is capable of measuring magnetic field in the order of subfemtotesla ($<$ fT), although it is still limited in laboratory [70]. However, for space application, the accuracy of attitude estimation using magnetometer alone is limited by the accuracy of the local magnetic field model itself, which is about 5° [28: 318].

3.4.1 Magnetometer Model

The magnetometer used in Aalto-2 ADCS is HMC5983 from Honeywell [33], based on anisotropic magnetoresistive (AMR) technology which is popular for nanosatellite navigation system because of its relatively small size, low cost, and low power consumption. Formulating the mathematical model of a magnetometer requires understanding on the various disturbance sources which affect the magnetometer reading.

Common disturbances in AMR magnetometer reading originate from two kinds of sources: instrument and environment disturbance sources [29, 47, 56, 64, 79]. These disturbances affect the true magnetic field vector $\boldsymbol{b}$ as such that the final magnetic field vector measured by the sensor, $\check{\boldsymbol{b}}$, is distorted. A general mathematical model for magnetometer measurement that incorporates these disturbances—broken down to more specific components—is shown in the following equation [64]:

$$\check{\boldsymbol{b}} = \boldsymbol{S}_m \boldsymbol{N}^{-1} \left(1 + \boldsymbol{C}_{\text{si}}\right) \left(\boldsymbol{b} + \boldsymbol{b}_{\text{hi}}\right) + \mathbf{off}_m + \boldsymbol{\eta}_m, \tag{7}$$

where the parameters can be differentiated based on their sources, whether the errors originate from the magnetometer itself or from the environment, as follows:

Instrument error parameters: Parameters which originate from inside the instrument itself. Consist of: *i*) *scale factor*, $\boldsymbol{S}_m$, describes the proportionality of the input to the output for each axis; *ii*) *nonorthogonality*, $\boldsymbol{N}$, a transformation matrix which describes the true x-, y-, and z-axis direction of the sensor in the defined sensor frame; and *iii*) *sensor bias*, $\mathbf{off}_m$, caused by the Wheatstone

bridge offset used in AMR magnetometer architecture. Other errors inherent to the instrument are also present, namely temperature fluctuation and cross-field effect, but these parameters are omitted from eq. (7) for simplicity since they can be mathematically represented by the present parameters [56, 47, 64].

Environment error parameters: Parameters which originate from magnetic sources outside the intended magnetic field that is supposed to be measured and can be divided into two categories based on their magnetic behavior: *i*) *hard iron parameter*, $\boldsymbol{b}_{\text{hi}}$, a vector modeled from permanent magnet dipoles in the environment and will appear as a linear shift of the magnetic field lines; and *ii*) *soft iron parameter*, $\boldsymbol{C}_{\text{si}}$, a matrix modeled from ferromagnetic materials that get induced by external magnetic field (i. e., the ambient magnetic field and hard iron disturbances) and, in effect, will distort the shape of the magnetic field lines.

In a spacecraft system, environment disturbances is dominated by the magnetic field generated by the spacecraft itself: its structure might contain magnetic materials and electrical currents on the electronics also produce magnetic fields—these disturbances are commonly referred as remanent magnetic field or residual magnetic moment (RMM). In this thesis, it will be addressed as RMM from here onwards. Finally, an additive measurement random noise $\boldsymbol{\eta}_m$ is also included in the final reading. Mathematical models other than eq. (7) are possible by assuming different behaviors in the disturbance sources (e. g., hysteresis model), since magnetic interaction are highly nonlinear.

Time properties of errors. Errors inherent in the magnetometer are generally time-invariant. Environment disturbances, however, are sensitive to changes in the condition of the system (e. g., electrical current in the electronics generating magnetic dipole or thermal expansion of the spacecraft structure changing the magnetic response of the structure). These effects in environment disturbances are more pronounced in smaller spacecraft, because the relatively small distance between components promotes stronger magnetic interference (magnetic field strength is inversely proportional to distance in the third power). This is especially true for magnetometers that are installed inside the spacecraft—many spacecrafts have their magnetometers installed on an external boom in order to mitigate this problem. However, this method adds complexity with the boom deployment system and consequently requires more costs, mass, and volume. Literature have provided basic guidelines in evaluating the magnetic properties of a spacecraft (usually referred as magnetic cleanliness program (MCP)) in spacecraft system design process [59].

3.4.2 Magnetometer State-of-the-art Calibration Methods

Significant number of studies have been conducted to minimize the effects of various disturbance sources on the magnetometer reading. The differences between these studies are in their mathematical model of the magnetometer, the geometric approach used, and the algorithm used in estimating the calibration parameters. Appropriate

mathematical model is determined by the basic assumptions on the magnetometer and the environment magnetic behavior, while the geometric approach in calibrating magnetometer can be divided into two domain: heading domain and magnetic domain.

Working in heading domain is an attitude-dependent approach, where a true heading information is required as a reference in order to calibrate the magnetometer—this is the same concept with the case of Sun sensor calibration discussed in section 3.2.2. For magnetometers, knowledge of true information vector of ambient magnetic field is required, and in order to calibrate the magnetometer in its full operational range, a Helmholtz cage setup can be used to perform the calibration in a magnetically clean environment. One way to do this is using a method called "compass swinging" where the magnetometer is leveled and rotated in one axis while maintaining the information on the heading offset from a reference direction [30]. However, the condition of leveling and rotating the sensor requires an accurate test setup in order to minimize errors from misalignments. Springmann et al. has provided a good documentation on the calibration procedure of magnetometer using Helmholtz coil, although this method relies on the reliability of the test facility, because the non-homogeneity of the magnetic field produced by the Helmholtz coil can introduce some error [82].

In contrast, working in magnetic domain is attitude-independent: no heading reference is needed. This method is also called "scalar checking" because only the known ambient magnetic field magnitude information is required as reference [88: 328, 52: 269]. This approach is possible because: *i*) local magnetic field magnitude in the testing facility can be measured directly from a much more accurate magnetometer in a lab-controlled environment—for on-orbit calibration, the local magnetic field magnitude, particularly on lower Earth orbit which is the region of interest in this study, can be calculated from IGRF model; and *ii*) an error-free magnetometer rotated in all direction arbitrarily inside a static magnetic field will produce measurement locus on the surface a perfect sphere, with its center coincides with the origin of sensor reference frame. In a real condition, various disturbance sources will shift and distort the sphere-shaped measurement locus into an ellipsoid [29, 30, 56, 64]. An example of how this measurement locus forms an ellipsoid and its calibrated sphere-shaped locus is depicted in fig. 4.

By exploiting these magnetic domain properties, calibrating magnetometer reading is made possible by fitting sets of measured magnetic field magnitude into the magnetometer model so that the scalar residual is minimized. Foster et al. adopted this approach and used two-step algorithm with least squares method to estimate intermediate parameters which is then derived further to obtain the real calibration parameters [29]; Springmann et al. used similar approach and introduced a new time-varying component as a function of the electronics current measurement [79]; Wu et al. in [91] proposed using particle swarm optimization (PSO) algorithm for solving the parameters estimation problem because of the better theoretical capability of the algorithm in finding global solution without a good initial estimate, and later optimized it with stretching PSO algorithm [90]; while Ali et al. optimized the standard PSO by combining it with a preliminary Range of Interest Selection Technique (RIST) in order to reduce the number of iterations [3].

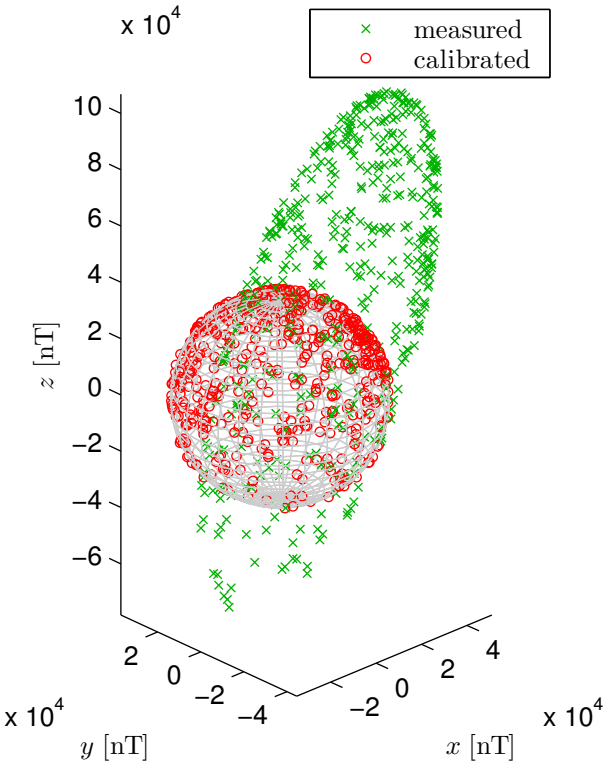


Figure 4: 3-D plot of magnetic field vectors from an example of erroneous magnetometer measurement locus (denoted with ×'s, forming an ellipsoid) and its calibrated/ideal measurement locus (denoted with ∘'s, forming a sphere). The axes represent the magnetic field strength in nT from the magnetometer frame. Note that measurement random noise is not included for viewing clarity.

Another approach in calibrating systematic disturbances inherent to the magnetometer (specifically its bias offset, temperature error, and cross-field effect) is by using the "flipping routine" with a set/reset mechanism commonly used in AMR magnetometers [47, 56, 64]. However, this flipping routine requires external circuitry (fortunately, many modern AMR magnetometer includes a set/reset circuit in their chip) and will increase the power consumption of the magnetometer, which is not favorable for small spacecraft application. Mohamadabadi proposed a workaround on this problem for solving the cross-field error by substituting the flipping routine with a numerical method, although it is based on the assumption of a perfectly orthogonal sensor triad with known anisotropy magnetic field inherent to each sensor axis [56].

3.4.3 Modeling The Spacecraft as Multiple Magnetic Dipoles

Another systematic approach in mitigating the RMM of the spacecraft is by directly modeling these external magnetic sources from MCP analysis, and then either compensating for them numerically from the measurements or placing permanent magnets on the spacecraft structure as such that the RMM is canceled at the measurement point. The latter technique can also be used for modifying the magnetic properties of the spacecraft, in order to manipulate the disturbance torque caused by

RMM of the spacecraft. Note that magnetic disturbance torque is a typical source of dominant disturbance torque for smaller spacecraft in Low Earth Orbit (LEO) [82]. In order to model the magnetic properties of the spacecraft, the spacecraft has to be decomposed into a finite numbers of magnetic dipoles, which are positioned at certain coordinates in the spacecraft frame with certain value of magnetic moment; this model is known as multiple magnetic dipole model (MDM) while the decomposing process is the inverse MDM problem.

The concept of MDM has been long introduced, although the proposed method at the time was geometrically limited and contained error if the limitation was not fulfilled [24]. The concept is based on the assumption that a spacecraft magnetic field can be represented by a finite numbers of magnetic moment dipoles, where their relationship is governed by [38, 82]

$$\boldsymbol{b}_s(\boldsymbol{r}_{ds}, \boldsymbol{m}_d) = \frac{\mu_0}{4\pi} \sum_{d=1}^{n_d} \left(\frac{3\boldsymbol{r}_{ds} \left(\boldsymbol{m}_d \cdot \boldsymbol{r}_{ds}\right)}{\left|\boldsymbol{r}_{ds}\right|^5} - \frac{\boldsymbol{m}_d}{\left|\boldsymbol{r}_{ds}\right|^3} \right), \tag{8a}$$

or in an expanded form (with the frame definition depicted in fig. 5)

$$\begin{aligned} &\boldsymbol{b}_s(\boldsymbol{p}_s, \boldsymbol{p}_d, \boldsymbol{m}_d) = \\ &\quad \frac{\mu_0}{4\pi} \sum_{d=1}^{n_d} \left(\frac{3\left(\boldsymbol{p}_s - \boldsymbol{p}_d\right)\left(\boldsymbol{p}_s - \boldsymbol{p}_d\right)^\top}{\left|\left(\boldsymbol{p}_s - \boldsymbol{p}_d\right)\right|^5} - \frac{\mathbf{1}_3}{\left|\left(\boldsymbol{p}_s - \boldsymbol{p}_d\right)\right|^3} \right) \boldsymbol{m}_d, \end{aligned} \tag{8b}$$

where $s = [1, n_s] \in \mathbb{Z}$ and $d = [1, n_d] \in \mathbb{Z}$ are the indexes of measurement points and magnetic dipoles, respectively; $\boldsymbol{b}_s$ is the theoretical magnetic field at measurement point s; μ_0 is the magnetic permeability of the medium (assumed in space); $\boldsymbol{r}_{ds}$ is the position vector of the s-th measurement points from the d-th dipole; $\boldsymbol{p}_s$ and $\boldsymbol{p}_d$ are the coordinates of s-th measurement points and d-th magnetic dipole, respectively; $\boldsymbol{m}_d$ is the magnetic moment of the d-th dipole; and $\mathbf{1}_3$ is a 3×3 identity matrix. For the algorithms studied in this report, the general task is finding the optimal pairs of dipole position $\boldsymbol{p}_d$ and its corresponding magnetic dipole moment $\boldsymbol{m}_d$ by minimizing the (least squares) error between the estimated magnetic field $\hat{\boldsymbol{b}}_s(\boldsymbol{p}_s, \hat{\boldsymbol{p}}_d, \hat{\boldsymbol{m}}_d)$ and its actual measurement $\check{\boldsymbol{b}}_s(\boldsymbol{p}_s, \check{\boldsymbol{p}}_d, \check{\boldsymbol{m}}_d)$.

Various studies have attempted to optimize this modeling process. Mehlem in [55] improved the concept by using Gauss-Newton algorithm for solving the least squares problem. Until recently, the same author improved the method to address different issues bound to the modeling process [54]. Another method was proposed by Kapsalis et al. to use stochastic algorithm such as PSO to solve the inverse MDM because of its better performance in nonlinear estimation [38]. Carrubba et al. then optimized the PSO implementation by refining the fitness functions that need to be minimized, varying the weighting coefficient, and introducing a recursive refinement step to help the algorithm escapes from local minima [14].

3.5 Magnetorquer

Magnetorquer is a magnetic dipole moment generator controlled by running electric current on a winded coil. A ferromagnetic rod is used as the center of the coil,

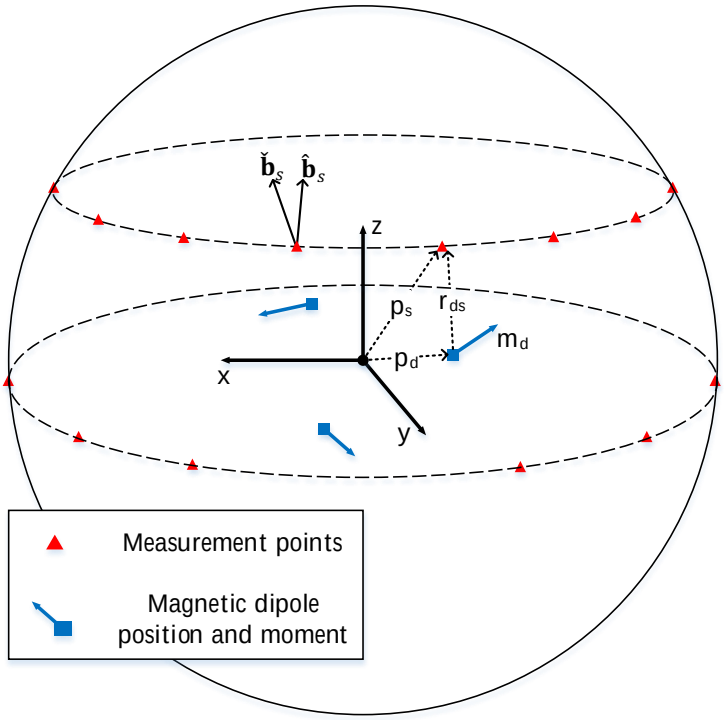


Figure 5: Definition of the magnetic dipole modeling reference frame. The measurement points are denoted with △'s, while the magnetic dipoles positions are denoted with □'s, with their respective magnetic moments denoted with the arrows originating from the magnetic dipoles.

while for an air-core magnetorquer the center of coil is hollow. It is a convenient attitude actuator for spacecrafts because it does not contain any moving parts, therefore not only improving its lifetime and reliability, but also lowering the cost and simplifying the procedure associated with its production. Magnetorquers also generally have lower power consumption compared to other active attitude actuator [44: 369]. However, magnetorquers have some limitations in its application, which are explained in the next section. Magnetorquers generate torque from the interaction between the generated magnetic moment of the magnetorquer and the ambient magnetic field. The generated torque is governed by [88]

$$\boldsymbol{\tau}_{\text{mtq}} = \boldsymbol{m}_{\text{mtq}} \times \boldsymbol{b}, \tag{9}$$

where $\boldsymbol{\tau}_{\text{mtq}}$ is the torque exerted from the magnetorquer; $\boldsymbol{b}$ is the ambient magnetic field; and $\boldsymbol{m}_{\text{mtq}}$ is the magnetic moment generated by the magnetorquer. In turn, the torque produced by the magnetorquer will affect the spacecraft dynamics as defined in eq. (1) with $\boldsymbol{\tau}_{\text{ctrl}} = \boldsymbol{\tau}_{\text{mtq}}$.

3.5.1 Magnetorquer Model

The magnetorquers for Aalto-2 are produced by Hyperion Technologies, fully assembled on the ADCS module with the driver circuitry which will communicate with

the OBC through I2C bus [2]. The magnetic moment generated by magnetorquers can be calculated from [88: 204]

$$m_{\mathrm{mtq},i} = \mu_{\mathrm{core},i} n_{\mathrm{mtq},i} i_{\mathrm{mtq},i} A_{\mathrm{mtq},i}, \tag{10a}$$

where the subscript $i \in \{x, y, z\}$ is the magnetorquer axis index, μ_{core} is the core permeability of the magnetorquer, n_{mtq} is the number of turns in magnetorquer coil, i_{mtq} is the electric current on the coil, and A_{mtq} is the cross-section area of the coil windings. The formula itself is simplified for magnetorquers in general—exact relationship depends on the coil geometry, while core material can introduce hysteresis. The model in eq. (10a) assumes hysteresis is negligible since manufactured magnetorquers with ferromagnetic core usually offer a residual dipole moment of less than 1% of the saturation moment [52: 168]. Generally, magnetic moments produced by COTS magnetorquers are already specified by the manufacturer (Aalto-2 magnetorquers are specified to produce maximum dipole moment of $0.2\,\mathrm{A\,m^2}$ in each axis [2]), hence it is more practical if the constants in eq. (10a) are simplified for calibration purpose into a matrix form of

$$\boldsymbol{m}_{\mathrm{mtq}} \equiv \begin{bmatrix} m_{\mathrm{mtq},x} \\ m_{\mathrm{mtq},y} \\ m_{\mathrm{mtq},z} \end{bmatrix} = \boldsymbol{C}_{\mathrm{mtq}} \boldsymbol{i}_{\mathrm{mtq}}. \tag{10b}$$

For Aalto-2 ADCS, which only has magnetorquer for active attitude control, the nature of the torque produced by magnetorquers defined in eq. (9) imposes several limitations in the magnetorquer operation:

- The magnetorquer can only produce torque perpendicular to the ambient magnetic field direction. This means that at any time instant, the magnetorquer can only produce torque in two axes, limiting the three-axis control of the ADCS in an underactuated condition.
- The magnitude of the torque produced is limited not only by the size and driving power of the magnetorquer, but also the ambient magnetic field strength. Therefore, its applications are mostly limited to LEO, where the ambient magnetic field strength is still sufficient for attitude control purposes (or around other space objects with strong magnetic field).
- The order of torque magnitude which can be produced by a magnetorquer is very small compared to the inertia of the spacecraft. This is a design lock in magnetorquer design, because it can be deducted from eq. (10a) that an attempt to increase its torque requires either an increase in power consumption, number of turns in the coil, or cross-section area of the coil, where the latter two options will increase the inertia of the system again. As a result, the ADCS reaction speed will be very slow compared to ADCSs that use actuators with higher torque capability.
- Additionally, controlling magnetorquer torque requires knowledge of the ambient magnetic field strength. The information of the ambient magnetic field

strength itself is obtained from magnetometers. For spacecrafts with their magnetometers installed near the magnetorquer, the magnetic field generated by the magnetorquer will disturb (and most likely saturate) the magnetometer reading. Thus, in such condition, the magnetorquer cannot be operated during the magnetometer measurements.

Because of its underactuated nature, magnetorquers in spacecraft ADCS are generally operated in cooperation with other actuators such as reaction wheels or CMGs, where the magnetorquer main role is as the momentum dumping device [52: 307]. Nonetheless, different studies have been conducted in order to find a workaround for these limitations—studies have shown, using periodic system analysis, that sufficient controllability in all three axes can be achieved for solely magnetic actuated spacecrafts, given that there is enough variations in the ambient magnetic field direction along the orbit of the spacecraft [48, 72, 89]. This means that in order to generate torque in arbitrary direction in three axes (at least on average in longer timescale), orbit with high inclination is more favored because of the larger variations in Earth magnetic field direction between the equator and polar region [72].

3.5.2 Magnetorquer State-of-the-art Calibration Methods

Based on the different types of relationships defined in eqs. (1) and (9), magnetorquer calibration can be performed either through *magnetic properties analysis* or *dynamics analysis*.

Magnetic properties analysis: Analyzing the magnetic properties of magnetorquers is performed by measuring the magnetic fields generated by the magnetorquer at some arbitrary measurement points and comparing it with the expected theoretical magnetic fields generated by the magnetorquer at those points, which can be computed by combining eqs. (8) and (10b). It is important, for this method to work, that the calibration took place in a magnetically clean space, or at least the ambient magnetic field is known and homogeneous so it can be compensated from the measurements.

Dynamics analysis: Analyzing the dynamics of magnetorquers requires more dedicated testing facility that is capable of, to some degree, simulating the attitude dynamics of spacecraft. The best ground-based setup for this is an air-bearing three-axis gimbal platform, although even a very simple setup using suspending wire is also useful for magnetorquer testing [6]. A Helmholtz coil is also useful in magnetorquer testing for amplifying the ambient magnetic field, so that the magnetorquer torque is amplified for easier observation in lab setup.

4 Estimation and Optimization Algorithms

Various optimization algorithms are available in literature for solving general parameter estimation problem [7, 12]. In this thesis, two main concepts which are applied for the ADCS instruments are the least squares method and PSO. This section explains the basic forms of these algorithms, while the specific implementations are explained in section 6. For the estimation filters which can be implemented as real-time calibration, the theoretical basics are explained for the sake of understanding the complete mechanism of the ADCS, although the implementation is not in the scope of this thesis.

4.1 Least Squares Method

Least squares method is formally defined as a solution to [7: 98]

$$\hat{c}^{\mathrm{LS}}(n_s) = \arg\min_{c} \left\{ \sum_{s=1}^{n_s} \left[z_s - h_s(x, c) \right]^2 \right\}, \tag{11a}$$

where z_s is a series of n_s number of measurements, which can be modeled by the measurement model $h_s(x, c)$ disturbed by noise η_s, as defined by

$$z_s = h_s(x, c) + \eta_s \text{ for } s = 1 \dots n_s, \tag{11b}$$

where the measurement model $h_s(x, c)$ itself is a function of the reference input x and parameters c.

This means that least squares method estimates the value of model parameters c (where c contains an n_c number of unknown parameters) that will minimize the summed square of residuals, which is the difference between the observed/measured quantities, z_s, and the expected/modeled quantities, $h_s(x, c)$. It is also possible to reverse the condition, where z_s is the reference and $h_s(x, c)$ is calculated from measurement data—the manner it is implemented depends on how the data are collected and the mathematical model. Note that $h_s(x, c)$ can be either linear or nonlinear, and the solution for both varies depending on the mathematical construct of the problem.

In a linear system with overdetermined problem ($n_s > n_c$), the most straightforward method in solving the linear least squares problem is using analytical approach by direct matrix manipulation called batch least squares estimator [7: 129–131]. In order to do this, eq. (11) (in vector form) need to be linearized and rearranged into a matrix equation in the form of

$$\boldsymbol{z}^{n_s} = \boldsymbol{H}^{n_s} \boldsymbol{c}, \tag{12a}$$

where the vector $\boldsymbol{z}^{n_s}$ is the stacked vector of n_s measurements with the dimension of $n_s n_z$ (where each measurement is an n_z-vector), $\boldsymbol{c}$ is the vector containing the unknown model parameters with dimension of n_c, and the matrix $\boldsymbol{H}^{n_s}$ contains the rearranged components from the vector form of eq. (11) with the dimension of $n_s n_z \times n_c$. The unknown model parameters vector $\boldsymbol{c}$ can then be estimated with the pseudoinverse method

$$\boldsymbol{c} = \left[\boldsymbol{H}^{n_s\top} \boldsymbol{H}^{n_s}\right]^{-1} \boldsymbol{H}^{n_s\top} z^{n_s}. \tag{12b}$$

Solving a nonlinear least squares problem requires different approach. Different methods have been studied in literature for ADCS instruments calibration, such as adaptive least squares, two-step nonlinear estimator, PSO, or Gauss-Newton method [30, 64, 79, 90].

4.2 Particle Swarm Optimization

Eberhart et al. and Kennedy et al. introduced PSO algorithm as nonlinear function optimization algorithm in [23] and [40]. The algorithm was initially derived from a social model of animals, e. g. flight behavior of a flock of birds, implemented in an abstract level. Since its inception, PSO has been used in very wide range of field. Studies such as ones by Eberhart et al. and Poli documented various fields of study where PSO has been implemented: automatic control tuning, neural network training, optimization problem applications in telecommunication networks, biomedical applications, robotics, automation technology, antenna design, mechanical analysis, and many other fields [22, 60]. The total number of papers published from 1997 up to 2006 is more than a thousand with incredible growth rate [22, 60]—a popularity gained because of its simplicity and high level of abstraction, which means it is easy to implement and combine with different methods in different kinds of applications.

However, through this very rapid developments over the years, PSO algorithm has been implemented with large amounts of variations: its simplicity also means that modifications, whether limited to minor modifications or major reworks on the algorithm mechanics, can be easily designed and implemented, resulting in unstandardized performance benchmarks across wide variations of papers that address different specific class of problems using PSO. To give a clear base in PSO development and implementation, Bratton et al. redefined the standard form of PSO and categorizes the different modification techniques that can be implemented on the algorithm in [13]. This section describes the basic version of PSO that is used as a ground for development in this thesis, along with some insights on the different modification concepts and parameter tuning techniques.

The standard PSO mechanics. PSO emulates swarm intelligence from animal social behavior in solving problems, e. g. flock of birds flying in search of an ideal food source. The location of the food source is initially unknown to all the birds, and the swarm will explore the area by following the bird that is most likely to be nearest

to the food source. In abstract level, each bird (called *particle*) is a single solution to the parameter optimization problem, which is the location of the food. The closer a particle is to the best solution (which provide the best *fitness*), the more likely the swarm will go into the direction of the said solution. However, as the swarm explore the solution space, each particle will maintain its own 'craziness' to emulate the randomness of animal search pattern, thus dividing its search effort between the swarm-wide best solution (the particle with best fitness among all particles) and the particle own best solution which it has found so far. If a more likely solution is found, the swarm will switch to the new, better solution until the swarm reach a certain threshold. Interpreting this abstraction into an algorithm can be implemented in several steps: initializing the swarm, updating the swarm positions, and polling the best solution. These steps (except the initialization) will be iterated until a certain threshold is reached:

1. Initialization of the swarm: the algorithm employs n_p numbers of particles (the *swarm size*), each placed in a random initial 'position' $\boldsymbol{p}_{\text{pso},i}(k=0)$ with a random initial 'velocity' $\boldsymbol{v}_{\text{pso},i}(k=0)$ in an abstract n^c-dimensional space (the *swarm dimension*), or, written in mathematical notation (the pso subscript is omitted for brevity)

$$\begin{aligned}\boldsymbol{p}_i(k) &= [p_{i1}(k), p_{i2}(k), \ldots, p_{in^c}(k)] \equiv [p_{ij}|_{j=1\ldots n^c}(k)]\,, \text{ and}\\ \boldsymbol{v}_i(k) &= [v_{i1}(k), v_{i2}(k), \ldots, v_{in^c}(k)] \equiv [v_{ij}|_{j=1\ldots n^c}(k)]\,,\end{aligned}$$

   where the subscript i is the particle index ($i = [1, n_p] \in \mathbb{Z}$), subscript j is the particle component index ($j = [1, n^c] \in \mathbb{Z}$), and k is the iteration index. In an optimization problem of a function where an n_c number of unknown parameters need to be estimated, the number of parameters corresponds to the swarm dimension n^c, i. e. solving n_c unknown parameters requires particles to explore n^c-dimensional space, or $n^c = n_c$. For simplicity, n_c will refer to both the swarm dimension and number of parameters from here onwards. Besides the position and velocity vector, each particle also has a third vector: its *local* best position $\boldsymbol{pbest}_{\text{pso},i}$, defined as the position that will produce the best fitness in the search history of each particle i. In the initialization phase, the initial local best position $\boldsymbol{pbest}_{\text{pso},i}(k=0)$ of each particle is the particle initial position $\boldsymbol{p}_{\text{pso},i}(k=0)$ itself. In swarm-wide scope, a *global* best position $\boldsymbol{gbest}_{\text{pso}}$ is defined as the best position in the whole swarm, i. e. the local best position of the particle with the best fitness.

2. Swarm positions update: the velocity and position of each particle are updated with the basic formula (the pso subscript is omitted for brevity) [13, 14]

$$\begin{aligned}v_{ij}(k+1) = wv_{ij}(k) &+ \text{rand}(0,1)c_1(pbest_{ij}(k) - p_{ij}(k)) +\\ &\text{rand}(0,1)c_2(gbest_j(k) - p_{ij}(k)), \text{ and}\end{aligned} \tag{13a}$$

$$p_{ij}(k+1) = p_{ij}(k) + v_{ij}(k+1), \tag{13b}$$

   where $\text{rand}(0,1)$ is a random number in the range of $[0\ldots1] \in \mathbb{R}$, w is the *inertia weight* parameter, c_1 is the *cognitive rate* parameter, and c_2 is the *social*

rate parameter. The inertia weight parameter defines the particle tendency to stay on its original course, resisting influence from the global and local best position. The cognitive rate parameter defines the influence of the particle memory of its best location, pulling the particle towards it. On the other hand, the social rate parameter defines the influence of the swarm global best position, pulling each particle search space towards it.

3. Best solution evaluation: $\boldsymbol{pbest}_i$ for each particle i is replaced by the new particle position if the fitness value is better than the last one. In this thesis, the PSO is used for optimizing minimization problems, thus the objective of the swarm is to minimize the fitness value, or mathematically written as

$$\boldsymbol{pbest}_i(k) = \arg\min_{\boldsymbol{p}_i} f(\boldsymbol{p}_i(1 \dots k)), \tag{14a}$$

while, as previously explained, the global best position is obtained from the local best position with the best fitness among all particles, or, in mathematical notation

$$\boldsymbol{gbest}(k) = \arg\min_{\boldsymbol{pbest}_i} f(\boldsymbol{pbest}_i(k)), \tag{14b}$$

where $f()$ is the *fitness function* which returns the fitness value

$$F_i = f(\boldsymbol{p}_i), \text{ for } i = 1 \dots n_p, \tag{14c}$$

with the notation for the fitness value of the local and global best positions are written as

$$F_{pi} \equiv f(\boldsymbol{pbest}_i) \text{ and } F_g \equiv f(\boldsymbol{gbest}). \tag{14d}$$

After the local best position for each particle is evaluated with eq. (14a), the global best position is then reevaluated with eq. (14b): if a new local best position with a fitness value lower than the current global best position exists, then the global best position will be updated with the new, better position—else, the global best position stays the same.

4. Iteration evaluation: the algorithm will determine whether the swarm has reached its goal by checking several predefined conditions, e. g. the current global best fitness is lower than a certain threshold, the number of iterations has reached a certain number, or the swarm has converged into a certain value and the search space is not explored to a certain range. If all of the predefined conditions are not met, then the iteration will continue. On the other hand, if any of the condition (or a combination of them) is met, then the algorithm will stop and return the solution contained in its $\boldsymbol{gbest}$. A pseudocode summarizes the basic PSO process in algorithm 1.

Algorithm 1: Basic PSO algorithm.

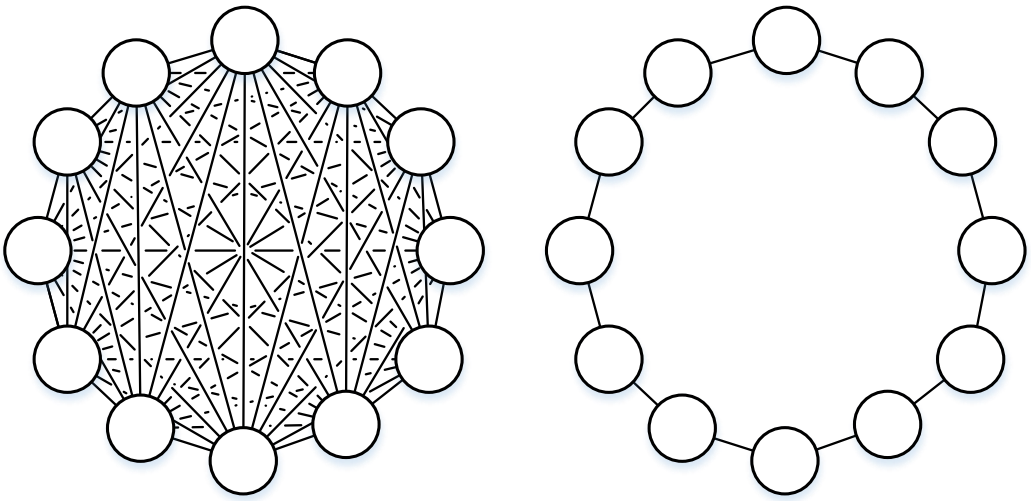
define PSO parameters and termination conditions
for $i = 1 \dots n_p$ **do**
    initialize $\boldsymbol{p}_i(k=0)$ and $\boldsymbol{v}_i(k=0)$ ▷ initial swarm properties
    $\boldsymbol{pbest}_i = \boldsymbol{p}_i$ ▷ initial local best positions
end for
evaluate $\boldsymbol{gbest}$ with eq. (14b) ▷ initial global best position
while *iterate* $= 1$ **do** ▷ *iterate* set to zero if termination conditions are fulfilled
    $k = k + 1$ ▷ update time step k
    for $i = 1 \dots n_p$ **do** ▷ each particle in the swarm
        update $\boldsymbol{v}_i$ using eq. (13a) ▷ swarm velocities
        update $\boldsymbol{p}_i$ using eq. (13b) ▷ swarm positions
        calculate F_i using eq. (14c) ▷ particle fitness
        reevaluate $\boldsymbol{pbest}_i$ using eqs. (14a) and (14d) ▷ particle local best position
    end for
    reevaluate $\boldsymbol{gbest}$ using eqs. (14b) and (14d) ▷ swarm global best position
    ▷ Now, the termination conditions are checked: if k reach maximum, F_g is better than threshold, or the swarm already converge and stabilize, which can be evaluated with different approaches, then *terminate* is set to 1.
    if *terminate* $= 1$ **then**
        iterate $= 0$
    else
        iterate $= 1$
    end if
end while

From this basic version of PSO, numerous methods of improvement and parameters selection guidelines have been described in literature [13, 19, 22, 63, 65]. Sections 4.2.1–4.2.6 discuss some of these methods, such as swarm communication topology, constriction factor, dynamic parameters, initialization and boundary conditions, number of particles, and multi-objective problem optimization. The list is by no means exhaustive, since each method itself may have many variations in its implementation and these methods can be implemented in combination with each other. Nevertheless, basic comprehension of these methods will help in understanding the behavior of PSO. The application-specific version of PSO implemented in this thesis, particularly for magnetometer calibration and the spacecraft MCP analysis, is explained in section 6.

4.2.1 Swarm Communication Topology

The standard PSO employs a *global topology*: every particle in the swarm can communicate with each other directly, enabling an instant propagation of the $\boldsymbol{gbest}$ information across the swarm. However, this communication topology can be modified into a *local topology*, which is a very significant modification to the PSO mechanics that has already been proposed in one of the first published PSO study [23]. In a

local topology swarm, a particle can only communicate with its neighbors, forming a local neighborhood of swarm clusters. Defining the requirements and limitations in neighbor identification can be implemented with numerous methods, thus there are many versions of local topology, e. g. ring, star network, and tree network topology [65]. The simplest version of a local topology is the ring topology, where each particle forms a neighborhood with its two nearest neighbors: fig. 6 illustrates the difference between a ring topology and a global topology.



(a) Global topology.

(b) Local best (ring) topology.

Figure 6: Depiction of global and local ring topology. Each circle is the individual particle, and each line connecting them with each other represents the available communication between particle.

The advantage of implementing a local topology PSO varies depending on the problem characteristics and the topology design itself [41]. The main consideration is to find the optimal balance between the convergence speed and the solution quality: for a function with single minimum point, fast convergence using global topology can result in a comparable quality solution with the local topology version, although for a multimodal function where multiple local minima exist as solutions, global topology PSO tends to converge into those suboptimal local minima, delivering poor quality solution compared to local topology PSO [13, 41].

4.2.2 Constriction Factor

Constriction factor, represented with χ, is a special case of inertia weight application, where the parameter values are determined analytically using the formula [13, 17, 21, 22]

$$\chi = \frac{2}{\left|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}\right|}, \text{ where } \varphi = c_1 + c_2, \ \varphi > 4. \tag{15}$$

This constriction factor is then implemented in the swarm velocities update equation, modified from eq. (13a) into

$$v_{ij}(k+1) = \chi\Big(v_{ij}(k) + \text{rand}(0,1)c_1(pbest_{ij}(k) - p_{ij}(k)) + \text{rand}(0,1)c_2(gbest_j(k) - p_{ij}(k))\Big). \tag{16}$$

Convergence characteristic of the swarm can be manipulated analytically by changing the value of φ, where a condition of $\varphi > 4$ guarantees swarm convergence, $\varphi \approx 4$ shows some delays in convergence, while $\varphi < 4$ shows oscillation of particles around the convergence point. The main purpose of the constriction factor is to guarantee convergence of the swarm, and it is recommended to implement this technique with the implementation of velocity boundary conditions to ensure stability of the swarm. [17, 21]

4.2.3 Dynamic Parameters

For simpler test functions, PSO main parameters (i. e., w, c_1, and c_2) can be approximated with some rule of thumb combined with velocity restrictions described in [71]. However, for many problems that involve more complex nonlinear functions to be optimized, performance of PSO can be significantly improved with simple implementation of dynamic parameters: the most widely used technique is by starting the iteration with a large inertia weight value ($w_s > 1$), promoting exploration of the search space early in the iteration, and then decreasing the value to an end value w_e, promoting a finer search after the swarm has found the general region of the best solution [13, 14, 22].

Besides varying the inertia weight value, Carrubba et al. also implemented similar behavior with the cognitive and social rate values for the same purpose, which is promoting exploration in the earlier iteration and later focusing the swarm on the region of the best solution. This is implemented by first setting w and c_1 with a high value (w_s, $c_{1s} > 1$) and c_2 with zero ($c_{2s} = 0$), so that in the beginning of the iteration, each particle will explore its own search space without influence from the global network by nullifying the $\boldsymbol{gbest}$ influence in eq. (13a), allowing only the $\boldsymbol{pbest}_i$ value to influence each particle exploration behavior. Then, w value is decreased gradually until it reach certain value w_f in a predetermined number of iterations w_v, while c_1 and c_2 are constant. After that, w stays in its final value w_f while c_1 is gradually decreased and c_2 is gradually increased for another predetermined number of iterations c_v until they reach a certain value of c_{1f} and c_{2f}, respectively, thus gradually introducing the $\boldsymbol{gbest}$ into the swarm, pulling the swarm to the best solution at the time. The values for w_f, c_{1f}, and c_{2f} are analytically determined with the constriction factor rules from eq. (15) for $\varphi \approx 4$ to promote more exploration. After this period of varying parameters, the values for w, c_1, and c_2 are set to their end values w_e, c_{1e}, and c_{2e} where their values follows the constriction factor rules for $\varphi > 4$ to promote swarm convergence in the end of the iterations. This control of PSO main parameters is illustrated in fig. 7. [14]

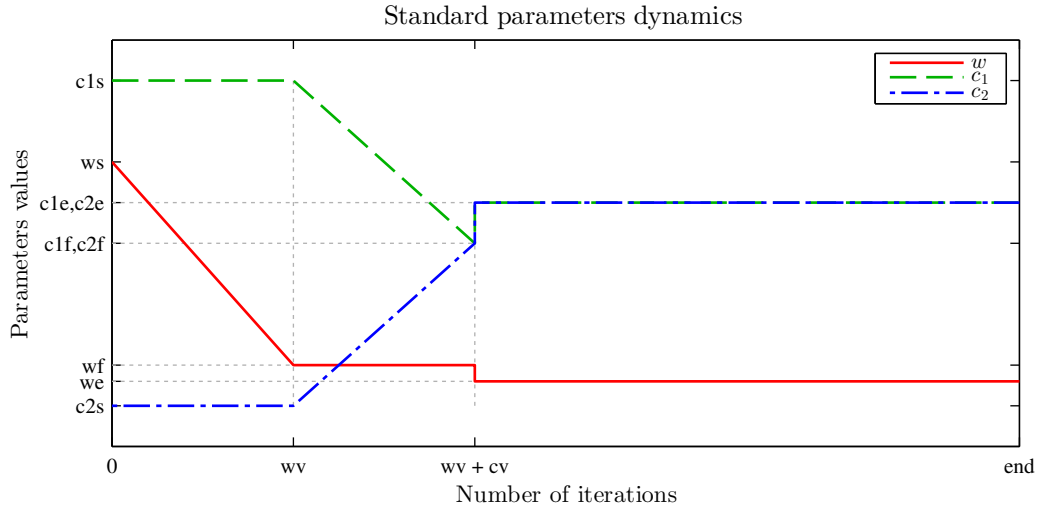


Figure 7: Plot of PSO parameters values (w denoted by straight line, c_1 denoted by dashed line, and c_2 denoted by dash-dotted line) against the number of iterations, depicting the dynamic parameters variation over the iterations. This technique is used for promoting exploration of the search space earlier in the iterations, as proposed in [14] with an addition of w_e definition.

4.2.4 Initialization and Boundary Conditions

Monson et al. in [57] has shown that for certain problem, PSO solution can be biased to the center of the region where the swarm is initialized. This creates problem for algorithm-benchmarking study to analyze the algorithm performance in an unbiased evaluation. However, for practical implementation of the algorithm, constricting the area where the swarm is initialized could help the algorithm to find the optimal solution, given there is a reasonable assumption on the position of the solution. Besides the initialization region, PSO performance can be improved by implementing boundary condition on the swarm [13].

There are several approaches to impose boundary conditions on the swarm, either implemented by limiting the swarm positions $\boldsymbol{p}_i$ below a value $\boldsymbol{pmax}$ or limiting the swarm velocities $\boldsymbol{v}_i$ below a value $\boldsymbol{vmax}$. These conditions prevent *swarm explosion*, where the particles fly into infinite by limiting the search space. Limiting $\boldsymbol{v}_i$ will indirectly limit the search space to the area around the initialized search space while allowing overshoot to some extent [13, 17]. Generally, imposing direct limitations on $\boldsymbol{v}_i$ using $\boldsymbol{vmax}$ is more acceptable, while imposing direct limitations on $\boldsymbol{p}_i$ using $\boldsymbol{pmax}$ is generally avoided because it is considered too 'artificial'—a more natural way to impose boundary condition on swarm positions is by ignoring the fitness of the particles located outside $\boldsymbol{pmax}$, thus avoiding any pull factor from a location outside the boundary [13, 22].

4.2.5 Number of Particles

The effect of the number of particles n_p on the algorithm performance varies depending on the problem—many literature starts within the guideline of 20–100 particles to

produce optimal performance, although determining the optimal number of particles for a specific application remains a trial-and-error problem [13].

4.2.6 Multi-Objective Optimization

PSO can be modified to optimize multi-objective problem by defining multiple fitness functions, each representing different objective in the problem. This introduces *Pareto-optimization* problem, where the swarm is required to find the optimal solution so that it produces the best fitness for all objectives without compromising the fitness of any objective (*Pareto optimal*), in contrast to a condition where one or several objectives dominate the solution, while other objectives ended in a suboptimal solution (*Pareto dominated*). Coello et al. published a paper proposing the solution for this multi-objective problem in [18], and later made some improvement on the method in [19]. Since then, various papers addressing different methods for multi-objective optimization using PSO have been published, and a survey study by Reyes-Sierra et al. provides a good starting point to understand the different methods of this multi-objective PSO algorithms, e. g. *aggregating* approaches, *lexicographic ordering*, *sub-population* approaches, *Pareto-based* approaches, and combined approaches [65].

Aggregating approaches: In this approach, the multi-objective problem is handled by aggregating all the objectives into one, which can be implemented by simply combining the fitness values from different objectives (effectively converting the multi-objective problem into a single objective), although one could assign different weighting scheme to balance the dominance between these objectives, e. g.fixed weight for each objective, dynamic weight (gradually changing weight such as alternating sinusoidal/saw pattern for each objective), or bang bang weight (where the weight alternate abruptly between the different objectives) [65].

Lexicographic ordering: In this approach, the objectives are ranked based on their priority, and then the algorithm will evaluate each objective fitness separately, starting from the one with highest priority and continued with the next highest prioritized one [65].

Sub-population approaches: In this approach, the swarm is separated into several subpopulations, where each subpopulation evaluates a single objective and communicates with the other subpopulations to exchange their results. Combining the results from every subpopulation can be performed using different swarm communication architectures and methods [65].

Pareto-based approaches: In this approach, the swarm will select certain particles as the swarm leaders, which have the best non-dominated solution compared to the swarm in various neighborhood topologies. However, identifying the leaders requires the processing of additional information, which can be obtained through various schemes, as compiled in [65].

Combined approaches: As the name describes, this approach combines many techniques for handling multiple objectives in PSO, e. g. Mahfouf et al. combined adaptive weight objectives aggregation with non-dominated sorting algorithm in [50], and Xiao-hua et al. implemented an intelligent PSO, where the particles are assigned with additional properties and goals in a lattice-like neighborhood topology in order to push the swarm into the Pareto optimal region by introducing particle selection pressure [92]. Numerous studies have proposed other multi-objective PSO approaches, as reviewed in [65].

4.3 Estimation Filter for Real-Time Calibration

Calibration methods which can be executed in real time while simultaneously taking measurements on orbit discussed in this thesis are performed with estimation filters. The estimation filter task is to directly estimate the states of the system (in some cases including the calibration parameters) from combined measurement data of different sensors while taking into account different errors—deterministic and/or stochastic—contained in those measurements [7]. The advantage of estimation filters is that any unknown behavior of the system which cannot be modeled properly can be taken as stochastic process and, to some degree, compensated by the filter. Huge numbers of studies are dedicated to optimize these filters for space applications. [20, 34, 66, 77, 80, 95]

The widely used real-time calibration algorithms for space applications are Kalman filter (KF)-based algorithms. Since the state dynamics of spacecraft is highly nonlinear, nonlinear KFs such as EKF or unscented Kalman filter (UKF) are typically used. EKF has been around for a long time, originally developed for implementing standard KF in nonlinear system, and has been documented as the standard nonlinear state estimator in textbooks for estimation theory such as [7], while UKF was initially proposed by Julier et al. in [37].

UKF was developed to increase the filter reliability in highly nonlinear system by including the fourth and higher order in the Taylor series expansion of the errors (EKF only approximate to the first order) into the estimator without the need of calculating the Jacobians—this is possible because UKF approach begin by approximating the probability distribution itself rather than approximating the nonlinear function of the system by calculating its Jacobians [37]. However, some papers also suggest the usefulness of modified EKF for some situations: Zeng et al. proposed a robust EKF scheme to compensate for orbit error that propagates into attitude estimation from magnetometer reading [95]; and Searcy et al. proposed a method for full attitude estimation only from magnetometer reading by using the attitude information and its derivative—this requires long orbital arc to converge, but ultimately gives real-time data [66].

Other papers have supported the theoretical advantage of UKF over EKF: Crassidis et al. compared the performance of three estimation algorithms—including EKF and UKF—in calibrating magnetometer reading using results from both simulated (in Monte-Carlo simulation) and real satellite flight data, and concluded that UKF has the most stable result [20]; Inamori used UKF to estimate the RMM of spacecraft

and used the RMM estimate in feed-forward control to compensate the disturbance torque caused by it, whereas EKF is too sensitive to small inaccuracies in the spacecraft dynamic model [34]; and Springmann et al. also implemented the UKF in real satellite flight data and pointed that UKF would perform better than EKF for wider initial estimate range, although it was also noted that for a close initial estimate, the difference in accuracy between the two algorithms is negligible and EKF has lower computational load in such application [80]. Adaptive UKF has also been proposed to improve the UKF performance when dealing with uncertain error covariance and changing parameters [74, 75, 77]. An overview of the difference between standard KF, EKF, and UKF is listed in table 2. Aalto-2 ADCS will implement a UKF as its attitude estimation filter. However, the technical details is not in the scope of this thesis work.

Table 2: Comparison of Kalman Filter algorithms.

STANDARD KF	EKF	UKF
Only for linear system.	For nonlinear system.	For nonlinear system.
The state function is linear from the beginning.	Operate by linearizing the state function with Taylor expansion approximation through calculating the Jacobians.	Operate with fully intact state function by selecting sample points according to the error distribution.
Initial estimate affects converging time; accuracy is optimal in linear functions.	Initial estimate affects final accuracy; sensitive to parameter errors	Better tolerance to larger range of initial estimate and parameter errors.

5 First Generation Aalto Nanosatellite Missions

This thesis focuses on the calibration of ADCS instruments in Aalto-2 nanosatellite mission. However, the hardware used for experimental tests comes from Aalto-1 EM, which is currently in a parallel development with Aalto-2 as the first generation nanosatellite missions in Aalto University. This is done for practical reason, since Aalto-1 EM is ready for testing during the time of this thesis work and both satellites share similar ADCS instruments, although the missions and payloads they carry are quite different [2, 42, 58, 62]. The spacecraft body reference frame defined for Aalto-1 is as follows: positive x-axis points to the direction of the camera payload, positive y-axis points to the direction opposite of the star tracker, and z-axis points to the side where the antenna is attached, completing a right-handed reference frame. This frame definition is illustrated in fig. 8. For analytical purposes and model considerations, Aalto-2 mission requirements are used as the reference.

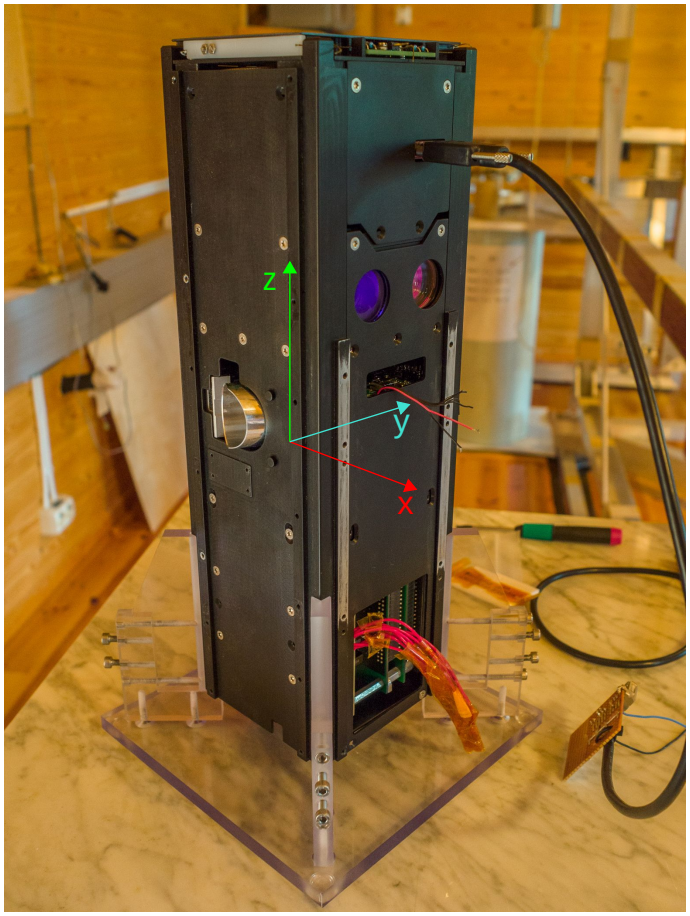


Figure 8: Aalto-1 engineering model with the spacecraft body reference frame definition.

This section gives a short overview on the Aalto-2 mission requirements and a more detailed analysis on the orbital geomagnetic characteristics. The latter is useful for design considerations in setting up the test environment for magnetic properties analysis of the spacecraft, including the magnetometer calibration.

5.1 Aalto-2 Mission Requirements

Aalto-2 satellite mission is to conduct in-situ atmospheric measurements in the lower thermosphere between 200–380 km (lower end of LEO) using a multi-Needle Langmuir Probe (mNLP) as its scientific payload [2]. The mission requires the payload to be positioned in the ram direction of the satellite—detailed mission requirements refer to the critical design review of Aalto-2 and QB50 mission descriptions in [2] and are subject to changes until the final flight model is built. In order to achieve this requirement, both passive and active attitude control will be implemented in different phases of the mission: active attitude control using three-axis magnetorquer in the earlier mission phase, where the orbit altitude is around 400 km, and passive attitude stabilization using atmospheric drag in the latter phase of the mission as the altitude decreases to 300 km and below, where atmospheric drag is predicted to become more dominant in the spacecraft dynamics [36].

In order to perform the three-axis active attitude control efficiently, the ADCS is programmed to behave in three different operation modes, each with its own prerequisites and performance requirements [2]:

a) *Recovery mode*: activated when the angular velocity of the satellite exceeds the threshold of 5 °/s, where the ADCS detumbles the satellite until the angular velocity goes down to 0.8 °/s—the minimum performance requirement is to recover from a tumbling rate of up to 10 °/s in two days. This mode is also the initial mode activated during Launch and Early Orbit Phase (LEOP) after subsystems self-check routine.

b) *Nominal attitude control mode*: activated after recovery mode is completed, where the ADCS performs three-axis pointing control with minimum pointing accuracy of ±10°.

c) *Safe mode*: activated when a malfunction occurs or when the battery charge drops below 60 %, where the ADCS limits its functionality: active attitude control is deactivated, and sensor telemetry will be collected only from the gyro and magnetometer.

Another ADCS performance requirement is a minimum pointing knowledge of ±2° from its initial launch altitude [2].

5.2 Orbital Geomagnetic Analysis

Achieving best results in spacecraft magnetic properties analysis, including calibration of magnetic-sensitive instruments, requires a magnetically 'clean' test environment: this means a uniform, accurately known ambient magnetic field strength in the test environment which can be manipulated and nullified, usually using a Helmholtz cage setup [14, 82]. For the purpose of this thesis work, the Helmholtz cage setup are used for two purposes:

a) Nullifying the ambient magnetic field in the test environment for spacecraft MDM analysis as described in section 3.4.3, so that the measured magnetic field

$\check{\boldsymbol{b}}_s$ at each measurement point s (as depicted in fig. 5) only contains magnetic field originating from the spacecraft RMM.

b) Simulating a range of magnetic field strength which will be experienced by the spacecraft in its defined orbit in order to detect any anomaly associated with ferromagnetic materials in the spacecraft that might behave differently in different magnetic field magnitude. This anomaly should show up in the magnetometer calibration parameter, since the magnetometer model in eq. (7) only include a constant soft iron coefficient, which should not change with varying ambient magnetic field—any detected anomaly is very likely to be associated with hysteresis effect, which might require anticipation strategies depending on the severity of the model error.

Appropriately manipulating the ambient magnetic field for the latter purpose requires a knowledge of the geomagnetic field model on the orbit altitude during a certain time period, which can be calculated from the IGRF model as defined in [26].

From implementing the 12th IGRF model—dubbed IGRF12—in Matlab, it can be shown that the Earth magnetic field at 350 km altitude (average of the expected orbit altitude), in a time period between the year 2016 and 2019, will vary from 19 502 to 56 203 nT. It is assumed that the satellite will experience the full range of this geomagnetic field magnitude, because the polar orbit of the spacecraft will eventually cover all of the Earth surface. An example of the geomagnetic field magnitude distribution over the Earth surface, calculated for 1^{st} of June 2016, is mapped in fig. 9.

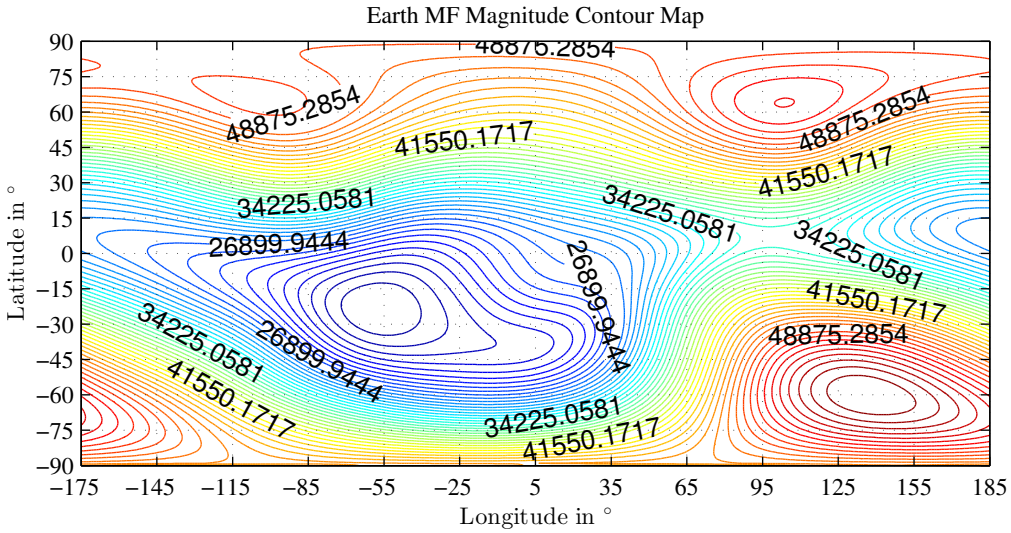


Figure 9: Contour map of geomagnetic field magnitude distribution at 350 km altitude on 1^{st} of June 2016, based on IGRF12 model. The plot x-axis denotes the longitude in ° and the y-axis denotes the latitude in °. The geomagnetic field magnitude is denoted with the contour lines in nT.

6 Development of PSO for Magnetic Tests

An improved PSO algorithm is developed in this thesis in order to solve the specific problem of magnetometer calibration parameters estimation and spacecraft MCP analysis. This section explains the concepts and techniques involved in the design of the improved PSO algorithm in contrast to the standard version of PSO discussed in section 4.2, i. e. objectives and fitness functions definition, dynamic parameters tuning, initialization and boundary condition tuning, and refinement procedure. Additionally, the development process of these concepts and their effects on algorithm performance improvement are demonstrated using simulated data.

6.1 Objectives and Fitness Functions Definition

Mathematical definition of the swarm fitness function is an important starting point of modeling a problem for PSO, since the chosen fitness function can significantly affect the performance of the algorithm [14]. Since the fitness function is derived from the objective of the optimization problem, the function definition differs for the magnetometer calibration parameters estimation and for the MCP analysis with inverse MDM problem.

6.1.1 Fitness Functions for Magnetometer Calibration

As discussed previously, the magnetometer calibration is performed in magnetic domain. Thus, as explained in section 3.4.2, the objective of the calibration algorithm is to optimize the calibration parameters from eq. (7) so that the magnitude of all the magnetic field vector measured by the magnetometer $\check{\boldsymbol{b}}$ is equal to a known reference magnetic field magnitude $\boldsymbol{b}$, independent of the magnetometer orientation itself. However, many of the calibration parameters defined in eq. (7) is not separable mathematically, hence eq. (7) can be simplified into

$$\check{\boldsymbol{b}} = \boldsymbol{S}_{m_c}\boldsymbol{b} + \mathbf{off}_{m_c}, \tag{17a}$$

where 3×3 matrix $\boldsymbol{S}_{m_c}$ and 3×1 vector $\mathbf{off}_{m_c}$ are the compounded calibration parameters from the combination of $\boldsymbol{S}_m$, $\boldsymbol{N}$, $\boldsymbol{C}_{\text{si}}$, $\boldsymbol{b}_{\text{hi}}$, and $\mathbf{off}_m$ in eq. (7). Note that in magnetometer calibration parameters estimation problem, the calibration parameters serve to transform measured magnetic field $\check{\boldsymbol{b}}$ into the estimated magnetic field $\hat{\boldsymbol{b}}$, thus the true magnetic field $\boldsymbol{b}$ in eq. (17a) is replaced by $\hat{\boldsymbol{b}}$ and the whole equation can be rearranged into

$$\hat{\boldsymbol{b}} = \boldsymbol{K}_m\check{\boldsymbol{b}} - \boldsymbol{k}_m, \tag{17b}$$

where $\boldsymbol{K}_m = \boldsymbol{S}_{m_c}^{-1}$ and $\boldsymbol{k}_m = \boldsymbol{S}_{m_c}^{-1}\mathbf{off}_{m_c}$. The estimated calibration parameters themselves are contained in the PSO solution with the mathematical structure

$$\begin{aligned}\boldsymbol{p}_{\text{pso},i} = [&K_{m11,i}, K_{m12,i}, K_{m13,i}, K_{m21,i}, K_{m22,i}, K_{m23,i},\\ &K_{m31,i}, K_{m32,i}, K_{m33,i}, k_{m1,i}, k_{m2,i}, k_{m3,i}]^\top,\end{aligned} \tag{18}$$

where

$$\boldsymbol{K}_{m,i} \equiv \begin{bmatrix} K_{m11,i} & K_{m12,i} & K_{m13,i} \\ K_{m21,i} & K_{m22,i} & K_{m23,i} \\ K_{m31,i} & K_{m32,i} & K_{m33,i} \end{bmatrix} \text{ and } \boldsymbol{k}_{m,i} \equiv \begin{bmatrix} k_{m1,i} \\ k_{m2,i} \\ k_{m3,i} \end{bmatrix}.$$

From here, the fitness function for magnetometer calibration can be defined as the difference between the reference magnetic field magnitude $|\boldsymbol{b}|$ and the calibrated magnetic field magnitude $\left|\hat{\boldsymbol{b}}\right|$, or written mathematically as

$$F_{1,i} = \sum_{s=1}^{n_s} \left[|\boldsymbol{b}_s| - |\hat{\boldsymbol{b}}_{is}| \right]^2, \tag{19a}$$

or, by substituting eq. (17b) into eq. (19a),

$$F_{1,i} = \sum_{s=1}^{n_s} \left[|\boldsymbol{b}_s| - |\boldsymbol{K}_{m,i} \check{\boldsymbol{b}}_{is} - \boldsymbol{k}_{m,i}| \right]^2, \tag{19b}$$

where $i = [1, n_p] \in \mathbb{Z}$ and $s = [1, n_s] \in \mathbb{Z}$ are the index for the particle in the swarm and the index for measurement data, respectively.

Comparing the form of eq. (19) and eq. (11), one can conclude that in this case, the PSO objective is to optimize a nonlinear least squares problem. However, the solution of the PSO up to this point still contains a rotational ambiguity: any arbitrary rotation matrix can be multiplied with the total scale matrix $\boldsymbol{K}_m$, and the measurement locus will stay on the surface of a sphere. Since no actual heading reference is used at any point of measurement, the algorithm cannot differentiate whether the sphere is 'looking' into a certain direction, resulting in infinite numbers of non-trivial solutions. To compensate for this problem, many studies which utilize attitude-independent magnetometer calibration technique impose some assumptions on the calibration parameters: Ali et al. simplified the scale matrix into a diagonal vector [3], Foster et al. and Springmann et al. assumed that one of the sensor axis is perfectly aligned with the reference axis, resulting in triangular scale matrix [29, 79], while Gebre-Egziabher et al. assumed that the misalignment coefficients are small, a condition used for initial estimate in two-step calibration method [30]. All of them also neglected soft iron coefficients in non-aligned axes, thus simplified the estimation problem and effectively eliminating the rotational ambiguity. Renaudin et al. proposed a complete magnetometer calibration algorithm which minimizes all those assumptions using adaptive least squares method, although the results show that the algorithm is still sensitive to non-diagonal components of the total scale matrix associated with large errors from soft iron and misalignment factors [64].

Resolving the rotational ambiguity when no simplification is made on the calibration parameters requires an additional information in the algorithm. An example of this is by incorporating the fact that the angle between the geomagnetic field vector and the gravity vector is always fixed on a fixed location on Earth, thus resolving the rotational ambiguity by estimating a new correction rotation matrix as proposed in [46]. However, this method is only suitable for land-based magnetometers setup

where the location of the instruments on Earth is relatively constant (at least during the data collection for calibration) and the gravity vector is large enough to be detected by accelerometers—this is especially important in the scope of this thesis work for the fact that Aalto-2 is not equipped with accelerometers.

New objective for magnetometer calibration. In this thesis, a novel approach is used by incorporating the spacecraft rotation axis information into the PSO algorithm. A basic version of this approach is as follows: consider a set of magnetic field measurements from the magnetometer as the spacecraft tumbles as such that the measurement locus can be separated into several loci l, where each locus forms a circle on a plane with different normal directions—at least two circles (not necessarily a complete circle) are required for a reason which will be explained later. If the vectors representing the rotation axes of the spacecraft which forms the circle loci are known (e. g., from gyroscope data), then the second objective of the PSO is to select the proper calibration parameters which will rotate those circles as such that the plane normal $\mathring{\boldsymbol{n}}_{il}$ (the plane containing a circle of measurement locus l after calibrated by particle i) is aligned with its respective known rotation axis $\mathring{\boldsymbol{\omega}}_l$. Note that the plane normal direction and the rotation axis are both represented by unit vectors $\mathring{\boldsymbol{a}} \equiv \frac{\boldsymbol{a}}{|\boldsymbol{a}|}$, since only the direction information is of interest for this purpose. This is also the reason for incomplete circle loci to be sufficient: since only the plane normal direction information is required for fulfilling this objective, then an incomplete circle, as long as it fits into a plane, could provide enough information on the plane direction. In the other hand, the reason for the requirement of at least two circle loci is because one circle locus only provides a single constraint for the data plane in two axes, leaving scale and offset ambiguity in one axis that is aligned with the plane normal as well as a rotational ambiguity along that axis. An example of this loss of scale and offset information is depicted in fig. 10, where the data is erroneously scaled and offset as such that the calibrated data still fulfill its ill-constrained objectives. The PSO solution can be drawn to a unique, correct solution that resolve the rotational ambiguity in three axes only if the requirements are met.

A more universal version of this approach is to fit any shape of measurement locus into the objective, especially when the loci curve cannot be continuously fitted into a single plane. This can be achieved by different methods, e. g. by using similar principle as the basic version where several segments of the curve is fitted into different planes (although this is limited to loci that can be averaged each to fit a plane) or by fitting the loci directly into the curve functions. This universal approach is more favorable for on-orbit batch calibration, where the tumbling motion of the satellite is not necessarily dividable into segments as such that each segment fits a single plane. However, for the ground-based calibration test in this thesis, the state of rotation can be easily determined, since the rotation was performed manually as described in section 7.3.1—the advanced, universal version of this approach is not in the scope of this thesis and will be left for future studies.

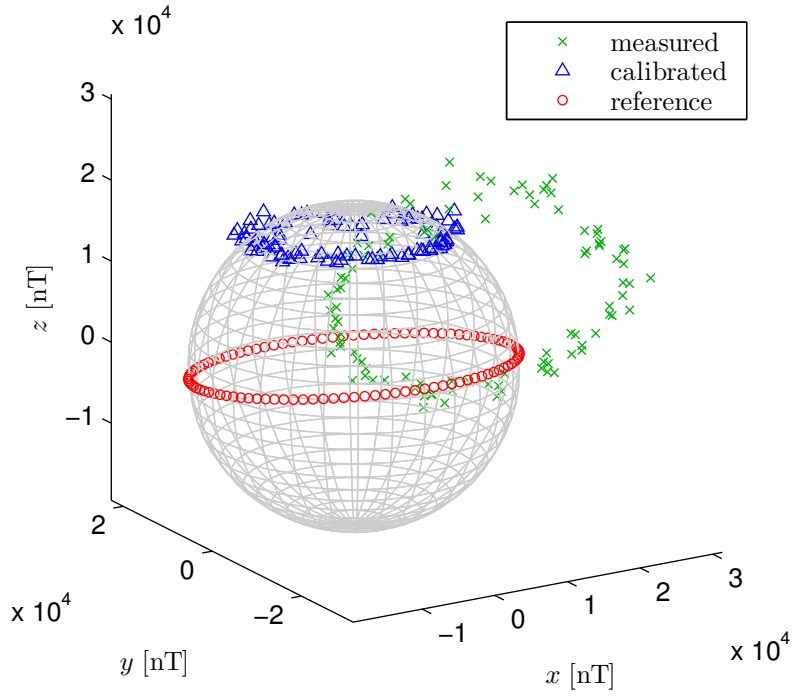


Figure 10: 3-D plot of magnetic field vectors (the axes represent the magnetometer frame in nT), depicting a case with scale and offset ambiguity. The calibration parameters forced the measured data (denoted with ×'s) into the calibrated data (denoted with △'s) that fits into a plane with correct normal direction and uniform distance from the center (fulfilling the requirement of a sphere surface) , although in a wrong offset and scale compared to the reference (denoted with ○'s), which is the simulated error-free data. The data shown contains 20% random noise relative to the reference.

Thus, a second fitness function can be defined from the new objective:

$$F_{2,i} = \sum_{l=1}^{n_l} \left[1 - \left(\mathring{\boldsymbol{n}}_{il} \cdot (-\mathring{\boldsymbol{\omega}}_l) \right) \right]^2, \tag{20}$$

where the subscripts i and l are the index for the particle in the swarm and the index for individual measurement locus that lies on a single plane, respectively, the unit vector $\mathring{\boldsymbol{n}}_{il}$ is the normal direction of the plane which contains the calibrated measurement locus l as estimated by particle i, where its positive direction follows the right-hand rule of the data sequence in the calibrated measurement locus, and the unit vector $\mathring{\boldsymbol{\omega}}_l$ is the rotation axis that corresponds to the measurement locus l, which can be obtained from gyroscope data or direct observation (e. g., in preflight test). Note that the direction of $\mathring{\boldsymbol{\omega}}_l$ is inverted with a minus sign because the direction of magnetometer rotation is the opposite of the measurement locus sequence. In summary, the second fitness function F_{m_2} represents the sum of angular difference between $\mathring{\boldsymbol{n}}_{il}$ and $\mathring{\boldsymbol{\omega}}_l$ for $i = 1 \ldots n_p$ and $l = 1 \ldots n_l$, which will be minimized by the PSO.

Mathematically extracting $\mathring{\boldsymbol{\omega}}_l$ and $\mathring{\boldsymbol{n}}_{il}$ information. As previously discussed, this thesis work only implements the basic version of the rotation vector fitting, where the rotation axis associated with each measurement locus is fixed. Thus, each measurement locus lies on a single plane—or at least the measurement data can be averaged to a single plane since noise presence will disturb the locus out of its plane. For the magnetometer calibration test performed in this thesis work, the value of $\mathring{\boldsymbol{\omega}}_l$ can be determined directly from direct observation during the calibration test, where the spacecraft is rotated manually along its spacecraft body axes , while the value of $\mathring{\boldsymbol{n}}_{il}$ is calculated using orthogonal distance regression plane fitting, where the sum of squared orthogonal distances from the points on the calibrated measurement locus to the plane is minimized. Mathematical tools to solve this least squares problem are well documented, and this thesis work implements a singular value decomposition (SVD) method [73, 78]. This method is summarized in algorithm 2.

Algorithm 2: Algorithm for orthogonal regression plane fitting using SVD [73].

1: calculate $\bar{\hat{\boldsymbol{b}}}_{i,l} = \frac{1}{n_{sl}} \sum_{s_l=1}^{n_{sl}} \hat{\boldsymbol{b}}_{is_l,l}$ ▷ data centroid in each locus l

2: construct $\boldsymbol{A} = [\hat{\boldsymbol{b}}_{i1,l} - \bar{\hat{\boldsymbol{b}}}_{i,l}, \ldots, \hat{\boldsymbol{b}}_{in_{sl},l} - \bar{\hat{\boldsymbol{b}}}_{i,l}]$

3: decompose $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$ ▷ singular value decomposition

4: find $\text{col}_{\min} = \arg\min_x \{\text{diag}_x(\boldsymbol{S})\}$ ▷ column with minimum singular value

5: $\mathring{\boldsymbol{n}}_{il} = \boldsymbol{U}\{:, \text{col}_{\min}\}$ ▷ plane normal vector corresponds to the column of minimum singular value

6: $\mathring{\boldsymbol{n}}_{il} = \text{sign}\left(\mathring{\boldsymbol{n}}_{il} \cdot \left(\hat{\boldsymbol{b}}_{i1,l} \times \hat{\boldsymbol{b}}_{i(2\ldots n_{sl}),l}\right)\right) \mathring{\boldsymbol{n}}_{il}$ ▷ correction of the plane normal positive direction by comparing it with the measurement sequence

Note that line 6 in algorithm 2 is an additional step for rectifying the plane normal positive direction so it follows the right-hand rule with respect to the measurement sequence. This is implemented by first extracting the general estimate of the rotation axis from the calibrated measurement data using the cross product between the first measurement in the locus $\left(\hat{\boldsymbol{b}}_{i1,l}\right)$ and the next measurement $\left(\hat{\boldsymbol{b}}_{i(2\ldots n_{sl}),l}\right)$ after the magnetometer frame is rotated with arbitrary angular difference as long as the angular displacement is between 0–180°, because the cross product of two vectors with angular difference outside of that range will point to the opposite direction. An ideal point would be at 90° angular displacement, since the general direction of the rotation axis will be less affected by noise in the data. For the calibration test performed in this thesis, the spacecraft was rotated with relatively constant angular velocity for one full rotation, thus the second point of measurement used for evaluating the general rotation axis can be approximated at the quarter point of measurement in the particular locus $\left(\hat{\boldsymbol{b}}_{i(2\ldots n_{sl}),l} = \hat{\boldsymbol{b}}_{i(n_{sl}/4),l}\right)$. For different case, the second point need to be evaluated manually as the total measurement locus is segmented into several loci. After the general estimate of the rotation axis is extracted, the sign of the dot product between the vector of rotation axis general estimate and the vector of previously estimated plane normal will indicate whether the estimated $\mathring{\boldsymbol{n}}_{il}$ is pointing to the right-hand positive direction as indicated by $\left(\mathring{\boldsymbol{n}}_{il} \cdot \left(\hat{\boldsymbol{b}}_{i1,l} \times \hat{\boldsymbol{b}}_{i(2\ldots n_{sl}),l}\right)\right) > 0$, or

vice versa for $\left(\mathring{\boldsymbol{n}}_{il} \cdot \left(\hat{\boldsymbol{b}}_{i1,l} \times \hat{\boldsymbol{b}}_{i(2\ldots n_{sl}),l}\right)\right) < 0$. If $\left(\mathring{\boldsymbol{n}}_{il} \cdot \left(\hat{\boldsymbol{b}}_{i1,l} \times \hat{\boldsymbol{b}}_{i(2\ldots n_{sl}),l}\right)\right) \approx 0$, then the first and/or second point of measurement used for evaluating the general rotation axis has to be changed with another point since it indicates either that those two points have angular displacement of $\approx 0°$ or $\approx 180°$ relative to each other or that the data at those points are affected by very large noise that it alters the estimated rotation axis vector.

Combining fitness functions for magnetometer calibration. Finally, the fitness values from the two objectives have to be combined: as discussed previously in section 4.2.6, there are numerous methods to combine multiple objectives in PSO to achieve a non-dominated solution. This thesis applies the fixed weight aggregation technique to combine the fitness values for magnetometer calibration parameters estimation problem because of its simplicity. Thus, the total fitness function combines $F_{1,i}$ from eq. (19) and $F_{2,i}$ from eq. (20) into

$$F_i = c_{f_1} \frac{\sqrt{F_{1,i}}}{n_s} + c_{f_2} \frac{\sqrt{F_{2,i}}}{n_l}, \tag{21}$$

where c_{f_1} and c_{f_2} are the fixed weight for each objective. Note that both fitness values are normalized with respect to their own squared sum index elements. After some trial and error with simulated data, a balanced solution can be achieved with $c_{f_1} = 1$ and $c_{f_2} = 4 \times 10^3$. Obviously, the important part for achieving a non-dominated solution is the ratio between the two weights, although the absolute values themselves are important in the context of fitness value scaling.

Twin global minima: reflection transformation. It is important to note that, with the defined objectives, the PSO solution space still contain two 'global' minima. One minimum point is the reflection of the other minimum, where the scalar checking objective from eq. (19) is fulfilled, while the measurement locus plane fitting objective from eq. (20) is fulfilled under a reflected frame definition with a left-handed triad. Correcting the estimated calibration parameter $\boldsymbol{K}_m$, which is a 3-D transformation matrix, can be performed by reflecting it back if it contains a reflection transformation using

$$\boldsymbol{K}_m = \text{sign}\left(|\boldsymbol{K}_m|\right) \boldsymbol{K}_m, \tag{22}$$

since a transformation matrix which contains a reflection transformation will have a negative determinant.

6.1.2 Fitness Functions for Inverse MDM Problem

Fitness function for solving the inverse MDM problem in magnetic cleanliness analysis can be derived directly from the concepts explained in section 3.4.3, where the PSO objective is minimizing the summed squares of difference between the measured magnetic field $\check{\boldsymbol{b}}_s$ and the estimated magnetic field $\hat{\boldsymbol{b}}_{is}$ on all measurement points $s = 1 \ldots n_s$. $\hat{\boldsymbol{b}}_{is}$ itself is computed from substituting the PSO solution estimated by

particle i into eq. (8) at measurement points s, where the PSO solution estimated by each particle $i = 1 \dots n_p$ is structured as

$$\begin{aligned}\boldsymbol{p}_{\text{pso},i} = [&\hat{p}_{x,i1}, \hat{p}_{y,i1}, \hat{p}_{z,i1}, \hat{m}_{x,i1}, \hat{m}_{y,i1}, \hat{m}_{z,i1}, \dots, \\ &\hat{p}_{x,in_d}, \hat{p}_{y,in_d}, \hat{p}_{z,in_d}, \hat{m}_{x,in_d}, \hat{m}_{y,in_d}, \hat{m}_{z,in_d}]^\top,\end{aligned} \tag{23}$$

where

$$\hat{\boldsymbol{p}}_{id} \equiv \begin{bmatrix}\hat{p}_{x,id} \\ \hat{p}_{y,id} \\ \hat{p}_{z,id}\end{bmatrix} \text{ and } \hat{\boldsymbol{m}}_{id} \equiv \begin{bmatrix}\hat{m}_{x,id} \\ \hat{m}_{y,id} \\ \hat{m}_{z,id}\end{bmatrix}$$

are the estimated magnetic dipoles position and moment, respectively, with $d = [1, n_d] \in \mathbb{Z}$ as the dipole index for n_d number of dipoles. This means that the number of magnetic dipoles has to be estimated manually outside the PSO algorithm, which dictates the swarm dimension as $n_c = 6n_d$ since each magnetic dipole is defined with 6 components.

However, the PSO objective of minimizing the summed squares of difference between $\check{\boldsymbol{b}}_s$ and $\hat{\boldsymbol{b}}_{is}$ can be implemented in different mathematical structures. Carrubba et al. proposed two fitness functions representing this objective in PSO algorithm, which are adapted in this thesis. These two fitness functions are [14]

$$F_{1,i} = \frac{\sqrt{\sum_{s=1}^{n_s}\left[\left(\check{b}_{x,s} - \hat{b}_{x,is}\right)^2 + \left(\check{b}_{y,s} - \hat{b}_{y,is}\right)^2 + \left(\check{b}_{z,s} - \hat{b}_{z,is}\right)^2\right]}}{\sqrt{\sum_{s=1}^{n_s}\left[\check{b}_{x,s}^2 + \check{b}_{y,s}^2 + \check{b}_{z,s}^2\right]}}, \tag{24}$$

and

$$\begin{aligned}F_{2,i} = &\frac{\sum_{s=1}^{n_s}\left[\left(\check{b}_{x,s} - \hat{b}_{x,is}\right)^2\right]}{\sum_{s=1}^{n_s}\check{b}_{x,s}^2} + \frac{\sum_{s=1}^{n_s}\left[\left(\check{b}_{y,s} - \hat{b}_{y,is}\right)^2\right]}{\sum_{s=1}^{n_s}\check{b}_{y,s}^2} + \\ &\frac{\sum_{s=1}^{n_s}\left[\left(\check{b}_{z,s} - \hat{b}_{z,is}\right)^2\right]}{\sum_{s=1}^{n_s}\check{b}_{z,s}^2},\end{aligned} \tag{25}$$

where

$$\check{\boldsymbol{b}}_s \equiv \begin{bmatrix}\check{b}_{x,s} \\ \check{b}_{y,s} \\ \check{b}_{z,s}\end{bmatrix} \text{ and } \hat{\boldsymbol{b}}_{is} \equiv \begin{bmatrix}\hat{b}_{x,is} \\ \hat{b}_{y,is} \\ \hat{b}_{z,is}\end{bmatrix}$$

by definition. These fitness functions are formulated so that eq. (24) promotes a balanced influence between the magnetic field difference in all axes by normalizing the magnitude of magnetic field error with respect to the measured magnetic field magnitude, while in the other hand, eq. (25) provides more axis-specific influence by normalizing the summed squares of magnetic field error with respect to the summed

squares of measured magnetic field, each in its own respective axis, thus pulling the PSO to the global solution when the measured magnetic field have very different magnitude in each axis. To provide a balanced influence between the two fitness function, Carrubba et al. also proposed a dynamic weight aggregation method in the form of [14]

$$F_i(k) = \left(\frac{F_{1,i}(k-1)}{F_{2,i}(k-1)}\right) F_{1,i}(k) + \left(\frac{F_{2,i}(k-1)}{F_{1,i}(k-1)}\right) F_{2,i}(k), \tag{26}$$

where the combined fitness value at k-th iteration is dynamically weighted using the fitness value ratio from the two fitness functions at the previous iteration.

6.2 Tuning The Dynamic Parameters

As discussed in section 4.2.3, the PSO parameters w, c_1, and c_2 need to be tuned properly for the swarm to converge into the global optimal solution. This is especially true for the optimization problems applied in this thesis, where the problems are highly nonlinear and driven by multiple objectives, resulting in multiple local minima in the solution space. Moreover, the swarm topology implemented for the PSO algorithm in this thesis is a standard global topology which is chosen because of its simplicity, although a simplistic global topology tends to converge too early into a suboptimal local minimum for highly nonlinear problem [13]. Thus, an extended dynamic parameters is implemented in this thesis for tuning the PSO parameters dynamically as the swarm explores the solution space.

First, a combination of gradually decreasing w early in the iteration followed by gradually converging c_1 and c_2 as demonstrated in [14] is adapted in this thesis. Then, this pattern of w, c_1, and c_2 dynamics is repeated over the iterations to extend the exploration of search space. This extended pattern of PSO parameters dynamics is described in fig. 11. The purpose behind this extended parameters dynamics is for promoting exploration of the search space earlier in the iteration, and later helping the swarm in escaping local minima as the swarm starts to stabilize, which can be evaluated from preliminary simulation of the algorithm, as depicted in fig. 12.

The changes in swarm behavior for different dynamic parameters shown in fig. 12 suggest that the extended parameters dynamics might contribute to wider exploration in the search space in the later part of iterations, improving the chance of finding the region of optimal solution. This can be seen in the difference between fig. 12a and fig. 12b, where in both cases the swarm have pretty much converges early in the iteration as the varying period for the parameters dynamics ends ($k \approx w_v + c_v = 150$ for both dynamics), although for the extended parameters dynamics case in fig. 12b, the swarm best position was switched to another region in the solution space at about 200 iterations, in the middle of the second cycle of varying parameters ($k = 150 \ldots 300$). However, this behavior is not consistent throughout the whole swarm and for different algorithm runs. This is an expected behavior because PSO itself is a stochastic algorithm which might give varying results for different runs, although there is a guarantee for stability and convergence of the swarm, given some requirements in the PSO parameters [17]. Using different problem model for the

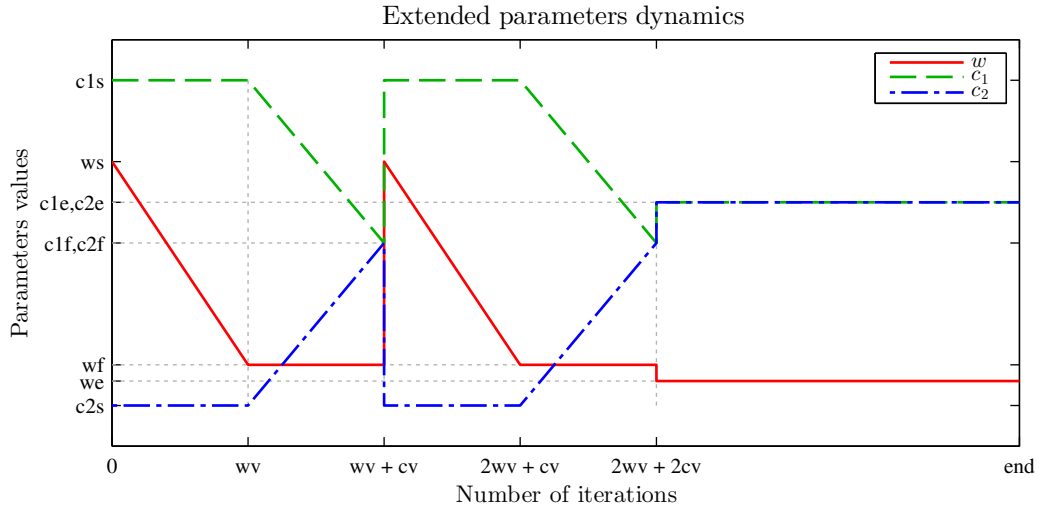


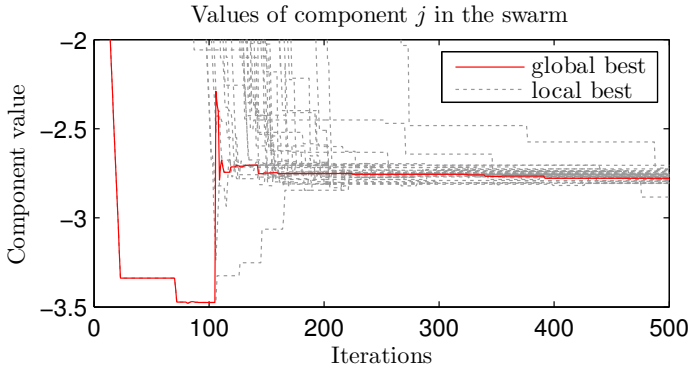
Figure 11: Plot of PSO parameters values (w denoted by straight line, c_1 denoted by dashed line, and c_2 denoted by dash-dotted line) against the number of iterations, depicting the extended dynamic parameters variations implemented in this thesis. This technique is used for promoting exploration of the search space and assisting the swarm in escaping local minima.

simulations also shows varying results about the benefit of applying the extended dynamic parameters compared to the standard one. Thus, more evaluations on the parameters dynamics are conducted using the real experimental data.

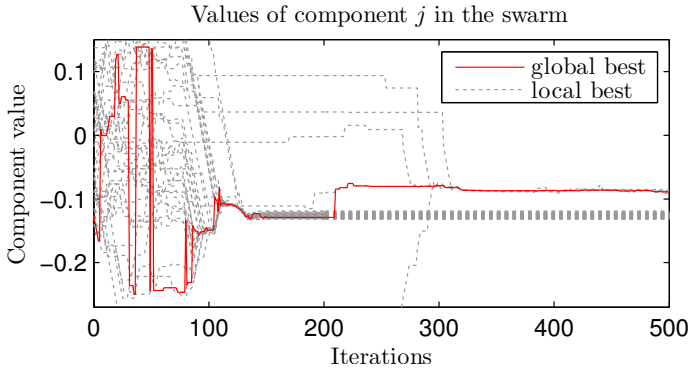
6.3 Setting The Initialization and Boundary Conditions

As discussed in section 4.2.4, the initialization and boundary conditions for PSO play some roles in preventing swarm explosion and affecting possible bias in the solution. General rules for defining the values of these conditions depend on the optimization problem and the objectives driving the PSO algorithm.

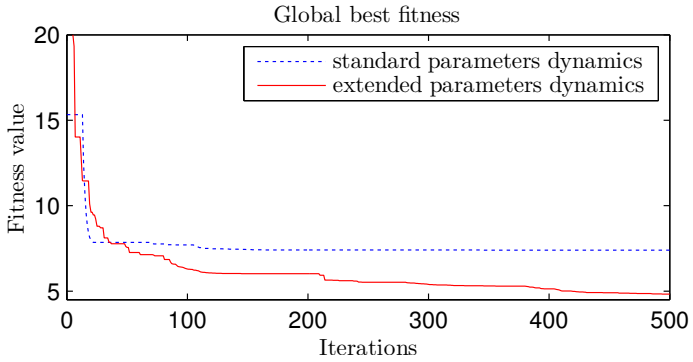
The initialization conditions for the problems discussed in this thesis is quite straightforward: for magnetometer calibration parameters estimation problem, the swarm is initialized in a certain range for the $\boldsymbol{K}_m$ matrix components and the $\boldsymbol{k}_m$ vector components, while for the spacecraft MCP evaluation using inverse MDM problem, the swarm is initialized in a certain range for the dipole(s) position $\hat{\boldsymbol{p}}_d$ and moment $\hat{\boldsymbol{m}}_d$ components, where the initialization region for each problem is represented by their respective $\boldsymbol{pmax}$. As a rule of thumb of setting the values of this initialization conditions, the initialization range for each component $\boldsymbol{pmax}_j$ is estimated around the realistic expected value of the solution, e. g. a range of $[-1, 1]$ for the $\boldsymbol{K}_m$ matrix components or $[-0.2, 0.2]\,\mathrm{A\,m^2}$ for the $\hat{\boldsymbol{m}}_d$ components. Note that the initialization ranges written so far are symmetrical and centered on zero for simplicity, although this may change in implementation. In the other hand, initialization of the swarm velocities is performed in a similar manner as the swarm positions, although the initialization range is defined by $\boldsymbol{vmax}$, which will also be used for the boundary conditions. This initialization process can be written



(a) Plot of the values of one component in the swarm against the number of iterations under standard parameters dynamics. Each component value is denoted with dashed line, and their global best value is denoted with solid line.



(b) Plot of the values of one component in the swarm against the number of iterations under extended parameters dynamics. Each component value is denoted with dashed line, and their global best value is denoted with solid line.



(c) Plot of swarm global best fitness F_g against the number of iterations under the standard (denoted with dashed line) and extended (denoted with solid line) parameters dynamics.

Figure 12: A simulation of swarm position evolution in the search space and its corresponding global best fitness, showing the difference in PSO algorithm performance under the standard parameters dynamics (from fig. 7) and the extended parameters dynamics (from fig. 11). Only the evolution of one component j from the swarm, which showed the most significant changes in behavior for different parameters dynamics, is shown for brevity.

mathematically as

$$p_{ij}(k=0) = pmax_{s,j} + \text{rand}(0,1)\,(pmax_{e,j} - pmax_{s,j})\,, \text{ and} \tag{27a}$$
$$v_{ij}(k=0) = -\,vmax_j + \text{rand}(0,2)vmax_j, \tag{27b}$$

where $\boldsymbol{pmax}_j$ and $vmax_j$ are the individual components of $\boldsymbol{pmax}$ and $\boldsymbol{vmax}$ for each swarm component j, respectively, with the structure $\boldsymbol{pmax}_j = [pmax_{s,j}, pmax_{e,j}]$ defining the start and end of swarm position initialization range and $vmax_j$ defining the maximum allowed velocity of the swarm. Note that $\boldsymbol{pmax}$ can define the swarm positions within any range in the search space, while $\boldsymbol{vmax}$ can only define the swarm velocities symmetrically, centered at zero.

The boundary conditions can be implemented in several ways, either imposed on the swarm positions, velocities, or both. The basic boundary condition which is implemented in this thesis is the swarm velocity limit represented by $\boldsymbol{vmax}$, whose implementation is described in algorithm 3, without imposing direct restrictions on

Algorithm 3: Procedure for limiting swarm velocities.

```
1:  procedure LimVel(v_pso, vmax)
2:      for i = 1...n_p do                          ▷ each particle in the swarm
3:          for j = 1...n_c do                      ▷ each component of the swarm
4:              if |v_pso,ij| > vmax_j then         ▷ over speed limit
5:                  v_pso,ij = sign(v_pso,ij) vmax_j   ▷ force to maximum speed
6:              else                                ▷ inside speed limit
7:                  v_pso,ij = v_pso,ij             ▷ no change
8:              end if
9:          end for
10:     end for
11:     return v_pso
12: end procedure
```

the swarm position using $\boldsymbol{pmax}$ which is only used for defining the initialization conditions discussed previously. A good starting point to define the velocity limit is by setting each component of $\boldsymbol{vmax}$, $vmax_j$, to a value equal with the range of its respective $\boldsymbol{pmax}_j$ (i. e., $vmax_j = \frac{pmax_{e,j} - pmax_{s,j}}{2}$), thus enabling the swarm to explore the region outside its initialization range to a certain extent. This is useful if the actual optimal solution happens to lie outside the initialization range, while limiting the possibilities of swarm explosion [13, 71]. However, some preliminary evaluation of the algorithm shows some advantage in setting the $\boldsymbol{vmax}$ much lower than its respective $\boldsymbol{pmax}$, where the solution reached by the end of the iterations is more likely to converge to the global optimal solution known in the simulated data, especially when the optimal solution still falls near the initialization region, although this characteristic still varies between runs and for different simulated model of the problems. As a rule of thumb, the algorithm is evaluated further and checked for consistency using both rules of $\boldsymbol{vmax}$, which are

$$vmax_j = \frac{pmax_{e,j} - pmax_{s,j}}{v_{\text{lim}}} \text{ with } v_{\text{lim}} = 2 \text{ and } v_{\text{lim}} \gg 2, \tag{28}$$

on real experimental data. Note that v_{lim} is inversely proportional with the velocity limit imposed on the swarm.

6.4 Refinement Procedure

Refinement procedure was first proposed in [14] for assisting the swarm in escaping suboptimal solutions in inverse MDM problem, which is basically re-running the algorithm with the initialization region $pmax_j$ imposed on each swarm component j set to an a priori knowledge taken from its respective solution of the previous PSO run $gbest_j(k)$, where each refinement run optimizes the parameters for only one dipole while the rest of the dipoles parameters are fixed to the initial values. In this thesis, a more general variation of this refinement procedure is adapted, where each refinement run can optimize all components in the swarm without the association with dipole parameters structure in inverse MDM problem, making the procedure applicable to the magnetometer calibration parameters estimation problem. An overview of this refinement procedure process is described in algorithm 4. Note that a new constant $p_{0,j}$, which value is determined manually, is defined as the approximated search space around the a priori solution $gbest_j$ for each component j.

Algorithm 4: General version of refinement procedure adapted from [14].

▷ The refinement procedure requires knowledge of the previous PSO run solution for an a priori values of its initialization conditions.

1: **procedure** REFINE($\boldsymbol{gbest}$)
2: **for** $j = 1 \dots n_c$ **do** ▷ for every swarm component j
3: $\boldsymbol{pmax}_j = [gbest_j - p_{0,j}\ ,\ gbest_j + p_{0,j}]$ ▷ define the initial position range
4: $vmax_j = \frac{pmax_{e,j} - pmax_{s,j}}{v_{\text{lim}}}$ ▷ define the velocity limit
5: **for** $i = 1 \dots n_p$ **do** ▷ for every swarm particle i
6: initialize p_{ij} and v_{ij} using eq. (27)
7: $\boldsymbol{p}_i|_{i=\text{randi}(1,n_p)} = \boldsymbol{gbest}$ ▷ Set one random particle in the swarm equal to previous global best.
8: evaluate initial $\boldsymbol{pbest}_i$ and $\boldsymbol{gbest}$ ▷ see lines 4–6 in algorithm 1
9: **end for**
10: **end for**
11: ...

▷ Continue with PSO iterations as in algorithm 5 starting from line 8.

12: **end procedure**

In principle, the refinement procedure serves to 'shake' the swarm from the previously reached solution by limiting the initialization region of the swarm, which in turn will assist the swarm in escaping possible local minima. Technically, this is done by introducing a solution bias into the new swarm by setting a small value for $\boldsymbol{p}_0$, generating a swarm around the a priori solution from previous PSO run with even smaller $\boldsymbol{vmax}$ from the previous run, since $\boldsymbol{vmax}$ is defined by the total range of $\boldsymbol{pmax}$ itself, as defined in eq. (28). Additional step might be implemented to ensure that the fitness value of the refinement procedure solution will at least have the same

value as the previous solution, by setting the values of an arbitrary particle from the swarm with the global best value of the previous solution described in algorithm 4 line 7.

6.5 Algorithm Validation with Simulated Data

Testing the calibration algorithm using simulated data is important for verifying its validity. In simulated data, the result of the calibration procedure can be compared directly to the model, because all disturbances and error models are defined in the simulation model. This is not possible with real, experimental data because the exact model parameters are originally unknown, except for some a priori assumptions. This is also important in interpreting the fitness value produced by the PSO solution, since the scale of the fitness value depends on the problem and its parameters definition. Thus, the purpose of the following sections is twofold: demonstrating the performance of the specific algorithm developed in this thesis as well as evaluating the proper threshold for the fitness value in a realistically defined problem.

For reference, a complete description of the developed PSO algorithm (improved from the basic version in algorithm 1) implemented in this thesis is given in algorithm 5. The parameters setting for both problems share some common values, which are listed in table 3. More problem-specific parameters are discussed in the next sections.

Algorithm 5: Developed PSO algorithm.

1: **define** PSO parameters and termination conditions
2: $w = w_s,\ c_1 = c_{1s},\ c_2 = c_{2s}$ ▷ initial PSO parameters
3: **for** $i = 1 \dots n_p$ **do**
4: initialize $\boldsymbol{p}_i$ and $\boldsymbol{v}_i$ using eq. (27) ▷ initial swarm properties
5: $\boldsymbol{pbest}_i = \boldsymbol{p}_i$ ▷ initial local best positions
6: **end for**
7: evaluate $\boldsymbol{gbest}$ with eq. (14b) ▷ initial global best position
8: **while** *iterate* $= 1$ **do** ▷ *iterate* set to zero if termination conditions are fulfilled
9: $k = k + 1$ ▷ update time step k
10: update $w, c_1,$ and c_2 ▷ according to parameters dynamics in fig. 7 or fig. 11
11: **for** $i = 1 \dots n_p$ **do** ▷ each particle in the swarm
12: update $\boldsymbol{v}_i$ using eq. (13a) ▷ swarm velocities
13: $\boldsymbol{v}_i = \text{LimVel}(\boldsymbol{v}_i,\ \boldsymbol{vmax})$ ▷ limit swarm velocities; see algorithm 3
14: update $\boldsymbol{p}_i$ using eq. (13b) ▷ swarm positions
15: calculate F_i with the defined fitness functions ▷ particle fitness
▷ For magnetometer calibration parameters estimation problem, use the fitness functions eqs. (19)–(21). For inverse MDM problem, use the fitness functions eqs. (24)–(26).
16: reevaluate $\boldsymbol{pbest}_i$ using eqs. (14a) and (14d) ▷ particle local best position
17: **end for**
18: reevaluate $\boldsymbol{gbest}$ using eqs. (14b) and (14d) ▷ swarm global best position
▷ Now, the termination conditions are checked: if k reaches allowed maximum $K_{\max}$, or F_g is better than threshold F_{TH}.
19: **if** $F_g < F_{\text{TH}}$ **then**
20: *iterate* $= 0,\ success = 1$
21: **else if** $k \geq k_{\max}$ **then**
22: *iterate* $= 0,\ success = 0$
23: **else**
24: *iterate* $= 1,\ success = 0$
25: **end if**
26: **end while**
27: **if** *success* $= 0$ **then** ▷ if previous run did not reach fitness threshold
28: **while** $refine = 1$ **do** ▷ repeat refinements until certain conditions reached
29: Refine($\boldsymbol{gbest}$) ▷ run a refinement loop
▷ Check the solution of refinement procedure as in lines 19–25.
30: **if** $F_g < F_{\text{TH}}$ **then** ▷ if fitness is below threshold or cannot be improved
31: $refine = 0$
32: **else**
33: $refine = 1$
34: **end if**
35: **end while**
36: **end if**

Table 3: Common values for PSO parameters and constants used in this thesis. Some values are not fixed and may be evaluated separately in order to check the algorithm for consistency and to identify key values associated with global optimal solution.

PARAMETER	VALUE	DESCRIPTION
$k_{\max}$	$500 \ldots 4000$	The typical maximum allowed number of iterations.
n_p	$30 \ldots 60$	Typical number of particles in the swarm.
w_s	$3 \ldots 5$	Typical starting value for inertia weight w.
w_f	0.5	Final value for w in the end of the varying phase.
w_e	0.382	End value for w after the varying phase.
c_{1s}	4	Starting value for cognitive rate c_1.
c_{2s}	0	Starting value for social rate c_2.
$c_{1f},\ c_{2f}$	2	Final value for c_1 and c_2 in the end of the varying phase.
$c_{1e},\ c_{2e}$	2.5	End value for c_1 and c_2 after the varying phase.
w_v	$(0.05 \ldots 0.4)k_{\max}$	Typical iteration length for one cycle of varying w.
c_v	$(0.05 \ldots 0.4)k_{\max}$	Typical iteration length for one cycle of varying c_1 and c_2.
$v_{\lim}$	$2 \ldots 30$	Typical speed limit modifier from eq. (28).

6.5.1 PSO Evaluation for Magnetometer Calibration

The PSO algorithm for magnetometer calibration parameters estimation is evaluated using the magnetometer model discussed in section 3.4.1 combined with the environmental model discussed in section 5.2. The algorithm is tested on several type of simulated data, differentiated by their number of circular measurement loci n_l and whether the magnitude of simulated ambient magnetic field is constant for all measurements or not. The latter variation is tested in order to demonstrate the capability of the algorithm in estimating the calibration parameters using a set of measurement data taken during long orbital period, where the magnitude of geomagnetic field $|\boldsymbol{b}|$ might vary as modeled in section 5.2 (on the designated orbit, $|\boldsymbol{b}|$ varies from 19 502 to 56 203 nT). For the case of constant geomagnetic field, $|\boldsymbol{b}|$ is set to 35 000 nT (approximately in the middle of the expected range of geomagnetic field magnitude), and a measurement random noise with maximum magnitude 20% of the simulated measurement data is added to the data. Other values of noise levels are also tested for evaluating the effect of noise to the algorithm accuracy. The combinations of tested model variations are (simulated noise level is 20% for all unless stated otherwise):

1. two simulated, full circle measurement loci with constant $|\boldsymbol{b}|$;
2. three simulated, full circle measurement loci with constant $|\boldsymbol{b}|$;
3. same simulation model as item 2, but with maximum noise level 0% and 50% of the measurement data; and
4. two simulated, full circle measurement loci with varying $|\boldsymbol{b}|$ in the range of 19 502 to 56 203 nT gradually increasing from the first measurement until the last one.

The estimated calibration parameters from each variation and its comparison with the true values from the simulated model is given in table 5. A graphical representation of the measured, calibrated, and true magnetic field vector locus is given in figs. 13a and 13b, which, for brevity, is taken from only two of the model variants, specifically the variant in item 2 and item 4, respectively. The earlier is selected since the three circle loci with constant ambient magnetic field is the closest model with the type of data taken from the experimental test of magnetometer calibration discussed in section 7.3, while the latter is selected to demonstrate the capability of the algorithm in solving problem with non-circular locus of varying magnetic field. The results presented are estimated with the problem-specific PSO parameters set to values given in table 4 with the number of measurement data n_s is modeled as 100, representing realistic number of measurement data taken in the experimental test, while $pmax_{j=1\ldots9}$ and $pmax_{j=10\ldots12}$ are the position boundary for the calibration matrix $\boldsymbol{K}_m$ and calibration vector $\boldsymbol{k}_m$ components, respectively. The values for $pmax_{j=1\ldots9}$ and $pmax_{j=10\ldots12}$ are assigned with different values, where the initialization region for $\boldsymbol{K}_m$ is set to 1, so that the swarm begins the search in the solution space from random values in the range of $[-1, 1]$, while for $\boldsymbol{k}_m$, the search begins from random values in the range of $[-5\,000, 5\,000]$ nT, since the calibration parameter values are expected around those figures. The results are obtained with varying numbers of refinement procedure loops, typically one to three loops.

The simulation results in table 5 show that the algorithm successfully estimated the calibration parameters for all simulation model with minimum error. As expected, the model with 50% noise level, the highest in this comparison, yields the worst result as indicated by its high fitness value F_g. However, the F_g value itself is not comparable between different models, in this case for models with different noise level: in this comparison, $F_g = 481.8$ for a model with 50% noise level still estimate acceptable value of calibration parameters (the heading error improved from 9.15–72.97° before calibration to 0.21–38.18° after calibration), but for a model with 20% noise level , such a high F_g indicates very erroneous calibration parameters estimate. For example, a PSO run, with the initialization region deliberately set to some wrong values on a simulation model with $n_l = 3$, constant $|\boldsymbol{b}|$, and 20% noise, yields the results $\boldsymbol{S}_{m_c} = \begin{bmatrix} 10.1 & -10.7 & 12.1 \\ -7.1 & 14 & -17 \\ 23.6 & -30.6 & 45.6 \end{bmatrix}$ and $\mathbf{off}_{m_c} = \begin{bmatrix} \text{-}6.15 \\ 7.64 \\ \text{-}19.8 \end{bmatrix} \times 10^5$ [nT] which are very different from the model parameters, although the fitness value is $F_g = 490.4$. It can

Table 4: Specific PSO parameters for magnetometer calibration parameters estimation problem. Unspecified parameters follow the common parameter values in table 3.

PARAMETER	VALUE
$k_{\max}$	4000
n_s	100
w_s	55
w_v	$0.075k_{\max}$
c_v	$0.1k_{\max}$
$v_{\lim}$	10
$pmax_{j=1\ldots9}$	1
$pmax_{j=10\ldots12}$	5×10^3

be seen that using F_g to estimate the absolute accuracy of the PSO is not reliable, especially in real experiment data where the error model of the magnetometer and its environment is not completely known. However, it is useful to determine the relative accuracy between different PSO runs where the data still contain identical error model. Moreover, detecting large error in the calibration parameters estimate is possible, even for experimental data where the true model is not known, by inspecting the graphical representation of the calibrated measurement locus. A good calibration result of the magnetic field vector data should have a locus on the surface of a sphere, while the erroneous data will have an ellipsoid locus, as demonstrated in fig. 13a.

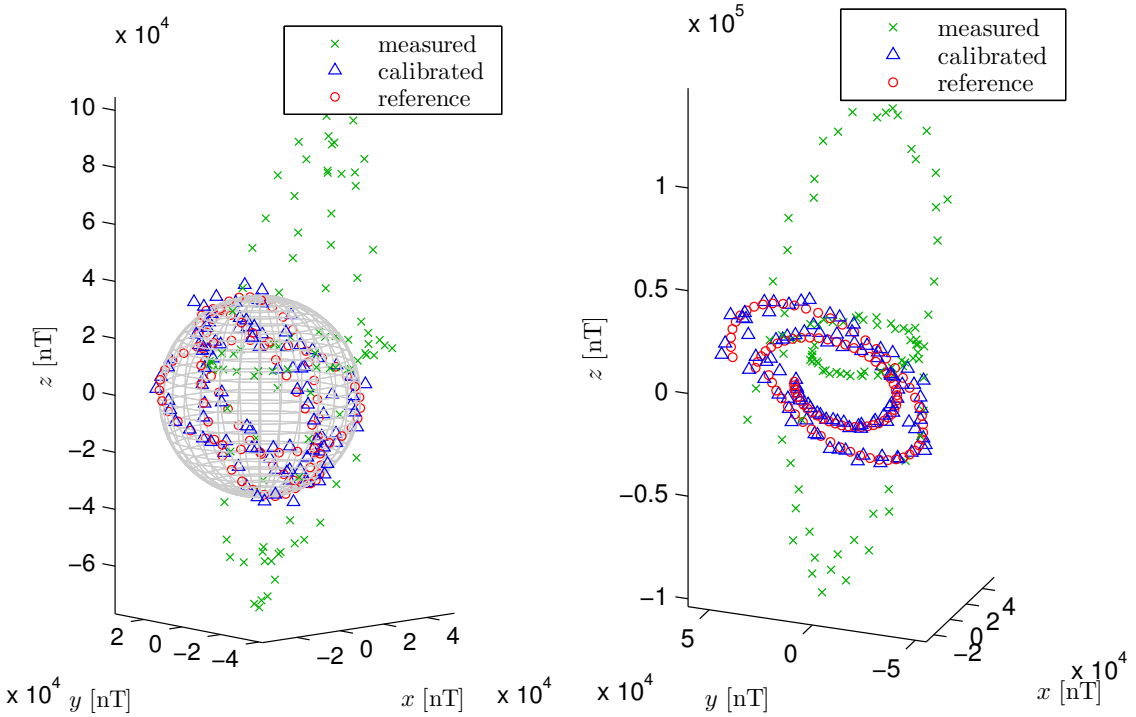
Varying ambient magnetic field magnitude. The simulation results also show that the PSO algorithm is capable of estimating the calibration parameters from set of measurement data with varying ambient magnetic field as depicted in fig. 13b, where the measurement loci are shaped like spirals instead of circles due to the gradually increasing ambient magnetic field—different locus shape is possible as the magnetic field magnitude varies up and down. Such data might be collected on orbit, especially when the data is collected from a long flight period, where the geomagnetic field magnitude changes according to the position of the spacecraft in Earth reference frame. However, during such long flight period, different parameters that contribute to the measurement errors might change (e. g., varying magnetic field bias from spacecraft electronics). Since the current magnetometer model does not include those time-varying components, whether the PSO algorithm can compensate for those unknown errors to estimate the calibration parameters of the magnetometer using data from a long flight period should be investigated in future works.

Table 5: Comparison of estimated model parameters from different simulation models in relation to the true reference value.

MODEL VARIANT	MODEL PARAMETERS				F_g
	$\boldsymbol{S}_{m_c}$			$\mathbf{off}_{m_c}$ [nT]	
Model reference	$\begin{bmatrix} 0.749 & 0.354 & 0.677 \\ -0.242 & 1.122 & -0.209 \\ -0.395 & -1.043 & 2.433 \end{bmatrix}$			$\begin{bmatrix} 13\,578.7 \\ -3\,825.7 \\ 13\,252 \end{bmatrix}$	
$n_l = 3$, $\lvert\boldsymbol{b}\rvert = 35\,000$ [nT], 0% noise	$\begin{bmatrix} 0.716 & 0.392 & 0.692 \\ -0.302 & 1.108 & -0.206 \\ -0.376 & -1.081 & 2.42 \end{bmatrix}$			$\begin{bmatrix} 13\,578.7 \\ -3\,825.7 \\ 13\,252 \end{bmatrix}$	2.3319
$n_l = 3$, $\lvert\boldsymbol{b}\rvert = 35\,000$ [nT], 20% noise	$\begin{bmatrix} 0.736 & 0.348 & 0.671 \\ -0.244 & 1.137 & -0.21 \\ -0.4 & -1.022 & 2.409 \end{bmatrix}$			$\begin{bmatrix} 12\,656 \\ -3\,427.1 \\ 12\,566.6 \end{bmatrix}$	281.1149
$n_l = 3$, $\lvert\boldsymbol{b}\rvert = 35\,000$ [nT], 50% noise	$\begin{bmatrix} 0.81 & 0.295 & 0.71 \\ -0.277 & 1.139 & -0.206 \\ -0.383 & -1.031 & 2.657 \end{bmatrix}$			$\begin{bmatrix} 14\,771 \\ -3\,220.9 \\ 10\,935 \end{bmatrix}$	481.8039
$n_l = 2$, $\lvert\boldsymbol{b}\rvert = 35\,000$ [nT], 20% noise	$\begin{bmatrix} 0.731 & 0.367 & 0.647 \\ -0.24 & 1.107 & -0.237 \\ -0.408 & -1.054 & 2.494 \end{bmatrix}$			$\begin{bmatrix} 13\,839.8 \\ -3\,514.9 \\ 14\,167.8 \end{bmatrix}$	211.9915
$n_l = 2$, $\lvert\boldsymbol{b}\rvert$ =19 502–56 203 [nT], 20% noise	$\begin{bmatrix} 0.735 & 0.364 & 0.644 \\ -0.283 & 1.128 & -0.157 \\ -0.455 & -1.008 & 2.336 \end{bmatrix}$			$\begin{bmatrix} 13\,651.2 \\ -2\,699.3 \\ 9\,637.9 \end{bmatrix}$	275.6383

6.5.2 PSO Evaluation for Inverse MDM

Inverse MDM problem is a highly nonlinear problem with numerous factors that increase its complexity, e. g.number of dipoles, noise in the measurements, and number of measurement points. The number of dipoles assigned to PSO for estimation ($\hat{n}_d$) seriously affects the nonlinearity of the problem, since each dipole has to be modeled with six independent components. The actual number of dipoles in the simulation model (n_d) itself can be different from the estimated dipoles ($\hat{n}_d$), although it is expected that the best result will be obtained when n_d and $\hat{n}_d$ are equal [14]. The noise presence in the measurement data will certainly affect the accuracy of the dipoles estimation, just as demonstrated previously for the magnetometer calibration parameters estimation problem in section 6.5.1. The number of measurements n_s of the spacecraft remanent magnetic field vector data $\check{\boldsymbol{b}}_s$ as well as the corresponding position of the measurement points $\boldsymbol{p}_s$ used as the set of data in the PSO algorithm will also affect the accuracy of the estimation results. In this algorithm validation for inverse MDM problem, the number of measurements and their corresponding measurement point positions can be modeled from the actual test described later in section 7.3.2, where the number of measurements can be approximated as $n_s = 100$,



(a) Data with three measurement loci, constant ambient magnetic field magnitude, and 20% noise level.

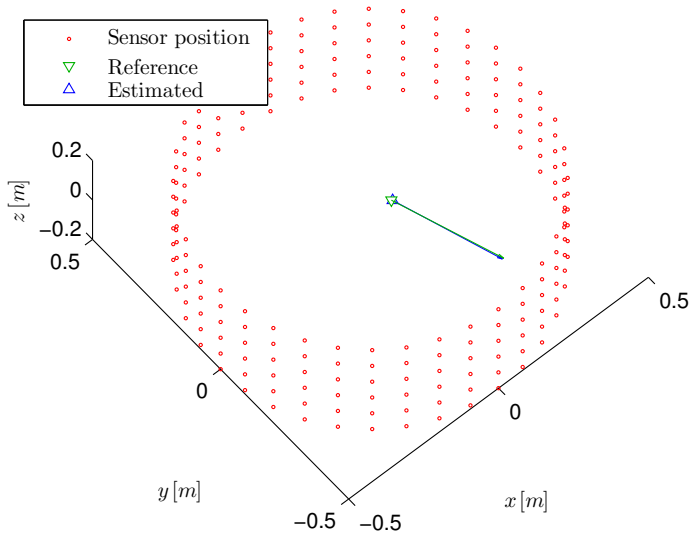
(b) Data with two measurement loci, gradually increasing ambient magnetic field magnitude, and 20% noise level.

Figure 13: 3-D plot of magnetic field vectors comparing the measured (denoted with ×'s), calibrated (denoted with △'s), and true (denoted with ○'s) magnetic field vector data points from simulated data with constant and varying magnetic field. The axes represent the magnetic field strength in nT from the magnetometer frame.

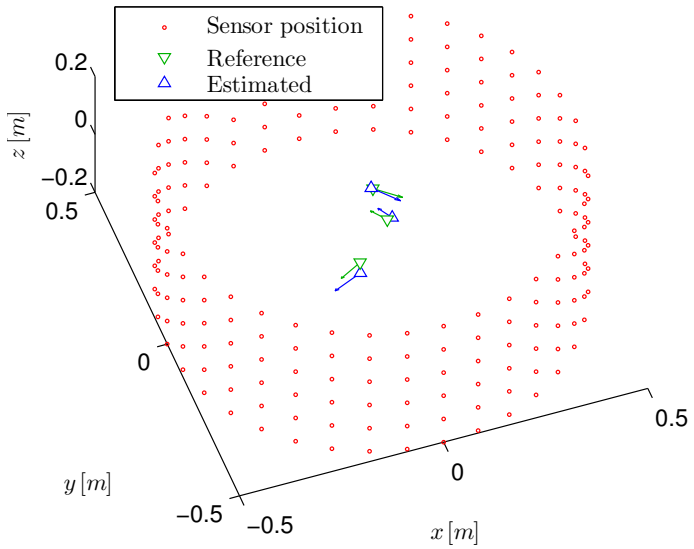
while the positions of those measurement points are evenly distributed around the spacecraft body frame, making a cylinder shape with a radius of 50 cm centered at the spacecraft body frame origin. Simulation results with varying noise levels and numbers of dipoles in the simulation model are presented in this section for analyzing the effects of noise level and the modeled number of dipoles to the performance of the algorithm.

Estimating a known number of dipoles. The increase in problem nonlinearity caused by the number of estimated dipoles can be seen from the results presented in fig. 14, where the estimation results are relatively accurate—in fact, for one dipole estimation, the estimation results are practically identical with the model (maximum error of 2 mm for dipole positions and 5 mA m^2 for dipole moments estimations). This result came from a single PSO run with 0% noise in the simulated data. However, estimating a three dipoles simulation model (with the PSO set to estimate the matching number of dipoles, or $\hat{n}_d = n_d$) and producing accurate result (although still comparably lower than the one dipole case) such as in fig. 14b requires much more extensive PSO runs compared to achieving similar results on one dipole simulation

model.



(a) One simulated dipole estimated by PSO with $F_g = 0.0253$.



(b) Three simulated dipoles estimated by PSO with $F_g = 0.0425$.

Figure 14: 3-D plots of the estimated (denoted with $\triangle$'s) and reference (denoted with $\triangledown$'s) dipoles positions from one dipole and three dipoles simulation models with the matching number of dipoles estimate assigned to the PSO ($\hat{n}_d = n_d$). The magnetic moments of the dipoles are denoted with arrows originating from the respective dipole positions, and the measurement points for collecting the data are denoted with $\circ$'s. The axes represent positions from the spacecraft frame in m. Both models simulate data with 0% noise.

In the three dipoles simulation model, the first PSO runs were very likely led into suboptimal solutions, which can be corrected by extensive refinement procedure. Refining the result after the first PSO run was performed with the general version of

refinement procedure explained in section 6.4, where all the estimated dipoles went through a finer PSO search by manipulating the initialization region using the a priori knowledge from the previous PSO run. If the result still converge in suboptimal solution, dipole-specific refinement procedure, which refines only one dipole in each run as originally proposed in [14], is executed alternatingly for different dipoles using different parameters (e. g. different parameters dynamics, $k_{\max}$, a priori initialization range $\boldsymbol{p}_0$ for the refinement loops, and velocity boundary modifier $v_{\lim}$) with typical values fall in the range described in table 3. The refinement procedure, both the general and the dipole-specific version, is repeated as necessary and every PSO run with better fitness value is used as the initialization a priori for the next run until the result is accurate enough.

Effect of noise level in algorithm performance. Increasing the noise level also reduces the best achievable accuracy of the algorithm. Simulations using data containing different noise levels to estimate three simulated dipoles, with the results shown in table 6, show that with increasing noise level (measured as maximum noise magnitude relative to the simulated data), the best achievable F_g and the accuracy of the estimation become worse.

Table 6: Simulated PSO performance for inverse MDM problem with different noise levels in the data. All simulations estimate three magnetic dipoles ($n_d = 3$) with the matching number of dipoles assigned to the PSO ($\hat{n}_d = 3$).

MODEL VARIANT	ERROR RANGE & AVERAGE				
	$\tilde{\boldsymbol{p}}_d$ [cm]	$\bar{\tilde{\boldsymbol{p}}}_d$ [cm]	$\tilde{\boldsymbol{m}}_d$ [mA m^2]	$\bar{\tilde{\boldsymbol{m}}}_d$ [mA m^2]	F_g
0% noise	0.03–2.94	0.87	24.5–57.4	30.5	0.0425
20% noise	1.4–6.73	2.52	16.1–105.1	50.8	0.1339
50% noise	1.29–5.08	2.43	4.1–138.5	73.6	0.4414

Estimating unknown number of dipoles. Determining the accuracy of the results directly to the reference dipoles is only possible in simulated data where the dipoles model is known, while in real experimental data, the actual dipoles model—both the parameters ($\boldsymbol{p}_d$ and $\boldsymbol{m}_d$) and the number of dipoles (n_d)—is not known and relying on the fitness values to determine the actual accuracy of the estimation results is not practical because the fitness values may have different scales for different data conditions and PSO parameters, although some assumptions on the actual dipoles model in experimental setting is possible: if the spacecraft subsystems or modules which are known to generate magnetic interference are tested individually, then the MCP analysis for the complete spacecraft using inverse MDM problem can be performed with the assumption that each subsystem/module represents one dipole, whose dipole model can be initially assumed equal with the results of the respective individual tests. However, such scrutiny in MCP tests for this thesis work

is not feasible because the spacecraft MCP analysis is performed directly with the whole assembled spacecraft. Thus, another approach in assessing the quality of the spacecraft residual magnetic dipoles estimation is used.

First of all, assessing the quality of spacecraft magnetic properties estimate is related to the purpose of the MCP analysis itself. It has been argued previously in section 3.4.3 that the purpose of modeling the spacecraft as multiple magnetic dipoles in the scope of this thesis is for mitigating the effects of the spacecraft RMM, which could negatively affect the mission. The specific part of the system that is negatively affected is the ADCS performance, where the disturbance magnetic torque will disturb the spacecraft attitude and its generated magnetic field will disturb the reading on the magnetometer. The magnetic field bias in magnetometer reading caused by the spacecraft RMM itself is already included in the magnetometer calibration parameters as demonstrated in section 6.5.1, while the disturbance magnetic torque information can only be obtained from this approach. The disturbance magnetic torque itself is an important information in the spacecraft design process because it has to be guaranteed that the disturbance torque experienced by the spacecraft can be compensated by the attitude actuators to ensure full controllability of the spacecraft attitude. Since the disturbance magnetic torque can be calculated with

$$\boldsymbol{\tau}_{\text{dist},m} = \sum_{d} \boldsymbol{m}_d \times \boldsymbol{b},$$

similar with the torque produced by magnetorquer in eq. (9) where $\boldsymbol{b}$ is the ambient magnetic field experienced by the spacecraft and $\sum_d \boldsymbol{m}_d$ is the total RMM of the spacecraft, then the maximum magnetic dipole that can be produced by the attitude actuator (in this case, the magnetorquer) has to be larger than the spacecraft RMM. Thus, the minimum information which is of interest from the MCP analysis result is the total magnetic dipoles moment magnitude of the spacecraft RMM, while the position and moment vector of each modeled dipole is of secondary interest.

To guarantee the accuracy of the estimated total RMM magnitude from the PSO algorithm, results from simulations with mismatched estimated number of dipoles and its real simulated number of dipoles ($\hat{n}_d \neq n_d$) are investigated. All the models simulate four or eight magnetic dipoles, whose data contain maximum 50% noise with different combinations of simulated and estimated dipoles number set in the PSO algorithm. The resulting estimated total RMM compared to the simulated total RMM is presented in table 7.

It can be seen from table 7 that the resulting total RMM is not as sensitive as the individual dipole properties to suboptimal solutions reached by the algorithm, even when there are mismatches between the estimated and the simulated number of dipoles. Generally, a better fitness value is observed as the number of estimated dipoles increase and reach the best value when $\hat{n}_d = n_d$, although its correlation with the accuracy of the total RMM estimate itself is not straightforward—in the case of the $n_d = 8$ simulation model, 4 dipoles estimate produce better result than 1 dipole estimate, which also shows in its fitness value, although for 8 and 10 dipoles estimate, the total RMM accuracy could be relatively lower than 4 dipoles estimate although the fitness value is better. This is expected, as more dipoles are modeled in

Table 7: Simulated PSO performance (with 50% noise in the data) in estimating the total RMM using inverse MDM approach for mismatched number of dipoles between the simulated (n_d) and estimated ($\hat{n}_d$) model.

MODEL		TOTAL RMM $[\mathrm{mA\,m^2}]$		
n_d	$\hat{n}_d$	$\sum_d \boldsymbol{m}_d$	$\lvert\sum_d \boldsymbol{m}_d\rvert$	F_g
4	*reference*	$\begin{bmatrix}57 & -49 & -159\end{bmatrix}^\top$	175.9	—
4	1	$\begin{bmatrix}54 & -49 & -146.1\end{bmatrix}^\top$	163.2	0.664
4	4	$\begin{bmatrix}56 & -46.5 & -156\end{bmatrix}^\top$	172.1	0.346
4	10	$\begin{bmatrix}49.5 & -49.4 & -150.6\end{bmatrix}^\top$	166	0.3836
8	*reference*	$\begin{bmatrix}72 & -183 & -171\end{bmatrix}^\top$	260.6	—
8	1	$\begin{bmatrix}70.3 & -175.1 & -173\end{bmatrix}^\top$	256.1	0.6736
8	4	$\begin{bmatrix}67.4 & -184.2 & -172.5\end{bmatrix}^\top$	261.2	0.4031
8	8	$\begin{bmatrix}87.5 & -173.1 & -163\end{bmatrix}^\top$	255.1	0.3791
8	10	$\begin{bmatrix}74 & -169.7 & -168.8\end{bmatrix}^\top$	250.5	0.4008

the estimate, reproducing the pattern of magnetic field vector on every measurement point by the estimated dipoles will become easier, although this also means that more possible solutions could exist where many of which are suboptimal from the perspective of individual dipole properties, affecting the accuracy of the total RMM estimate. Of course, theoretically, the 8 dipoles estimate will give the best result in total RMM accuracy, given that the swarm converge in the optimal region. However, the data shown in table 7 have only went through several refinement loops, in order to show the impact of suboptimal local minima in the solution space to the accuracy of the estimated total RMM. Thus, in analyzing the real experimental data, similar conduct of estimating the total RMM with different number of estimated dipoles has to be performed, and the results are checked for consistency.

7 Tests Setups, Procedures, and Results

7.1 Sun Sensor Calibration

In this thesis, only one DSS unit (depicted in fig. 15 with its frame definition) is tested to investigate any manufacturing defect, which covers two main problems:

a) direct estimation of calibration parameters defined in eq. (5) using least squares method by comparing the known Sun reference angle obtained in Sun simulator test with the measured elevation angle from the DSS; and

b) investigation on the characteristic of the light intensity reading from the DSS, which is expected to be dependent on the Sun angle. This is required because from all six sensors, angle measurement data from the sensor with highest light intensity value will be used in order to differentiate the light source coming from the Earth albedo, which can be modeled with known coordinate and orbit altitude as given by [9], with actual light from the Sun.

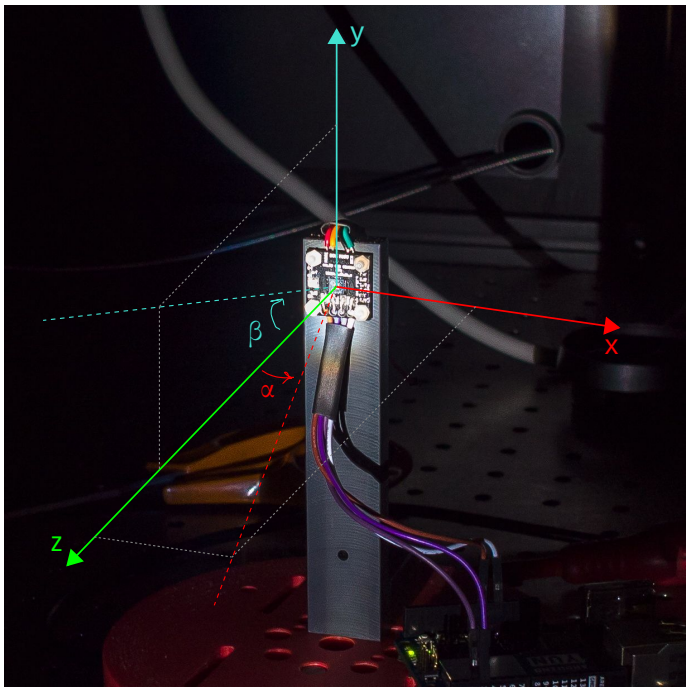


Figure 15: The digital Sun sensor module from Aalto-1 qualification model with its defined reference frame.

Test procedure. The test is performed with a Sun simulator setup that includes a single-axis rotation platform and a 'Sun lamp' fixed on a calibration table as depicted in fig. 16. The single-axis rotation platform is a CR1-Z7 motorized rotation stage from Thorlabs, whose angular displacement and velocity can be controlled digitally through TDC001 DC servo motor driver from the same company [85, 86]. The Sun

lamp is actually a xenon lamp that acts as a wide spectrum light source emulating the Sun, although the light intensity and the spectrum range is not identical with the Sun. The test procedure is as follows:

1. Position the DSS as such so that the beam from the Sun lamp is focused on the DSS surface. The goal is to simulate actual sunlight which comes from practically infinite distance: if the beam from the Sun lamp is too wide, the light beam might not be focused enough on a single sensor detector.

2. Adjust the orientation of the DSS as such that its z-axis points directly to the Sun lamp and that the rotation axis is aligned with the sensor x- or y-axis (both axes will be used).

3. Change the orientation of the DSS using the rotation stage by 2° steps in both direction while taking measurements of the Sun angle and intensity in every step.

4. Repeat step 3 until the Sun angle measurements does not increase/decrease as it should be, which means the Sun is out of the sensor effective FOV.

5. Repeat the step 2–4 with the rotation axis changed, either aligned with x- or y-axis depending on the previous alignment, by rotating the sensor on its mounting.

Test results and analysis. The plot of Sun angle α and β measured by the DSS against the reference angle, separately for variation in x- and y-axis, is shown in fig. 17. The reference line represents the ideal plot of the measured Sun angle and the reference angle. The nominal data is the non-calibrated Sun angle reading of the DSS that was converted from raw data into Sun angle using the conversion formula in eq. (4) with $\mathbf{FoV}_0 = -75°$, $\mathbf{FoV} = 150°$, and $2^m = 56$ derived from the DSS specifications [1]. The calibrated data is produced from calibrating the nominal data using eq. (5), with the calibration parameters obtained (shown in table 8) with linear batch least squares method described in section 4.1. Note that the measurement data used in the calibration algorithm is limited to the linear region of the data, which is in the range of $[-60, 60]°$, where the Sun angle reading starts to flat out as the light source is going outside the effective FOV of the DSS (theoretical FOV in DSS specification is $[-75, 75]°$ [1]), as evaluated from the nominal reading in fig. 17.

Table 8: Model parameters for DSS calculated with linear batch least squares algorithm.

SCALE $\boldsymbol{S}_{ss}$	OFFSET $\mathbf{off}_{ss}$ [°]
$\begin{bmatrix} 0.989 & 0.0343 \\ -0.0187 & 1.0131 \end{bmatrix}$	$\begin{bmatrix} -1.2814 \\ -0.6497 \end{bmatrix}$

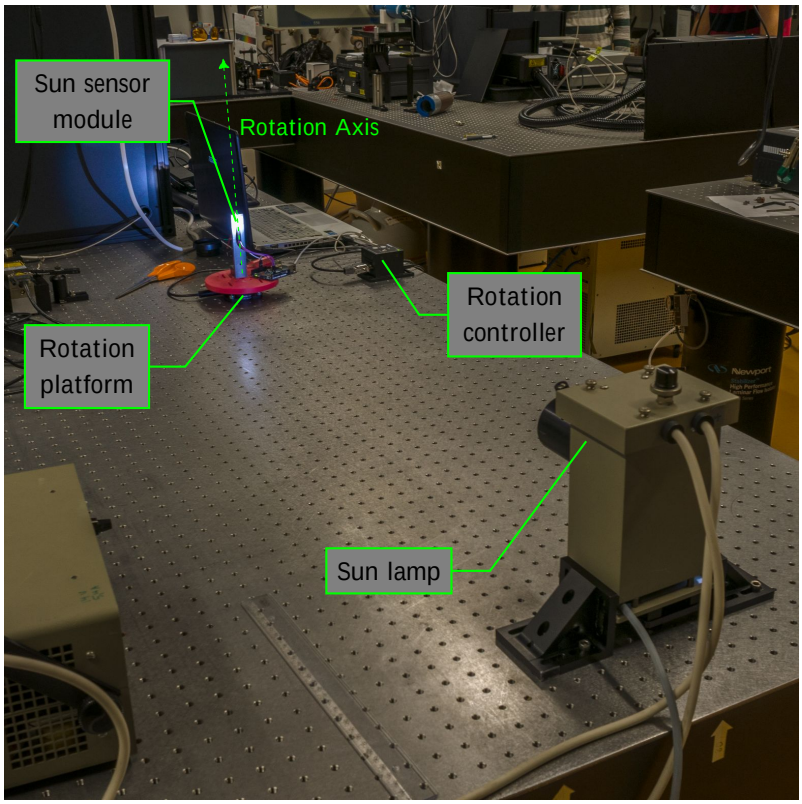
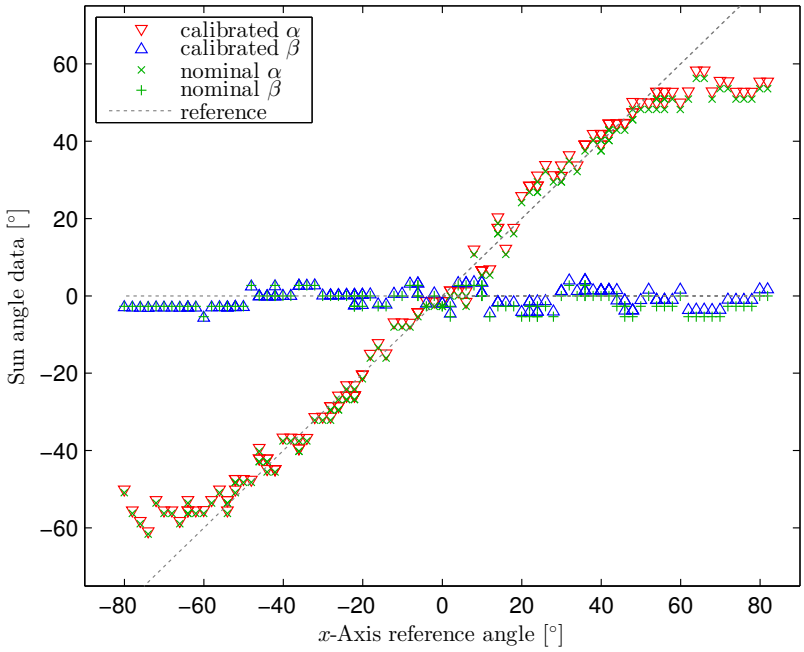


Figure 16: The Sun simulator setup in Aalto University. The single-axis rotation platform angular displacement is controlled digitally. All light sources except for the Sun lamp are turned off during the test, minimizing error in the Sun sensor reading. The distance from the Sun lamp to the sensor is 110 cm.

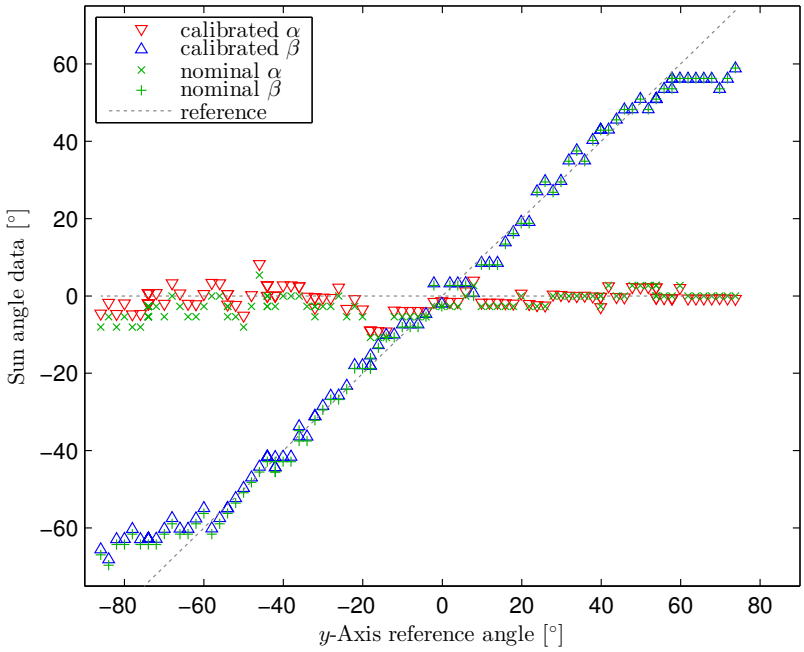
The results in table 8 and fig. 17 show that the calibration parameters have small impact to the Sun angle reading, since the nominal reading itself is already accurate with minimal scale factor (diagonal elements of $\boldsymbol{S}_{ss}$), misalignment (non-diagonal elements of $\boldsymbol{S}_{ss}$), and offset ($\mathbf{off}_{ss}$) errors. Numerically, the worst accuracy of the DSS in x-axis improved from 11.71° before calibration to 9.93° after calibration, while in y-axis it only improved from 7.89° before calibration to 7.18° after calibration. The average accuracy only improved from 2.81° before calibration to 2.57° after calibration for x-axis reading and from 2.03° before calibration to 1.98° after calibration for y-axis reading. The inaccuracies come from non-uniform distributed noise that is visible in fig. 17, which reduces the effective accuracy of the DSS into two or three times the native resolution (2.68°/LSB) of the sensor.

Besides the Sun angle calibration, the brightness reading of the DSS need to be evaluated in order to distinguish the light coming from the Sun and the one from Earth albedo. The plot of the brightness reading against the reference angle is presented in fig. 18, where the brightness reading obviously decreases as the light source deviates from the sensor normal direction. Note that the brightness reading is in raw value (LSB), which can be converted into light intensity with specified gain settings [25, 39]. However, this information is not crucial, since the Sun lamp

brightness in the test was not calibrated to exactly emulate the Sun in orbit, and the drop ratio of the brightness in relation with the Sun angle is sufficient. It can also be seen that the brightness sensor have small difference in sensitivity when the Sun angle is varied in the x- and y-axis, although this is not a major problem. At 60° deviation, the highest Sun angle in the linear region, the lowest brightness reading falls down to 35.84% of the highest brightness reading. This may pose a problem if other DSS facing the Earth directly measured a brightness value from the Earth albedo larger than the brightness reading from the Sun itself. The solution for this problem, whether by calibrating the brightness readings from the DSS before the values are compared or by reducing the effective Sun angle until the brightness level drop is on a safe level, will be implemented in the attitude estimation method outside of this thesis work.

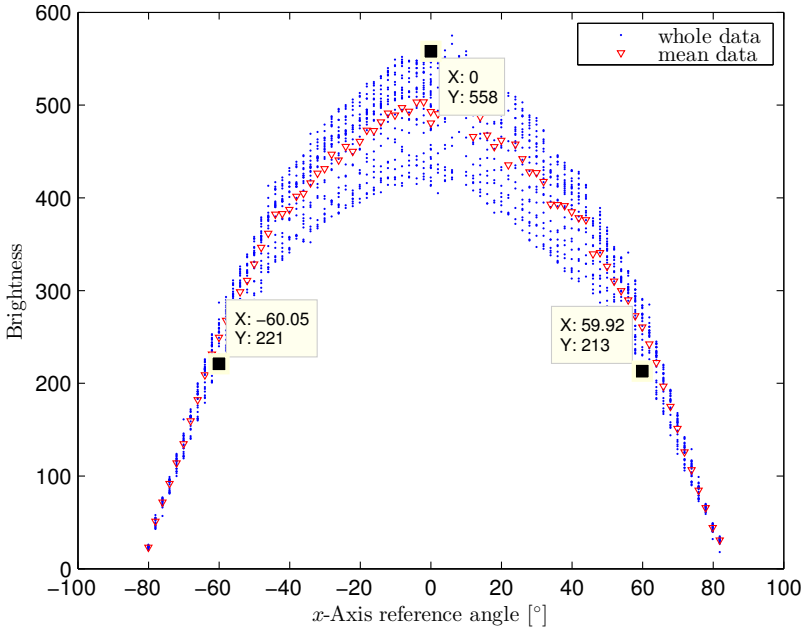


(a) Varying reference angle in x-axis with y-axis reference angle stays at 0°.

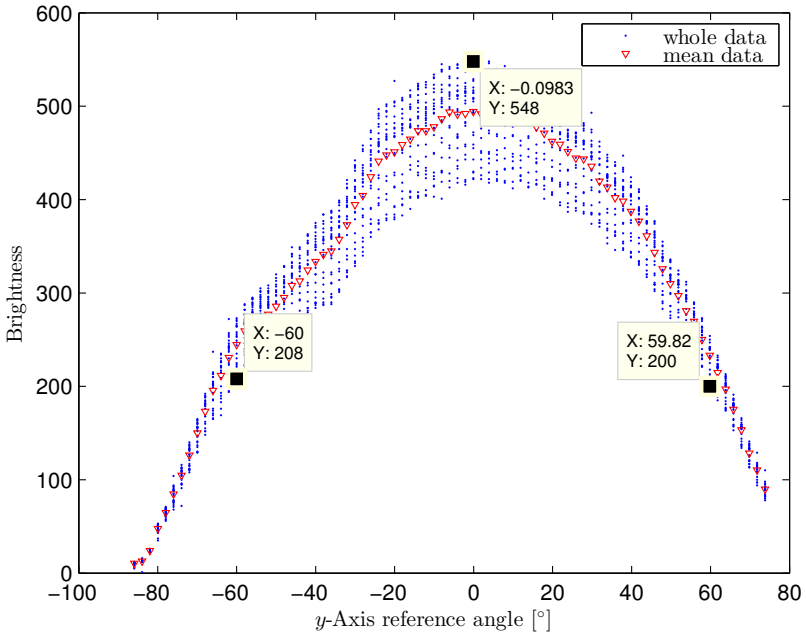


(b) Varying reference angle in y-axis with x-axis reference angle stays at 0°.

Figure 17: Plot of Sun angle (in °) against the reference angle (in °). The nominal Sun angle values in xz- and yz-plane are denoted with $\times$'s and $+$'s, respectively, while the calibrated values are denoted with $\triangledown$'s and $\triangle$'s, respectively. denoted Note that the linear region of the DSS FOV falls in the range of $[-60, 60]°$.



(a) Varying reference angle in x-axis, with y-axis reference angle stays at 0°.



(b) Varying reference angle in y-axis, with x-axis reference angle stays at 0°.

Figure 18: Plot of DSS brightness reading (in raw value (LSB)) against the reference angle (in °). The original data is denoted with •'s, while the data mean in each reference angle step is denoted with △'s.

7.2 Gyroscope Calibration

In this thesis, the gyroscope calibration procedure solves only the time-invariant part of the calibration parameters (i. e., scale factor $\boldsymbol{S}_g$ and initial offset $\mathbf{off}_g(0)$) because compensation for the time-varying part (i. e., $\boldsymbol{\eta}_{\mathrm{ARW}}$ and $\boldsymbol{\eta}_{\mathrm{RRW}}$) will be performed by the UKF in attitude estimation process. Aalto-2 ADCS has two gyros for redundancy, thus it is also possible to implement gyros array technique. However, this approach is not implemented in this thesis because the improvement in data quality from the implementation of gyros array depends on the number of gyros and their data correlation [8].

Test procedure. The gyroscope calibration test setup, pictured in fig. 19, consists of two gyroscopes mounted inside the Aalto-1 EM and a single-axis rotation platform identical to the one used in the DSS calibration setup from section 7.1.

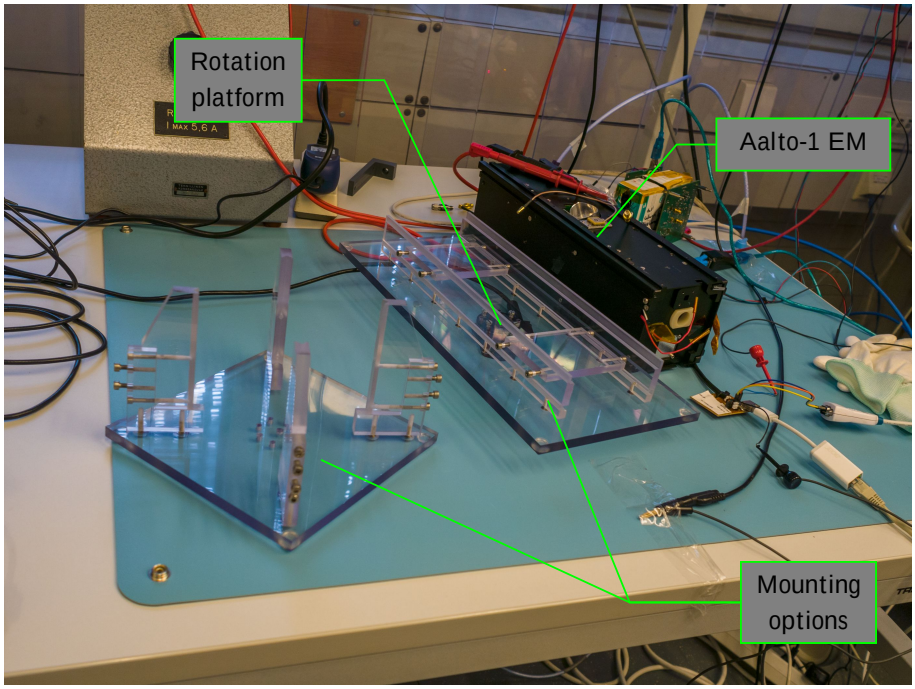


Figure 19: Gyroscope calibration test setup. The gyroscopes are installed inside the Aalto-1 engineering model and the single-axis rotation platform angular velocity is controlled digitally.

The calibration model is described in eq. (6), where the initial offset $\mathbf{off}_g$ and the total scale error $\boldsymbol{S}_g$ are estimated with linear batch least squares method explained in section 4.1 by taking series of measurements $\check{\boldsymbol{\omega}}$ while the gyroscope is rotated on a rotation platform with digitally controlled angular velocity. This angular velocity data commanded to the rotation platform is used as the reference angular velocity $\boldsymbol{\omega}$. The detailed test procedure is:

1. Position the spacecraft EM on the suitable mounting as such that the rotation axis of the rotation platform is aligned with the spacecraft body x-axis.

2. Collect gyroscopes readings $\breve{\boldsymbol{\omega}}_1$ and $\breve{\boldsymbol{\omega}}_2$ (there are two gyroscopes on the Aalto-2 EM) in stationary condition.

3. Set the rotation platform angular velocity to different values in the range of $\boldsymbol{\omega} = [-6, 6]\,[^\circ/\mathrm{s}]$, which is the maximum angular speed of the platform [86], while collecting measurements $\breve{\boldsymbol{\omega}}$ from the gyroscopes with the rotation platform angular velocity as the reference $\boldsymbol{\omega}$.

4. Repeat from step 1 with the rotation axis aligned to the other axes once at a time. Make sure sufficient number of angular velocity variation taken in step 3 is collected so that the estimation problem is overdetermined, i. e. the total number of angular velocity variation in all axes are larger than the number of unknown parameters, which is 12 in this case (3×3 matrix and 3×1 vector).

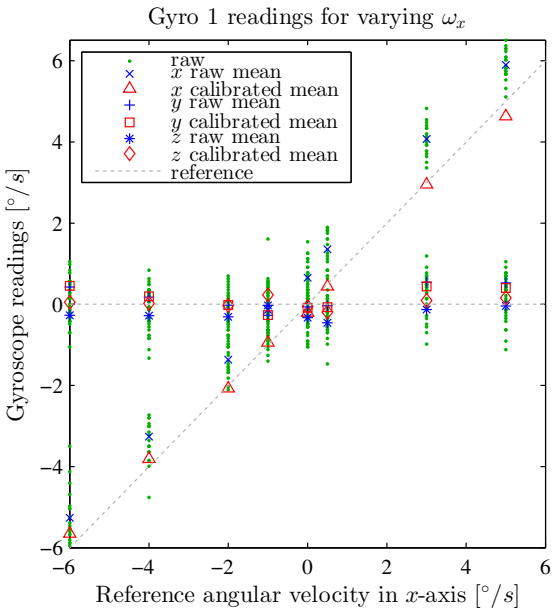
Test results and analysis. The gyroscope model parameters defined in eq. (6a) obtained with least squares method from the measurement data for each gyroscope are given in table 9. The plots of gyroscopes reading before calibration ($\breve{\boldsymbol{\omega}}$) and after calibration ($\hat{\boldsymbol{\omega}}$) against the reference angular velocity $\boldsymbol{\omega}$ are shown in figs. 20 and 21 for gyroscope 1 and 2, respectively.

Table 9: Model parameters for calibrating each gyroscope as calculated with linear batch least squares algorithm.

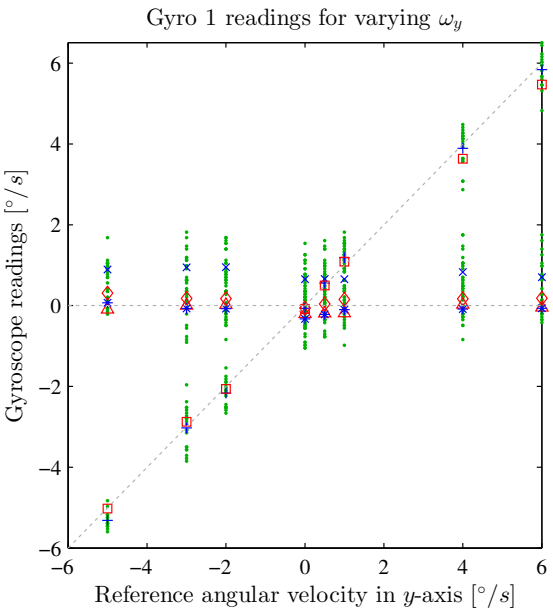
GYROSCOPE	SCALE $\boldsymbol{S}_g$	OFFSET $\mathbf{off}_g\,[^\circ\,\mathrm{s}^{-1}]$
1	$\begin{bmatrix} 1.085 & -0.023 & 0.014 \\ 0.016 & 1.063 & -0.014 \\ 0.011 & 0 & 1.061 \end{bmatrix}$	$\begin{bmatrix} 0.88 \\ 0.03 \\ -0.26 \end{bmatrix}$
2	$\begin{bmatrix} 1.076 & 0.003 & -0.026 \\ -0.01 & 1.065 & -0.009 \\ 0.002 & -0.003 & 1.018 \end{bmatrix}$	$\begin{bmatrix} 0.57 \\ -0.4 \\ 0.14 \end{bmatrix}$

The estimated model parameters in table 9 show that only minor correction is needed for the gyroscopes readings, as the total scale error matrix $\boldsymbol{S}_g$ values are close to identity matrix and the offset values in $\mathbf{off}_g$ are small. However, examining the plot of gyroscopes readings in figs. 20 and 21, the calibration process visibly improved the gyroscopes readings, moving the mean of the data closer to the ideal reference line. Analyzing the mean uncalibrated and calibrated data, it can be calculated that the rotation axis direction error improved from 5.12–63.81° before calibration to 0.92–26.9° after calibration for gyroscope 1 and from 4.78–69.79° before calibration to 0.25–27.83° after calibration for gyroscope 2, while the angular speed error improved from 0.018–1.11 $^\circ\,\mathrm{s}^{-1}$ before calibration to 0.006–0.528 $^\circ\,\mathrm{s}^{-1}$ after calibration for gyroscope 1 and from 0.017–0.778 $^\circ\,\mathrm{s}^{-1}$ before calibration to 0.003–0.354 $^\circ\,\mathrm{s}^{-1}$ after calibration for gyroscope 2. The presence of the large range of error in rotation axis accuracy even with the data mean comes from the RRW error, which defines a drift in the

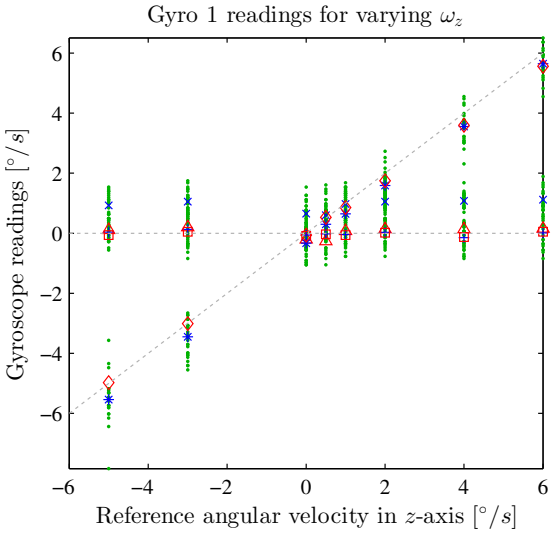
offset error itself. This time-varying error requires proper filtering in order to remove it from the data.



(a) Angular velocity plot for varying reference in x-axis.

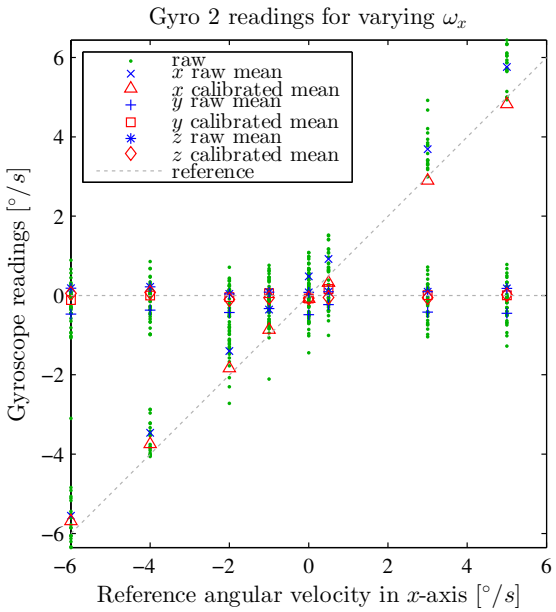


(b) Angular velocity plot for varying reference in y-axis.

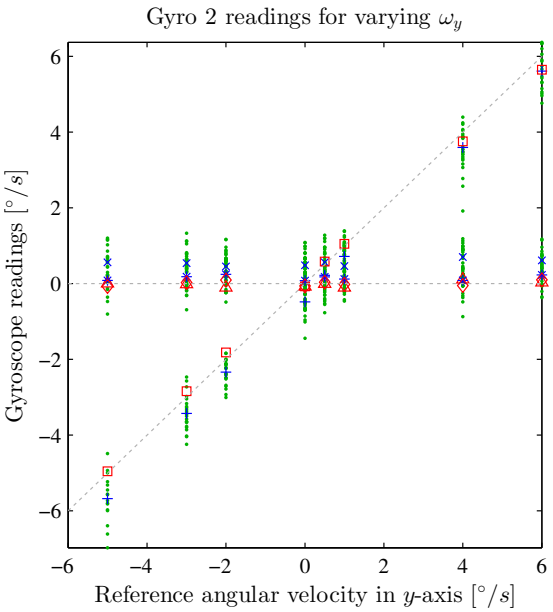


(c) Angular velocity plot for varying reference in z-axis.

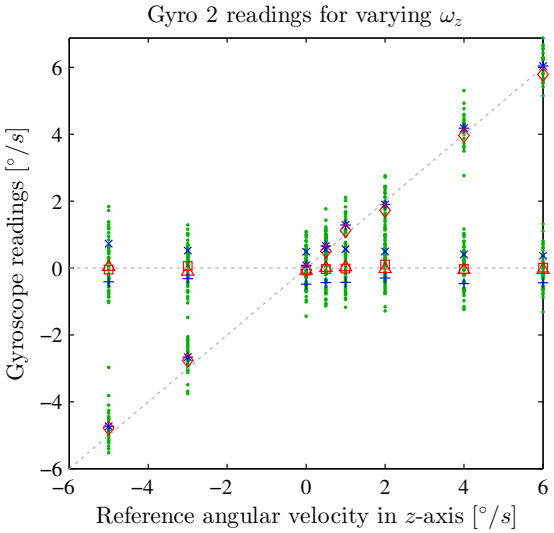
Figure 20: Plot of gyroscope 1 readings in $^\circ s^{-1}$ (raw data denoted with $\bullet$'s; mean of uncalibrated data denoted with $\times$'s, $+$'s, and $*$'s for its x-, y-, and z-axis components, respectively; and mean of calibrated data denoted with $\triangle$'s, $\square$'s, and $\Diamond$'s for its x-, y-, and z-axis components, respectively) against the reference angular velocity in $^\circ s^{-1}$. The angular velocity is varied for one axis at a time, while the other axes are set to zero.



(a) Angular velocity plot for varying reference in x-axis.



(b) Angular velocity plot for varying reference in y-axis.



(c) Angular velocity plot for varying reference in z-axis.

Figure 21: Plot of gyroscope 2 readings in $^\circ\,\mathrm{s}^{-1}$ raw data denoted with •'s; mean of uncalibrated data denoted with ×'s, +'s, and ∗'s for its x-, y-, and z-axis components, respectively; and mean of calibrated data denoted with △'s, □'s, and ♢'s for its x-, y-, and z-axis components, respectively) against the reference angular velocity in $^\circ\,\mathrm{s}^{-1}$. The angular velocity is varied for one axis at a time, while the other axes are set to zero.

7.3 Magnetometer Calibration

Advanced calibration procedure based on PSO algorithm for the magnetometer has been developed in this thesis. The calibration process includes direct calibration parameters estimation (defined in eq. (7)) in magnetic domain and multiple MDM analysis for further RMM manipulation. Both tests were performed with a Helmholtz cage setup at the magnetic test facility operated by Finnish Meteorological Institute located in Nurmijärvi, Finland [27], as depicted in fig. 22. The test setup includes:

a) Aalto-1 EM, as illustrated in fig. 8 with the defined reference frame;

b) a three-axis Helmholtz cage setup;

c) a one-axis manual rotation platform; and

d) a custom LEMI-CLE fluxgate magnetometer (part number N9512)—referred to as *independent magnetometer* from this point. This independent magnetometer is used for calibrating the Helmholtz coil output and taking measurements for MCP analysis.

The test procedure is separated for the magnetometer calibration parameters estimation and for the MCP evaluation, which are given in sections 7.3.1 and 7.3.2, respectively.

7.3.1 Calibration Parameters Estimation

The data obtained from the magnetometer calibration test procedure is a series of magnetic field measurements taken by the satellite magnetometer while rotating the spacecraft. The magnetometer calibration is performed in the magnetic domain, thus the process is attitude independent, where no information of the actual spacecraft heading and its relationship to the magnetic field vector is required to perform the calibration, as discussed in section 3.4.2. However, this preflight calibration procedure also served as verification phase of the calibration algorithm developed in this thesis, thus several measurements with known reference heading are also obtained.

Another aspect to analyze in this preflight test is whether any inconsistency in the calibration parameters appears when the magnetometer is calibrated in different ambient magnetic field magnitude. This is due to the fact that as given in eq. (7), the proposed magnetometer model parameters is not dependent on the ambient magnetic field magnitude itself, thus any inconsistency in the calibration parameters when the spacecraft is subjected to different magnitude of ambient magnetic field might indicate some problems: the presence of noise in the data affecting the accuracy of the calibration algorithm solution and the presence of hysteresis effect from ferromagnetic materials, which is not included in the magnetometer model as defined in eq. (7).

Test procedure. Taking those requirements into considerations, the test procedure is performed as follows:

1. Calibrate Helmholtz cage output, setting the ambient magnetic field inside the cage to: *i*) 20 000 nT or *ii*) 50 000 nT, representing the lowest and highest expected magnetic field magnitude in the mission orbit as modeled in section 5.2.

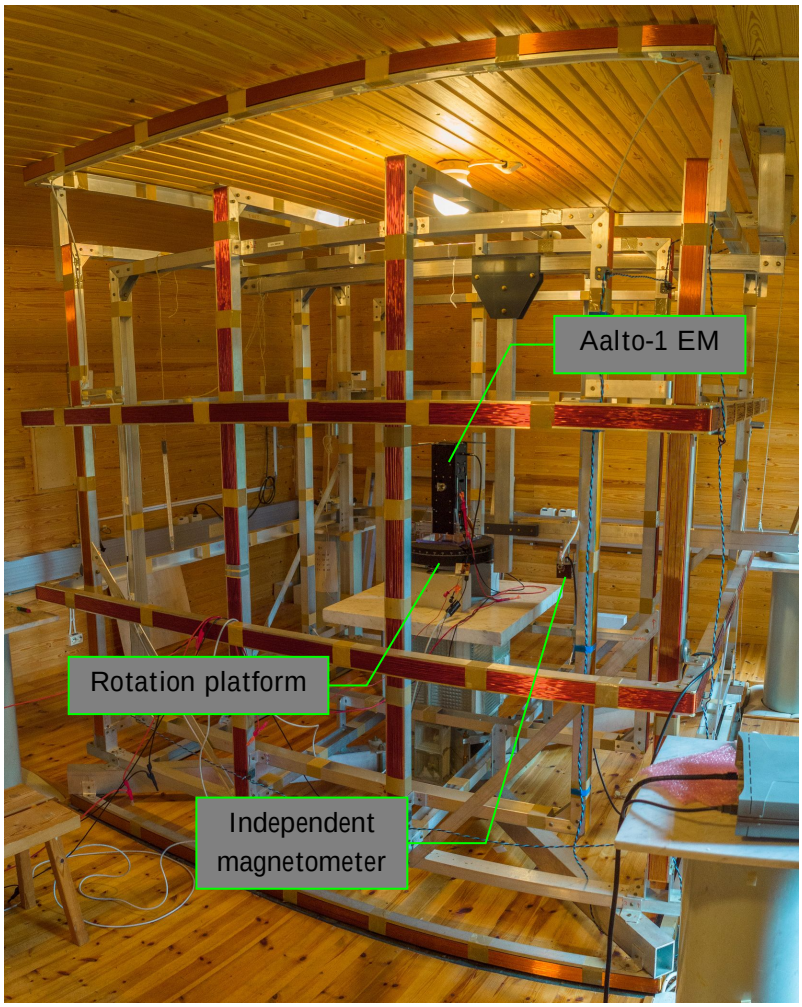


Figure 22: The magnetic test facilities with Helmholtz cage setup in Nurmijärvi Geophysical Observatory, operated by Finnish Meteorological Institute. The specifications of the test setup are available in [27].

2. Position the spacecraft EM in the center of the Helmholtz cage setup and orient it to a certain direction while taking note of the spacecraft orientation. This initial orientation provides the data with known reference heading for calibration verification.

3. Set the satellite operation mode to: *i*) nominal mode or *ii*) full-power mode for the purpose of detecting changes in calibration parameters caused by varying spacecraft electronics current in different operation modes.

4. Start recording the magnetic field vector using the spacecraft magnetometer and start rotating the spacecraft around the rotation axis for one full rotation.

5. Repeat step 3 and 4 with different satellite operation mode than the previous one.

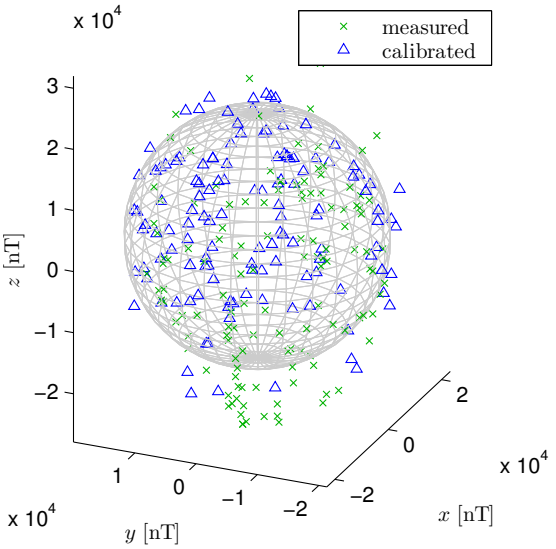
6. Repeat step 2–4 with different rotation axis in spacecraft reference frame by changing the orientation of the spacecraft on the rotation platform.

7. Repeat from step 1 using different magnetic field magnitude.

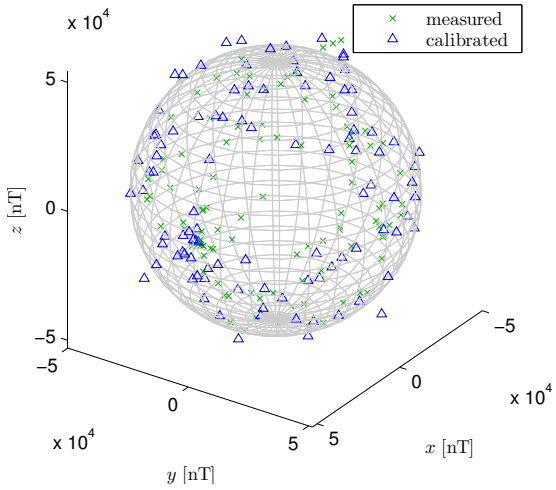
Test results and analysis. The measurements taken from the magnetometer calibration test consist of four sets of data from the combination of two ambient magnetic field magnitude $|\boldsymbol{b}|$ (20 000 nT and 50 000 nT) and two operation mode of the spacecraft (nominal operating mode and full-power operating mode). The results of the estimated magnetometer model parameters $\boldsymbol{S}_{m_c}$ and $\mathbf{off}_{m_c}$ from eq. (17a) are given in table 10. Graphical representations of the measured and calibrated magnetic field vector loci are provided in fig. 23. Using the known angular reference from the initial orientation data (note that this is unfiltered data), it can be calculated that the heading error of the magnetometer improved from 5.24–13.24° before calibration to 1.9–7.3° after calibration.

Table 10: Estimated magnetometer model parameters under different combinations of ambient magnetic field $|\boldsymbol{b}|$ and spacecraft operation mode in the test setups.

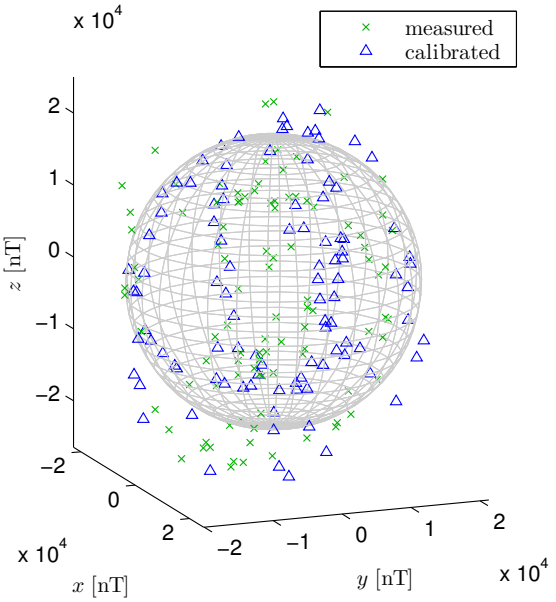
TEST SETUP	MODEL PARAMETERS				F_g
	$\boldsymbol{S}_{m_c}$			$\mathbf{off}_{m_c}$ [nT]	
Nominal mode, $\lvert\boldsymbol{b}\rvert = 20\,000$ [nT], $n_l = 3$, $n_s = 130$.	0.8742	−0.0643	−0.0323	−4 496.3	347.77
	0.0024	−0.8932	−0.0328	−3 155.5	
	0.0223	0.0849	−1.1746	−1 717.3	
Nominal mode, $\lvert\boldsymbol{b}\rvert = 50\,000$ [nT], $n_l = 3$, $n_s = 102$.	0.8635	−0.0387	0.025	−4 647	368.78
	−0.0295	−0.915	−0.0013	−2 848.1	
	−0.0293	−0.0465	−0.9335	−2 346.7	
Full-power mode, $\lvert\boldsymbol{b}\rvert = 20\,000$ [nT], $n_l = 2$, $n_s = 100$.	0.77	−0.0694	−0.0026	−2 515.1	369.87
	−0.0391	−0.9294	−0.0935	−1 482.7	
	0.1408	0.1837	−1.0349	−3 013.2	
Full-power mode, $\lvert\boldsymbol{b}\rvert = 50\,000$ [nT], $n_l = 3$, $n_s = 118$.	0.8402	−0.0102	−0.0285	−3 513.4	322.31
	−0.0374	−0.924	0.0425	−1 656	
	−0.0702	−0.0006	−0.992	−959.1	



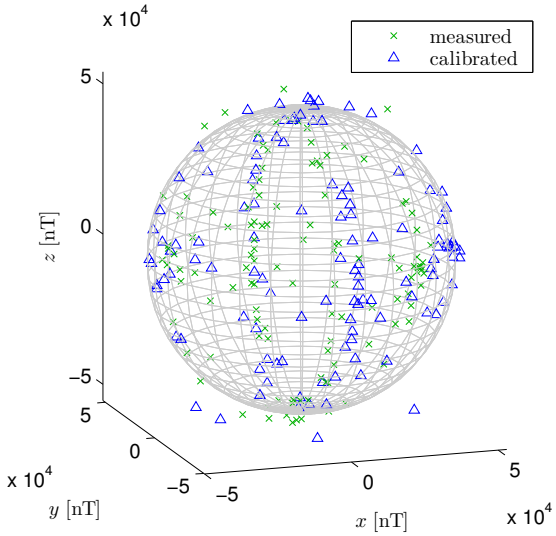
(a) Nominal mode, $|\boldsymbol{b}| = 20\,000\,[\mathrm{nT}]$, $n_l = 3$, $n_s = 130$.



(b) Nominal mode, $|\boldsymbol{b}| = 50\,000\,[\mathrm{nT}]$, $n_l = 3$, $n_s = 102$.



(c) Full-power mode, $|\boldsymbol{b}| = 20\,000\,[\mathrm{nT}]$, $n_l = 2$, $n_s = 100$.



(d) Full-power mode, $|\boldsymbol{b}| = 50\,000\,[\mathrm{nT}]$, $n_l = 3$, $n_s = 118$.

Figure 23: 3-D plot of magnetic field vectors showing calibrated (denoted with △'s) and measured (denoted with ×'s) magnetic field vector loci under different combinations of ambient magnetic field $|\boldsymbol{b}|$ and spacecraft operation mode in the test setups. The axes represent the magnetic field strength in nT from the magnetometer frame.

The estimated model parameters in table 10 show that there is no serious misalignment, nonorthogonality, or cross-axis sensitivity problem, as indicated by the very small non-diagonal components values of the combined scale matrix $\boldsymbol{S}_{m_c}$. Note that the diagonal components for y- and z-axis scale factor are negative because the frame definition of the ADCS module is inverted from the spacecraft body frame definition

in the y- and z-axis. It also seems that the different test setups do not significantly affect the parameters. This is expected for test setups with the same operating mode under different ambient magnetic field magnitude, since the model parameters are independent from the ambient magnetic field as modeled in eq. (7). The estimated combined offset parameter $\mathbf{off}_{m_c}$, however, is quite inconsistent across the different test setups, especially for tests under different operating mode. This is also expected since the spacecraft modules that are switched on in full-power mode will generate magnetic field disturbance. However, the result shows that the estimated $\mathbf{off}_{m_c}$ for full-power operating mode is not consistent under different ambient magnetic field magnitude, which is most likely caused by the poor data quality for the lower ambient magnetic field magnitude.

PSO solution with the current set of objectives is very likely to converge in the global optimum, since the current results for each test setup are consistent across numerous runs with different parameters. Inconsistencies in results under different ambient magnetic field magnitudes at this point are contributed by the noise level and unmodeled time-varying errors. The signal-to-noise ratio is especially higher for tests under $|\boldsymbol{b}| = 20\,000\,[\mathrm{nT}]$ which is practically less than half the tests under $|\boldsymbol{b}| = 50\,000\,[\mathrm{nT}]$. A closer look to the result for test setup with full-power operating mode and $|\boldsymbol{b}| = 20\,000\,[\mathrm{nT}]$ in fig. 23c also suggest an especially noisy data, where the sphere/ellipsoid shape of the measurement locus is badly scattered.

7.3.2 Magnetic Cleanliness Evaluation using Multiple Magnetic Dipole Modeling

The data obtained from the spacecraft MCP evaluation procedure is a series of measurements taken by the independent magnetometer outside the satellite structure on different measurement points, as depicted in fig. 5. However, the measurement points forming a sphere surface such as in fig. 5 is not practical for the current test setup, where the independent magnetometer position and the rotation platform angular displacement have to be adjusted manually by hand. Therefore, the measurement points will form a cylinder instead, which can be obtained by rotating the spacecraft for a full rotation while taking measurements every several degrees of rotation, continued by repositioning the independent magnetometer vertically for some interval, repeating the rotation and measurement taking sequence.

Test procedure. The test procedure is as follows:

1. Calibrate Helmholtz cage output, setting the ambient magnetic field inside the cage to zero. This is because only the magnetic field generated by the spacecraft is of interest for MCP analysis.

2. Set the spacecraft to a certain operating mode. The test was conducted only in nominal operation mode, although different operation mode will be tested for future studies.

3. Position the spacecraft in the center of the Helmholtz cage setup and orient it to a certain direction—this initial orientation is important because it defines the

relationship between the spacecraft and the independent magnetometer frame, where the data from the independent magnetometer need to be transformed into the spacecraft reference frame.

4. Position the independent magnetometer in a suitable location near the spacecraft. In the test, the independent magnetometer was initially positioned at $[x, y, z] = [50, 0, -23.5]$ [cm] in the spacecraft reference frame.

5. Rotate the rotation platform in 10° intervals for one full rotation while taking measurements from the independent magnetometer in every interval.

6. Repeat step 4 and 5 with repositioned independent magnetometer by moving it vertically in a certain interval until the measurement points adequately cover the spacecraft. The vertical interval used in the test was 8 cm.

With this test procedure, the total number of measurement points $n_s = 254$ was obtained.

Test results and analysis. As discussed in section 6.5.2, the focus of MCP analysis in this thesis is to determine the estimate total RMM (which is sum of all the dipoles moment) of the spacecraft. Since the number of dipoles are unknown, the PSO algorithm is run several times while increasing the number estimated of dipoles $\hat{n}_d$ until the lowest fitness value can be reached with minimum number of dipoles. With this process, it is found that the optimal number is 10 dipoles. It is important to note that inverse MDM estimation results from simulated data with 10 dipoles as in section 6.5.2 as well as from the real experimental data show that the problem becomes highly nonlinear at 10 dipoles (which means the PSO need to estimate 60 independent variables) and numerous local minima exist in the solution space. These local minima in the results appear as different combination of estimated dipoles properties from different PSO runs with similar best fitness value, which, in PSO runs using the experimental data, most of these minima have a fitness value $F_g \approx 0.5$. The result with lowest fitness achieved is $F_g = 0.4826$ with more details of the dipoles properties given in table 11 and its 3-D plot depicted in fig. 24.

Since the minimum information required from this analysis is to obtain a consistent result for the total RMM, inidividual dipole properties (positions and moments) is not crucial. However, the results show that many estimated dipoles positions are located outside the spacecraft body, which occupies a cuboid with the dimension $[x, y, z] \approx [10, 10, 30]$ [cm] centered in the frame origin. These pattern is consistent with the best solution achieved with different n_d, which signifies the possibility of disturbance outside the spacecraft body, possibly coming from measurement noise and bias in the magnetometer reading. The effect from noise presence could be reduced by moving the sensor position closer to the spacecraft body (reducing the radius of the measurement cylinder), thus increasing the signal-to-noise ratio.

The magnitude of the total RMM itself is estimated around $\sum_d |\boldsymbol{m}_d| \approx 58.5\,[\mathrm{mA\,m^2}]$, which is lower than the maximum magnetic moment that can be produced by the magnetorquer ($|\boldsymbol{m}_{\mathrm{mtq,max}}| = 200\,[\mathrm{mA\,m^2}]$). Thus, further strategies for magnetic disturbance torque mitigation is not required at this point.

Table 11: Estimated total remanent magnetic moment of Aalto-1 engineering model with the lowest fitness value. The PSO is set to estimate 10 dipoles.

DIPOLE	POSITION	RMM [mA m^2]		
	$\boldsymbol{p}_d$ [cm]	$\boldsymbol{m}_d$	$\lvert\boldsymbol{m}_d\rvert$	F_g
1	$\begin{bmatrix}-7.1 & -10.7 & 10.8\end{bmatrix}^\top$	$\begin{bmatrix}-133.2 & -58.9 & -58.7\end{bmatrix}^\top$	157	
2	$\begin{bmatrix}-10.2 & 1.9 & 12.2\end{bmatrix}^\top$	$\begin{bmatrix}47.9 & -52.5 & -19.2\end{bmatrix}^\top$	73.6	
3	$\begin{bmatrix}1.4 & -6.1 & 14.3\end{bmatrix}^\top$	$\begin{bmatrix}155.2 & -90.6 & -69.2\end{bmatrix}^\top$	192.5	
4	$\begin{bmatrix}-11.1 & 12.5 & 5.6\end{bmatrix}^\top$	$\begin{bmatrix}-84.2 & -59.2 & 74.4\end{bmatrix}^\top$	127	
5	$\begin{bmatrix}-12.9 & 8.2 & 17.1\end{bmatrix}^\top$	$\begin{bmatrix}-17.6 & 209.4 & -66.4\end{bmatrix}^\top$	220.4	
6	$\begin{bmatrix}-9.7 & -3.5 & -7\end{bmatrix}^\top$	$\begin{bmatrix}19.5 & 103.8 & 38.4\end{bmatrix}^\top$	112.4	
7	$\begin{bmatrix}-10 & 14.6 & 19.9\end{bmatrix}^\top$	$\begin{bmatrix}-160.6 & -147.2 & -71.6\end{bmatrix}^\top$	229.3	
8	$\begin{bmatrix}-2.4 & 8.2 & 15.9\end{bmatrix}^\top$	$\begin{bmatrix}-61.5 & 175.2 & 8.5\end{bmatrix}^\top$	185.9	
9	$\begin{bmatrix}-14.6 & 14 & 17.2\end{bmatrix}^\top$	$\begin{bmatrix}221.9 & -62.2 & 35.6\end{bmatrix}^\top$	233.2	
10	$\begin{bmatrix}-4.2 & -5.6 & 32.7\end{bmatrix}^\top$	$\begin{bmatrix}27 & 4 & 180.5\end{bmatrix}^\top$	182.6	
Total		$\begin{bmatrix}14.5 & 21.9 & 52.3\end{bmatrix}^\top$	58.5	0.4826

7.4 Magnetorquer Calibration

For this thesis, magnetorquer calibration is performed using magnetic property analysis approach. Note that the magnetorquers are COTS integrated and that they receive command directly in magnetic moment m_{mtq} for each axis where the required driving current i_{mtq} is automatically calculated with eq. (10b) with the constants already predefined. Instead of estimating those constants, the test procedure goal is to obtain set of magnetic field measurement vectors $\check{\boldsymbol{b}}_{\text{mtq}}$ produced by the magnetorquer as the magnetorquer output is set to different magnetic moment $\boldsymbol{m}_{\text{mtq}}$. The relationship between the magnetorquer moment and its magnetic field measured at a certain point of measurement $\boldsymbol{p}_s$ in space is defined in eq. (8), where the magnetic field strength at a constant point in space should be directly proportional to the magnetic moment—the data obtained from the test (measured magnetic field magnitude and the corresponding magnetic moment commanded to the magnetorquers) is for verifying this relationship. Another aspect to consider is that since the magnetorquers under test are integrated with the ADCS module inside the spacecraft EM as COTS package, misalignment errors are assumed to be negligible.

The test setup includes the three-axis magnetorquers integrated inside the Aalto-1

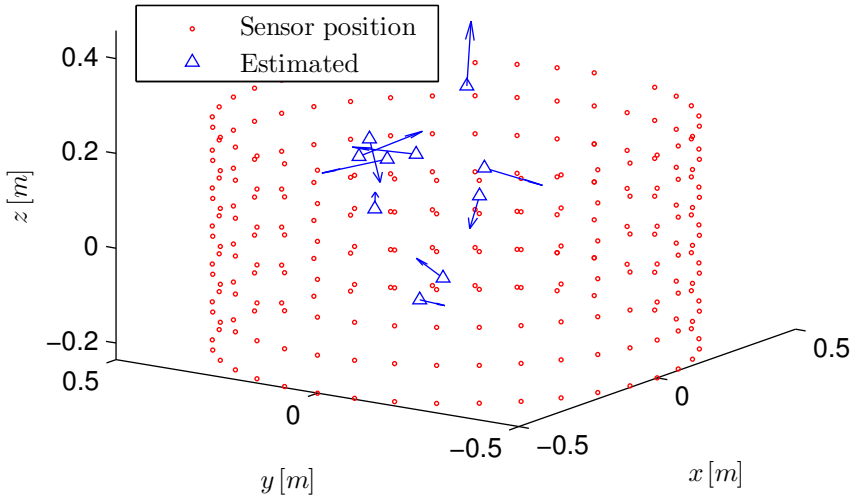


Figure 24: 3-D plots of the estimated dipoles positions (denoted with △'s) and moments (denoted with arrows originating from the dipoles positions) for magnetic cleanliness evaluation with 10 dipoles estimate. The measurement points for collecting the data are denoted with ○'s. The axes represent positions from the spacecraft frame in m.

EM and an independent magnetometer (HMC5883 three-axis digital magnetometer) to measure the magnetic field strength produced by the magnetorquer, as depicted in fig. 25.

Test procedure. The test procedure for gathering sufficient data in verifying the relationship between the magnetic moment commanded to the magnetorquer and its generated magnetic field is as follows:

1. Calibrate the independent magnetometer using the same principle as explained in section 6.5.1 by rotating the magnetometer in several axes by hand near the Aalto-1 EM to compensate the error factors.

2. Position the independent magnetometer near the magnetorquers as such that it is not too far from the magnetorquers that it cannot detect the magnetic field generated by the magnetorquer or too close that the generated magnetic field saturates the magnetometer reading.

3. Set the magnetorquer moment output to zero for all axis and activate the self test function of the magnetorquer to remove magnetization in the core material.

4. Collect and calibrate reading from the independent magnetometer. This reading is the total ambient magnetic field.

5. Set the magnetorquer moment output for x-axis to values between -0.2 to $0.2\,\mathrm{A\,m^2}$ with $10\,\mathrm{mA\,m^2}$ steps, while the y- and z-axis magnetorquer set to zero. Collect and calibrate reading from the independent magnetometer after running the magnetorquer self test function for each magnetorquer moment

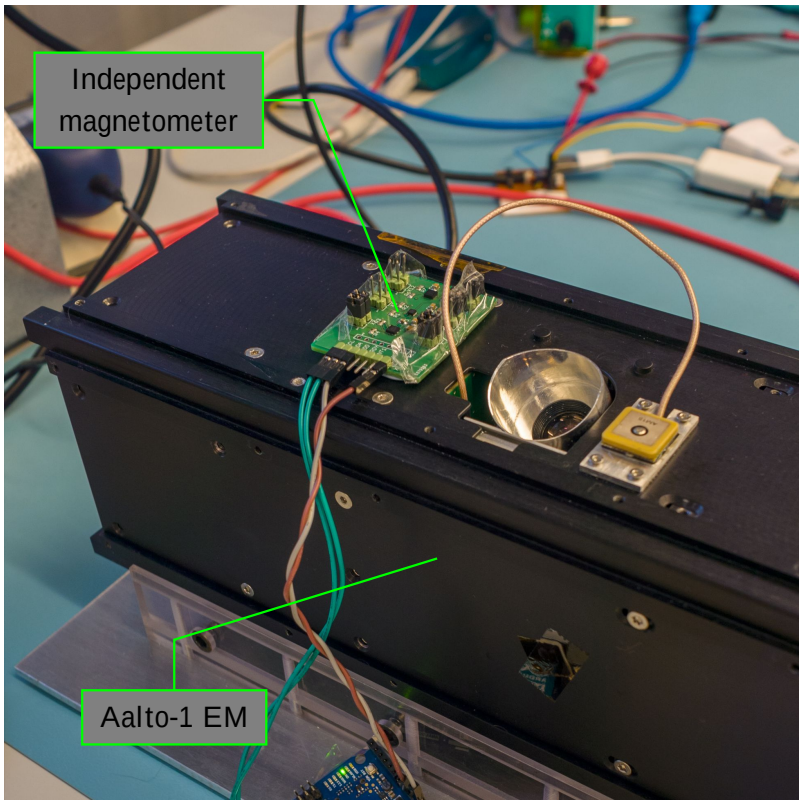


Figure 25: Test setup for magnetorquer calibration by directly measuring the magnetic field generated from the magnetorquers.

step and substract it with the ambient magnetic field obtained from step 4 to isolate the magnetic field generated by the magnetorquer from the ambient magnetic field.

6. Repeat step 5 with the magnetorquer moment output controlled for y- and z-axis once at a time while setting the magnetorquer moment output on other axis to zero.

Test results and analysis. First, the independent magnetometer need to be calibrated using measurement data obtained near the actual measurement point. The estimated magnetometer calibration parameters as defined in eq. (17b) are $\boldsymbol{K}_m = \begin{bmatrix} 0.97 & 0 & -0.02 \\ -0.01 & 1.13 & 0 \\ -0.19 & -0.01 & 1.05 \end{bmatrix}$ and $\boldsymbol{k}_m = \begin{bmatrix} 2862 \\ -12501 \\ -12391 \end{bmatrix}$ [nT], with the measurement locus plot for the uncalibrated and calibrated data shown in fig. 26. With these calibration parameters, the magnetic field reading is calibrated and the magnetic field generated by the magnetorquer is calculated as explained in the procedure. The plot of the measured magnetic field magnitude $\left|\check{\boldsymbol{b}}_{\text{mtq}}\right|$ against the commanded magnetorquer moment $\boldsymbol{m}_{\text{mtq}}$ in each axis is shown in fig. 27. For comparison, the theoretical relationship between the magnetic moment $\boldsymbol{m}_{\text{mtq}}$ and the generated magnetic field strength $\boldsymbol{b}_{\text{mtq}}$ at a point $\boldsymbol{p}_s$ in space is plotted in fig. 28, where the

theoretical magnetic field magnitude $|\boldsymbol{b}_{\text{mtq}}|$ generated by a simulated dipole moment with varying magnitude $\boldsymbol{m}_{\text{mtq}} = \begin{bmatrix}(-0.2 \ldots 0.2) & 0 & 0\end{bmatrix}^\top$ [A m^2] at the center of the frame is measured at a measurement point located at $\boldsymbol{p}_s = \begin{bmatrix}0.1 & 0.1 & 0\end{bmatrix}^\top$ [m] in the frame.

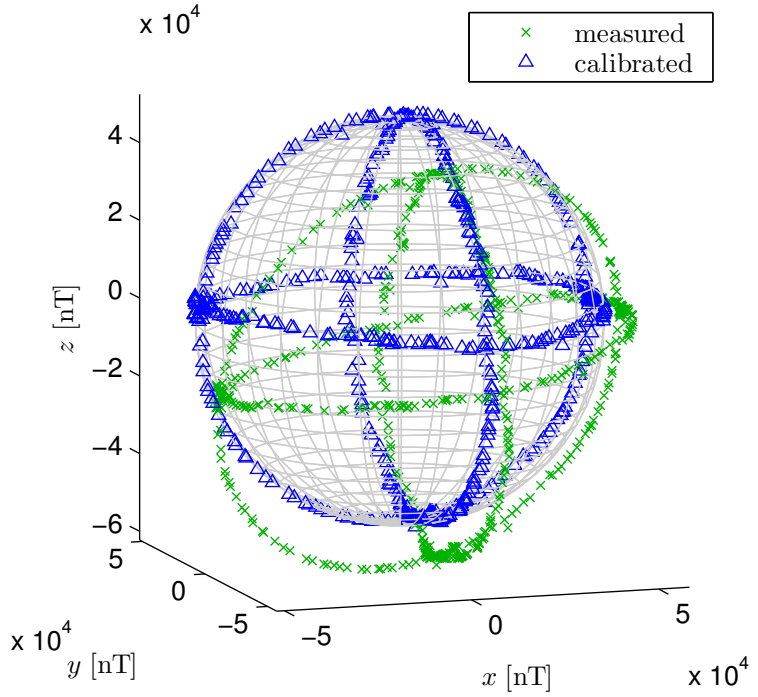
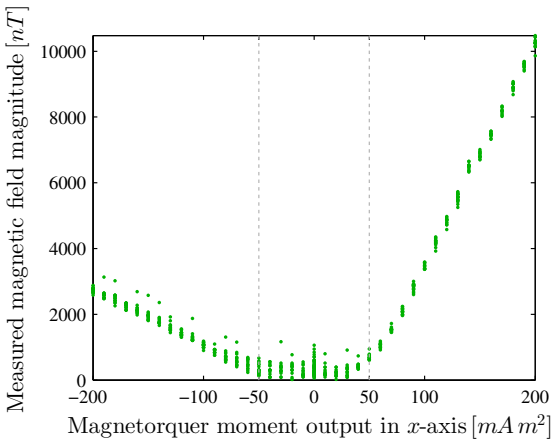
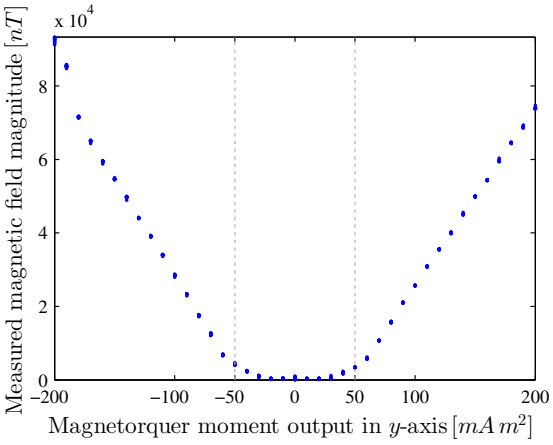


Figure 26: 3-D plot of magnetic field vectors for the measured (denoted with ×'s) and calibrated (denoted with △'s) data taken by HMC5883 independent magnetometer used for measuring the magnetic field strength generated by magnetorquer. The plot axes represent the magnetic field strength in nT from the magnetometer frame. The data was taken by rotating the magnetometer by hand around its three axes ($n_l = 3$), and the estimated calibration parameters have a best fitness value of $F_g = 14.937$.

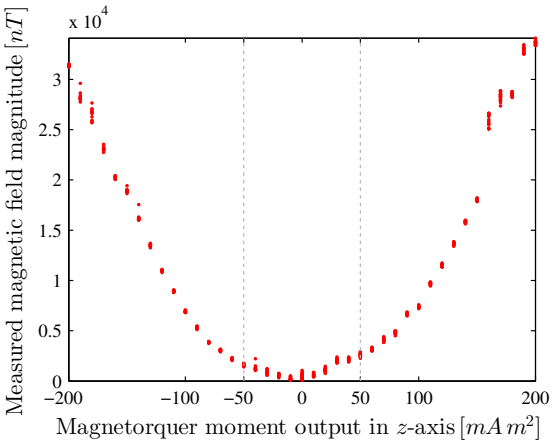
Figure 27 shows that the relationship between the commanded magnetorquer moment and its generated magnetic field is in fact nonlinear, unlike the theoretical relationship depicted in fig. 28 where the relationship is linear. However, the relationship is approximately linear for $|\boldsymbol{m}_{\text{mtq}}| \geq 50\,[\text{mA m}^2]$. The nonlinear region for smaller commanded magnetorquer moment ($|\boldsymbol{m}_{\text{mtq}}| < 50\,[\text{mA m}^2]$) is likely due to the nonlinearity of the magnetorquer core where eq. (10) is not valid. Thus, for linear region of the magnetorquer moment, the actual magnetorquer moment $\check{m}_{\text{mtq},i}$ produced by the i-axis magnetorquer as measured by the generated magnetic field can be approximated with $|\check{m}_{\text{mtq},i}| \approx |m_{\text{mtq},i}| - 50$ for $|\boldsymbol{m}_{\text{mtq}}| \geq 50\,[\text{mA m}^2]$, assuming there is no scaling error between the commanded and the actual magnetorquer moment. Acquiring more detailed error parameters such as scaling and misalignment factors requires exact knowledge of the independent magnetometer relative position and orientation with respect to each magnetorquer frame, which was not taken during the test.



(a) Varying magnetorquer moment in x-axis.



(b) Varying magnetorquer moment in y-axis.



(c) Varying magnetorquer moment in z-axis.

Figure 27: Plot of measured magnetic field magnitude $\left|\check{\boldsymbol{b}}_{\text{mtq}}\right|$ generated by the magnetorquer against the commanded magnetorquer moment $m_{\text{mtq},i}$ in i axis. The magnetorquer moment in other axes are set to zero.

Another abnormality is observed for varying magnetorquer moment in x-axis shown in fig. 27a, where the gradient of the plot is not symmetrical between positive and negative magnetorquer moment value, in contrast to the symmetrical theoretical plot lines given in fig. 28. There are several possible causes for this abnormality: hysteresis of the magnetorquer core and the surrounding materials, or scaling error in the magnetorquer control originating from software or hardware bugs. Since this behavior does not appear in the other magnetorquer axes, it is unlikely that hysteresis of the magnetorquer core or other surrounding materials can cause this. Unless more meticulous testing is performed, the results seem to be inconclusive. Thus, further testing is recommended for verifying any error in the current driving the magnetorquer.

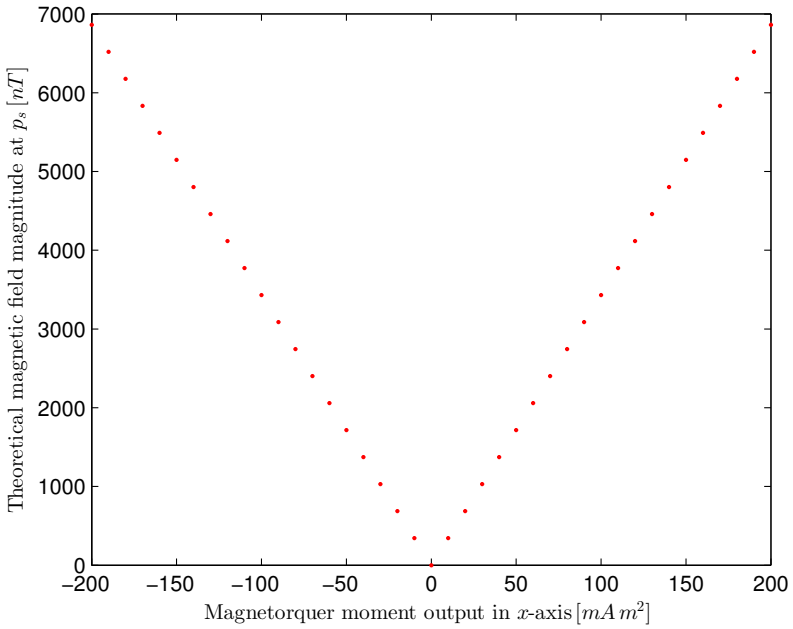


Figure 28: Plot of theoretical magnetic field magnitude $|\boldsymbol{b}_{\text{mtq}}|$, measured at a measurement point as calculated from a simulated magnetorquer dipole moment at frame origin using eq. (8), against the magnetorquer moment magnitude in the x-axis. The magnetic moment in other axes are kept at zero.

8 Summary

8.1 Conclusions

Calibration and testing of ADCS instruments on Aalto-1 EM (i. e., two-axis DSS, three-axis gyroscopes, three-axis magnetometer including MCP analysis, and three-axis magnetorquer) was performed and the results were analyzed. Several findings were discovered during the whole calibration and testing process, from the design and testing phase of calibration algorithms up to the analysis of experimental data:

- A stochastic estimation method based on biological swarm behavior called PSO was specifically developed for estimating the magnetometer calibration parameters and solving the inverse MDM problem in spacecraft MCP analysis.
- A novel approach was developed to improve the standard PSO implementation for magnetometer calibration parameters estimation problem in magnetic domain without the need of compromising the number of estimated parameters: the novel PSO is designed to estimate a full 3×3 calibration matrix $\boldsymbol{K}_m$ and 3×1 calibration vector $\boldsymbol{k}_m$ (a total of 12 parameters) by adding a new objective definition. The current implementation of the new feature, however, still has some limitations:
  - individual measurement error factors (i. e., scale factor, nonorthogonality, misalignment, soft iron error, hard iron error, and magnetometer offset error) are still mathematically indistinguishable inside $\boldsymbol{K}_m$ and $\boldsymbol{k}_m$ unless more rigorous testing is performed,
  - the measurement data has to be dividable into at least two segments ($n_l \geq 2$) where the measurement locus of each segment l can be averaged into a single plane with unique normal direction $\mathring{\boldsymbol{n}}_l$ (implemented in this thesis with orthogonal distance regression plane fitting) and the average angular velocity axis of the magnetometer $\mathring{\boldsymbol{\omega}}_l$ when the measurement was taken for that segment is known (in this thesis work it was known from the test procedure), and
  - the calibration parameters do not include time-varying errors.
- The improved PSO for magnetometer calibration parameters estimation is capable of estimating the calibration parameters under varying ambient magnetic field magnitude as shown using simulated data.
- Magnetometer calibration results using experimental data showed a consistent estimated calibration parameters under different ambient magnetic field. However, estimation results under larger ambient magnetic field magnitude ($|\boldsymbol{b}| = 50\,\mu\text{T}$) is more accurate since the signal-to-noise ratio of the data is larger. In the other hand, minor differences was detected in the offset calibration vector under different spacecraft operating mode, which came from changing hard iron bias of the spacecraft electronics under different load. The calibration

process improved the magnetometer reading accuracy from 5.24–13.24° before calibration to 1.9–7.3° after calibration according to the evaluation of the raw, unfiltered experimental data.

- The PSO developed for MCP analysis using inverse MDM approach is based on the PSO developed in [14] with some small differences:
  - the parameters dynamics is extended by repeating the parameters variation pattern in one PSO loop to promote re-exploration of the solution space after the swarm starts converging to a minimum point and
  - the refinement procedure is generalized so that it can be implemented for all components of the swarm instead of the components associated only with a particular dipole—this also means that the refinement procedure can be implemented for magnetometer calibration parameters estimation.

- The focus of the MCP analysis was to determine the total RMM of the spacecraft, while the individual dipole properties were not crucial. Extensive PSO runs were executed on the experimental data to determine the optimal number of estimated dipoles $\hat{n}_d$, until a global minimum solution is obtained (it is found that the optimal $\hat{n}_d = 10$). The results from experimental data show that the total RMM magnitude of the spacecraft is $|\hat{\boldsymbol{m}}_d| = 58.5\,[\mathrm{mA\,m^2}]$, which is lower than the maximum magnetic moment that can be generated by each magnetorquer in i-axis ($|m_{\mathrm{mtq},i}| \leq 0.2\,[\mathrm{A\,m^2}]$).

- Calibration results of the single DSS module showed no serious problem in the Sun angle readings. Calibration parameters estimation using linear batch least squares method did not significantly improve the Sun angle accuracy, mostly because the low native resolution of the DSS itself and the presence of noise.

- The brightness readings of the Sun sensor was observed to drop as the Sun angle deviates from the sensor normal direction (z-axis). This information is necessary to calibrate the brightness readings used for distinguishing the readings from the Sun and the readings from the Earth albedo.

- The estimated gyroscope model parameters showed that only minor correction was needed for the gyroscopes reading, as the total scale error matrix $\boldsymbol{S}_g$ values are close to identity matrix and the offset values in $\mathbf{off}_g$ are small.

- The magnetorquer calibration was performed with magnetic property analysis by evaluating the relationship between the magnetic field magnitude $\left|\check{\boldsymbol{b}}_{\mathrm{mtq}}\right|$ measured at a single measurement point and the magnetic moment commanded to the i-axis magnetorquer $m_{\mathrm{mtq},i}$. Theoretically, the relationship should be linear as described by eq. (8), although experimental data showed that the relationship is approximately linear only for $m_{\mathrm{mtq}} \geq 50\,[\mathrm{mAm^2}]$, which is most likely caused by the nonlinearity of the magnetorquer core permeability. Further evaluation of the control loop is required to determine whether more detailed calibration is necessary for improving the control accuracy.

- Anomaly in the magnetorquer moment behavior was observed in the test results of x-axis magnetorquer (fig. 27a), where the gradient of the plot is not symmetrical between the positive and negative commanded magnetorquer moment. Further testing is required to determine the actual cause of this anomaly.

8.2 Future Work

Some suggestions for future work are derived from the findings discovered during the calibration and testing process:

- The novel PSO developed for magnetometer calibration parameters estimation can be improved further using a more general curve fitting algorithm to fit the measurement locus into the angular velocity information so that the requirements on the measurement locus can be relaxed. This is also beneficial for implementation of the algorithm using real flight data for on-orbit calibration of the magnetometer reading, because the state of rotation during flight is not as controllable as preflight calibration process.

- Time-varying errors in magnetometer, which are related to the magnetic fields generated by spacecraft electronics, can be added into the measurement model to improve accuracy. However, real-time information of the spacecraft load current for individual on-board electronic module is required.

- The accuracy of magnetometer calibration parameters estimations results can be improved by adding more measurement data and filtering the data from outliers before the calibration process.

- Implementing PSO for MCP analysis with inverse MDM approach using Aalto-1 EM proves to be limited only to total RMM estimation because of the highly nonlinear problem in estimating 10 dipoles. To accurately estimate individual dipole properties, initial assumption on the dipole properties is needed to distinguish local minima in the solutions. This initial assumptions can be obtained by MCP testing of individual building blocks of the spacecraft prior to the MCP analysis for the whole spacecraft.

- More detailed calibration parameters for DSS brightness reading can be defined by implementing curve fitting algorithm to the brightness vs Sun reference angle plot.

- The test procedure for magnetorquer is only useful for verifying the validity of magnetic moment generated by the magnetorquer. Properly calibrating any scaling and misalignment errors requires further testing with precise knowledge of the independent magnetometer position and orientation relative to the individual magnetorquer frame.

References

[1] *Aalto-1 Sun Sensor Interface Control Document*. Version 3.0. Aalto University, May 5, 2015.

[2] *Aalto-2/FI1 CubeSat Design Overview Report*. QB50 Critical Design Review. Version 2. Aalto University, Mar. 17, 2014.

[3] A. Ali et al. "Swarm Optimization-Based Magnetometer Calibration for Personal Handheld Devices". In: *Sensors* 12.9 (Sept. 13, 2012), pp. 12455–12472. DOI: 10.3390/s120912455.

[4] P. Appel. "Attitude estimation from magnetometer and earth-albedo-corrected coarse sun sensor measurements". In: *Acta Astronautica* 56.1–2 (Oct. 28, 2005). 4th {IAA} International Symposium on Small Satellites for Earth Observation, pp. 115–126. DOI: 10.1016/j.actaastro.2004.09.001.

[5] G. A. Aydemir and A. Saranlı. "Characterization and calibration of MEMS inertial sensors for state and parameter estimation applications". In: *Measurement* 45.5 (June 2012), pp. 1210–1225. DOI: 10.1016/j.measurement.2012.01.015.

[6] A. Aydinlioglu. "Design, Development and Production of Electromagnetic Coils for Attitude Control of a Pico Satellite". Diploma thesis. University of Applied Sciences Aachen, Feb. 2006. URL: http://www.raumfahrt.fh-aachen.de/compass-1/download/Diploma_Thesis_Ali_Aydinlioglu.pdf (visited on 03/06/2015).

[7] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. New York, USA: John Wiley & Sons, Inc., 2001. ISBN: 0-471-41655-X.

[8] D. S. Bayard and S. R. Ploen. "High accuracy inertial sensors from inexpensive components". US Patent US6882964 B2. Apr. 19, 2005.

[9] D. Bhanderi and T. Bak. "Modeling Earth Albedo for Satellites in Earth Orbit". In: *Proceedings of AIAA Conference on Guidance, Navigation and Control*. (Aug. 15–18, 2005). San Francisco, CA: AIAA, Aug. 2005. DOI: 10.2514/6.2005-6465.

[10] C. de Boom et al. "Micro Digital Sun Sensor: System in a Package". In: *Proceedings of the 2004 International Conference on MEMS, NANO and Smart Systems (ICMENS '04)*. (Aug. 12–27, 2004). IEEE, Aug. 2004, pp. 322–328. DOI: 10.1109/ICMENS.2004.1508968.

[11] J. Bouwmeester and J. Guo. "Survey of worldwide pico- and nanosatellite missions, distributions and subsystem technology". In: *Acta Astronautica* 67.7–8 (Oct.–Nov. 2010), pp. 854–862. DOI: 10.1016/j.actaastro.2010.06.004.

[12] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge: Cambridge University Press, Mar. 2004. ISBN: 978-0-521-83378-3.

[13] D. Bratton and J. Kennedy. "Defining a Standard for Particle Swarm Optimization". In: *Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS '07)*. (Apr. 1–5, 2007). Honolulu, HI, Apr. 2007, pp. 120–127. DOI: 10.1109/SIS.2007.368035.

[14] E. Carrubba et al. "Particle swarm optimization for multiple dipole modeling of space equipment". In: *IEEE Transactions on Magnetic* 50.12 (Dec. 9, 2014). DOI: 10.1109/TMAG.2014.2334277.

[15] H. Chang et al. "An Integrated MEMS Gyroscope Array with Higher Accuracy Output". In: *Sensors* 8.4 (Apr. 28, 2008), pp. 2886–2899. DOI: 10.3390/s8042886.

[16] Y.-K. Chang, M.-Y. Yun, and B.-H. Lee. "A new modeling and validation of two-axis miniature fine sun sensor". In: *Sensors and Actuators A: Physical* 134.2 (Mar. 15, 2007), pp. 357–365. DOI: 10.1016/j.sna.2006.05.037.

[17] M. Clerc and J. Kennedy. "The particle swarm - explosion, stability, and convergence in a multidimensional complex space". In: *IEEE Transactions on Evolutionary Computation* 6.1 (Feb. 2002), pp. 58–73. DOI: 10.1109/4235.985692.

[18] C. A. C. Coello and M. S. Lechuga. "MOPSO: a proposal for multiple objective particle swarm optimization". In: *Proceedings of the 2002 Congress on Evolutionary Computation (CEC '02)*. (May 12–17, 2002). Vol. 2. Honolulu, HI: IEEE, May 2002, pp. 1051–1056. DOI: 10.1109/CEC.2002.1004388.

[19] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga. "Handling multiple objectives with particle swarm optimization". In: *IEEE Transactions on Evolutionary Computation* 8.3 (June 2004), pp. 256–279. DOI: 10.1109/TEVC.2004.826067.

[20] J. L. Crassidis, K.-L. Lai, and R. R. Harman. "Real-time attitude independent three-axis magnetometer calibration". In: *Journal of Guidance, Control, and Dynamics* 28.1 (Jan.–Feb. 2005), pp. 115–120. DOI: 10.2514/1.6278.

[21] R. C. Eberhart and Y. Shi. "Comparing inertia weights and constriction factors in particle swarm optimization". In: *Proceedings of the 2000 Congress on Evolutionary Computation*. (July 16–19, 2000). Vol. 1. La Jolla, CA: IEEE, July 2000, pp. 84–88. DOI: 10.1109/CEC.2000.870279.

[22] R. Eberhart and Y. Shi. "Particle swarm optimization: developments, applications and resources". In: *Proceedings of the 2001 Congress on Evolutionary Computation*. (May 27–30, 2001). Vol. 1. Seoul: IEEE, May 2001, pp. 81–86. DOI: 10.1109/CEC.2001.934374.

[23] R. Eberhart and J. Kennedy. "A new optimizer using particle swarm theory". In: *Proceedings of the Sixth International Symposium on Micro Machine and Human Science (MHS) 1995*. (Oct. 4–6, 1995). Nagoya, Japan: IEEE, Oct. 1995, pp. 39–43. DOI: 10.1109/MHS.1995.494215.

[24] W. L. Eichhorn. *Magnetic dipole moment determination by near-field analysis*. Technical Report NASA TN D-6685. NASA Goddard Space Flight Center, July 1972. URL: http://hdl.handle.net/2060/19720020082 (visited on 03/06/2015).

[25] ELMOS Semiconductor. *Integrated Solar Angle Sensor*. E910.86 datasheet. Dec. 10, 2010. URL: http://www.mouser.com/ds/2/594/910_86-224506.pdf (visited on 08/11/2015).

[26] C. C. Finlay et al. "International Geomagnetic Reference Field: the eleventh generation". In: *Geophysical Journal International* 183.3 (Sept. 2010), pp. 1216–1230. DOI: 10.1111/j.1365-246X.2010.04804.x.

[27] Finnish Meteorological Institute. *Nurmijärvi Magnetic Calibration and Test Laboratory*. NuMCTL brochure. URL: http://en.ilmatieteenlaitos.fi/c/document_library/get_file?uuid=6b00a65f-31d5-4fe4-a8da-410e1059a65c&groupId=30106 (visited on 08/11/2015).

[28] P. Fortescue, J. Stark, and G. Swinerd. *Spacecraft Systems Engineering*. 4th ed. Chichester: Wiley, 2011. ISBN: 978-0-470-75012-4.

[29] C. C. Foster and G. H. Elkaim. "Extension of a two-step calibration methodology to include nonorthogonal sensor axes". In: *IEEE Transactions on Aerospace and Electronic Systems* 44.3 (July 2008), pp. 1070–1078. DOI: 10.1109/TAES.2008.4655364.

[30] D. Gebre-Egziabher et al. "Calibration of Strapdown Magnetometers in Magnetic Field Domain". In: *Journal of Aerospace Engineering* 19.2 (Apr. 2006), pp. 87–102. DOI: 10.1061/(ASCE)0893-1321(2006)19:2(87).

[31] N. W. Green and A. R. Hoffman. "Anomaly trends for missions to Mars: Mars Global Surveyor and Mars Odyssey". In: *46th AIAA Aerospace Sciences Meeting and Exhibit*. (Jan. 7–10, 2008). Reno, Nevada, Jan. 2008. URL: http://hdl.handle.net/2014/41381 (visited on 02/15/2015).

[32] A. R. Hoffman, N. W. Green, and H. B. Garrett. "Assessment of in-flight anomalies of long life outer planet missions". In: *Proceedings of the 5th International Symposium on Environmental Testing for Space Programmes*. ESA SP-558 vols. Noordwijk, The Netherlands, June 15, 2004, pp. 43–50. (Visited on 02/17/2015).

[33] Honeywell. *3-axis digital compass IC*. HMC5983 datasheet. URL: https://aerospace.honeywell.com/~/media/Images/Plymouth%20Website%20PDFs/Magnetic%20Sensors/Data%20Sheets/HMC5983_3_Axis_Compass_IC.ashx (visited on 08/11/2015).

[34] T. Inamori. "In-Orbit Magnetic Disturbance Estimation and Compensation Using UKF in Nano-Satellite Mission". In: *Guidance, Navigation, and Control and Co-located Conferences*. AIAA Guidance, Navigation, and Control Conference. (Aug. 10–13, 2009). Chicago, Illinois: American Institute of Aeronautics and Astronautics, Aug. 2009. DOI: 10.2514/6.2009-5905.

[35] InvenSense. *Triple-Axis Industrial Gyroscope*. MPU-3300 product specification. Version 1.0. May 31, 2012. URL: http://store.invensense.com/datasheets/invensense/PS-MPU-3300.pdf (visited on 08/11/2015).

[36] N. Jovanovic. "Aalto-2 satellite attitude control system". Master thesis. Aalto University, Aug. 25, 2014. URL: http://urn.fi/URN:NBN:fi:aalto-201409012592 (visited on 02/15/2015).

[37] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte. "A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators". In: *IEEE Transactions on Automatic Control* 45.3 (Mar. 2000), pp. 477–482. DOI: 10.1109/9.847726.

[38] N. C. Kapsalis, S.-D. J. Kakarakis, and C. N. Capsalis. "Prediction of multiple magnetic dipole model parameters from near field measurements employing stochastic algorithms". In: *Progress In Electromagnetics Research Letters* 34 (2012), pp. 111–122. DOI: 10.2528/PIERL12030905.

[39] K. Karvinen, T. Tikka, and J. Praks. "Using Hobby Prototyping Boards and Commercial-off-the-shelf (COTS) Components for Developing Low-cost, Fast-delivery Satellite Subsystems". In: *Journal of Small Satellites* 4.1 (2015), pp. 301–314. ISSN: 2327-4123. URL: http://www.jossonline.com/wp-content/uploads/2015/06/Using-hobby-prototyping-boards-and-commercial-off-the-shelf-components-for-developing-low-cost-and-fast-delivery-satellite-subsystems.pdf.

[40] J. Kennedy and R. Eberhart. "Particle swarm optimization". In: *Proceedings of the 1995 IEEE International Conference on Neural Networks*. (Nov. 27–Dec. 1, 1995). Vol. 4. Perth, WA: IEEE, Nov. 1995, pp. 1942–1948. DOI: 10.1109/ICNN.1995.488968.

[41] J. Kennedy and R. Mendes. "Neighborhood topologies in fully informed and best-of-neighborhood particle swarms". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 36.4 (July 2006), pp. 515–519. DOI: 10.1109/TSMCC.2006.875410.

[42] A. Kestilä et al. "Aalto-1 nanosatellite – technical description and mission objectives". In: *Geoscientific Instrumentation, Methods and Data Systems* 2.1 (Feb. 21, 2013), pp. 121–130. DOI: 10.5194/gi-2-121-2013.

[43] J. B. Kuipers. *Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace, and Virtual Reality*. Princeton University Press, 1999. ISBN: 0-691-05872-5.

[44] W. J. Larson and J. R. Wertz. *Space Mission Analysis and Design*. 3rd ed. 8 vols. Space Technology Library Series. El Segundo (CA): Microcosm, 1999. ISBN: 1-881883-10-8.

[45] K. Liu et al. "The development of micro-gyroscope technology". In: *Journal of Micromechanics and Microengineering* 19.11 (2009). DOI: 10.1088/0960-1317/19/11/113001.

[46] Y. X. Liu et al. "Novel Calibration Algorithm for a Three-Axis Strapdown Magnetometer". In: *Sensors* 14.5 (May 14, 2014), pp. 8485–8504. DOI: 10.3390/s140508485.

[47] L.-Y. Liu et al. "The Magneto-Resistive Magnetometer of BCU on the Tatiana-2 Satellite". In: *Terrestrial, Atmospheric and Oceanic Sciences* 23.3 (June 2012), pp. 317–326. DOI: 10.3319/TAO.2011.11.07.01(AA).

[48] M. Lovera and A. Astolfi. "Spacecraft attitude control using magnetic actuators". In: *Automatica* 40.8 (Aug. 2004), pp. 1405–1414. DOI: 10.1016/j.automatica.2004.02.022.

[49] H. Lühr, F. Yin, and R. Bock. "Magnetic properties of CHAMP and their effects on in-orbit calibration". In: *Journal of Sensors and Sensor Systems* 2.1 (2013), pp. 9–17. DOI: 10.5194/jsss-2-9-2013.

[50] M. Mahfouf, M.-Y. Chen, and D. A. Linkens. "Adaptive Weighted Particle Swarm Optimisation for Multi-objective Optimal Design of Alloy Steels". English. In: *Parallel Problem Solving from Nature - PPSN VIII*. Ed. by X. Yao et al. Vol. 3242. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2004, pp. 762–771. ISBN: 978-3-540-23092-2. DOI: 10.1007/978-3-540-30217-9_77.

[51] I. Maqsood and T. Akram. "Development of a low cost sun sensor using quadphotodiode". In: *2010 IEEE/ION Position Location and Navigation Symposium (PLANS '10)*. (May 4–6, 2010). Indian Wells, CA, USA, May 2010, pp. 639–644. DOI: 10.1109/PLANS.2010.5507186.

[52] F. L. Markley and J. L. Crassidis. *Fundamentals of Spacecraft Attitude Determination and Control*. 33 vols. Space Technology Library. New York: Springer, 2014. ISBN: 978-1-4939-0802-8. DOI: 10.1007/978-1-4939-0802-8.

[53] H. Martin et al. "A New Approach to Better Low-cost MEMS IMU Performance Using Sensor Arrays". In: *Proceedings of the 26th International Technical Meeting*
2013). (Sept. 16–19, 2013). Nashville, TN, Sept. 2013, pp. 2125–2142.

[54] K. Mehlem, A. Wiegand, and S. Weickert. "New developments in magnetostatic cleanliness modeling". In: *2012 Proceedings ESA Workshop on Aerospace EMC*. (May 12–23, 2012). Venice, Italy, May 2012.

[55] K. Mehlem. "Multiple magnetic dipole modeling and field prediction of satellites". In: *IEEE Transactions on Magnetics* 14.5 (Sept. 1978), pp. 1064–1071. DOI: 10.1109/TMAG.1978.1059983.

[56] K. Mohamadabadi. "Anisotropic Magnetoresistance Magnetometer for inertial navigation systems". PhD thesis. Ecole Polytechnique X, Nov. 2013. URL: https://tel.archives-ouvertes.fr/tel-00946970 (visited on 02/21/2015).

[57] C. K. Monson and K. D. Seppi. "Exposing origin-seeking bias in PSO". In: *Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation Conference (GECCO) 2005*. (June 25–29, 2005). Washington DC, USA: ACM, June 2005, pp. 241–248. DOI: 10.1145/1068009.1068045.

[58] A. Näsilä et al. "Aalto-1: a hyperspectral Earth observing nanosatellite". In: *Proc. SPIE*. Sensors, Systems, and Next-Generation Satellites XV 8176 (Oct. 3, 2011), p. 81760C. DOI: 10.1117/12.898125.

[59] National Aeronautics and Space Administration. *Assessment and control of spacecraft magnetic fields*. Technical Report NASA-SP-8037. NASA, Sept. 1970. URL: http://hdl.handle.net/2060/19710003603 (visited on 02/15/2015).

[60] R. Poli. "Analysis of the Publications on the Applications of Particle Swarm Optimisation". In: *Journal of Artificial Evolution and Applications* 2008, 685175 (2008), p. 10. DOI: 10.1155/2008/685175.

[61] M. A. Post, J. Li, and R. Lee. "A Low-Cost Photodiode Sun Sensor for CubeSat and Planetary Microrover". In: *International Journal of Aerospace Engineering* 2013 (Nov. 4, 2013). DOI: 10.1155/2013/549080.

[62] J. Praks et al. "Aalto-1 – An experimental nanosatellite for hyperspectral remote sensing". In: *2011 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. (July 24–29, 2011). Vancouver, BC: IEEE, July 2011, pp. 4367–4370. DOI: 10.1109/IGARSS.2011.6050199.

[63] Z. Qin and Y. Liu. "Optimal combination for multi-objective Particle Swarm Optimization". In: *2014 IEEE 7th Joint International Information Technology and Artificial Intelligence Conference (ITAIC '14)*. (Dec. 20–21, 2014). Chongqing, China: IEEE, Dec. 2014, pp. 11–15. DOI: 10.1109/ITAIC.2014.7064996.

[64] V. Renaudin, M. H. Afzal, and G. Lachapelle. "Complete triaxis magnetometer calibration in the magnetic domain". In: *Journal of Sensors* (2010). DOI: 10.1155/2010/967245.

[65] M. Reyes-Sierra and C. A. C. Coello. "Multi-Objective particle swarm optimizers: A survey of the state-of-the-art". In: *International Journal of Computational Intelligence Research* 2.3 (2006), pp. 287–308.

[66] J. D. Searcy and H. J. Pernicka. "Magnetometer-Only Attitude Determination Using Novel Two-Step Kalman Filter Approach". In: *Journal of Guidance, Control, and Dynamics* 35.6 (Nov.–Dec. 2012), pp. 1693–1701. DOI: 10.2514/1.57344.

[67] J. Sedlak and J. Hashmall. "Improved fine sun sensor field of view calibration". In: *Flight Mechanics Symposium*. (Oct. 28–30, 2003). Greenbelt, MD, United States: NASA Goddard Space Flight Center, Oct. 2003.

[68] D. Selva and D. Krejci. "A survey and assessment of the capabilities of Cubesats for Earth observation". In: *Acta Astronautica* 74 (May–June 2012), pp. 50–68. DOI: 10.1016/j.actaastro.2011.12.014.

[69] N. El-Sheimy, H. Hou, and X. Niu. "Analysis and Modeling of Inertial Sensors Using Allan Variance". In: *IEEE Transactions on Instrumentation and Measurement* 57.1 (Jan. 2008), pp. 140–149. DOI: 10.1109/TIM.2007.908635.

[70] D. Sheng et al. "Subfemtotesla Scalar Atomic Magnetometry Using Multipass Cells". In: *Physical Review Letters* 110.16 (Apr. 2013). DOI: 10.1103/PhysRevLett.110.160802.

[71] Y. Shi and R. C. Eberhart. "Parameter selection in particle swarm optimization". English. In: *Evolutionary Programming VII*. Ed. by V. Porto et al. Vol. 1447. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 1998, pp. 591–600. ISBN: 978-3-540-64891-8. DOI: 10.1007/BFb0040810.

[72] E. Silani and M. Lovera. "Magnetic spacecraft attitude control: a survey and some new results". In: *Control Engineering Practice* 13.3 (Mar. 2005), pp. 357–371. DOI: 10.1016/j.conengprac.2003.12.017.

[73] I. Söderkvist. *Using SVD for some fitting problems*. URL: http://www.ltu.se/cms_fs/1.51590!/svd-fitting.pdf (visited on 07/22/2015).

[74] H. E. Söken and C. Hajiyev. "Adaptive Unscented Kalman Filter with Multiple Fading Factors for Pico Satellite Attitude Estimation". In: *4th International Conference on Recent Advances in Space Technologies (RAST) 2009*. (June 11–13, 2009). Istanbul, Turkey: IEEE, June 2009, pp. 541–546. DOI: 10.1109/RAST.2009.5158254.

[75] H. E. Söken and S.-i. Sakai. "Adaptive Tuning of the Unscented Kalman Filter for Satellite Attitude Estimation". In: *Journal of Aerospace Engineering* (Feb. 7, 2014), p. 04014088. DOI: 10.1061/(ASCE)AS.1943-5525.0000412.

[76] H. E. Söken and S.-i. Sakai. "UKF Based On-Orbit Gyro and Magnetometer Bias Estimation as a part of the Attitude Determination Procedure for a Small Satellite". In: *2011 11th International Conference on Control, Automation and Systems (ICCAS '11)*. (Oct. 26–29, 2011). Gyeonggi-do, Korea: IEEE, Oct. 2011, pp. 1891–1896.

[77] H. E. Söken, S.-i. Sakai, and R. Wiśniewski. "In-Orbit Estimation of Time-Varying Residual Magnetic Moment". In: *IEEE Transactions on Aerospace and Electronic Systems* 50.4 (Oct. 2014), pp. 3126–3136. DOI: 10.1109/TAES.2014.130225.

[78] H. Späth. "Orthogonal least squares fitting with linear manifolds". English. In: *Numerische Mathematik* 48.4 (July 1986), pp. 441–445. DOI: 10.1007/BF01389650.

[79] J. C. Springmann and J. W. Cutler. "Attitude-Independent Magnetometer Calibration with Time-Varying Bias". In: *Journal of Guidance, Control, and Dynamics* 35.4 (July–Aug. 2012), pp. 1080–1088. DOI: 10.2514/1.56726.

[80] J. C. Springmann and J. W. Cutler. "Flight results of a low-cost attitude determination system". In: *Acta Astronautica* 99 (June–July 2014), pp. 201–214. DOI: 10.1016/j.actaastro.2014.02.026.

[81] J. C. Springmann and J. W. Cutler. "On-Orbit Calibration of Photodiodes for Attitude Determination". In: *Journal of Guidance, Control, and Dynamics* 37.6 (Nov.–Dec. 2014), pp. 1808–1823. DOI: 10.2514/1.G000175.

[82] J. C. Springmann, J. W. Cutler, and H. Bahcivan. "Magnetic Sensor Calibration and Residual Dipole Characterization for Application to Nanosatellites". In: *Guidance, Navigation, and Control and Co-located Conferences*. AIAA/AAS Astrodynamics Specialist Conference. (Aug. 2–5, 2010). Toronto, Ontario Canada: American Institute of Aeronautics and Astronautics, Aug. 2010. DOI: 10.2514/6.2010-7518.

[83] J. C. Springmann et al. "The attitude determination system of the RAX satellite". In: *Acta Astronautica* 75 (June–July 2012), pp. 120–135. DOI: 10.1016/j.actaastro.2012.02.001.

[84] R. Strietzel. "Two-dimensional calibration of a sun attitude sensor". In: *Proceedings of the 15th IFAC World Congress*. (July 21–26, 2002). Ed. by L. Basañez and J. A. de la Puente. Barcelona, Spain: International Federation of Automatic Control, July 2002, p. 1266. DOI: 10.3182/20020721-6-ES-1901.01268.

[85] Thorlabs. *DC Servo Motor Driver*. TDC001 user guide. Nov. 17, 2014. URL: http://www.thorlabs.de/thorcat/15700/TDC001-Manual.pdf (visited on 08/11/2015).

[86] Thorlabs. *Motorized Rotation Stage*. CR1-Z7 operating manual. June 16, 2015. URL: http://www.thorlabs.de/thorcat/17000/CR1-Z7-Manual.pdf (visited on 08/11/2015).

[87] R. J. Vaccaro. "Statistical Modeling of Rate Gyros". In: *IEEE Transactions on Instrumentation and Measurement* 61.3 (Nov. 2, 2011), pp. 673–684. DOI: 10.1109/TIM.2011.2171609.

[88] J. R. Wertz. *Spacecraft attitude determination and control*. Astrophysics and Space Science Library : a series of books on the recent developments of space science and of general geophysics and astrophysics. Dordrecht: Reidel, 1978. ISBN: 90-277-0959-9.

[89] R. Wiśniewski and M. Blanke. "Fully magnetic attitude control for spacecraft subject to gravity gradient". In: *Automatica* 35.7 (July 1999), pp. 1201–1214. DOI: 10.1016/S0005-1098(99)00021-7.

[90] Z. Wu et al. "Calibration of Three-Axis Magnetometer Using Stretching Particle Swarm Optimization Algorithm". In: *IEEE Transactions on Instrumentation and Measurement* 62.2 (Feb. 2013), pp. 281–292. DOI: 10.1109/TIM.2012.2214951.

[91] Z. Wu et al. "Calibration of three-axis strapdown magnetometers using Particle Swarm Optimization algorithm". In: *2011 IEEE International Symposium on Robotic and Sensors Environments (ROSE '11)*. (Sept. 17–18, 2011). Montreal, QC: IEEE, Sept. 2011, pp. 160–165. DOI: 10.1109/ROSE.2011.6058522.

[92] Z. Xiao-hua, M. Hong-yun, and J. Li-cheng. "Intelligent particle swarm optimization in multiobjective optimization". In: *The 2005 IEEE Congress on Evolutionary Computation*. (Sept. 5, 2005). Vol. 1. Edinburgh, Scotland: IEEE, Sept. 2005. ISBN: 714–719. DOI: 10.1109/CEC.2005.1554753.

[93] N. Xie and A. Theuwissen. "A Miniaturized Micro-Digital Sun Sensor by Means of Low-Power Low-Noise CMOS Imager". In: *IEEE Sensors Journal* 14.1 (Jan. 2014), pp. 96–103. DOI: 10.1109/JSEN.2013.2280453.

[94] L. Xue et al. "A novel Kalman filter for combining outputs of MEMS gyroscope array". In: *Measurement* 45.4 (May 2012), pp. 745–754. DOI: 10.1016/j.measurement.2011.12.016.

[95] Z. Zeng et al. "Robust Adaptive Filter for Small Satellite Attitude Estimation Based on Magnetometer and Gyro". In: *Abstract and Applied Analysis* 2014 (May 26, 2014). DOI: 10.1155/2014/159149.