

MASTER OF SCIENCE THESIS

REAL-TIME POSITION AND ATTITUDE DETERMINATION
OF THE STRATOS II SOUNDING ROCKET

BRYAN TONG MINH



MASTER OF SCIENCE THESIS

**Real-time position and attitude determination of the
Stratos II sounding rocket**

BRYAN TONG MINH

Delft University of Technology

JULY 24, 2012

Cover image:

- Globus by Hannes Grobe, CC-BY-3.0 (https://commons.wikimedia.org/wiki/File:Globus_book-shelf_hg.jpg)
- Dragon's Lair background image by NASA JPL
- Stratos II render by Jeroen van Straten

Preface

At the time of writing, it has been five years since I visited the presentation that would have a profound impact on the further course of my educational career. In the 2007 academic year, I visited a presentation by DARE, and I was immediately fascinated by the prospect of being part of a group that designed, built, and launcher their own rockets. My involvement in DARE eventually culminated in several apoapsides, such as the launch of the Stratos rocket with its successful second-stage ignition, the evitable failure of the parachute deployment, and the eventual impact with the ground. During the development of the software of the Stratos flight computer, I found out about the difficulty of knowing where you are, what is going on, and what is going to happen, being a rocket. Unfortunately, we found out the hard way the importance of these topics. When we first started to brainstorm about the successor of the Stratos, the Stratos II, it was clear that we would need a more sophisticated system to know the state of the rocket. Fortunately, at that time, I was about to choose a topic for my MSc. thesis, and I quickly decided that for my thesis, I wanted to research the position and attitude determination of Stratos II.

I have been fortunate that the chair of Astrodynamics and Satellite Missions was fully supportive of this research topic. Prof. Ambrosius, Dr. Mooij and Dr. Verhoeven were immediately enthusiastic of this research. Moreover, they also supported putting my research into practice, rather than only relying on simulations and theoretical results, for which I am grateful to have had the opportunity, and which was the part of the thesis research that I enjoyed the most. Of course, aside from my graduation supervisors – in particular Erwin Mooij – also other people have helped me complete this thesis. First of all, with the entire Stratos II graduation room – Hein and Frank – we had many fruitful and enjoyable discussions, both on- and off-topic. Secondly, I would like to thank my friends from DARE for all their work and the great collaboration we had the last five years, in particular the electronics group (Eric, Jeroen, Bart, John-John) and the capsule team (Saskia, Joost, Mats and Jork) and the Stratos I and II teams. Without The Room 903 the duration of my graduation would have been a much quieter time – and of course I would not have found an SAM taped on my wall. Finally, despite the slightly longer duration of my graduation than expected, my family has always supported me, for which I am thankful.

I think most readers will have read enough of this off-topic text, and are without doubt eager to dive into the wonderful world of position and attitude determination of sounding rockets. So, the only thing for me left to say, is, enjoy the read and I hope you have as much fun reading, as I had writing it.

— Bryan Tong Minh
July 24, 2012

Abstract

Currently, the student amateur rocket association Delft Aerospace Rocket Engineering (DARE), is designing a small sounding rocket to carry payloads to over 50 km, the Stratos II. Proper operation of this rocket will require an on-board position and attitude determination system. In this thesis, it will be researched how this position and attitude determination can be performed. In essence, the research can be divided into three main topics: *simulation*, *measurement* and *estimation*. Flight estimation is the main goal of the research, while simulation and measurements are prerequisites to estimation.

A measurement system with low-cost sensors has been designed and built, and during calibration the performance of these sensors has been established. This measurement system has flown in an actual small sounding rocket, the Stratos II concept launcher, and acceleration, angular velocity and atmospheric pressure data have been acquired. Several estimators have been researched, from which the extended Kalman filter was selected as the most suitable to estimate the position and attitude. The accelerometer and gyroscope measurements are used to propagate the state, and the other measurement data to improve the state estimate, and to estimate various noise factors of the sensors. Measurements are generated from simulations of nominal and non-nominal flights, with nominal and non-nominal sensors, to verify the performance of the estimator. It was found that both the flight trajectory, as well as sensor noise factors such as bias and non-orthogonality strongly affect the estimation result.

As only coarse requirements on the position and attitude determination system are available, it cannot be concluded with certainty that the developed measurement system and estimator are suitable for the Stratos II. However, preliminary analysis performed in this thesis, indicates that this is the case. Furthermore, in this thesis clear guidelines are established to improve the estimator performance, should the developed system not satisfy the final requirements.

Contents

Preface	v
Abstract	vii
List of symbols	xiii
List of abbreviations	xvii
List of figures	xxiii
List of tables	xxvi
1 Introduction	1
1.1 Scope of the thesis	1
1.2 Heritage	2
1.3 Flight profile of Stratos II	3
1.4 Requirements on the navigation system	6
1.5 Thesis research question and methodology	6
1.6 Thesis outline	7
2 Flight mechanics and trajectory simulation	9
2.1 Reference systems	9
2.1.1 List of reference frames	9
2.1.2 Transformations between reference frames	12
2.1.3 Coordinates in WGS84	14
2.2 Attitude representations	14
2.2.1 Euler axis/angle	15
2.2.2 Quaternions	16
2.2.3 Modified Rodrigues parameters	17
2.2.4 Quantifying attitude differences	18
2.2.5 Presentational representations for attitude	19
2.3 Equations of motion	19
2.3.1 Translational equations of motion	19
2.3.2 Rotational equations of motion	21
2.4 Forces and moments	22
2.4.1 Variable mass forces and moments	22
2.4.2 Aerodynamic forces	23
2.4.3 Gravitational forces	25
2.4.4 Constraint forces	26
2.4.5 Flight phases	26
2.5 Modeled rockets	28
2.5.1 Stratos II	28

2.5.2	Stratos II concept launcher	29
2.6	Rocket Simulation and Impact Estimation	29
2.7	Simplified simulator model	31
2.7.1	Equations of motion	32
2.7.2	Gravity	32
2.7.3	Thrust	32
2.7.4	Aerodynamics	33
2.7.5	Mass properties	35
2.7.6	Tower constraints	36
2.8	Summary	36
3	Estimation theory	37
3.1	Least-squares estimation	37
3.2	Kalman filtering	38
3.2.1	Extended Kalman filter	39
3.2.2	Unscented Kalman filter	41
3.2.3	Rauch-Tung-Striebel smoother	42
3.3	Outlook	43
4	Navigation models	45
4.1	Simplified estimator model	45
4.2	Measurement models	47
4.2.1	Accelerometer	47
4.2.2	Gyroscope	48
4.2.3	Magnetometer	49
4.2.4	Barometric pressure sensor	49
4.2.5	GPS	50
4.3	Kinematic estimator model	52
4.4	Flight computer model	53
5	Measurement system design	55
5.1	Measurement system development	55
5.1.1	Component selection and system level design	55
5.1.2	Circuit and PCB design	56
5.1.3	Manufacturing	58
5.1.4	Software development	58
5.2	Sensor calibration	58
5.2.1	Misalignment	60
5.2.2	Noise	62
5.2.3	Pressure sensor	62
5.2.4	Gyroscope	63
5.2.5	Accelerometer	65
6	Verification and validation	67
6.1	Simulator	67
6.2	Measurement system	69
6.3	Estimator	70
6.3.1	Falling body estimation	70
6.3.2	Model testing	72
6.3.3	System testing	73

7 Stratos II simulation and estimation results	75
7.1 Estimator performance evaluation	75
7.2 Experimental design	76
7.2.1 Orthogonal arrays	77
7.2.2 Response surface	77
7.3 Nominal trajectory simulation	78
7.4 Model and estimator type	79
7.4.1 Computational requirements	82
7.4.2 Simplified model estimator	83
7.4.3 Kinematic model estimator	93
7.4.4 Model comparison and selection	98
7.5 Sensor factors	105
7.5.1 GPS measurements and bias estimation	105
7.5.2 Sensor control factors	111
7.5.3 Sensor noise factors	113
7.5.4 Sensor failure	120
7.6 Non-nominal trajectory	126
7.7 Non-nominal flight estimation	129
7.8 Result summary	134
8 Post-flight Stratos II concept launcher results	135
8.1 Flight results	136
8.1.1 Raw measurement data	137
8.2 Trajectory simulation	139
8.3 Estimation results of simulated trajectory	142
8.3.1 Pressure measurements	142
8.3.2 Estimation with nominal sensors	142
8.4 Sensor noise factors	148
8.5 Estimation results of flight	154
8.5.1 Comparison with simulation	154
8.5.2 Validity and accuracy of the estimation result	156
9 Conclusions and recommendations	159
9.1 Conclusions	159
9.2 Recommendations	161
Bibliography	165
A Attitude derivations	167
A.1 Derivation of the MRP time derivative equation	167
B Partial derivatives	169
B.1 Rules of differentiation	169
B.1.1 Differentiation of vector norms	169
B.1.2 Differentiation of frame transformations	170
B.1.3 Differentiation of the cross product	170
B.2 Auxiliary partial derivatives	171
B.2.1 Aerodynamic partial derivatives	171
B.2.2 Attitude partial derivatives	172
B.3 State derivative partial derivatives	173
B.3.1 Translational state derivative partial derivatives	173
B.3.2 Rotational state derivative	174

C Estimator covariance properties	177
C.1 Simplified estimator	177
C.2 Kinematic estimator model	180
D Estimation with ROSIE	181
D.1 Motor underperformance	181
D.2 Roll motion	182
D.3 Performance of the estimator	183
E Orthogonal arrays	185
E.1 3-level, 81-run orthogonal array	185
F Results	189
F.1 GPS tight coupling	189
F.2 Sensor control factors	192
F.3 Sensor noise factors	195

List of symbols

Symbol	Description	Units
a	Semi-major axis	m
a	Speed of sound	m/s
\mathbf{b}	Measurement bias vector	[various]
\mathbf{C}	Direction cosine matrix	[–]
C_{l_0}	Aerodynamic static rolling moment coefficient	[–]
C_X	Aerodynamic x-force coefficient	[–]
C_Y	Aerodynamic y-force coefficient	[–]
C_{Y_β}	Aerodynamic y-force due to side slip angle coefficient	rad ⁻¹
C_Z	Aerodynamic z-force coefficient	[–]
C_{Z_α}	Aerodynamic z-force due to angle of attack coefficient	rad ⁻¹
\mathbf{D}	Diagonal part of UDU decomposition of covariance	[various]
\mathbf{D}	Dilution of precision matrix	[–]
d_{ij}	Non-orthogonality	[–]
\mathbf{e}	Euler rotation axis	[–]
e	Eccentricity	[–]
\mathbf{E}	Skew-symmetric matrix of \mathbf{e}	[–]
E	Eastwards coordinate	m
f	Specific force	m/s ²
\mathbf{F}	Force	N
\mathbf{F}_{k-1}	State transition matrix of \mathbf{x} from t_{k-1} to t_k	[various]
h	Altitude / height above geoid	m
H	Scaling height	m
h_A	Apogee altitude	m
\mathbf{H}_k	Measurement matrix at time t_k	[various]
\mathbf{I}	Identity matrix	[–]
\mathbf{I}	Inertia matrix	kg · m ²
l_{ref}	Reference length	m
$\hat{\mathbf{i}}$	Unit vector in x-direction	[–]
$\hat{\mathbf{j}}$	Unit vector in y-direction	[–]
\mathbf{K}	(Kalman) gain	[various]
$\hat{\mathbf{k}}$	Unit vector in z-direction	[–]
\mathcal{L}	Moment around the x_B -axis	Nm
\mathbf{m}	Magnetic field vector	T

m	Mass	kg
\mathcal{M}	Moment around the y_B -axis	Nm
\boldsymbol{M}	Moment	Nm
M	Mach number	[−]
\dot{m}_e	Exhaust mass flow	kg/s
\mathcal{N}	Moment around the z_B -axis	Nm
N	Noise spectral density	[various]
N	Northwards coordinate	m
\boldsymbol{p}	Linear momentum	kg · m/s
p	Angular velocity component around x-body axis	rad/s
p	Pressure	N/m ²
\mathbf{P}_k	Error covariance matrix at time t_k	[various]
\boldsymbol{q}	Quaternion	[−]
q	Angular velocity component around y-body axis	rad/s
q_i	Quaternion with $i = 1 \dots 4$	[−]
\mathbf{Q}	Covariance matrix of the process noise	[various]
\mathbf{Q}_{xx}	Covariance matrix of \boldsymbol{x}	[various]
\boldsymbol{r}	Response	[various]
r	Angular velocity component around z-body axis	rad/s
\mathbf{R}	Covariance matrix of the measurement noise	[various]
R	Range	m
R	Specific gas constant	J/kg/K
R_e	Earth radius	m
R_N	Normal radius	m
s	Scale factor	[−]
\mathbf{S}	Scale factor matrix	[various]
s_i	Scale factor error	[−]
S_{ref}	Reference area	m ²
T	Temperature	K
\mathbf{U}	Upper triangular part of UDU decomposition of covariance	[various]
U	Geopotential	m ² /s ²
$\hat{\mathbf{u}}_L$	Unit vector in launch rail direction	[−]
\mathbf{V}_A	Aerodynamic velocity	m/s
\mathbf{V}_e	Mean exhaust velocity	m/s
\mathbf{V}_W	Wind velocity	m/s
\boldsymbol{w}	Zero-mean, Gaussian noise	[various]
\mathbf{W}	Weight matrix	[various]
\boldsymbol{x}	State vector	[various]
$\hat{\boldsymbol{x}}$	State estimation	[various]
$\tilde{\boldsymbol{x}}^{(i)}$	i-th sigma point of \boldsymbol{x}	[various]
\boldsymbol{y}	Measurement vector	[various]
y_{CoM}	Center of mass offset in y-direction	m
$\hat{\boldsymbol{y}}$	Measurement estimation	[various]

α	Angle of attack	rad
β	Response surface coefficient	[various]
β	Side slip angle	rad
$\bar{\beta}$	Normalized response surface coefficient	[–]
γ	Elevation	rad
γ	Ratio of specific heats	[–]
δ	Geocentric latitude	rad
$\delta\mathbf{a}$	Accelerometer non-orthogonality matrix	[–]
$\delta\mathbf{m}$	Magnetometer non-orthogonality matrix	[–]
$\delta\mathbf{q}$	Difference quaternion	[–]
δq_4	Scalar part of difference quaternion	[–]
$\delta\mathbf{S}_a$	Accelerometer scale factor error matrix	[–]
$\delta\mathbf{S}_m$	Magnetometer scale factor error matrix	[–]
$\delta\mathbf{S}_\omega$	Gyroscope scale factor error matrix	[–]
$\delta\boldsymbol{\omega}$	Gyroscope non-orthogonality matrix	[–]
$\delta\theta$	Angular difference	rad
ε	Estimation residual	[various]
θ	Rotation around Euler axis	rad
η_{C_X}	Aerodynamic x-force correction coefficient	[–]
η_T	Thrust magnitude correction coefficient	[–]
λ	Temperature gradient	K/m
μ_e	Standard gravitational parameter of the Earth	m^3/s^2
ρ	Atmospheric density	kg/m^3
ϱ	Square root density	$\text{kg}^{0.5}\text{m}^{-1.5}$
$\boldsymbol{\varrho}$	Vector part of the quaternion	[–]
σ	Modified Rodrigues parameter	[–]
σ	Standard deviation	[various]
τ	Geocentric longitude	rad
χ	Azimuth	rad
Ψ	State transition matrix	[various]
ω	Rotational rate along an axis	rad
ω_E	Rotational rate of the Earth along its polar axis	rad

Notations

Symbol	Description
x	Vector x
\hat{x}	Estimate of x
\dot{x}	Time derivative of x
\ddot{x}	Second time derivative of x
$\ x\ $	Norm of x
X	Matrix X
$ X $	Determinant of X

List of abbreviations

ADC	Analog-to-Digital Convertor
BLUE	Best Linear Unbiased Estimator
CEST	Central European Summer Time
COCOM	Coordinating Committee for Multilateral Export Controls
CoM	Center of Mass
DARE	Delft Aerospace Rocket Engineering
DCM	Direction Cosine Matrix
DRC	Design Rule Check
ECEF	Earth Centered Earth Fixed
ECI	Earth Centered Inertial
EKF	Extended Kalman Filter
EME2000	Earth Mean Equator and Equinox of J2000
GPS	Global Positioning System
I ² C	Inter-Integrated Circuit
IEKF	Iterated Extended Kalman Filter
IERS	International Earth Rotation and Reference Systems Service
IIP	Instantaneous Impact Point
ISA	International Standard Atmosphere
ITRS	International Terrestrial Reference System
JGM-3	Joint Gravity Model 3
LGA	Land Grid Array
LSB	Least Significant Bit
LTI	Linear Time Invariant
LU	Lower-Upper
MEMS	Microelectromechanical System
MRP	Modified Rodrigues Parameters
NASA	National Aeronautics and Space Administration
NED	North-East-Down
PCB	Printed Circuit Board
PIGA	Pendulum Integrating Gyroscopic Accelerometer
RK	Runge-Kutta
RKF	Runge-Kutta-Fehlberg
RMS	Root-Mean-Squared
ROSIE	Rocket Simulation and Impact Estimation

RTC	Real-time Clock
RTS	Rauch-Tung-Striebel
SPI	Serial Peripheral Interface
Tudat	TU Delft Astrodynamics Toolbox
UART	Universal Asynchronous Receiver/Transmitter
UD	Upper-triangular-diagonal
UERE	User Equivalent Range Error
UKF	Unscented Kalman Filter
WGS84	World Geodetic System 1984
WLSE	Weighted Least Squares Estimator
WMM2010	World Magnetic Model 2010

List of figures

1.1	Launch of Stratos I at Esrange. Source: SSC	2
1.2	Part of the Stratos I flight computer, with in the center a measurement board with pressure and accelerometers. Source: Frank Engelen, DARE	4
1.3	Preliminary flight profile of Stratos II. Source: Engelen (2012)	5
2.1	Body-fixed reference frame for a rocket. Source: Siouris (2004)	10
2.2	Bottom view of a rocket attached to a launch rail, showing the axes definition.	11
2.3	Earth Inertial, Earth Centered and Vehicle Vertical reference frames. Source: Mooij (1997)	13
2.4	Angles between the aerodynamic and body-fixed reference frame. Source: Siouris (2004)	14
2.5	Euler axis in A. Source: Chu (2010)	15
2.6	Definition of azimuth (χ) and elevation (γ) in a north-east-down frame (NED)	19
2.7	Schematic drawing of relative force and moment direction and the center of mass of a rocket	23
2.8	Drawing of rocket in tower with the forces applying to it	27
2.9	Dimensional drawing of the Stratos II. Source: Engelen (2012)	28
2.10	CATIA model of the concept launcher, showing the separate components for which the mass was specified	30
2.11	Picture sequence showing the release of the clamp band, the structural connection between the nose cone and the motor module. Source: Roel Vandeberg, DARE	30
2.12	Angles between the aerodynamic and body-fixed reference frame. Source: Siouris (2004)	35
3.1	The prediction-update cycle of a Kalman filter	38
4.1	State flow of the flight computer. Threshold values are indicative.	54
5.1	Block diagram of the preliminary measurement system	56
5.2	Schematic and footprint of gyroscope in Altium Designer	57
5.3	Partially routed measurement board	57
5.4	Respectively the top and bottom view of the designed PCB. Dimensions in mm.	59
5.5	A measurement board without accelerometers	60
5.6	Flowchart of the measurement board software	61
5.7	Pressure noise	63
5.8	Measurement boards mounted on the spin table	63
5.9	Gyroscope noise	64
5.10	Accelerometer noise	65
6.1	Distribution of accelerometer noise	68
6.2	Gyroscope measurements in the frequency domain	68
6.3	Simulated trajectory of a falling body	71

6.4	Estimation errors of the trajectory of a falling body	72
6.5	Difference between analytical and numerical partial derivatives of MRP time derivative	73
6.6	Attitude estimation error with noise-free accelerometer and gyroscope measurements	73
7.1	Nominal flight trajectory	79
7.2	Nominal velocity	80
7.3	Nominal Mach number	80
7.4	Nominal attitude	80
7.5	Nominal angular velocity	81
7.6	Nominal simulated measurements	81
7.7	Estimated altitude and ground track with the simplified model EKF	84
7.8	Position estimation error with the simplified model EKF	84
7.9	Estimated velocity with the simplified model EKF	84
7.10	Velocity estimation error with the simplified model EKF	85
7.11	Estimated attitude with the simplified model EKF	85
7.12	Attitude estimation error with the simplified model EKF	86
7.13	Estimated elevation and azimuth with the simplified model EKF	86
7.14	Estimated angular velocity with the simplified model EKF	86
7.15	Estimated augmented state variables with the simplified model EKF	87
7.16	Estimated altitude and ground track with the simplified model UKF	87
7.17	Estimated augmented state variables with the simplified model UKF	88
7.18	Estimated altitude and ground track with biased sensors with the simplified model EKF	89
7.19	Attitude estimation error with biased sensors with the simplified model EKF	89
7.20	Estimated bias with the simplified model EKF	90
7.21	Non-ideal rocket estimated altitude and ground track with the simplified model EKF	91
7.22	Non-ideal rocket velocity estimation error with the simplified model EKF	91
7.23	Non-ideal rocket attitude estimation error with the simplified model EKF	92
7.24	Non-ideal rocket estimated elevation and azimuth with the simplified model EKF	92
7.25	Non-ideal rocket estimated augmented state variables with the simplified model EKF	92
7.26	Estimated altitude and ground track with the kinematic model EKF	93
7.27	Position estimation error with the kinematic model EKF	94
7.28	Estimated velocity with the kinematic model EKF	94
7.29	Velocity estimation error with the kinematic model EKF	95
7.30	Estimated attitude with the kinematic model EKF	95
7.31	Attitude estimation error with the kinematic model EKF	95
7.32	Estimated elevation and azimuth with the kinematic model EKF	96
7.33	Position estimation error with the kinematic model UKF	97
7.34	Velocity estimation error with the kinematic model UKF	97
7.35	Estimated attitude with the kinematic model UKF	98
7.36	Estimated altitude and ground track with biased sensors with the kinematic model EKF	99
7.37	Attitude estimation error with biased sensors with the kinematic model EKF	99
7.38	Estimated bias with the kinematic model EKF	99
7.39	Estimated bias with accelerometer bias estimation with the kinematic model EKF	100
7.40	Non-ideal rocket position estimation error with the kinematic model EKF	100
7.41	Non-ideal rocket velocity estimation error with the kinematic model EKF	101
7.42	Non-ideal rocket attitude estimation error with the kinematic model EKF	101
7.43	Estimated altitude and ground track with biased sensors and loose GPS coupling	106

7.44	Estimated altitude and ground track with bias estimation and loose GPS coupling	107
7.45	Estimated bias with loose GPS coupling	107
7.46	Estimated bias with tight GPS coupling	108
7.47	Position and attitude error for various levels of GPS coupling	109
7.48	Translational error dependence on noise and frequency	111
7.49	Attitudinal error dependence on noise and frequency	112
7.50	Normalized response surface coefficients of sensor control factors	113
7.51	Normalized response surface coefficients with only bias estimation	114
7.52	Estimated accelerometer x-scale factor error	115
7.53	Estimated accelerometer x-scale factor and zx-non-orthogonality error	116
7.54	Estimated accelerometer x-scale factor and yx- and zx-non-orthogonality error	116
7.55	Estimation errors with bias estimation and sensor noise factors varied by orthogonal array	117
7.56	Estimation errors with accelerometer x-scale factor error estimation and sensor noise factors varied by orthogonal array	118
7.57	Estimation errors with accelerometer zx-non-orthogonality estimation and sensor noise factors varied by orthogonal array	118
7.58	Estimation errors with accelerometer x-scale factor and magnetometer zx-non-orthogonality estimation and sensor noise factors varied by orthogonal array	119
7.59	Estimation errors with accelerometer x-scale factor and yx- and zx-non-orthogonality estimation and sensor noise factors varied by orthogonal array	119
7.60	Position estimate with a failed accelerometer and without accelerometer bias estimation	121
7.61	Position estimate with a failed accelerometer with accelerometer bias estimation	121
7.62	Attitude estimate with a failed accelerometer with accelerometer bias estimation	122
7.63	Accelerometer bias estimate with a failed accelerometer with accelerometer bias estimation	122
7.64	Position estimate with a failed gyroscope	122
7.65	Attitude estimate with a failed gyroscope	123
7.66	Attitude estimation error with a failed gyroscope	123
7.67	Attitude estimation error with a failed magnetometer	124
7.68	Non-nominal flight trajectory	127
7.69	Non-nominal velocity	127
7.70	Non-nominal Mach number	127
7.71	Non-nominal attitude	128
7.72	Non-nominal angular velocity	128
7.73	Position error for non-nominal flight	130
7.74	Altitude error for non-nominal flight	130
7.75	Attitude error for non-nominal flight	130
7.76	Elevation error for non-nominal flight	130
7.77	Estimation error versus maximum angular velocity	131
7.78	Position error at characteristic points for non-nominal flight	133
7.79	Altitude error at characteristic points for non-nominal flight	133
7.80	Elevation error at characteristic points for non-nominal flight	133
8.1	The Stratos II concept launcher in the launch tower. Source: B.A.C. Ambrosius .	135
8.2	Rocket in launch tower just after ignition	136
8.3	Tumbling of the nose cone with undeployed parachute	136
8.4	Motor casing with burn hole	137
8.5	Raw measurement data	138
8.6	Non-nominal flight trajectory	139
8.7	Non-nominal velocity	140

8.8	Non-nominal Mach number	140
8.9	Non-nominal attitude	140
8.10	Non-nominal angular velocity	141
8.11	Position estimation with no pressure measurements	142
8.12	Position estimation with pressure measurements	143
8.13	Position error for concept launcher flight	146
8.14	Altitude error for concept launcher flight	146
8.15	Attitude error for concept launcher flight	146
8.16	Elevation error for concept launcher flight	146
8.17	Position error for concept launcher flight with pressure measurements	147
8.18	Altitude error for concept launcher flight with pressure measurements	147
8.19	Attitude error for concept launcher flight with pressure measurements	147
8.20	Elevation error for concept launcher flight with pressure measurements	147
8.21	Altitude error without and with pressure measurements for non-nominal sensors	148
8.22	Normalized response surface coefficients with pressure measurements	149
8.23	Estimation of the accelerometer non-orthogonalities	149
8.24	Normalized response surface coefficients with pressure measurements and accelerometer non-orthogonality estimation	150
8.25	Estimation of the accelerometer non-orthogonalities and scale factor	151
8.26	Normalized response surface coefficients with pressure measurements and accelerometer non-orthogonality and scale factor estimation	152
8.27	Estimation errors with sensor noise factors varied by orthogonal array and with pressure measurements	152
8.28	Estimation errors with sensor noise factors varied by orthogonal array and with pressure measurements and accelerometer non-orthogonality estimation	153
8.29	Estimation errors with sensor noise factors varied by orthogonal array and with pressure measurements and accelerometer non-orthogonality and x-scale factor error estimation	153
8.30	Estimated position from flight measurements	154
8.31	Estimated velocity from flight measurements	155
8.32	Estimated attitude from flight measurements	155
8.33	Estimated and nominal ground track of the flight at 't Harde, the Netherlands. Background by Google Earth	155
8.34	Estimated accelerations	156
8.35	Pressure measurements estimations	157
8.36	Flight sensor noise estimation	158
8.37	Estimated and nominal ground track of the flight at 't Harde, the Netherlands for different estimators. Background by Google Earth	158
D.1	Altitude and position error for varying thrust levels	182
D.2	Ground track and attitude error for varying thrust levels	182
D.3	Ground track and position error for various roll coefficients	183
D.4	Estimated rolling moment coefficient and attitude error for various roll coefficients	183
F.1	Annotated position error for various levels of GPS coupling	189
F.2	Estimated bias with tight GPS coupling case 1 (0 satellites)	190
F.3	Estimated bias with tight GPS coupling case 2 (1 satellite)	190
F.4	Estimated bias with tight GPS coupling case 3 (1 satellite)	190
F.5	Estimated bias with tight GPS coupling case 4 (1 satellite)	190
F.6	Estimated bias with tight GPS coupling case 31 (4 satellite)	191
F.7	Estimated bias with tight GPS coupling case 32 (5 satellite)	191
F.8	Normalized response surface coefficients with only bias estimation	196

F.9 Normalized response surface coefficients with accelerometer x-scale factor error estimation	198
F.10 Normalized response surface coefficients with accelerometer zx-non-orthogonality estimation	200
F.11 Normalized response surface coefficients with magnetometer zx-non-orthogonality estimation	202
F.12 Normalized response surface coefficients with accelerometer zx- and yx-non-orthogonality estimation	204

List of tables

2.1	United States Standard Atmosphere 1976 model. Source: Anderson (2005)	25
2.2	Properties of the used Stratos II model	28
2.3	Properties of the first generation Stratos II concept launcher	29
2.4	Moment of inertia for different models (kg m^2)	36
4.1	UERE standard deviation components. Source: Farrell (2008)	51
5.1	Gyroscope noise results	64
5.2	Gyroscope calibration results	64
5.3	Gyroscope mounting misalignment	64
5.4	Accelerometer calibration results	66
5.5	Accelerometer mounting misalignment	66
7.1	Orthogonal array of 4 runs and 3 factors	77
7.2	Estimator computational time comparison	82
7.3	Estimation errors for simplified model EKF	103
7.4	Estimation errors for simplified model UKF	103
7.5	Estimation errors for simplified model EKF with biased sensors	103
7.6	Estimation errors for simplified model EKF of non-ideal flight	103
7.7	Estimation errors for kinematic model EKF	104
7.8	Estimation errors for kinematic UKF	104
7.9	Estimation errors for kinematic model EKF with biased sensors	104
7.10	Estimation errors for kinematic model EKF of non-ideal flight	104
7.11	Kinematic model EKF estimation errors with loose GPS coupling	108
7.12	Kinematic model EKF estimation errors with bias estimation and loose GPS coupling	108
7.13	Kinematic model EKF estimation errors with bias estimation and tight GPS coupling	110
7.14	Response surface coefficients for sensor control factors	114
7.15	Kinematic model EKF estimation errors with a failed accelerometer	125
7.16	Kinematic model EKF estimation errors with a failed gyroscope	125
7.17	Kinematic model EKF estimation errors with a failed pressure sensor	125
7.18	Kinematic model EKF estimation errors with a failed magnetometer	125
7.19	Flight factors	126
7.20	Flight characteristics non-nominal flight	129
7.21	Non-nominal estimation response surface coefficients	132
7.22	Non-nominal estimation response surface coefficients characteristic points	134
8.1	Flight characteristics non-nominal flight	141
8.2	Concept launcher estimation error response surface coefficients	145
8.3	Concept launcher with pressure measurement estimation error response surface coefficients	145

8.4	Concept launcher with pressure measurement estimation error second order response surface coefficients	145
C.1	Initial covariance for the simplified estimator model	178
C.2	Process noise matrix for simplified estimator model	179
C.3	Initial covariance for the kinematic estimator model	180
F.1	Position mean error for varying noise and frequency [m]	192
F.2	Position max error for varying noise and frequency [m]	192
F.3	Altitude mean error for varying noise and frequency [m]	192
F.4	Altitude max error for varying noise and frequency [m]	193
F.5	Attitude mean error for varying noise and frequency [degrees]	193
F.6	Attitude max error for varying noise and frequency [degrees]	193
F.7	Elevation mean error for varying noise and frequency [degrees]	194
F.8	Elevation max error for varying noise and frequency [degrees]	194
F.9	Response surface coefficients with bias estimation	197
F.10	Response surface coefficients with accelerometer x-scale factor error estimation .	199
F.11	Response surface coefficients with accelerometer zx-non-orthogonality estimation	201
F.12	Response surface coefficients with magnetometer zx-non-orthogonality estimation	203
F.13	Response surface coefficients with accelerometer zx- and yx-non-orthogonality estimation	205

Chapter 1

Introduction

In the early days of the space age, space missions were exclusive to national space agencies. It followed naturally that also the sounding rocketry field was the domain of the national space agencies. This basically has been the case the entire twentieth century, however, as the twenty-first century progresses, space no longer is a domain exclusive to national agencies.

The same trend also applies to sounding rocketry. Currently, the student amateur rocket association Delft Aerospace Rocket Engineering (DARE), is designing a small sounding rocket to carry payloads to over 50 km. The system that is responsible for the on-board position and attitude determination, the navigation system, is one of the many subsystems that comprises a sounding rocket. In this thesis, it will be researched how this position and attitude determination can be performed.

1.1 Scope of the thesis

In 2001, a group of students at the Faculty of Aerospace Engineering at Delft University of Technology, decided that they would like to build and launch their own rockets. Therefore, they founded their own amateur rocketry student association, named *Delft Aerospace Rocket Engineering* (DARE). In the years following the founding of DARE, the group grew, and more rockets were launched every year, although limited to 1 km apogee altitude, the maximum possibility in the Netherlands. As the group grew, and more manpower and expertise became available, a more ambitious goal was sought. It was decided to try to break the European altitude record for amateur rocketry, at that time still set at 10.7 km altitude.

Early 2008, this ambition and the group effort culminated into the launch of the Stratos I rocket, to an apogee altitude of 12.3 km. The launch is pictured in figure 1.1. The Stratos I was by far the largest rocket ever designed and flown by DARE, with a length of over 3 m and a total mass of 50 kg. This mass was divided between a first and a second stage, each weighing 25 kg. The first stage contained 17 kg of solid propellant and delivered a maximum thrust of over 6 kN. The second stage contained 20 kg of solid propellant, and delivered a thrust of 4 kN (Uitendaal, 2009).

The launch of the Stratos I, however, was not the end of ambition. The next boundary was obvious: space. Still, ambition always has to be coupled with reality, and going to space at once might be one step too large. Therefore, for Stratos II an intermediate step was chosen: to launch a sounding rocket to 50 km *and* carry a scientific payload to that altitude and return it in one piece. Whereas for Stratos I simple timers and sensors sufficed, for the Stratos II, the *position and attitude determination system*, or the *navigation system* is an integral part of the mission.

The navigation system has multiple functions. Firstly, in its primary function it is responsible for triggering important flight events, such as stage separation, second-stage ignition and parachuting. Traditionally, in rockets this is performed by mechanical timers. However, perfor-

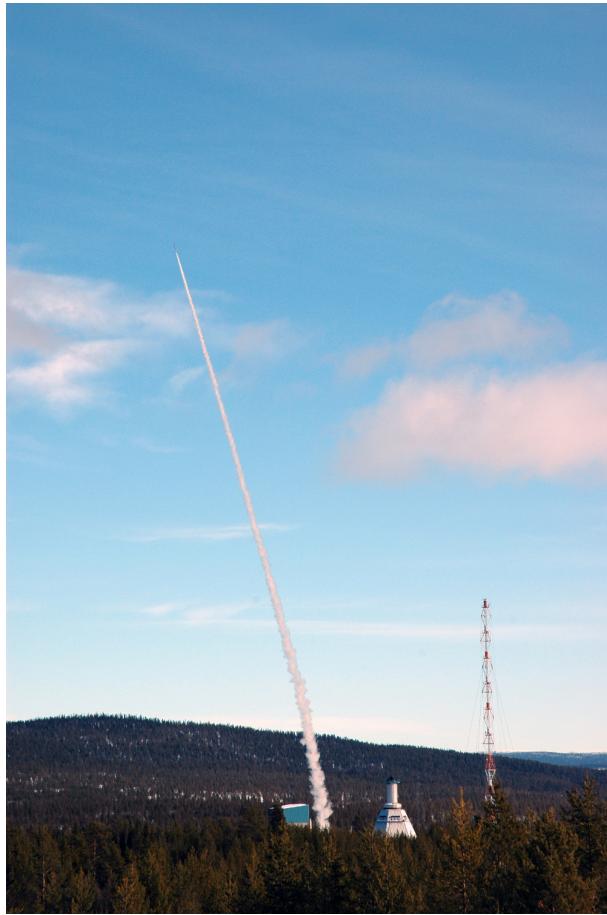


Figure 1.1: Launch of Stratos I at Esrange. Source: SSC

mance may improve if those events are triggered by certain flight conditions, such as a minimum altitude, which can be provided by the navigation system. Secondly, the navigation system will provide the scientific payloads with information on the current position and attitude. Thirdly, it may be used in post-flight analysis to compare the flight performance with the actual performance, which may help in designing any successors of the Stratos II. Finally, the navigation system may be used for range safety purposes. During the flight, the operations section at the launch site will continuously monitor the rocket flight and continuously calculate the instantaneous impact point. This point is the best estimate of the impact of the rocket based on the current flight data, assuming that the powered part of the flight would stop immediately. Range safety issues can be readily determined if the impact point is known. By applying data from the navigation system, it is possible to continuously improve the estimate of the impact point (Montenbruck, 2007).

1.2 Heritage

Attitude and position determination systems have a long heritage in the field of rocketry. The first rockets were designed as weapon-delivery systems and thus required some kind of system that could roughly determine the direction of the target as seen from the rocket. While the first systems only consisted of simple attitude determination, with the advance of technology more sophisticated systems were designed and built. Nevertheless, still today many sounding rockets only fly with a very coarse attitude determination system, or none at all, simply because they do not require it. The earliest example of this is perhaps the PIGA, a mechanical acceleration sensor with mechanical integration techniques, allowing a very rough estimation of the length

of the flight path flown (Haeussermann, 1981). It was used on the earliest ballistic missiles to provide fuel control.

With the advance of manned spaceflight it became apparent that better, full position and attitude estimations would require both an accurate flight model and sensors to incorporate deviations of the predetermined flight path. This led to research in filtering techniques pioneered by the Apollo program, resulting in the development of the widely used Kalman filter. The Kalman filter was proposed by Rudolf Kalman in 1960 (Kalman, 1960). Meanwhile, at NASA Ames research was being performed into the mid-course navigation and guidance of Apollo: the injection of the spacecraft in a circumlunar trajectory, around the Moon and back to Earth (McGee and Schmidt, 1985). Kalman's paper (Kalman, 1960) formulated the algorithm in state-space form, which was a new formulation at that time. A visit of Kalman to the research group brought insight to this paper, and the team immediately understood the practical use of Kalman's filter for their navigation problem. The linearized Kalman filter was selected, modified for practical applications and later extended to the extended Kalman filter. The measurement system that fed measurements into the Kalman filter was rather primitive. The onboard astronauts would look where the Earth and Moon were and measure the elevation, azimuth and subtended angles. These angles would then be manually typed into the Kalman filter. Since the Apollo program, the Kalman filter, extensions of it and also other estimation techniques have been widely studied (Simon, 2006).

Another part of the heritage of this thesis, is of course the Stratos I rocket. The Stratos I rocket did not possess a full attitude or position determination system. Nevertheless some sensors were present to observe part of the state. The rocket was equipped with 3 MEMS acceleration sensors which were able to measure the acceleration in 3 perpendicular directions. This was used to detect lift off and motor burn out. Furthermore a barometric pressure sensor was present, which could measure the static pressure, which was then used to derive the altitude. Finally also a GPS module was present, which was used to determine the maximum altitude and the impact point of the rocket. One of the measurement boards containing the sensors is shown in figure 1.2.

1.3 Flight profile of Stratos II

In order to be able to estimate the position and attitude of the rocket in flight, a description of the flight profile is required on beforehand. Currently, only a rough overview of the Stratos II flight is known. As stated before, Stratos II will be a two stage rocket, consisting of several flight phases. Exact values for altitude, velocity and acceleration are not yet known. Simulations to obtain these values will be performed in this thesis, but already some general characteristics of the flight can be given. The flight phases are presented here and shown in figure 1.3.

Pre-launch phase During the pre-launch phase the navigation system is measuring, but the rocket is at rest in the launch tower. This phase can be used to partially calibrate sensors.

First-stage burn During the first-stage burn, the rocket will accelerate with an estimated acceleration of 6 g and burn out after 10 seconds. It will then have reached an altitude of 3 km and flying with supersonic velocities. After burnout, the first-stage will separate from the rest of the rocket.

First coasting phase After first-stage burnout, the rocket will continue an unpowered flight for a few seconds until it has reached 8 km altitude. There, the second stage will be ignited. The decision to ignite the second stage will be based on the navigation system: the rocket should have reached sufficient altitude and should still be flying upwards.

Second-stage burn The second stage burn will be 10 seconds like the first stage, resulting in a maximum acceleration of 10 g and reach velocities around Mach 5 at burn-out at an

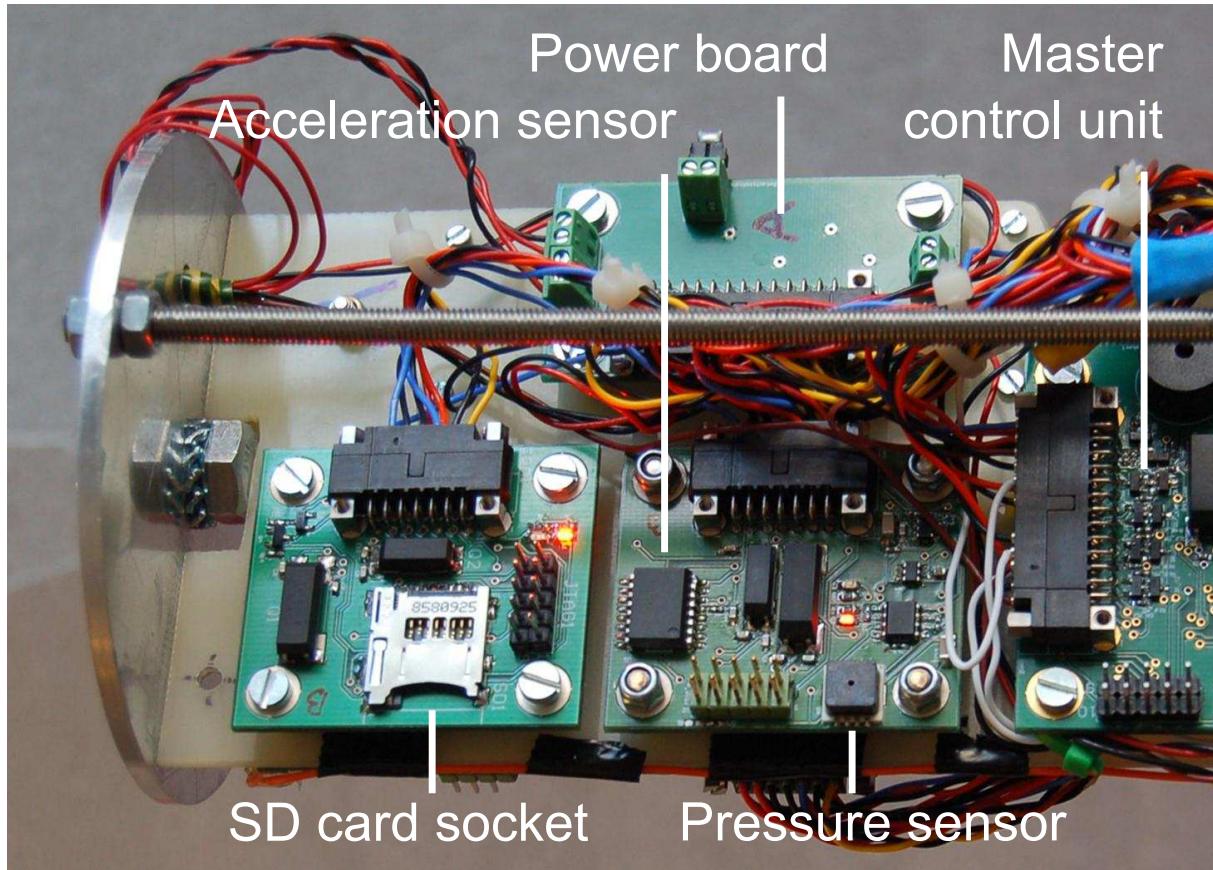


Figure 1.2: Part of the Stratos I flight computer, with in the center a measurement board with pressure and accelerometers. Source: Frank Engelen, DARE

altitude of 20 kilometers.

Coasting until apogee The second unpowered part of the flight will be until reaching apogee, somewhere between 50 km and 100 km altitude.

Free falling descent After reaching apogee the rocket will continue a free fall until approximately 8 km altitude.

Separated descent When the rocket has crossed the 8 km line, the nose cone will separate from the second stage and continue in free fall. The separation instance will be based on the altitude data provided by the navigation system.

Parachuted descent At 3 km altitude the parachute will deploy, which will initially lead to very high decelerations. Previous experience has shown shocks up to 40 g in magnitude. After parachute deployment, the rocket will continue descending at subsonic velocities until reaching ground.

Post-impact phase After impact the rocket will stay at the impact location until recovered.

The exact flight profile will be determined by the flight mechanics. For an ascent and re-entry systems the mechanics include the influence of gravity, drag, thrust and other environmental elements on the position, velocity, attitude and angular velocity of the vehicle.

With a description of the flight mechanics as determined pre-flight, the rocket will only know its position and attitude accurately if the flight does not deviate from its predetermined profile. This is of course an unrealistic assumption, which leads to the presence of sensors which may provide some information on the current state of the rocket. There are many things that can be

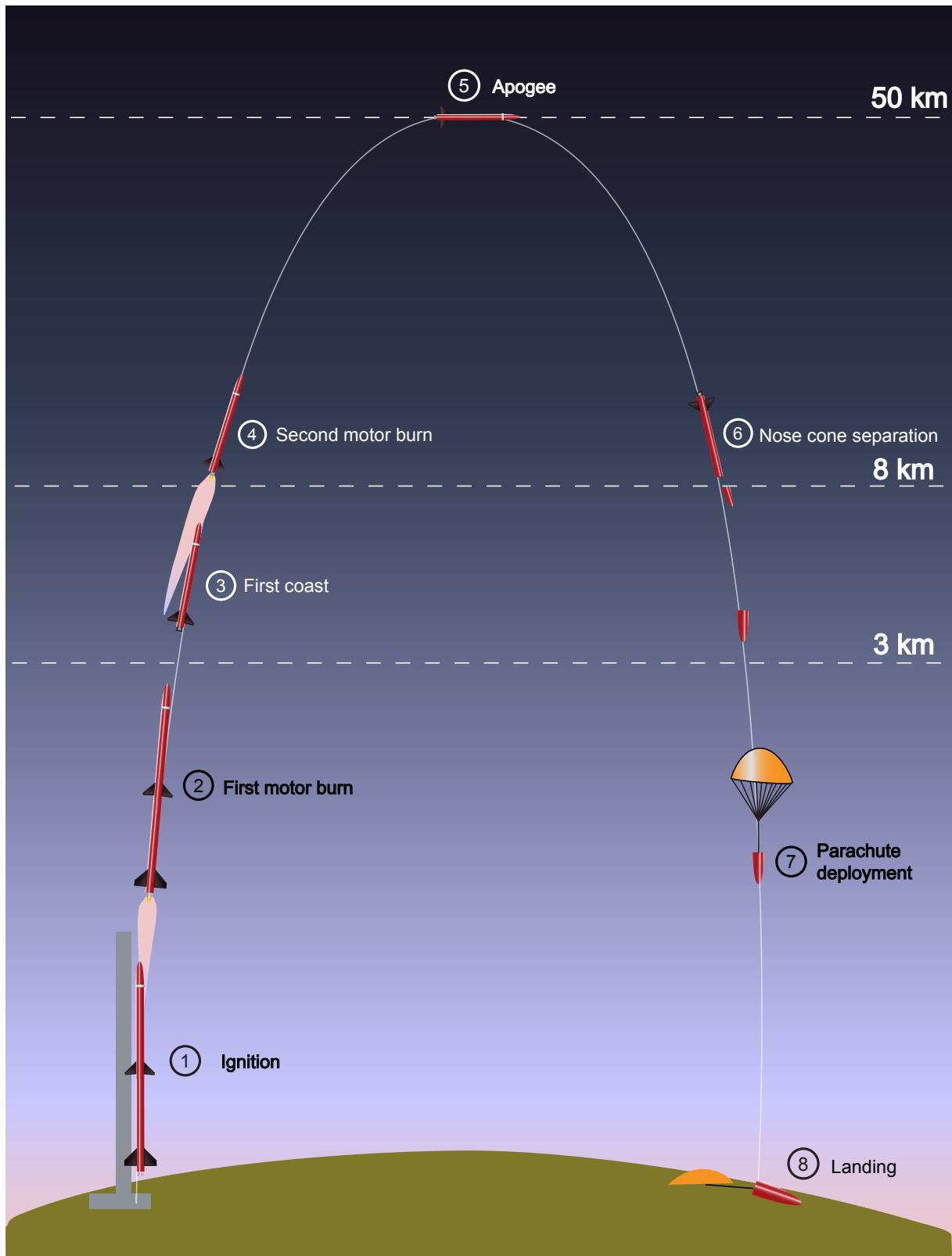


Figure 1.3: Preliminary flight profile of Stratos II. Source: Engelen (2012)

sensed, of which some, such as processed output from a GPS receiver, may be of direct interest, while others such as the raw range data of GPS and output of a pressure sensor are only of interest when it is known how to link this quantity with the quantities of interest. In the case of pressure an atmosphere model is required to link pressure to altitude.

Linking the measurements during flight and the flight mechanics determined beforehand to estimate the state of the rocket, can be done by filtering techniques such as the aforementioned Kalman filter.

1.4 Requirements on the navigation system

The most important requirements on the navigation system are imposed by safety regulations. NASA Wallops Flight Facility and Esrange Space Center impose the following safety requirement: *At least one data system is required to provide real time positional/IIP data during launch.* Furthermore if the rocket contains a *flight termination system*, there are several data sources that are allowed to base a decision about flight termination on. These are a video camera images, telemetry, radar and GPS. Data from an inertial measurement system may be used once it has been verified for at least 30 seconds by other systems (Viertotak, 2010).

In addition to this, as described in section 1.3, several events are triggered on navigation data. For second-stage ignition the navigation system should be able to determine altitude and attitude. For the separation and parachuting events the altitude should be known. Section 1.3 also describes the dynamic properties of the flight which the navigation system should be able to deal with. The triggering of events may impose performance requirements on minimum accuracy or minimum update rate, however exact numbers are not yet known.

1.5 Thesis research question and methodology

The development of the position and attitude determination system of the Stratos II is the core subject of this thesis. This leads to the main research question:

How can the position and attitude of the Stratos II sounding rocket be determined on-board and in real-time?

To be able to answer this question, the main question has been subdivided into the following subquestions:

1. How can the flight of a rocket be simulated beforehand?
2. How can in-flight data be obtained during the rocket flight?
3. How can a real-time state estimate of the rocket be made using the data obtained in flight?
4. How can post-flight analysis be performed using data obtained in flight?
5. How can the performance of the real-time state estimator and the acquisition of in-flight data be simulated beforehand?
6. How do the results of pre-flight simulation, in-flight state estimation and post-flight analysis compare?

The methodology to answering the research question follows directly from these subquestions. In essence, the research can be divided into three main topics: *simulation*, *measurement* and *estimation*. Flight estimation is the main goal of the research, while simulation and measurements are prerequisites to estimation.

Before estimation of the flight can be performed, measurements of the flight have to be available. Therefore, a measurement system is developed. This measurement system will be able

to obtain information about the flight, that the estimation process can use. The development of the measurement system contains the entire process: from design of the electronics system that performs the measurements, manufacturing of this system, software development, and calibration of the sensors. Eventually, the measurement system will be flown on a small rocket in order to test the functionality.

Simulation of the flight is important for multiple purposes. First, for the development of the measurement system it is important to know the characteristics of the parameters to be measured. Secondly, in order to study the performance of the estimator, "pseudo-measurements" have to be generated for various flight and sensor variations.

With measurements and simulations available, the real-time, in-flight estimator can be developed. The development of the estimator contains determination of the model, selection of a specific estimator, and finally study various combinations of sensors and flights. Also, with flight measurements available from the test flight, post-flight analysis can be performed.

1.6 Thesis outline

From the order in which these three topics are dependent on each other, the outline of this thesis follows. After this introduction, a chapter about simulation follows: chapter 2 covers the basic mechanics as applied to actual rocket flight. After this chapter, the background of estimation is treated. Chapter 3 elaborates on the mathematical theory behind estimation, whereas chapter 4 treats the flight and measurement models required for estimation. Following these topics, in chapter 5, the development of the preliminary measurement system is covered. This chapter is followed by chapter 6 about verification and validation of the simulation, measurement system and estimators. Then, two chapters with results follow. Chapter 7 extensively discusses flight estimator, the performance of it, and the choices that can be made. Chapter 8 discusses the test flight of the measurement system, and uses the measurements obtained during this flight to obtain an estimate of the test flight trajectory. Finally, chapter 9 finishes this thesis with conclusions and the answer on the thesis research question, followed by some concluding recommendations.

Chapter 2

Flight mechanics and trajectory simulation

Trajectory simulation is an essential prerequisite to the development of a measurement system and flight estimator. The flight mechanics describe the trajectory in the form of equations of motion. This description is done in a certain reference system, and in the form of an attitude representation. Thrust, aerodynamic and gravity forces and moments apply to the rocket, and form part of the equations of motion. Also part of equation of motion, and prerequisites to the calculation of forces and moments are properties such as mass and moment of inertia of the rockets. These are the topics covered in this chapter.

This chapter starts with the basic definitions of reference systems in section 2.1 and attitude parametrization in section 2.2. Section 2.3 then follows with the equations of motion that describe the trajectory, followed by the forces and moments that apply to a rocket in section 2.4. Then in section 2.5, the two rockets for which simulation is performed are described. Finally two simulators, the ROSIE simulator and the simplified simulator, which implement the concepts and models introduced earlier in the chapter are presented in section 2.6 and 2.7 respectively.

2.1 Reference systems

Position and attitude are relative quantities. They need to be defined with respect to a reference in order to make any sense. A reference system defines an origin and a reference orientation. A reference frame is a specific realization of a reference system (Montenbruck and Gill, 2000). In this section, the reference frames that will be used in this thesis, and the transformations between them are treated.

2.1.1 List of reference frames

There are several types of reference frames, such as celestial, Earth-centered, local and body-fixed reference frames. For the purpose of a sounding rocket, a celestial reference frame is not of interest, and thus not covered. The Earth-centered reference frames will be used to propagate the state by the simulator and the estimator. Certain quantities are expressed in local and body-fixed reference frames, and therefore these reference frames are covered.

Earth-Centered Inertial

An Earth-Centered Inertial (ECI) reference frame (F_I) has its origin in the center of the Earth. The x-axis is pointing towards the vernal equinox and the z-axis is aligned with the Earth rotation axis and pointing up northwards. The y-axis complements the right handed system (Mulder et al., 2007).

Due to lunisolar precession the vernal equinox is not fixed with respect to the ecliptic. Also reference concepts such as equator and rotation axis are not fixed in time. Therefore, a mean vernal equinox, ecliptic and equator are used for the definition of reference systems. These mean quantities are defined as those on the beginning of the year 2000 and called the *Earth Mean Equator and Equinox of J2000* (EME2000) (Montenbruck and Gill, 2000).

Earth-Centered Earth-Fixed

An Earth-Centered Earth-Fixed (ECEF) reference frame (F_C) has its origin in the center of the Earth and the z-axis pointing up to the pole, similar to the ECI frame. Its x-axis however points towards a reference meridian. The y-axis complements the right handed system (Mulder et al., 2007). The ECEF reference frame is fixed with respect to the Earth surface, but rotating with respect to inertial space. The rotation rate is one revolution per sidereal day. The rotation is in the positive direction around the z_I -axis.

A definition for an ECEF reference system is the International Terrestrial Reference Systems (ITRS). The ITRS defines an *IERS Reference Pole* and an *IERS Reference Meridian*. New versions are published annually. A specific version, the World Geodetic System 84 (WGS84) is used by GPS.

Body-Fixed

A body-fixed reference frame (F_B) is fixed to the vehicle and has its origin at a reference point on the vehicle. The choice of axes can be arbitrary, but the system is right handed (Mulder et al., 2007). The x-axis is chosen to be on the symmetry line and positive in the direction of the nose. This is shown in figure 2.1. The choice of the other axes is deferred to section 2.1.1.

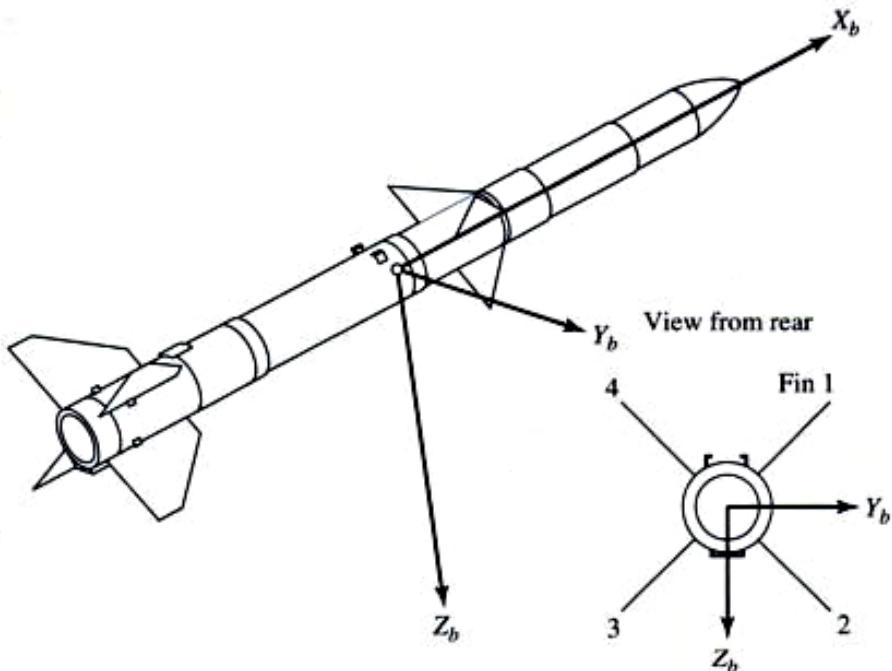


Figure 2.1: Body-fixed reference frame for a rocket. Source: Siouris (2004)

Vehicle Vertical

A vehicle vertical reference frame (F_V) has its origin at a reference point on the vehicle, similar to the body fixed reference frame. The z-axis points along the gravity vector and the x-axis points northwards. The y-axis complements the right handed reference frame (Mooij, 1997).

Aerodynamic

The aerodynamic reference frame (F_A) has its origin in the center of mass of the vehicle and has its x-axis aligned with the aerodynamic velocity. Due to the presence of wind, this is not necessarily the same as the ground velocity (Mulder et al., 2007). As with the body-fixed reference frame, the other axes are defined in section 2.1.1. The aerodynamics reference frame is important, as this is the frame where the aerodynamic forces are defined in.

Definition of the y- and z-axes for the Body-Fixed and Aerodynamic reference frames

For an aircraft, typically the center of mass is chosen as origin of the body-fixed and aerodynamic reference frames and the x-axis lying in the symmetry plane. A rocket, however, is subject to large shifts in center of mass during its flight and does not have a single or uniquely defined symmetry plane.

Because a rocket has multiple symmetry planes, the definition of the y- and z- axis needs to be made based on other body references. A suitable reference point is the launch lug. This is the attachment of the rocket to the launch tower. This attachment will be in one of the symmetry planes. The z-axis is chosen to lie in the symmetry plane that goes through the centerline and the launch lug. The y-axis complements the right handed reference frame. This is shown in figure 2.2.

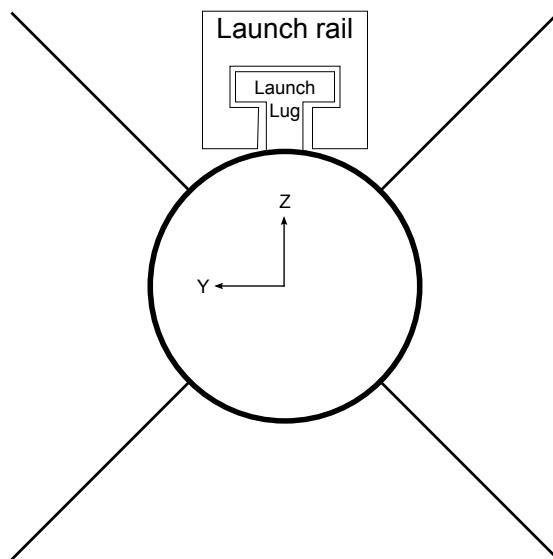


Figure 2.2: Bottom view of a rocket attached to a launch rail, showing the axes definition.

Before launch, the launch tower is set to a certain tower elevation. In the body-fixed reference frame, this is a rotation around the y-axis and the tower elevation is then equal to the pitch angle.

2.1.2 Transformations between reference frames

In general, two reference frames can be spatially separated and rotated with respect to each other. The vector from the origin of frame A to the origin of frame B is \mathbf{r}_{AB} . The rotation of frame B with respect to frame A is given by rotation matrix \mathbf{C}_{AB} . Then if the position of a point in frame B is given by vector \mathbf{r}_B , the equivalent vector in frame A is given by:

$$\mathbf{r}_A = \mathbf{r}_{AB} + \mathbf{C}_{AB}\mathbf{r}_B \quad (2.1)$$

For reference frames with coinciding origins, \mathbf{r}_{AB} is zero, yielding a single matrix multiplication:

$$\mathbf{r}_A = \mathbf{C}_{AB}\mathbf{r}_B \quad (2.2)$$

Transformations between reference frames are always expressed as a sequence of rotations around different axes, except for the transformations to and from the body-fixed frame. The transformations to and from the body-fixed frame are the subject of section 2.2. The transformation matrices of a rotation with angle θ about the x-, y- and z-axes are respectively:

$$\mathbf{C}_1(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix} \quad (2.3)$$

$$\mathbf{C}_2(\theta) = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \quad (2.4)$$

$$\mathbf{C}_3(\theta) = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

The following sections treat transformations between specific frames using these transformation matrices.

Transformation from F_C to F_I

The ECI and ECEF reference frames have coinciding origins, and thus only rotation is applicable. Furthermore, the z-axes of both reference frames coincide. Both reference frames can be found in figure 2.3. Let t be the time such that at $t = 0$ the two reference frames fully coincide and ω_E the Earth rotational rate (one revolution per sidereal day). The transformation is then a pure rotation around the z-axis by angle $-\omega_E t$. Then the transformation matrix \mathbf{C}_{IC} is given as (Mooij, 1997):

$$\mathbf{C}_{IC} = \mathbf{C}_3(-\omega_E t) = \begin{bmatrix} \cos \omega_E t & -\sin \omega_E t & 0 \\ \sin \omega_E t & \cos \omega_E t & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

Transformation from F_C to F_V

The conversion between the ECEF and Vehicle Vertical reference frame can be described by the latitude and longitude of the vehicle. Let τ be the longitude and δ the latitude of the vehicle, as shown in figure 2.3. The transformation is first a rotation around the y-axis for the colatitude

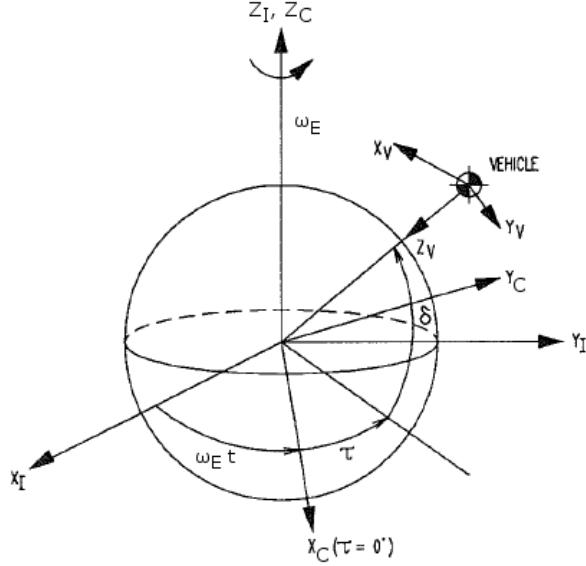


Figure 2.3: Earth Inertial, Earth Centered and Vehicle Vertical reference frames. Source: Mooij (1997)

and a rotation around the z-axis for the longitude. Then the matrix C_{CV} is given as (Mooij, 1997):

$$\mathbf{C}_{CV} = \mathbf{C}_3(-\tau)\mathbf{C}_2(\frac{\pi}{2} + \delta) = \begin{bmatrix} -\cos \tau \sin \delta & -\sin \tau & -\cos \tau \cos \delta \\ -\sin \tau \sin \delta & \cos \tau & -\sin \tau \cos \delta \\ \cos \delta & 0 & -\sin \delta \end{bmatrix} \quad (2.7)$$

Transformation from F_B to F_A

The transformation of the body-fixed frame to the aerodynamic frame is important, because it introduces two angles which often are used as dependent angles for the calculation of the aerodynamic forces. The transformation from F_B to F_A is obtained by a rotation sequence over two angles: first a rotation over the y-axis with the negative angle of attack $-\alpha$ and then a rotation over the z-axis over the side slip angle β , as shown in figure 2.12. This gives the following transformation matrix (Mulder et al., 2007):

$$\mathbf{C}_{AB} = \mathbf{C}_3(\beta)\mathbf{C}_2(-\alpha) = \begin{bmatrix} \cos \beta \cos \alpha & \sin \beta & \cos \beta \sin \alpha \\ -\sin \beta \cos \alpha & \cos \beta & -\sin \beta \sin \alpha \\ -\sin \alpha & 0 & \cos \alpha \end{bmatrix} \quad (2.8)$$

Figure 2.12 shows the aerodynamic and body fixed reference frame and the angles between them.

Let \mathbf{V}_A be the aerodynamic velocity expressed in the body frame. Then respectively the angle of attack and the side slip angle can be found:

$$\tan \alpha = \frac{V_{A_z}}{V_{A_x}} \quad (2.9)$$

$$\tan \beta = \frac{V_{A_y}}{\sqrt{V_{A_z}^2 + V_{A_x}^2}} \quad (2.10)$$

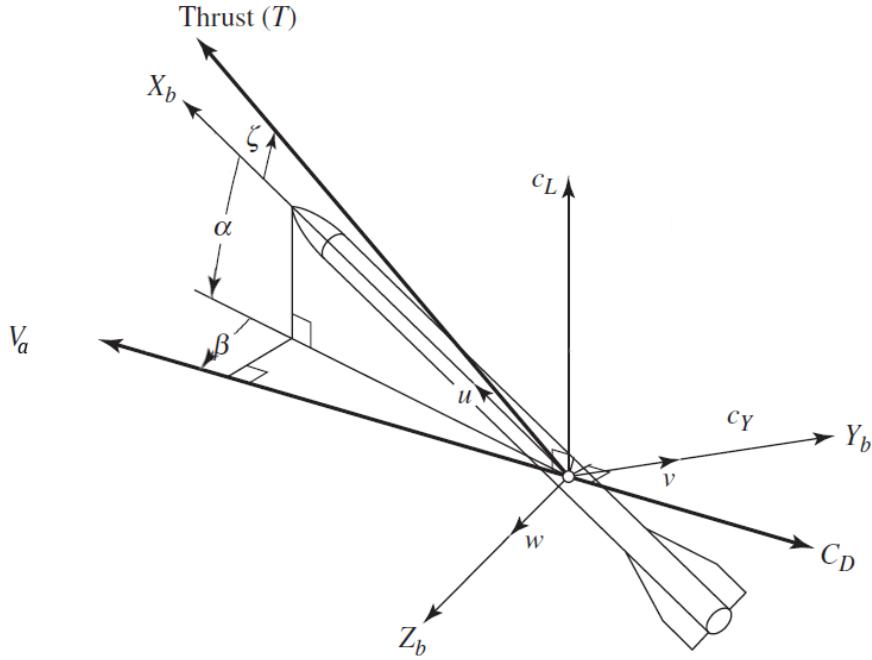


Figure 2.4: Angles between the aerodynamic and body-fixed reference frame. Source: Siouris (2004)

2.1.3 Coordinates in WGS84

The specific reference system used for the ECEF frame is the World Geodetic System 84 (WGS84). WGS84 defines a reference ellipsoid, which can be seen as an approximation of the geoid or the mean sea level of the Earth. Coordinates in WGS84 are often expressed in the spherical coordinates longitude τ , latitude δ and altitude above the geoid h . Conversion to Cartesian coordinates is given by the following equations (Farrell, 2008):

$$x = (R_N + h) \cos \delta \cos \tau \quad (2.11)$$

$$y = (R_N + h) \cos \delta \sin \tau \quad (2.12)$$

$$z = (R_N(1 - e^2) + h) \sin \delta \quad (2.13)$$

Here R_N is the normal radius, e the eccentricity and a the equatorial radius. They are given by:

$$R_N = \frac{a}{\sqrt{1 - e^2 \sin^2 \delta}} \quad (2.14)$$

$$e = 0.08181919 \quad (2.15)$$

$$a = 6378137 \text{ m} \quad (2.16)$$

2.2 Attitude representations

Attitude represents the rotation of a body with respect to a predefined reference frame. This is essentially the transformation between a defined reference frame F_A and a body-fixed reference frame F_B . There are several ways to parametrize this transformation. In this thesis, *quaternions* and *modified Rodrigues parameters* (MRP) are used. The advantage of either attitude parametrization, is that they both can express any possible attitude without encountering singularities, as opposed to other parametrizations such as Euler angles. This section covers how

these parametrizations can be used to transform vectors between reference frames, and how they change in time, subject to rotational motion. This rotational motion is expressed as the instantaneous angular velocity ω , expressed in reference frame F_B (Chu, 2010).

2.2.1 Euler axis/angle

The Euler rotation theorem states that any rotation in space can be obtained by a single rotation around a fixed axis. This axis is called the Euler (eigen) axis and this theorem can be used to define attitude. The Euler axis/angle are fundamental in the derivation of the quaternion and MRP representation, and is therefore covered here. Let the Euler axis be e . The components of e in A are e_1 , e_2 and e_3 :

$$\mathbf{e} = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} \quad (2.17)$$

Furthermore a rotation around e is defined as θ (Chu, 2010). Figure 2.5 shows this definition. These four parameters can be used to represent attitude. As there are only three independent parameters required to represent attitude, there is a constraint equation. This equation follows from the fact that e is a unit vector:

$$e_1^2 + e_2^2 + e_3^2 = 1 \quad (2.18)$$

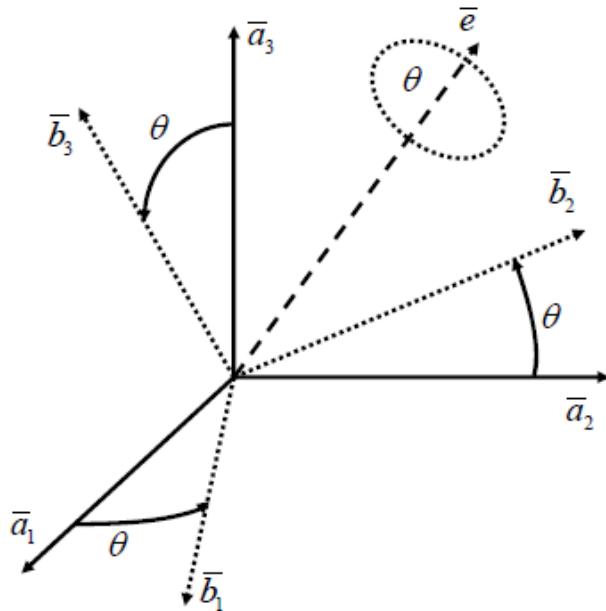


Figure 2.5: Euler axis in A. Source: Chu (2010)

2.2.2 Quaternions

Quaternions are a set of parameters that can be derived from the Euler rotation theorem. They are defined from the Euler axis/angle as follows (Chu, 2010):

$$q_1 = e_1 \sin \frac{\theta}{2} \quad (2.19)$$

$$q_2 = e_2 \sin \frac{\theta}{2} \quad (2.20)$$

$$q_3 = e_3 \sin \frac{\theta}{2} \quad (2.21)$$

$$q_4 = \cos \frac{\theta}{2} \quad (2.22)$$

The first three quaternion parameters are often called the vector part of the quaternion, while the fourth is called the scalar part. In literature, another common notation is to designate the scalar part as q_0 . This thesis will however always designate the scalar part as q_4 . The vector part of the quaternion will be defined as \boldsymbol{q} , while the full quaternion is \boldsymbol{q} :

$$\boldsymbol{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (2.23)$$

$$\boldsymbol{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \quad (2.24)$$

Quaternions possess more parameters than degrees of freedom. Therefore, the following constraint equation applies, which can be derived from the fact that the vector \boldsymbol{e} is a unit vector:

$$q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1 \quad (2.25)$$

The quaternion can be converted to a direction cosine matrix. This is the DCM that transforms a vector from frame A to frame B, and is given by:

$$\mathbf{C} = \begin{bmatrix} 1 - 2(q_2^2 + q_3^2) & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_2q_1 - q_3q_4) & 1 - 2(q_1^2 + q_3^2) & 2(q_2q_3 + q_1q_4) \\ 2(q_3q_1 + q_2q_4) & 2(q_3q_2 - q_1q_4) & 1 - 2(q_1^2 + q_2^2) \end{bmatrix} \quad (2.26)$$

If the vector part of the quaternion is zero, and the scalar part is either 1 or -1 , then the transformation of the quaternion is an identity transform, and the quaternion an identity quaternion. This identity quaternion is designated \boldsymbol{q}^I . The corresponding direction cosine matrix then becomes an identity matrix.

There always exist at least two quaternions that are numerically different, but represent the same rotation. For a given quaternion, the other quaternion that represents the same rotation, is called the shadow quaternion \boldsymbol{q}^S (Schaub and Junkins, 1996). The shadow quaternion is the negation of the quaternion:

$$\boldsymbol{q}^S = -\boldsymbol{q} = \begin{bmatrix} -q_1 \\ -q_2 \\ -q_3 \\ -q_4 \end{bmatrix} \quad (2.27)$$

An equation exists to get the quaternion \mathbf{q} that represents two consecutive rotations \mathbf{q}' and \mathbf{q}'' . This operation is also called quaternion multiplication, and given by:

$$\mathbf{q} = \mathbf{q}' \otimes \mathbf{q}'' = \begin{bmatrix} q'_4 & -q'_3 & q'_2 & q'_1 \\ q'_3 & q'_4 & -q'_1 & q'_2 \\ -q'_2 & q'_1 & q'_4 & q'_3 \\ -q'_1 & -q'_2 & -q'_3 & q'_4 \end{bmatrix} \begin{bmatrix} q''_1 \\ q''_2 \\ q''_3 \\ q''_4 \end{bmatrix} \quad (2.28)$$

From the multiplication, also the inverse can be defined. The inverse is defined such that multiplication with the original quaternion yields the identity quaternion, i.e., $\mathbf{q}^{-1} \otimes \mathbf{q} = \mathbf{q}^I$. This quaternion is obtained by either negating the scalar or the vector part:

$$\mathbf{q}^{-1} = \begin{bmatrix} -q_1 \\ -q_2 \\ -q_3 \\ q_4 \end{bmatrix} \cong \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ -q_4 \end{bmatrix} \quad (2.29)$$

The time derivative of a quaternion that represents the transformation from frame A to B, with rotational rate vector $\boldsymbol{\omega}$ defined in B, can be found by differentiating equation (2.26) and substituting the time derivative of the direction cosine matrix. Solving these equations give two equivalent equations in a matrix-vector multiplication form: firstly such that a quaternion matrix is multiplied with the rotational rate vector, secondly such that the skew-symmetric matrix of the rotational rate vector is multiplied with the quaternion vector. Neither form is preferred above the other. This is given by the following equation (Chu, 2010):

$$\begin{aligned} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} &= \frac{1}{2} \begin{bmatrix} q_4 & -q_3 & q_2 & q_1 \\ q_3 & q_4 & -q_1 & q_2 \\ -q_2 & q_1 & q_4 & q_3 \\ -q_1 & -q_2 & -q_3 & q_4 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ 0 \end{bmatrix} \\ &= \frac{1}{2} \begin{bmatrix} 0 & \omega_3 & -\omega_2 & \omega_1 \\ -\omega_3 & 0 & \omega_1 & \omega_2 \\ \omega_2 & -\omega_1 & 0 & \omega_3 \\ -\omega_1 & -\omega_2 & -\omega_3 & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \end{aligned} \quad (2.30)$$

Alternatively, this can be written in vector form:

$$\begin{aligned} \dot{\boldsymbol{\varrho}} &= \frac{1}{2} \mathbf{q} \times \boldsymbol{\omega} + \frac{1}{2} q_4 \boldsymbol{\omega} \\ \dot{q}_4 &= -\frac{1}{2} \mathbf{q}^T \boldsymbol{\omega} \end{aligned} \quad (2.31)$$

2.2.3 Modified Rodrigues parameters

The modified Rodrigues parameters (MRP) $\boldsymbol{\sigma}$ are a set of three parameters. The advantage of them is that there is no norm constraint, which is very convenient for estimation purposes. The MRP are derived from the Euler axis/angle (Schaub and Junkins, 1996):

$$\boldsymbol{\sigma} = \mathbf{e} \tan \frac{\theta}{4} \quad (2.32)$$

The transformations between MRP and quaternions can be derived from quaternions using the standard double angle equations for trigonometric functions (Schaub and Junkins, 1996):

$$\boldsymbol{\sigma} = \frac{\boldsymbol{\varrho}}{1 + q_4} \quad (2.33)$$

$$q_4 = \frac{1 - \|\boldsymbol{\sigma}\|^2}{1 + \|\boldsymbol{\sigma}\|^2} \quad (2.34)$$

$$\boldsymbol{\varrho} = \boldsymbol{\sigma}(1 + q_4)$$

As seen from equation (2.32), the MRPs go to infinity as the rotation angle goes to 360° . To avoid this singularity, the shadow modified Rodrigues parameters are used. The shadow MRP $\boldsymbol{\sigma}^S$ are derived by filling in the shadow quaternion \mathbf{q}^S in (2.33), giving (Schaub and Junkins, 1996):

$$\boldsymbol{\sigma}^S = \frac{\boldsymbol{\varrho}}{q_4 - 1} = -\frac{\boldsymbol{\sigma}}{\|\boldsymbol{\sigma}\|^2} \quad (2.35)$$

Numerically the shadow parameters and the normal parameters are different, even though they represent the same rotation. The singularity of the shadow parameters lies at a rotation angle of 0° . Switching between the normal MRP and shadow MRP near 0° and 360° allows avoiding all singularities. Both versions of the MRP yield the same DCM, which can be derived from equation (2.26) as:

$$\mathbf{C} = \frac{1}{Q} \mathbf{R} \quad (2.36)$$

$$P = 1 - \sigma_1^2 - \sigma_2^2 - \sigma_3^2 \quad (2.37)$$

$$Q = (1 + \sigma_1^2 + \sigma_2^2 + \sigma_3^2)^2 \quad (2.38)$$

$$\mathbf{R} = \begin{bmatrix} 4(\sigma_1^2 - \sigma_2^2 - \sigma_3^2) + P^2 & 8\sigma_1\sigma_2 + 4\sigma_3P & 8\sigma_1\sigma_3 - 4\sigma_2P \\ 8\sigma_2\sigma_1 - 4\sigma_3P & 4(-\sigma_1^2 + \sigma_2^2 - \sigma_3^2) + P^2 & 8\sigma_2\sigma_3 + 4\sigma_1P \\ 8\sigma_3\sigma_1 + 4\sigma_2P & 8\sigma_3\sigma_2 - 4\sigma_1P & 4(-\sigma_1^2 - \sigma_2^2 + \sigma_3^2) + P^2 \end{bmatrix} \quad (2.39)$$

The time derivative of the modified Rodrigues parameters can be obtained from the derivative of the quaternions. Appendix A.1 derives this from equation(2.30), resulting in:

$$\dot{\boldsymbol{\sigma}} = \frac{1}{2}\boldsymbol{\sigma}\boldsymbol{\sigma}^T\boldsymbol{\omega} + \frac{1}{4}(1 - \boldsymbol{\sigma}^2)\boldsymbol{\omega} + \frac{1}{2}\boldsymbol{\sigma} \times \boldsymbol{\omega} \quad (2.40)$$

2.2.4 Quantifying attitude differences

Often it is necessary to compare two different attitudes, e.g., to compare the results between simulation and estimation. The most straightforward way to do this is with quaternion multiplication. This means that if either attitude is in MRP, they need to be converted to quaternions using equation (2.34). Let \mathbf{q}_1 and \mathbf{q}_2 represent the attitudes to be compared. Then, the difference quaternion $\delta\mathbf{q}$ is given by (Crassidis and Markley, 1996):

$$\delta\mathbf{q} = \mathbf{q}_1 \otimes \mathbf{q}_2^{-1} \quad (2.41)$$

Then, as seen in equation (2.22), the scalar part δq_4 is representative for the attitude difference. A value for δq_4 of 1 or -1 means that the two compared attitudes are equal; the more it differs from either value, the larger difference between the two attitudes is. The angular difference $\delta\theta$ can be derived by solving equation (2.22) for θ :

$$\delta\theta = 2 \arccos \delta q_4 \quad (2.42)$$

2.2.5 Presentational representations for attitude

Presenting attitude in quaternion or MRP form provides little to no insight. For presentational purposes, Euler angles are much more suited. This is done using the Euler angles that represent the transformation from the vehicle vertical frame to the body frame: the azimuth (χ), elevation (γ) and bank angle, which represent a 3-2-1 rotation sequence. For an axial symmetric rocket the bank angle is of little interest, and therefore not used any further. Figure 2.6 shows the definition of these angles. From this figure, the physical meaning of the angles can be deduced: the azimuth is the direction of the x-body axis in the horizontal plane, while the elevation is the direction of this axis with respect to the horizontal plane.

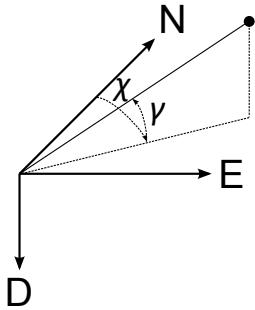


Figure 2.6: Definition of azimuth (χ) and elevation (γ) in a north-east-down frame (NED)

2.3 Equations of motion

The flight of a rocket is described by the equations of motion. The equations of motion are differential equations, which relate the position and attitude to applied forces and moments. Two types of equations of motion are distinguished: the translational, and the rotational equations of motion.

2.3.1 Translational equations of motion

Translational equations of motion deals with the relation of forces and mass with translational motion. In this section the general equations of motion are summarized for particles, rigid bodies and variable mass systems.

For a particle, the linear momentum can be defined as the product of a the mass and the velocity (Török, 2000):

$$\mathbf{p} = m\mathbf{v} \quad (2.43)$$

Then, Newton's second law states that the time derivative of the linear momentum is equal to the sum of forces acting on this particle:

$$\sum \mathbf{F} = \frac{d\mathbf{p}}{dt} \quad (2.44)$$

Alternatively, if the mass is constant, this results in the well known equation that the sum of forces is equal to the mass times the acceleration, given in an inertial reference frame:

$$\sum \mathbf{F} = m \frac{d^2\mathbf{r}}{dt^2} \quad (2.45)$$

A particle does not have dimensions. A rigid body however does have dimensions and consequently also an orientation. To apply Newton's laws to a rigid body, first the center of

mass needs to be defined. The center of mass is the point where all the mass would reside if the body with a point mass. It is given by the following integral over the body:

$$\int_m \mathbf{r}_{rel} dm = 0 \quad (2.46)$$

The integration is performed over the entire body. The vector \mathbf{r}_{rel} is the distance from a point on the body to the center of mass. From this equation the position of the center of mass can be determined.

Then, the acceleration of the center of mass, in an inertial reference frame times the mass, assuming a constant mass, is equal to the sum of forces. The position of the center of mass in an inertial frame is \mathbf{r}_C . This gives the equation of translational motion for a rigid body:

$$\sum \mathbf{F} = m \frac{d^2 \mathbf{r}_C}{dt^2} \quad (2.47)$$

The previous equations hold for a rigid body. For a variable mass systems extra terms are introduced in both equations of motion. The equation for translational motion can be derived as (Mooij, 1997):

$$\mathbf{F}_I = m \frac{d^2 \mathbf{r}_C}{dt^2} + 2\boldsymbol{\omega} \times \int_m \frac{\delta \mathbf{r}_{rel}}{\delta t} dm + \int_m \frac{\delta^2 \mathbf{r}_{rel}}{\delta t^2} dm \quad (2.48)$$

In this equation a number of terms appear, which are described here:

\mathbf{F}_I Sum of the total externally applied forces, expressed in an inertial frame in N

m Total instantaneous mass in kg

\mathbf{r}_C Instantaneous center of mass of the body in m

$\frac{d^2 \mathbf{r}_C}{dt^2}$ Acceleration of the center of mass with respect to the inertial frame in m/s²

$\boldsymbol{\omega}$ Rotation of the body with respect to the inertial frame in rad/s

$\frac{\delta \mathbf{r}_{rel}}{\delta t}$ Instantaneous velocity with respect to the body fixed frame in m/s

$\frac{\delta^2 \mathbf{r}_{rel}}{\delta t^2}$ Instantaneous acceleration with respect to the body fixed frame in m/s²

The components of equation (2.48) can be separated by introducing the following apparent forces:

$$\mathbf{F}_C = -2\boldsymbol{\omega} \times \int_m \frac{\delta \mathbf{r}_{rel}}{\delta t} dm \quad (2.49)$$

$$\mathbf{F}_{rel} = - \int_m \frac{\delta^2 \mathbf{r}_{rel}}{\delta t^2} dm \quad (2.50)$$

Here \mathbf{F}_C is the Coriolis force due to a variable mass and \mathbf{F}_{rel} the relative force due to variable mass. It should be noted that if these two forces are zero, equation (2.48) reduces to the equation of motion for a rigid body. This leads to the *principle of solidification*, by which these two internal forces can be introduced as external forces acting on a rigid body, with the mass properties discretely updated at each instance of time (Mooij, 1997). If \mathbf{F}_R is the sum of external forces, then the sum of forces, which, when substituted in equation (2.47) gives the equation of motion for a variable mass system, becomes:

$$\sum \mathbf{F} = \mathbf{F}_I + \mathbf{F}_C + \mathbf{F}_R \quad (2.51)$$

2.3.2 Rotational equations of motion

To obtain the equation of rotational motion of a rigid body, first the angular momentum needs to be defined. The angular momentum is the matrix-vector product of the inertia matrix \mathbf{I} and the rotational rate vector $\boldsymbol{\omega}$:

$$\mathbf{H} = \mathbf{I}\boldsymbol{\omega} \quad (2.52)$$

$$\begin{aligned} \mathbf{I} &= \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \\ &= \begin{bmatrix} \int_m (y^2 + z^2) dm & -\int_m xy dm & -\int_m xz dm \\ -\int_m xy dm & \int_m (x^2 + z^2) dm & -\int_m yz dm \\ -\int_m xz dm & -\int_m yz dm & \int_m (x^2 + y^2) dm \end{bmatrix} \end{aligned} \quad (2.53)$$

The inertia matrix is taken with respect to the center of mass and depends on the choice of axes. It is always possible to find an axis system where the moment of inertia matrix is a diagonal matrix. The elements on the diagonal are then called the *principal moments of inertia*, and the axes the *principal body axes*. If the body has an axis of symmetry, then this axis is a principal axis. Also, any plane of symmetry is perpendicular to a principal axis. Alternatively, the principal axes can be found by finding the eigenvectors of the inertia matrix. The eigenvectors are then the principal axes of the body (Török, 2000).

The equation of motion for rotational motion, also called Euler's dynamic equation, is then (Mooij, 1997):

$$\sum \mathbf{M} = \mathbf{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} \quad (2.54)$$

Solving for the angular acceleration gives:

$$\dot{\boldsymbol{\omega}} = \mathbf{I}^{-1} (\sum \mathbf{M} - \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega}) \quad (2.55)$$

If the moment of inertia matrix is a diagonal matrix, this reduces to:

$$\dot{p} = \frac{M_x + (I_{yy} - I_{zz})qr}{I_{xx}} \quad (2.56)$$

$$\dot{q} = \frac{M_y + (I_{zz} - I_{xx})rp}{I_{yy}} \quad (2.57)$$

$$\dot{r} = \frac{M_z + (I_{xx} - I_{yy})pq}{I_{zz}} \quad (2.58)$$

For the rotation of a variable mass body, a similar principle applies as to the translation. In this case, a Coriolis moment due to variable mass \mathbf{M}_C and a relative moment due to variable mass \mathbf{M}_{rel} should be added to the external moments. These moments are given as:

$$\mathbf{M}_C = -2 \int_m \mathbf{r}_{rel} \times \left(\boldsymbol{\omega} \times \frac{\delta \mathbf{r}_{rel}}{\delta t} \right) dm \quad (2.59)$$

$$\mathbf{M}_{rel} = - \int_m \mathbf{r}_{rel} \times \frac{\delta^2 \mathbf{r}_{rel}}{\delta t^2} dm \quad (2.60)$$

This results in the sum of moments for a variable mass system:

$$\sum \mathbf{M} = \mathbf{M}_C + \mathbf{M}_{rel} + \mathbf{M}_R \quad (2.61)$$

2.4 Forces and moments

The previous sections elaborated on how applied forces and moments affect a body. In this section, the forces that typically apply to rockets are described. Those are forces and moments due to variable mass, aerodynamics forces and moments and gravitational forces and moments.

2.4.1 Variable mass forces and moments

It was noted in section 2.3 that the variable mass of a rocket introduces two forces and two moments: the Coriolis force and moment, and the relative force and moment. These forces and moments are expanded upon in more detail in this section.

Coriolis force due to variable mass

The Coriolis force from equation (2.49) has a cross product between the angular rate $\boldsymbol{\omega}$ and the integral of $\frac{\delta \mathbf{r}_{rel}}{\delta t}$. This latter term represents a change in position of small mass particles with respect to the center of mass. For a rocket, this change is mostly along the x_B -axis in the form of the exhaust of the motor. The angular rate vector is also mostly along the x_B -axis; in normal rocket flight, pitching and yawing maneuvers are very small. The resulting cross product of the two vectors is thus small. It is shown in Ambrosius and Wittenberg (2009) that the Coriolis force is negligible compared to the thrust. Thus the Coriolis force due to variable mass is zero:

$$\mathbf{F}_C = 0 \quad (2.62)$$

Thrust forces

In a rocket the relative force is caused by the exhaust of the motor, and can be called the impulsive thrust. The relative force can be reduced to the following equation (Mooij, 1997):

$$\mathbf{F}_{rel} = -\dot{m}_e \mathbf{V}_e \quad (2.63)$$

This gives the relative or thrust force in vacuum, with \dot{m}_e the mass flow and \mathbf{V}_e the mean exhaust velocity. For a flight in non-vacuum an additional pressure thrust term needs to be added:

$$\mathbf{F}_{rel} = -\dot{m}_e \mathbf{V}_e - \int_{A_e} (p - p_a) \mathbf{n} dA \quad (2.64)$$

where A_e is the exhaust area, p the pressure in the exhaust, p_a the atmospheric pressure and \mathbf{n} the unit normal outward vector of the exhaust.

Coriolis moment due to variable mass

Equation (2.59) can be worked out resulting in the following equation:

$$\mathbf{M}_C = -\frac{\delta I}{\delta t} \cdot \boldsymbol{\omega} - \dot{m}_e \mathbf{r}_e \times (\boldsymbol{\omega} \times \mathbf{r}_e) \quad (2.65)$$

The Coriolis moment consists of two parts. The first part is the change in angular momentum in the direction of the rotational velocity vector. The second part is a damping moment due to the exhaust jet and therefore called the jet damping moment.

Thrust misalignment moment

The relative moment due to variable mass is essentially the moment due to thrust. If the thrust vector does not pass through the center of mass, a moment appears, called the thrust misalignment. This moment is called the thrust misalignment and given by:

$$\mathbf{M}_{rel} = -\dot{m}_e \mathbf{r}_e \times \mathbf{V}_e \quad (2.66)$$

In this equation \mathbf{r}_e is the vector from the center of mass to the point of application of the thrust force. This is also drawn in figure 2.7.

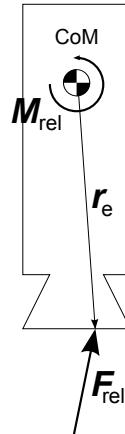


Figure 2.7: Schematic drawing of relative force and moment direction and the center of mass of a rocket

2.4.2 Aerodynamic forces

The aerodynamic forces are caused by the vehicle having a velocity relative to the air. The aerodynamic force \mathbf{F}_A is defined in the body frame. The force components are defined as a function of the dynamic pressure $\frac{1}{2}\rho V_A^2$, a reference area S_{ref} and a coefficient: C_X for the x-component, C_Y for y-component, C_Z for the z-component:

$$\mathbf{F}_A = \begin{bmatrix} C_X \\ C_Y \\ C_Z \end{bmatrix} \frac{1}{2}\rho V_A^2 S_{ref} \quad (2.67)$$

Each of the coefficients is a function of the geometry, Mach number M , Reynolds number Re , aerodynamic angle of attack α , side slip angle β :

$$C_X = f_1(M, Re, \alpha, \beta, \dots) \quad (2.68)$$

$$C_Y = f_2(M, Re, \alpha, \beta, \dots) \quad (2.69)$$

$$C_Z = f_3(M, Re, \alpha, \beta, \dots) \quad (2.70)$$

The aerodynamic velocity V_A is the relative velocity of the vehicle with respect to the air. This may differ from the ground velocity if wind is present. If \mathbf{V}_G is the ground velocity, \mathbf{V}_W the wind velocity, then the aerodynamic velocity \mathbf{V}_A is defined as:

$$\mathbf{V}_A = \mathbf{V}_G - \mathbf{V}_W \quad (2.71)$$

The aerodynamic forces are specified in the aerodynamic center, which is a defined as the point where the first and second derivatives of the moment coefficient with respect to angle of

attack are zero (Mulder et al., 2007). Because the aerodynamic center does not coincide with the center of mass, a moment is introduced. If \mathbf{r}_{CA} is the vector from the center of mass to the aerodynamic center, then the aerodynamic moment due to force is given as:

$$\mathbf{M}_{A_F} = \mathbf{r}_{CA} \times \mathbf{F}_A$$

Also aerodynamic moments that are not due to off-center definition of forces are present. These are the rolling moment, pitch moment and yawing moment, designated \mathcal{L} , \mathcal{M} and \mathcal{N} respectively. The moments are also given in the form of a coefficient, dynamic pressure and reference area, but also include a reference length l_{ref} :

$$\mathcal{L} = c_l \frac{1}{2} \rho V_A^2 S_{ref} l_{ref} \quad (2.72)$$

$$\mathcal{M} = c_m \frac{1}{2} \rho V_A^2 S_{ref} l_{ref} \quad (2.73)$$

$$\mathcal{N} = c_n \frac{1}{2} \rho V_A^2 S_{ref} l_{ref} \quad (2.74)$$

Similar to the forces, the coefficients are functions of geometry, Mach number, Reynolds number, aerodynamic angle of attack and side slip angle. The total aerodynamic moment, including the contribution of forces is given by:

$$\mathbf{M}_A = \begin{bmatrix} \mathcal{L} \\ \mathcal{M} \\ \mathcal{N} \end{bmatrix} + \mathbf{r}_{CA} \times \mathbf{F}_A \quad (2.75)$$

Atmosphere

The atmosphere plays part in the aerodynamic force with the introduction of the density ρ in the equations. There are several models for the atmosphere. Here, two models are presented: the exponential atmosphere and the international standard atmosphere (ISA).

Exponential atmosphere The exponential atmosphere is an atmosphere model that assumes the atmosphere to be isothermal at a mean temperature. It introduces a scaling height H , which is related to the mean temperature as (Ambrosius and Wittenberg, 2009):

$$H = \frac{RT}{g} \quad (2.76)$$

with $R = 287 \text{ m}^2/\text{s}^2 \text{ K}$ and $g = 9.81 \text{ m/s}^2$. The scaling height normally has a value around 7000 m, resulting in a mean temperature around 239 K and a mean speed of sound of 310 m/s. The following simple relation follows for the pressure p as a function of the geopotential altitude h and the pressure at sea level p_0 , and equivalently for the density, since constant temperature is assumed:

$$\frac{\rho}{\rho_0} = \frac{p}{p_0} = e^{-\frac{h}{H}} \quad (2.77)$$

International standard atmosphere Standard atmosphere models model the atmosphere as a set of alternating gradient layers and isothermal layers. In the gradient layers the temperature is linear with the geopotential altitude, while in the the isothermal layers the temperature is constant. This fact can be used with the hydrostatic equation to obtain an expression for pressure with altitude (Anderson, 2005). For each layer, the properties of the gas at the top of the layer form the properties of the gas at the bottom of the layer above it. The United States Standard Atmosphere 1976 is one of such standard atmosphere models with the layers given in table 2.1.

Table 2.1: United States Standard Atmosphere 1976 model. Source: Anderson (2005)

Base altitude h_0 [km]	Temperature lapse rate λ [K/m]
0	-0.0065
11	0
20	0.0010
32	0.0028
47	0
51	-0.0028
71	-0.0020
84.8520	Undefined

For a gradient layer and isothermal layer the expressions for pressure with geopotential altitude are respectively (Anderson, 2005):

$$\frac{p}{p_0} = \left(1 + \frac{\lambda(h - h_0)}{T_0}\right)^{-\left(\frac{g_0}{R\lambda} + 1\right)} \quad (2.78)$$

$$\frac{p}{p_0} = \exp\left(-\frac{g_0}{RT_0}(h - h_0)\right) \quad (2.79)$$

Then the density can be derived using the ideal gas law:

$$\rho = \frac{p}{RT} \quad (2.80)$$

where:

p is the atmospheric pressure in N/m²

p_0 is the atmospheric pressure at the bottom of the layer in N/m²

λ is the temperature gradient with geopotential altitude in K/m

h is the geopotential altitude in m

h_0 is the geopotential altitude at the bottom of the layer in m

T_0 is the standard temperature at the bottom of the layer for a gradient layer and the standard temperature in the entire layer for an isothermal layer in K

g_0 is the gravitational constant at sea level m/s²

R is the standard gas constant of air m²/s²K

2.4.3 Gravitational forces

The gravity is commonly expressed as the gradient of the geopotential U :

$$\mathbf{g} = \nabla U \quad (2.81)$$

Equation (2.81) gives the gravity vector in an ECEF reference frame. The geopotential can be expanded in spherical harmonics as an infinite sum of Legendre polynomials. In terms of the distance of the center of the Earth r , the longitude τ and latitude δ this is (Montenbruck and Gill, 2000):

$$U = \frac{\mu_e}{r} \sum_{n=0}^{\infty} \sum_{m=0}^n \frac{R_E^n}{r^n} P_{nm}(\sin \delta) (C_{nm} \cos(m\tau) + S_{nm} \sin(m\tau)) \quad (2.82)$$

where μ_e is the standard gravitational parameter of the Earth, R_E the radius of the Earth, and C_{nm} and S_{nm} are constants. The function $P_{nm}(\sin \delta)$ is the Legendre polynomial of degree nm with function argument $\sin \delta$.

For all practical purposes the geopotential is truncated after a number of terms. The first term of the expansion ($n = 0; m = 0$) yields a gravitational acceleration in the order of 10^1 m/s². The second non-zero term ($n = 2; m = 0$) yields a gravitational acceleration in the order of 10^{-2} m/s², while the third ($n = 2; m = 2$) yields 10^{-4} m/s² (Montenbruck and Gill, 2000). Therefore, only the first and the second term are considered in the simulation model. This results in the following expression for the gravitational acceleration, in the ECEF frame (Montenbruck and Gill, 2000; Wakker, 2007):

$$\mathbf{g}^C = -\frac{\mu_e}{r^3} \left(\begin{bmatrix} x \\ y \\ z \end{bmatrix} + \frac{3}{2} \frac{R_e^2}{r^2} J_2 \begin{bmatrix} x \left(1 - 5 \frac{z^2}{r^2} \right) \\ y \left(1 - 5 \frac{z^2}{r^2} \right) \\ z \left(3 - 5 \frac{z^2}{r^2} \right) \end{bmatrix} \right) \quad (2.83)$$

2.4.4 Constraint forces

Rockets that are not actively stabilized, use the fact that in case an angle of attack or side slip angle is present, an aerodynamic moment is generated that counteracts this angle of attack or angle of side slip. However, as stated in section 2.4.2, the generation of aerodynamic forces requires the vehicle to have a velocity. To stabilize the rocket during its initial flight, it is launched in a launch tower from a launch rail. The rail imposes a kinematic constraint, which is enforced by a tower force \mathbf{F}_T and tower moment \mathbf{M}_T . Figure 2.8 shows a rocket in the tower, with the forces and moments applying to it.

A tower force is generated such that the ground velocity of the rocket is in the direction of the rail. This constraint force is such that the sum of forces is in the direction of the rail. The constraint force will be always present when the tower is not exactly vertical, because it needs to counteract the component of the gravity force perpendicular to the launch rail. Furthermore, if the thrust of the rocket is not aligned with the launch rail, the tower force also needs to counteract the component of the thrust force perpendicular to the launch rail as well. The aerodynamic force can be neglected, because the velocity while the rocket is still in the tower is low, resulting in a very small aerodynamic force. Furthermore, the dominating aerodynamic force is the drag force, which is always in the direction of the velocity and thus in the direction of the launch rail.

The launch rail is almost always straight (the helical rail of the Super Loki is a notable exception), and therefore during the part of the trajectory in the launch rail the vehicle does not exhibit rotation. This kinematic constraint is enforced by the moment \mathbf{M}_T , which enforces the sum of moments to be zero. The tower moment depends on the other moments, of which thrust misalignment is most likely the only significant one. Aerodynamic moments can be neglected, similar to the aerodynamic forces. Also the Coriolis moment can be neglected, because for that to be present, the vehicle must have angular velocity. While the rocket is rotating in inertial space, because the launch tower is fixed in an Earth-fixed frame, this rotation is sufficiently small to neglect the Coriolis moment.

2.4.5 Flight phases

In general the sum of forces is given as follows:

$$\sum \mathbf{F} = \mathbf{F}_C + \mathbf{F}_{rel} + \mathbf{F}_A + \mathbf{F}_G + \mathbf{F}_T \quad (2.84)$$

The resulting sum of forces can then be substituted in equation (2.47) to obtain the equation of translational motion. The dominant force during powered flight is the relative force. During

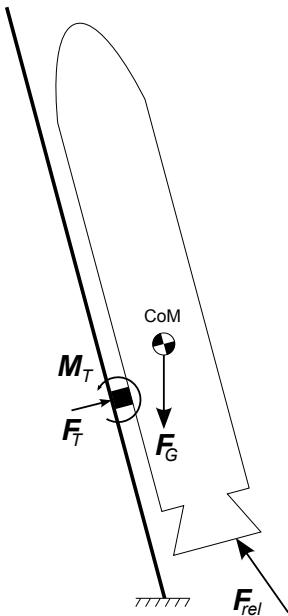


Figure 2.8: Drawing of rocket in tower with the forces applying to it

almost the entire powered flight it is several times larger than the other forces. The gravity force and aerodynamic forces dominate during the unpowered flight. Finally the tower force depends on the sum of forces excluding the tower forces. If this sum results in a vector parallel to the launch rail, the tower force will be zero. Physically, this means that the tower is vertical and that the direction of thrust is parallel to the launch rail. On the other hand, if the tower is under a large angle with respect to vertical, or if the direction of thrust is not aligned with the tower, then the tower force is large.

The sum of moments is given as follows:

$$\sum \mathbf{M} = \mathbf{M}_C + \mathbf{M}_{rel} + \mathbf{M}_A + \mathbf{r}_{CA} \times \mathbf{F}_A + \mathbf{M}_T \quad (2.85)$$

The resulting sum of moments can then be substituted in equation (2.54) to obtain the equation of rotational motion. For the moments it is much harder to make a qualitative statement about the individual forces. Thrust misalignment is likely one of the more dominant forces. For an aerodynamically stable rocket the aerodynamic moment counteracts deviations of the rocket's attitude with respect to the relative wind, making it indirectly dependent on the other moments. The Coriolis moment is only applicable if the rocket is rotating, and then only if the exhaust direction and the rotation axis are not aligned. Even though they are not exactly aligned, their mutual misalignment is small, making the Coriolis moment small as well. The tower moment most likely largely depends on the thrust misalignment and makes the sum of moments zero.

In different phases of the flight only some forces and moments are present:

- The tower constraint force and moment are only non-zero during the first part of the flight, while the rocket is still in the tower. While the rocket is in the tower, the sum of forces is in the direction of the launch tower and the sum of moments is zero.
- After leaving the tower in powered flight, all forces and moments except the tower force and moment are present: $\mathbf{F}_T = 0; \mathbf{M}_T = 0$
- When the motor has burnt out, it will keep a constant mass, therefore all forces and moments due to variable mass are zero: $\mathbf{F}_{rel} = 0; \mathbf{M}_C = \mathbf{M}_{rel} = 0$
- Finally if the rocket reaches such an altitude that no atmosphere is present anymore, the aerodynamic forces and moments become zero: $\mathbf{F}_A = 0; \mathbf{M}_A = 0$

2.5 Modeled rockets

The aforementioned equations of motion are used to simulated rocket trajectories. Two rockets are modeled for this thesis: the Stratos II rocket and the Stratos II concept launcher. In this section, vehicle properties of these rockets are presented. The Stratos II rocket is the rocket for which the position and attitude determination system is the subject of this thesis. The Stratos II concept launcher is a small rocket, developed to test subsystems for the final Stratos II rocket.

2.5.1 Stratos II

At the time of writing of this thesis, the design of the Stratos II has not yet been finalized. The model used for the Stratos II is the model used by Engelen (2012), which closely corresponds to the conceptual design of Stratos from Eiche et al. (2012). There are currently two models for the Stratos II, one variant with two solid rocket motors, and one variant with one or two hybrid rocket motors. As the technical readiness level of the solid variant is the higher, this concept will be used.

The Stratos II is a two stage rocket, propelled by two identical solid rocket motors. The rocket has a length of approximately 5.4 m, and a diameter of 0.16 m. It is passively stabilized by sets of fins. The fin configuration is shown in figure 2.9. The characteristic values of the Stratos II is given in table 2.2.

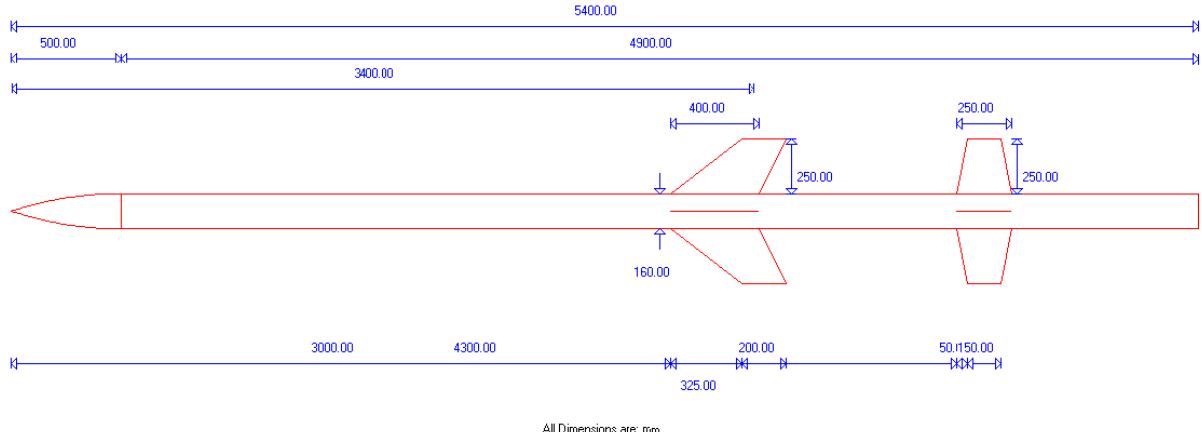


Figure 2.9: Dimensional drawing of the Stratos II. Source: Engelen (2012)

Table 2.2: Properties of the used Stratos II model

	Rocket section 1		Rocket section 2	
	Full	Empty	Full	Empty
Length	5.4 m		3.4 m	
Diameter	0.16 m		0.16 m	
Sea-level thrust	11 kN		11 kN	
Mass	112 kg	77.4 kg	63.1 kg	28.7 kg
I_{xx}	0.36 kg m ²	0.24 kg m ²	0.19 kg m ²	0.08 kg m ²
I_{yy}, I_{zz}	229.2 kg m ²	113.0 kg m ²	58.79 kg m ²	31.86 kg m ²

2.5.2 Stratos II concept launcher

In order to test subsystems of the final Stratos II rocket, several concept launchers have been developed. These rockets are relatively small, and can be launched in the Netherlands up to 1 km altitude. The concept launcher described here is the first generation concept launcher. This rocket has actually been built and has flown, allowing a much more detailed modeling than that of the final Stratos II rocket.

The rocket has a diameter of 0.20 m and a length of 2.2 m, and is shown in figure 2.10. The first generation concept launcher was based on an older Stratos II design, explaining its diameter of 0.20 m. It has a length of 2.2 m. The motor is a sorbitol/potassiumnitrate solid propellant motor developed for the Stratos I, and yields a sea-level thrust of 1.2 kN (Uitendaal, 2009). The motor is mounted in the lower section of the rocket, through the lower centering ring. The rocket is aerodynamically stabilized by three fins. The motor section and nose cone section are separable and held together during flight by the clampband, which is also shown in action in figure 2.11. Part of the nose cone is reserved for the parachute and the main electronics. On top of this is the measurement electronics section, where the measurement system of chapter 5 is located. Finally, to ensure stability of the rocket and to test the structural capabilities of the nose cone section, a heavy mass is fixed in the top.

The concept launcher was flown two times, with the second flight described in section 8.1. The reason that it only flew two flights, is that the second flight of this rocket reduced it to smaller pieces than it originally was in. This unfortunate event however, allowed a detailed piece-by-piece measuring of the rocket, allowing a very accurate mass model. The characteristic features of the rocket are tabulated in table 2.3, and these parameters are used as input for the flight simulations of this rocket.

Table 2.3: Properties of the first generation Stratos II concept launcher

	Full	Empty
Length	2.2 m	
Diameter	0.2 m	
Sea-level thrust	1.3 kN	
Mass	24.2 kg	19.8 kg
I_{xx}	0.116 kg m ²	0.112 kg m ²
I_{yy}	8.763 kg m ²	7.995 kg m ²
I_{zz}	8.764 kg m ²	7.997 kg m ²
I_{xy}	-0.027 kg m ²	-0.022 kg m ²
I_{yz}	0.000 kg m ²	0.000 kg m ²
I_{zx}	-0.045 kg m ²	0.040 kg m ²

2.6 Rocket Simulation and Impact Estimation

Rocket Simulation and Impact Estimation (ROSIE) is the main simulation tool used. ROSIE was developed by Engelen (2012) to analyze the impact dispersion of the Stratos II. It is a full six-degrees of freedom ascent simulator. For this thesis it was modified, such that measurement data could be generated. This section provides a summary of its capabilities, as fully described in Engelen, 2012.

Position and attitude The position of the rocket is defined by ROSIE in an ECI frame. The Earth is assumed to be rotating with a constant angular velocity. For the surface model

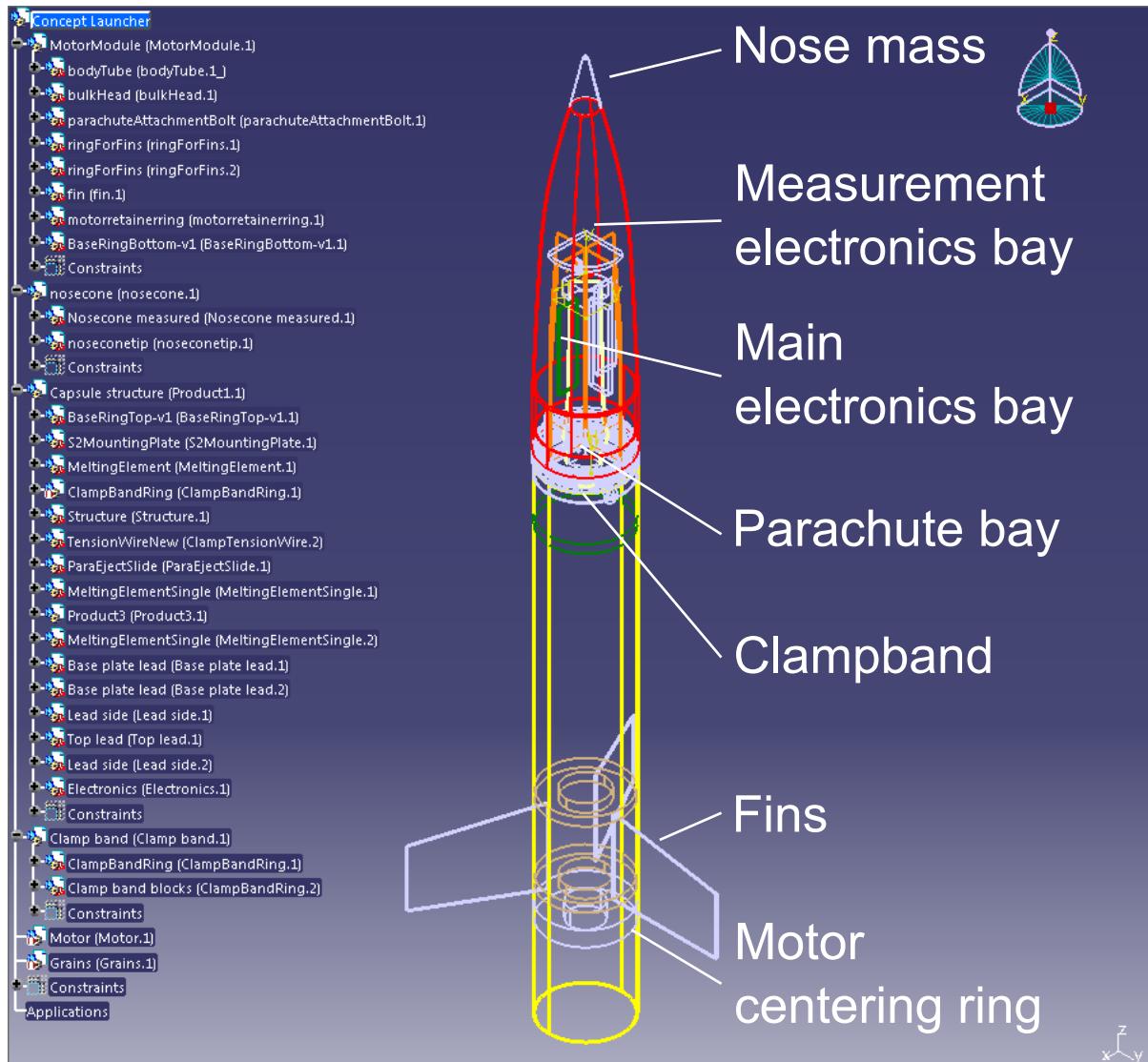


Figure 2.10: CATIA model of the concept launcher, showing the separate components for which the mass was specified

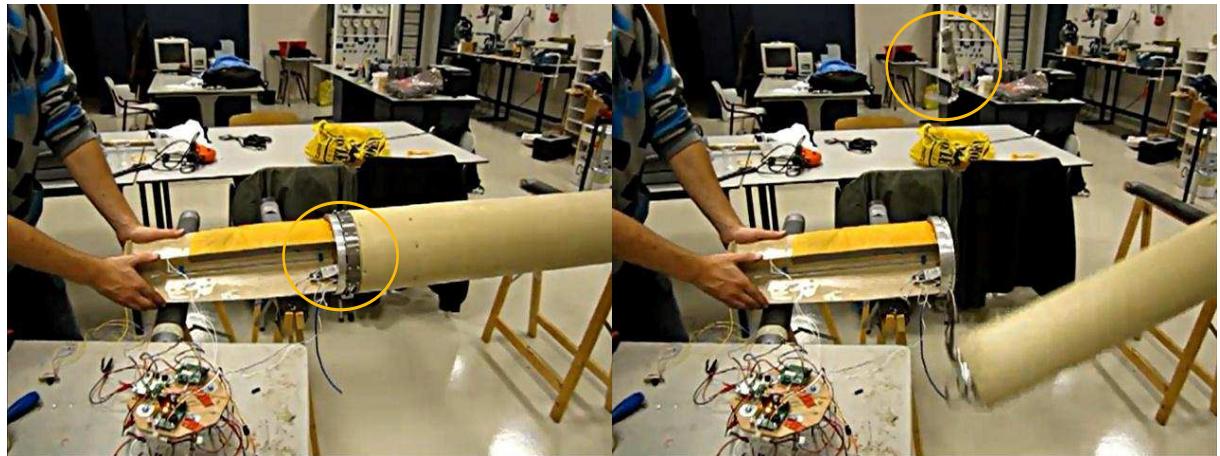


Figure 2.11: Picture sequence showing the release of the clamp band, the structural connection between the nose cone and the motor module. Source: Roel Vandeberg, DARE

the WGS84 spheroid is used. Attitude is represented by the body-to-inertial transformation quaternions.

Force and moment models The forces and moments modeled by ROSIE are the gravity force, the thrust force, the thrust misalignment moment, the jet damping moment, the aerodynamic force and the aerodynamic moment. The gravity force is modeled up to the J2-effect. The thrust force is obtained from tabulated data, and corrected for the atmospheric pressure. The thrust misalignment moment is obtained by assuming that the thrust does not necessarily act through the center of mass. The aerodynamic force and moment are generated by coefficients obtained from Missile DATCOM. The atmosphere is modeled using the US1976 model, and also wind can be read from tabulated data.

Simulation The rocket is simulated as a system of differential equations, solved by a numerical integrator. Several fixed- and variable-step size integrators are available, all from the family of Runge-Kutta integrators. For the nominal simulation of the Stratos II flight, Engelen, 2012 found the Runge-Kutta-Fehlberg-5(6) to be most suitable. However, due to the required measurement generation at small intervals, the maximum step size that the integrator can take is much smaller than required for the RKF5(6). Therefore, in order to reduce the simulation time, the integrator was switched to an RK4 integrator, which at the used step size yields equivalent integration errors as the RKF5(6) integrator at its optimal step size. ROSIE is also able to handle several discrete events, such as enabling and disabling of tower events, staging and parachuting. The simulator is wrapped in a Monte-Carlo layer, which allows varying up to 42 parameters of the simulation.

Measurement generation Originally ROSIE did not support generation of measurement data from the simulated trajectory. Therefore, the simulator was extended to allow for this. Generation of accelerometer, gyroscope, atmospheric pressure, GPS and magnetic field measurements is supported, according to the models in section 4.2. For each measurement type a bias and the amount of Gaussian noise can be specified.

2.7 Simplified simulator model

The *simplified simulator model* is a six degrees-of-freedom model to simulate rocket flight. Various simplifications with respect to a real flight and the ROSIE simulator model have been made, such that the model becomes relatively computationally inexpensive, and attractive to obtain partial derivatives of the differential equation with respect to state in an analytical fashion. As will be treated in chapter 3, these partial derivatives are required for the development of an extended Kalman filter. It is required that these partial derivatives can be obtained analytically, as it was found in the development of the estimator, that an estimator based on ROSIE, and partial derivatives obtained by numerical differentiation of ROSIE, would lead to a very sensitive and unstable estimator.

The model is given as a system of second-order differential equations of position and attitude. Furthermore, the state is augmented with some variables that are necessary to closely fit the model to reality. Position \mathbf{r} is expressed in Cartesian coordinates in an ECEF frame, while attitude is given as MRP $\boldsymbol{\sigma}$, as described in section 2.2.3, expressing the transformation from ECEF to body frame. MRPs are used rather than quaternions, since the model will be used in an estimator later, and the fact that one fewer parameter is present, has positive consequences to estimation error and estimation speed. The augmented variables are the thrust correction coefficient η_T , axial force correction coefficient η_{C_X} , the static rolling moment coefficient C_{l_0} , and the square root density ϱ , which are described in sections 2.7.3, 2.7.4, 2.7.4 and 2.7.4 respectively.

To write the system of differential equations in first order, the state variables become:

$$\mathbf{r} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad \mathbf{V} = \begin{bmatrix} V_x \\ V_y \\ V_z \end{bmatrix} \quad \boldsymbol{\sigma} = \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \end{bmatrix} \quad \boldsymbol{\omega} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (2.86)$$

$$\mathbf{x} = \begin{bmatrix} \mathbf{r} \\ \mathbf{v} \\ \boldsymbol{\sigma} \\ \boldsymbol{\omega} \\ \eta_T \\ \eta_{CX} \\ C_{l_0} \\ \varrho \end{bmatrix} \quad (2.87)$$

2.7.1 Equations of motion

For the translational equation of motion, the only forces that are considered are the gravity, thrust and aerodynamic force. Other forces, such as the variable mass forces other than thrust, are significantly smaller than those three forces. The mass is assumed to decrease linearly with time during motor burn, and constant otherwise. With \mathbf{a}_G the acceleration due to gravity, \mathbf{a}_T the acceleration due to thrust and \mathbf{a}_A the acceleration due to aerodynamics – all in the inertial frame – the translational equation of motion becomes:

$$\ddot{\mathbf{r}} = \dot{\mathbf{V}} = \mathbf{a}_G + \mathbf{a}_T + \mathbf{a}_A \quad (2.88)$$

For the rotational equations of motion, only the moments due to aerodynamics are considered. Furthermore, it is assumed that the rocket is axisymmetric, and thus that the cross terms of inertia are zero. With M_x , M_y and M_z respectively the moment components in x-, y- and z-direction in the body frame, the equations of rotational motion become:

$$\begin{aligned} \dot{p} &= \frac{M_x + (I_{yy} - I_{zz})qr}{I_{xx}} \\ \dot{q} &= \frac{M_y + (I_{zz} - I_{xx})rp}{I_{yy}} \\ \dot{r} &= \frac{M_z + (I_{xx} - I_{yy})pq}{I_{zz}} \end{aligned} \quad (2.89)$$

2.7.2 Gravity

As described in section 2.4.3, gravity can be expressed as a potential of an infinite amount of spherical harmonics. For the simplified model, terms of higher order, such as J2, are neglected, since they are several orders of magnitude smaller than the central term, and because of the complexity of their partial derivatives. This results the gravitational acceleration in an Earth centered frame as:

$$\mathbf{a}_G = -\frac{\mu_e}{\|\mathbf{r}\|^3} \mathbf{r} \quad (2.90)$$

2.7.3 Thrust

For the simplified model it is assumed that the thrust is constant, with a burn time equal to the actual burn time, and the thrust magnitude T equal to the average thrust. As this simplification

will significantly affect the simulation, a thrust correction coefficient η_T will be introduced, which the estimator can use to correct this constant thrust to the real thrust. Furthermore, it is assumed that misalignments are not present, and that the thrust acts along the x_B -axis, exactly through the center of mass. A unit vector along this axis is designated $\hat{\mathbf{i}}_B$. The transformation matrix from body to inertial frame, derived from the MRP by equation (2.36), is \mathbf{C}_{IB} . This results in the following equation for the thrust acceleration:

$$\mathbf{a}_T = \frac{\eta_T T}{m} \mathbf{C}_{IB} \hat{\mathbf{i}}_B \quad (2.91)$$

2.7.4 Aerodynamics

As previously described, aerodynamic forces and moments are caused by the vehicle having a velocity with respect to the air. To be able to specify the simplified aerodynamic force and moment models, first simplifications for atmosphere, speed of sound and aerodynamic angles are discussed.

Atmosphere

The exponential atmosphere is used as atmosphere model. The accuracy of this model depends on the chosen scale height. Literature gives 7200 m (Ambrosius and Wittenberg, 2009), but it can be shown that this leads to large deviations at higher altitudes. It is possible to closely match the exponential atmosphere with the international standard atmosphere. A solution to this would be using different scaling heights for different pressure ranges.

An alternative solution to the density and pressure mismatch, is to include the density in the state vector. When used in an estimator, it allows matching the density to the actual density. In order to use the density in the state, the time derivative of the density must be derived:

$$\begin{aligned} \dot{\rho} &= \frac{d}{dt} \left(\rho_0 e^{-\frac{h}{H}} \right) \\ &= -\frac{\rho_0}{H} e^{-\frac{h}{H}} \frac{dh}{dt} \\ &= -\frac{\rho}{H} (\mathbf{r} \cdot \mathbf{V}) \end{aligned} \quad (2.92)$$

The density is a physical quantity that can never be lower than zero. However, when estimating the density with an estimator, this constraint cannot be enforced. Therefore, instead of the density, the square root of density ϱ is used in the state vector, defined by:

$$\varrho = \sqrt{\rho} = \sqrt{\rho_0} e^{-\frac{h}{2H}} \quad (2.93)$$

The time derivative in terms of the square root density then becomes:

$$\begin{aligned} \dot{\varrho} &= \frac{d}{dt} \left(\sqrt{\rho_0} e^{-\frac{h}{2H}} \right) \\ &= -\frac{\sqrt{\rho_0}}{2H} e^{-\frac{h}{2H}} \frac{dh}{dt} \\ &= -\frac{\varrho}{2H} (\mathbf{r} \cdot \mathbf{V}) \end{aligned} \quad (2.94)$$

Speed of sound

Keeping the temperature constant, as done by the exponential atmosphere, will also mean that the speed of sound is constant, as the speed of sound a is given by the ratio of specific heats γ , the specific gas constant R and the temperature:

$$a = \sqrt{\gamma RT} \quad (2.95)$$

However, in practice the temperature of the atmosphere is not constant: it decreases linearly with a slope of approximately $dT/dh = 6.5 \times 10^{-3}$ K/m for the first 11 km. The speed of sound will not decrease linearly because the temperature appears in the square root in equation (2.95). Nevertheless, the speed of sound can be approximated by a linear relation, because it can be shown that the second- and higher-order derivatives of equation (2.95) are much smaller than the first-order derivative, because for the n -th derivative the term $(dT/dh)^n$ appear in them. The first-order derivative can be derived as:

$$\frac{\partial a}{\partial h} = \frac{1}{2} \frac{dT}{dh} \sqrt{\frac{\gamma R}{T_0 + \frac{dT}{dh} h}} \quad (2.96)$$

Then, the linearized speed of sound \tilde{a} around \tilde{h} becomes:

$$\tilde{a} = a(\tilde{h}) + \left. \frac{\partial a}{\partial h} \right|_{\tilde{h}} (h - \tilde{h}) \quad (2.97)$$

Aerodynamic angles

The magnitude of the aerodynamic forces depends greatly on the angle of attack α and angle of side slip β . These angles are given by the aerodynamic velocity \mathbf{V}_A in the body frame. Equations (2.98) and (2.99) and figure 2.12 give these angles.

$$\alpha = \tan^{-1} \frac{V_{A_z}}{V_{A_x}} \quad (2.98)$$

$$\beta = \tan^{-1} \frac{V_{A_y}}{\sqrt{V_{A_z}^2 + V_{A_x}^2}} \quad (2.99)$$

The aerodynamic velocity is normally dependent on both the ground speed and the velocity of the wind. However, assuming that no wind is present, the aerodynamic velocity reduces to the velocity in the body frame:

$$\mathbf{V}_A = \mathbf{C}_{IB}^T \mathbf{V} \quad (2.100)$$

Aerodynamic force

The aerodynamic force can be found from the aerodynamic coefficients, the dynamic pressure and a reference area. Since the aerodynamic force is defined in the body frame, to obtain the aerodynamic acceleration in the inertial frame, a reference frame transformation must be performed. Since the axial force coefficient is only known with moderate accuracy, even though it is the driving parameter for the non-powered part of the flight, its value needs to be estimated in-flight. Therefore, the axial force coefficient correction coefficient η_{C_X} is introduced. This results in the following equations:

$$\mathbf{a}_A = \frac{1}{m} \mathbf{C}_{IB} \mathbf{F}_A \quad (2.101)$$

$$\mathbf{F}_A = \begin{bmatrix} \eta_{C_X} C_X \\ C_{Y_\beta} \beta \\ C_{Z_\alpha} \alpha \end{bmatrix} \frac{1}{2} \rho V^2 S_{ref} \quad (2.102)$$

The coefficients are assumed to be only Mach-dependent; this follows from the coefficients obtained from Missile DATCOM. Since the fins of the vehicle are flat plates, and the rest of the rocket is basically a cylinder, at zero angle of attack and side slip angle, only a retarding force in

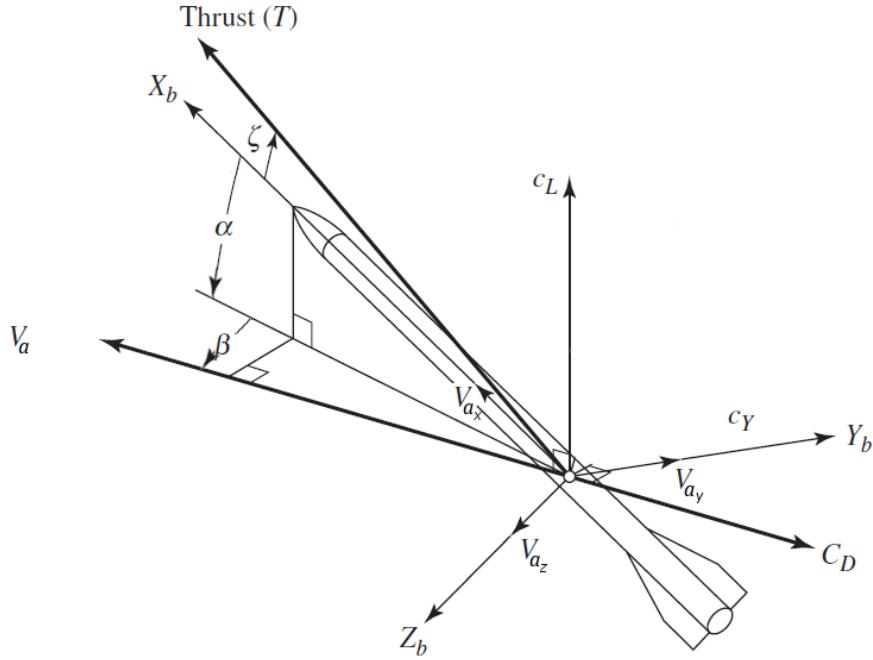


Figure 2.12: Angles between the aerodynamic and body-fixed reference frame. Source: Siouris (2004)

x-direction is present. The Missile DATCOM output shows that the C_{Y_β} and C_{Z_α} are relatively constant for small angles of attack and side slip, up to 15 degrees for either angle. Since the vehicle is aerodynamically stable around zero angle of attack and side slip, these two coefficients can be assumed to be constant.

Aerodynamic moment

The sum of moment consists of two parts: first, a moment due to aerodynamic force, and an aerodynamic moment. The moment due to aerodynamic force depends on the force from equation (2.102) and the arm of the force. It is assumed that the point of application for the force does not move in the body frame, and lies on the center line. This results in the following equation:

$$\sum \mathbf{M} = r_A \hat{\mathbf{i}}_B \times \mathbf{F}_A + \begin{bmatrix} C_{l_0} + c_{l_p} \frac{pl_{ref}}{2V} \\ c_{m_\alpha} \alpha + c_{m_q} \frac{ql_{ref}}{2V} \\ c_{n_\beta} \beta + c_{n_r} \frac{rl_{ref}}{2V} \end{bmatrix} \frac{1}{2} \rho V^2 S_{ref} l_{ref} \quad (2.103)$$

For an ideal rocket which is not designed to spin, c_{l_0} is zero. The coefficients obtained from Missile DATCOM also show this. However, in practice a rocket is not built as perfect as it is modeled, and therefore spin is present. To account for this, the static roll coefficient c_{l_0} is included in the state vector.

2.7.5 Mass properties

The mass of the rocket is assumed to decrease linearly with time during motor burn. The center of mass is assumed to lie at the centerline of the rocket, and to shift linearly with time during motor burn. It is assumed that the rocket is axially symmetric, which has as consequence that the non-diagonal terms in the inertia matrix are zero. In table 2.4, the moment of inertia is

stated for a model where the mass is considered to be equally distributed along a thickwalled cylinder, and a model obtained from measurements on the concept launcher, showing that the non-diagonal terms can be reasonably approximated by zero.

Table 2.4: Moment of inertia for different models (kg m^2)

	I_{xx}	I_{yy}	I_{zz}	I_{xy}	I_{yz}	I_{zx}
Cylinder model	0.151	9.952	9.952	0	0	0
Measured	0.116	8.763	8.764	-0.027	-0.045	0.000

2.7.6 Tower constraints

In section 2.4.4, it was explained that the tower exerts constraint forces on the rocket. For the simplified simulator, an analytical expression for the tower constraints is used. Let $\hat{\mathbf{u}}_L$ be a unit vector along the launch tower, and \mathbf{F}_C be the unconstrained sum of forces. The dot product of these two vectors is the projection of the force on the launch rail, and thus equal to the constrained force. The sum of forces and sum of moments are respectively given as:

$$\begin{aligned}\sum \mathbf{F} &= (\mathbf{F}_C \cdot \hat{\mathbf{u}}_L) \hat{\mathbf{u}}_L \\ &= \hat{\mathbf{u}}_L (\hat{\mathbf{u}}_L \cdot \mathbf{F}_C) \\ &= \hat{\mathbf{u}}_L \hat{\mathbf{u}}_L^T \mathbf{F}_C\end{aligned}\tag{2.104}$$

$$\sum \mathbf{M} = \mathbf{0}\tag{2.105}$$

2.8 Summary

In this chapter, flight mechanics and trajectory simulation were covered. First, the definitions to parametrize position and attitude were covered in sections about reference systems, coordinates and attitude representations. This was followed by the equations of motion and the forces and moments on a rocket. Then, the two rockets that will be simulated were presented, followed by the main simulator, ROSIE, and the simplified simulator.

The concepts of this chapter will be further used in the following parts of this thesis. In chapter 4, estimator models will be derived that use the covered reference systems and attitude representations. Also the simplified simulator will be used in to derive an estimator model in that chapter. Verification and validation of the simulators is part of chapter 6. ROSIE will be used in chapters 7 and 8, to obtain flight trajectories for which the estimator performance can be evaluated.

Chapter 3

Estimation theory

An important aspect in the development of a position- and attitude-determination system, is estimation theory. Estimation theory is used when a certain state needs to be estimated, and measurements, which have a known relation with the state, are available. An estimator is an algorithm that performs this estimation. Two types of estimators can be distinguished: estimators that estimate a single, non-varying state, and estimators that estimate a time-varying state. The least-squares estimator is an estimator of the first kind, whereas the Kalman filters are estimators of the second kind. This chapter treats the least-squares estimator in section 3.1 and the Kalman filters in section 3.2.

3.1 Least-squares estimation

The least-squares estimator is an estimator that aims to obtain the best estimation of the state \mathbf{x} , given measurements \mathbf{y} . It is a batch estimator, that will be used to estimate sensor parameters during sensor calibration. The least-squares estimator assumes the measurement equation to be linear and given in the following form (Simon, 2006):

$$\mathbf{y} = \mathbf{H}\mathbf{x} \quad (3.1)$$

The estimator yields the estimate of \mathbf{x} , designated $\hat{\mathbf{x}}$. With some positive definite weight matrix \mathbf{W} , the state estimate is given as:

$$\hat{\mathbf{x}} = (\mathbf{H}^T \mathbf{W} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{W} \mathbf{y} \quad (3.2)$$

The estimation residual is given by:

$$\boldsymbol{\varepsilon} = \mathbf{y} - \mathbf{H}\hat{\mathbf{x}} \quad (3.3)$$

Equation (3.2) is called the *weighted least-squares estimator* (WLSE). It can be proven that if \mathbf{W} is chosen to be a diagonal matrix with on the diagonal the reciprocal of the variance of the corresponding measurement error, then the WLSE is the *best linear unbiased estimator* (BLUE). It is the best estimator in the sense that it is the estimator that minimizes the variance of the estimation error (best) and has expectation value equal to the true value (unbiased) and this is valid for linear measurement models (Teunissen et al., 2006). Least-squares estimation requires the system to be fully observable. This is mathematically given by the condition that the matrix product $\mathbf{H}^T \mathbf{H}$ is invertible.

It is possible to rewrite equation (3.2) in a recursive form, where new measurements can be processed as they come in. First, an estimation with initial measurements \mathbf{y}_0 must be done using equation (3.2), resulting in $\hat{\mathbf{x}}_0$. The covariance of the estimation is then defined using (Simon, 2006):

$$\mathbf{P}_0 = (\mathbf{H}_0^T \mathbf{W}_0 \mathbf{H}_0)^{-1} \quad (3.4)$$

Each time a new measurement is processed, the covariance can be propagated:

$$\mathbf{P}_{k+1} = (\mathbf{P}_k^{-1} + \mathbf{H}_{k+1}^T \mathbf{W}_{k+1} \mathbf{H}_{k+1})^{-1} \quad (3.5)$$

The gain and measurement update are then given by the following two equations:

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1} \mathbf{H}_{k+1}^T \mathbf{W}_{k+1} \quad (3.6)$$

$$\hat{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_k + \mathbf{K}_{k+1} (\mathbf{y}_{k+1} - \mathbf{H}_{k+1} \hat{\mathbf{x}}_k) \quad (3.7)$$

The recursive form of the least-squares estimator is very useful when there are many measurements available, which would for a direct least-squares estimation require enormous amounts of processing power and memory to store the measurement matrix. It is not suitable as a sequential estimator because it estimates the state at a fixed instance, rather than the current state like a sequential estimator.

3.2 Kalman filtering

Kalman filters and smoothers estimate a time-varying state, based on a system model of the state and measurements. There are several filters and smoothers in the family of Kalman filters. This section treats the extended Kalman filter (EKF), the most commonly used Kalman filter, the unscented Kalman filter (UKF) and Rauch-Tung-Striebel (RTS) smoother.

A Kalman filter defines an *a-priori* state estimate $\hat{\mathbf{x}}_k^-$ and an *a-posteriori* state estimate $\hat{\mathbf{x}}_k^+$ at time t_k . From the state estimate at t_{k-1} , a Kalman filter will predict the *a-priori* estimate using the system model. Then, it will use the current measurement to update the estimate, which will become the *a-posteriori* estimate. The aim of a Kalman filter is converge to the true state. Figure 3.1 shows this prediction-update cycle.

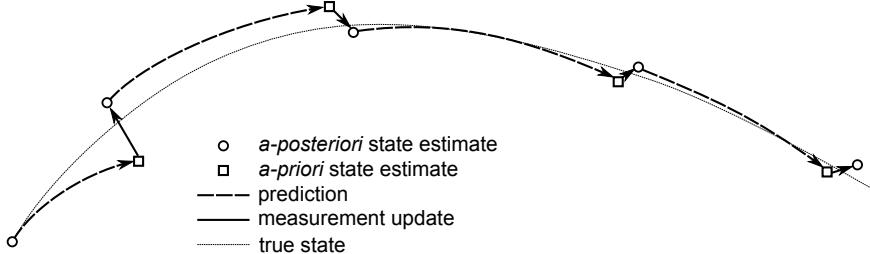


Figure 3.1: The prediction-update cycle of a Kalman filter

The time-dependence of the state can be represented in the form of a differential equation and a measurement function. In the most general form this can be written as follows (Olsder and Woude, 2005):

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}(t), \mathbf{u}(t)) \quad (3.8)$$

$$\mathbf{y}(t) = \mathbf{g}(t, \mathbf{x}(t), \mathbf{u}(t)) \quad (3.9)$$

Here \mathbf{x} the state vector, $\dot{\mathbf{x}}$ the state derivative, \mathbf{u} the control input and \mathbf{y} the output or measurement. These equations are the general form for a controlled non-linear system. If the system is linear, it can be written in the form of a matrix-vector multiplication:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) \quad (3.10)$$

$$\mathbf{y}(t) = \mathbf{C}(t)\mathbf{x}(t) + \mathbf{D}(t)\mathbf{u}(t) \quad (3.11)$$

Here \mathbf{A} is the system matrix, \mathbf{B} the control matrix, \mathbf{C} the state output matrix and \mathbf{D} the control output matrix. If these four matrices are constant in time, then the system is time invariant and called a linear time invariant system (LTI).

If \mathbf{u} is zero, then the system is uncontrolled. In this case equations (3.8) and (3.9) become functions of state and time only, and the matrices \mathbf{B} and \mathbf{D} disappear from equations (3.10) and (3.11). From now on it will be assumed that all systems are uncontrolled.

The functions \mathbf{f} and \mathbf{g} can be locally linearized to \mathbf{A} and \mathbf{C} to obtain the system in the form of equations (3.10) and (3.11). This is done by taking a Taylor series expansion around $\tilde{\mathbf{x}}$ and truncating after the first term. The entries in the system function \mathbf{f} and measurement function \mathbf{g} are defined as:

$$\mathbf{f} = \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} \quad \mathbf{g} = \begin{bmatrix} g_1 \\ \vdots \\ g_m \end{bmatrix} \quad (3.12)$$

The resulting \mathbf{A} and \mathbf{C} are given by (Olsder and Woude, 2005):

$$\mathbf{A} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{\tilde{\mathbf{x}}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{\tilde{\mathbf{x}}} \quad (3.13)$$

$$\mathbf{C} = \left. \frac{\partial \mathbf{g}}{\partial \mathbf{x}} \right|_{\tilde{\mathbf{x}}} = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \cdots & \frac{\partial g_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial g_m}{\partial x_1} & \cdots & \frac{\partial g_m}{\partial x_n} \end{bmatrix}_{\tilde{\mathbf{x}}} \quad (3.14)$$

3.2.1 Extended Kalman filter

The extended Kalman filter (EKF) is a form of the Kalman filter for non-linear systems given by equations (3.8) and (3.9). The principle of the extended Kalman filter is straightforward: at each time step the *a-priori* estimate is calculated by the state function:

$$\hat{\mathbf{x}}_k^- = \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1}^+) \quad (3.15)$$

It is assumed that the state function has a process noise, which is zero-mean, uncorrelated and has covariance matrix \mathbf{Q} . The matrix \mathbf{P}_k is the covariance matrix of the estimation error. As with the state estimate, there is both an *a-priori* and *a-posteriori* covariance. The *a-priori* error covariance is calculated with the matrix \mathbf{F}_{k-1} , which is the linearization of the state function from equation (3.8) around the *a-posteriori* estimate of the previous time step (Simon, 2006):

$$\mathbf{F}_{k-1} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_{k-1}^+} \quad (3.16)$$

$$\mathbf{P}_k^- = \mathbf{F}_{k-1} \mathbf{P}_{k-1}^+ \mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1} \quad (3.17)$$

The measurement function \mathbf{h} is linearized to obtain \mathbf{H}_k :

$$\mathbf{H}_k = \left. \frac{\partial \mathbf{h}_k}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_k^-} \quad (3.18)$$

This can be used with the error covariance matrix to determine the Kalman gain. The Kalman gain is a measure for the reliability of the measurements compared to the reliability of the system model. It is given by:

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \quad (3.19)$$

Next, the *a-priori* state estimate is updated for the measurements, using the Kalman gain:

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{y}_k - \mathbf{h}_k(\hat{\mathbf{x}}_k^-)) \quad (3.20)$$

The quantity $(\mathbf{y}_k - \mathbf{h}_k(\hat{\mathbf{x}}_k^-))$ is called the *innovation* for the filter step. Finally the covariance of the *a-posteriori* state estimate needs to be calculated. There are two alternative formulations for this, given in equations (3.21) and equations (3.22):

$$\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^T \quad (3.21)$$

$$\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- \quad (3.22)$$

Both formulations have advantages and disadvantages. The disadvantage of equation (3.21) is that it is computationally expensive, but it does guarantee the matrix to be always symmetric positive definite, making it numerically more stable. Equation (3.22) is computationally cheap but does not guarantee the matrix to be any specific form.

The extended Kalman filter uses first order approximations of the covariance matrices. This is the primary error component of the extended Kalman filter. It is possible to increase the order to two or higher, by taking higher order terms into account when calculating the covariance matrices. This will result in a more accurate and more stable filter (Simon, 2006).

U-D filtering

Problems with Kalman filtering are often caused by the propagation of the error covariance matrix (Grewal and Andrews, 2008). The error covariance is under normal circumstances a positive-definite symmetric matrix. However, due to limited numerical precision, the covariance matrix may loose these properties. A solution to this is propagating the covariance in an upper-triangular form. The U-D filter is derived from the sequential Kalman filter, a form of the Kalman filter where the measurement update is performed one entry at a time. This form requires the measurement noise matrix to be diagonal, which is the case if the measurements are independent (Simon, 2006).

Let \mathbf{H}_{ik} be the i-th row of \mathbf{H}_k and y_{ik} the i-th element of \mathbf{y}_k . An iteration of the U-D Kalman filter starts performing an \mathbf{UDU}^T -decomposition on the *a-priori* covariance calculated by equation (3.17) using the square-root-free Cholesky algorithm:

$$\mathbf{P}_k^- = \mathbf{U}_0 \mathbf{D}_0 \mathbf{U}_0^T \quad (3.23)$$

Next for each measurement $i = 1 \dots r$ the \mathbf{UDU}^T -decomposition of the covariance is updated. This is done by taking another \mathbf{UDU}^T -decomposition (Simon, 2006):

$$\bar{\mathbf{U}}_i \bar{\mathbf{D}}_i \bar{\mathbf{U}}_i^T = \mathbf{D}_{i-1} - \frac{\mathbf{D}_{i-1} \mathbf{U}_{i-1}^T \mathbf{H}_{ik}^T (\mathbf{D}_{i-1} \mathbf{U}_{i-1}^T \mathbf{H}_{ik}^T)^T}{R_{ik} + (\mathbf{U}_{i-1}^T \mathbf{H}_{ik}^T)^T \mathbf{D}_{i-1} (\mathbf{U}_{i-1}^T \mathbf{H}_{ik}^T)} \quad (3.24)$$

Finally, for this iteration, \mathbf{U} and \mathbf{D} are updated:

$$\mathbf{U}_i = \mathbf{U}_{i-1} \bar{\mathbf{U}}_i \quad (3.25)$$

$$\mathbf{D}_i = \bar{\mathbf{D}}_i \quad (3.26)$$

After performing this iteration r times, the *a-posteriori* covariance and Kalman gain can be calculated:

$$\mathbf{P}_k^+ = \mathbf{U}_r \mathbf{D}_r \mathbf{U}_r^T \quad (3.27)$$

$$\mathbf{K}_k = \mathbf{P}_k^+ \mathbf{H}^T \mathbf{R}_k^{-1} \quad (3.28)$$

Since \mathbf{R}_k is diagonal, its inverse is simply the inverse of the diagonal elements.

3.2.2 Unscented Kalman filter

In the derivation of the extended Kalman filter, the mean and variance are calculated using a linear approach. Discarding the higher order partial derivatives may however lead to large inaccuracies for highly non-linear functions. The unscented Kalman filter aims to solve this by determining a more accurate representation of the probability density functions. A number of points around the previous state estimate is chosen and propagated with the system equations. The statistics of the propagated points yield the new state estimate (Julier and Uhlmann, 1997).

Instead of immediately determining the *a-priori* state estimate, first $2n$ sigma points are determined, based on the previous *a-posteriori* covariance (Simon, 2006):

$$\hat{\mathbf{x}}_{k-1}^{(i)} = \hat{\mathbf{x}}_{k-1}^+ + \tilde{\mathbf{x}}^{(i)} \quad i = 1 \dots 2n \quad (3.29)$$

$$\tilde{\mathbf{x}}^{(i)} = \left(\sqrt{n\mathbf{P}_{k-1}^+} \right)_i^T \quad i = 1 \dots n \quad (3.30)$$

$$\tilde{\mathbf{x}}^{(n+i)} = - \left(\sqrt{n\mathbf{P}_{k-1}^+} \right)_i^T \quad i = 1 \dots n \quad (3.31)$$

The matrix $\sqrt{n\mathbf{P}}$ is a square root of matrix $n\mathbf{P}$. A computationally efficient matrix square root is the Cholesky decomposition. The subscript i in $(\sqrt{n\mathbf{P}})_i$ denotes the i -th row of $\sqrt{n\mathbf{P}}$. The sigma points are propagated to the next time instance:

$$\hat{\mathbf{x}}_k^{(i)} = \mathbf{f}(\hat{\mathbf{x}}_{k-1}^{(i)}) \quad (3.32)$$

Finally they are assembled to a single *a-priori* state estimate:

$$\hat{\mathbf{x}}_k^- = \frac{1}{2n} \sum_{i=1}^{2n} \hat{\mathbf{x}}_k^{(i)} \quad (3.33)$$

The *a-priori* covariance is given by:

$$\mathbf{P}_k^- = \frac{1}{2n} \sum_{i=1}^{2n} (\hat{\mathbf{x}}_k^{(i)} - \hat{\mathbf{x}}_k^-)(\hat{\mathbf{x}}_k^{(i)} - \hat{\mathbf{x}}_k^-)^T + \mathbf{Q}_{k-1} \quad (3.34)$$

Next, new sigma points are chosen based on the *a-priori* covariance. This step is optional, as the previously calculated sigma points can be used at the cost of filter performance.

$$\hat{\mathbf{x}}_k^{(i)} = \hat{\mathbf{x}}_k^- + \tilde{\mathbf{x}}^{(i)} \quad i = 1 \dots 2n \quad (3.35)$$

$$\tilde{\mathbf{x}}^{(i)} = \left(\sqrt{n\mathbf{P}_k^-} \right)_i^T \quad i = 1 \dots n \quad (3.36)$$

$$\tilde{\mathbf{x}}^{(n+i)} = - \left(\sqrt{n\mathbf{P}_k^-} \right)_i^T \quad i = 1 \dots n \quad (3.37)$$

Measurement estimates are generated based on the sigma points:

$$\hat{\mathbf{y}}_k^{(i)} = \mathbf{h}(\hat{\mathbf{x}}_k^{(i)}) \quad (3.38)$$

The measurement predictions are assembled to a single measurement estimate:

$$\hat{\mathbf{y}}_k = \frac{1}{2n} \sum_{i=1}^{2n} \hat{\mathbf{y}}_k^{(i)} \quad (3.39)$$

Next the covariance of the predicted measurements and the cross covariance with the *a-priori* state estimate should be established:

$$\mathbf{P}_y = \frac{1}{2n} \sum_{i=1}^{2n} (\hat{\mathbf{y}}_k^{(i)} - \hat{\mathbf{y}}_k)(\hat{\mathbf{y}}_k^{(i)} - \hat{\mathbf{y}}_k)^T + \mathbf{R}_k \quad (3.40)$$

$$\mathbf{P}_{xy} = \frac{1}{2n} \sum_{i=1}^{2n} (\hat{\mathbf{x}}_k^{(i)} - \hat{\mathbf{x}}_k^-)(\hat{\mathbf{y}}_k^{(i)} - \hat{\mathbf{y}}_k)^T + \mathbf{R}_k \quad (3.41)$$

Finally the measurement update can be performed to obtain the *a-posteriori* state estimate and covariance:

$$\mathbf{K}_k = \mathbf{P}_{xy} \mathbf{P}_y^{-1} \quad (3.42)$$

$$\hat{\mathbf{x}}_{k+1}^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k) \quad (3.43)$$

$$\mathbf{P}_{k+1}^+ = \mathbf{P}_k^- - \mathbf{K}_k \mathbf{P}_y \mathbf{K}_k^T \quad (3.44)$$

The algorithm can be fine tuned by assigning non-equal weights to the sigma points, which can be treated as additional design parameters. Furthermore, depending on the probability distribution of the system model, $\sqrt{n}\mathbf{P}$ can be replaced by $\sqrt{(n + \kappa)}\mathbf{P}$, where κ is a design parameter.

The advantage of the unscented Kalman filter is that no partial derivatives need to be calculated, which is potentially difficult. The unscented Kalman filter performs better than the extended Kalman filter, especially for highly nonlinear functions. It comes at the price of some computationally expensive operations, such as finding the square root of a matrix and multiple system model evaluations.

The mean and covariance are correct to the second order, in contrast to the extended Kalman filter, whose mean is correct to the first order. Furthermore, it can be shown that the unscented filter is able to partially incorporate information from the higher orders, leading to even greater accuracy (Julier and Uhlmann, 1997).

3.2.3 Rauch-Tung-Striebel smoother

Kalman smoothing extends the Kalman filter to also take into account future measurements. There are different formulations, but they all work by first performing a standard Kalman filter estimation using one of the Kalman filters, and then performing a smoothing pass (Simon, 2006; Simon, 2006). The most often used smoother is the Rauch-Tung-Striebel (RTS) smoother. This smoother first performs a Kalman filter pass using the EKF. This is called the forward pass. The estimated *a-posteriori* state in the forward pass is designated $\hat{\mathbf{x}}_{fk}^+$ and the covariance \mathbf{P}_{fk} . The smoother then performs another pass, which results in the smoothed estimate $\hat{\mathbf{x}}_k$. The smoother is initialized as (Simon, 2006):

$$\hat{\mathbf{x}}_N = \hat{\mathbf{x}}_{fN}^+ \quad (3.45)$$

$$\mathbf{P}_N = \mathbf{P}_{fn}^+ \quad (3.46)$$

Then backwards in time, for $k = N - 1, \dots, 0$ the RTS smoother equations are performed:

$$\mathbf{K}_k = \mathbf{P}_{fk}^+ \mathbf{F}_k^T \left(\mathbf{P}_{f,k+1}^- \right)^{-1} \quad (3.47)$$

$$\mathbf{P}_k = \mathbf{P}_{fk}^+ - \mathbf{K}_k (\mathbf{P}_{f,k+1}^- - \mathbf{P}_{k+1}) \quad (3.48)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{fk}^+ + \mathbf{K}_k (\hat{\mathbf{x}}_{k+1} - \hat{\mathbf{x}}_{f,k+1}^-) \quad (3.49)$$

The stability of the smoother will in general be determined by the stability of the backwards pass. It can be found that it is stable if the following equation is stable:

$$\hat{\mathbf{x}}_k = \mathbf{P}_{fk}^+ \mathbf{F}_k^T \mathbf{P}_{f,k+1}^- \hat{\mathbf{x}}_{k+1} \quad (3.50)$$

3.3 Outlook

In this chapter, two types of estimators have been presented: the least-squares estimator and the Kalman filter. The least-squares estimator can be used to estimate a single state using multiple measurements. Kalman filters can be used to estimate a state that varies in time using system models and measurements. The least-squares estimator primary purpose in this thesis, is the calibration of sensors, as will be described in chapter 5. There, it will be used to estimate sensor characteristics such as scale factors and bias. The Kalman filters will be used for the real-time and post-flight estimation of the rocket trajectories, in chapters 7 and 8, respectively. The system and measurement models required for the Kalman filters will be covered in the following chapter, chapter 4.

Chapter 4

Navigation models

As described in chapter 3, a Kalman filter requires a state transition function and measurement functions. In addition, the extended Kalman filter requires a state transition matrix, and a measurement matrix. The state transition function and transition matrix together form the system model, whereas the measurement function and matrix form the measurement model. In this chapter, two system models are developed, namely the simplified model, and the kinematic model. Furthermore, measurement models for an accelerometer, gyroscope, magnetometer, pressure sensor and GPS are presented. This chapter starts with the simplified estimator model in section 4.1. The kinematic model uses some measurements as system model, and is therefore deferred to section 4.3, after the measurement models have been treated in section 4.2. Finally, this chapter finishes with the flight computer model in section 4.4.

4.1 Simplified estimator model

As implied from the name, the simplified estimator model derives from the simplified simulator model. The state transition function consists of the integral of the derivative of equation (2.87). This integration is performed using the Runge-Kutta 4 numerical integrator.

The state transition matrix from t_{k-1} to t_k , \mathbf{F}_{k-1} , can be obtained from the state derivative $\dot{\mathbf{x}}$ by partial differentiation. To the first order, this matrix can be given by (Zarchan and Musoff, 2005):

$$\mathbf{F}_{k-1} = \mathbf{I} + \frac{\partial \dot{\mathbf{x}}}{\partial \mathbf{x}} (t_k - t_{k-1}) \quad (4.1)$$

The partial derivative matrix in this equation has the following form:

$$\frac{\partial \dot{\mathbf{x}}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial \mathbf{V}}{\partial \mathbf{r}} & \frac{\partial \mathbf{V}}{\partial \mathbf{V}} & \frac{\partial \mathbf{V}}{\partial \boldsymbol{\sigma}} & \frac{\partial \mathbf{V}}{\partial \boldsymbol{\omega}} & \frac{\partial \mathbf{V}}{\partial \eta_T} & \frac{\partial \mathbf{V}}{\partial c_{l_0}} & \frac{\partial \mathbf{V}}{\partial \varrho} \\ \frac{\partial \mathbf{a}}{\partial \mathbf{r}} & \frac{\partial \mathbf{a}}{\partial \mathbf{V}} & \frac{\partial \mathbf{a}}{\partial \boldsymbol{\sigma}} & \frac{\partial \mathbf{a}}{\partial \boldsymbol{\omega}} & \frac{\partial \mathbf{a}}{\partial \eta_T} & \frac{\partial \mathbf{a}}{\partial c_{l_0}} & \frac{\partial \mathbf{a}}{\partial \varrho} \\ \frac{\partial \dot{\boldsymbol{\sigma}}}{\partial \mathbf{r}} & \frac{\partial \dot{\boldsymbol{\sigma}}}{\partial \mathbf{V}} & \frac{\partial \dot{\boldsymbol{\sigma}}}{\partial \boldsymbol{\sigma}} & \frac{\partial \dot{\boldsymbol{\sigma}}}{\partial \boldsymbol{\omega}} & \frac{\partial \dot{\boldsymbol{\sigma}}}{\partial \eta_T} & \frac{\partial \dot{\boldsymbol{\sigma}}}{\partial c_{l_0}} & \frac{\partial \dot{\boldsymbol{\sigma}}}{\partial \varrho} \\ \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{r}} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \mathbf{V}} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \boldsymbol{\sigma}} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \boldsymbol{\omega}} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \eta_T} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial c_{l_0}} & \frac{\partial \dot{\boldsymbol{\omega}}}{\partial \varrho} \\ \frac{\partial \dot{\eta}_T}{\partial \mathbf{r}} & \frac{\partial \dot{\eta}_T}{\partial \mathbf{V}} & \frac{\partial \dot{\eta}_T}{\partial \boldsymbol{\sigma}} & \frac{\partial \dot{\eta}_T}{\partial \boldsymbol{\omega}} & \frac{\partial \dot{\eta}_T}{\partial \eta_T} & \frac{\partial \dot{\eta}_T}{\partial c_{l_0}} & \frac{\partial \dot{\eta}_T}{\partial \varrho} \\ \frac{\partial \dot{c}_{l_0}}{\partial \mathbf{r}} & \frac{\partial \dot{c}_{l_0}}{\partial \mathbf{V}} & \frac{\partial \dot{c}_{l_0}}{\partial \boldsymbol{\sigma}} & \frac{\partial \dot{c}_{l_0}}{\partial \boldsymbol{\omega}} & \frac{\partial \dot{c}_{l_0}}{\partial \eta_T} & \frac{\partial \dot{c}_{l_0}}{\partial c_{l_0}} & \frac{\partial \dot{c}_{l_0}}{\partial \varrho} \\ \frac{\partial \dot{\varrho}}{\partial \mathbf{r}} & \frac{\partial \dot{\varrho}}{\partial \mathbf{V}} & \frac{\partial \dot{\varrho}}{\partial \boldsymbol{\sigma}} & \frac{\partial \dot{\varrho}}{\partial \boldsymbol{\omega}} & \frac{\partial \dot{\varrho}}{\partial \eta_T} & \frac{\partial \dot{\varrho}}{\partial c_{l_0}} & \frac{\partial \dot{\varrho}}{\partial \varrho} \end{bmatrix} \quad (4.2)$$

Several entries in this partial derivative matrix are zero or trivial. First, all partial derivatives of the velocity are zero, except with respect to itself, which is identity. Secondly, the acceleration does not depend on the angular velocity and the rolling moment coefficient. The kinematic relation of the MRP does only depend on the MRP itself and the angular velocity. The angular velocity does not depend on the thrust magnitude, since the thrust misalignment moment was

assumed to be zero. The nominal time derivative of the thrust magnitude and rolling moment coefficient are zero, and therefore also their partial derivatives are zero. Finally, the square root density only depends on position, velocity and itself. This results in the following equation:

$$\frac{\partial \dot{\mathbf{x}}}{\partial \mathbf{x}} = \begin{bmatrix} 0 & \mathbf{I}_3 & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial \mathbf{a}}{\partial r} & \frac{\partial \mathbf{a}}{\partial V} & \frac{\partial \mathbf{a}}{\partial \sigma} & 0 & \frac{\partial \mathbf{a}}{\partial \eta_T} & 0 & \frac{\partial \mathbf{a}}{\partial \varrho} \\ 0 & 0 & \frac{\partial \dot{\sigma}}{\partial \sigma} & \frac{\partial \dot{\sigma}}{\partial \omega} & \frac{\partial \dot{\sigma}}{\partial \eta_T} & 0 & 0 \\ \frac{\partial \dot{\omega}}{\partial r} & \frac{\partial \dot{\omega}}{\partial V} & \frac{\partial \dot{\omega}}{\partial \sigma} & \frac{\partial \dot{\omega}}{\partial \omega} & 0 & \frac{\partial \dot{\omega}}{\partial c_{t_0}} & \frac{\partial \dot{\omega}}{\partial \varrho} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial \dot{\varrho}}{\partial r} & \frac{\partial \dot{\varrho}}{\partial V} & 0 & 0 & 0 & 0 & \frac{\partial \dot{\varrho}}{\partial \varrho} \end{bmatrix} \quad (4.3)$$

The remaining partial derivatives of equation (4.3) are worked out in appendix B.3.

The tower constraints from equation (2.104) are applied when the distance from the launch tower to the rocket is smaller than the launch tower length. This simplification has as consequence that the constraints are applied when the rocket is next to the tower, even though the rocket is not in the tower. This is, however, a situation which is unlikely to arise. As stated in section 2.7.6, the time derivative of velocity while the rocket is subject to tower constraints is given as follows, with $\hat{\mathbf{u}}_L$ the tower unit vector, and \mathbf{a} the unconstrained acceleration:

$$\dot{\mathbf{V}} = \hat{\mathbf{u}}_L \hat{\mathbf{u}}_L^T \mathbf{a} \quad (4.4)$$

Since the tower unit vector is constant, the matrix $\hat{\mathbf{u}}_L \hat{\mathbf{u}}_L^T$ is also constant, and as given by the rules of differentiation in appendix B.1.2, the partial derivative can be found as:

$$\frac{\partial \dot{\mathbf{V}}}{\partial \mathbf{x}} = \hat{\mathbf{u}}_L \hat{\mathbf{u}}_L^T \frac{\partial \mathbf{a}}{\partial \mathbf{x}} \quad (4.5)$$

It was also found in section 2.7.6, that the angular acceleration is zero, i.e., $\dot{\boldsymbol{\omega}} = 0$, and with an initial angular velocity of zero, also the time derivative of the MRP is zero. Therefore, the partial derivatives of angular acceleration and of the MRP time derivative are zero. Summing this up, while the rocket is subject to tower constraints, equation (4.3) simplifies to:

$$\frac{\partial \dot{\mathbf{x}}}{\partial \mathbf{x}} = \begin{bmatrix} 0 & \mathbf{I}_3 & 0 & 0 & 0 & 0 & 0 \\ \hat{\mathbf{u}}_L \hat{\mathbf{u}}_L^T \frac{\partial \mathbf{a}}{\partial r} & \hat{\mathbf{u}}_L \hat{\mathbf{u}}_L^T \frac{\partial \mathbf{a}}{\partial V} & \hat{\mathbf{u}}_L \hat{\mathbf{u}}_L^T \frac{\partial \mathbf{a}}{\partial \sigma} & 0 & \hat{\mathbf{u}}_L \hat{\mathbf{u}}_L^T \frac{\partial \mathbf{a}}{\partial \eta_T} & 0 & \hat{\mathbf{u}}_L \hat{\mathbf{u}}_L^T \frac{\partial \mathbf{a}}{\partial \varrho} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial \dot{\varrho}}{\partial r} & \frac{\partial \dot{\varrho}}{\partial V} & 0 & 0 & 0 & 0 & \frac{\partial \dot{\varrho}}{\partial \varrho} \end{bmatrix} \quad (4.6)$$

The only remaining part of the system model is the process noise. Given a process noise \mathbf{Q}_D associated with $\dot{\mathbf{x}}$, then the process noise for the state transition function can be found as (Simon, 2006):

$$\mathbf{Q} = \mathbf{F}_{k-1} \mathbf{Q}_D \mathbf{F}_{k-1}^T \cdot (t_k - t_{k-1}) \quad (4.7)$$

It was found that the simplified estimator model required different matrices for \mathbf{Q}_D for each of the phases described in section 4.4. Proper values for these matrices were found empirically, and can be found in table C.2. The initial error covariance is given in table C.1.

4.2 Measurement models

The second part of the estimator are the measurement models. These models provide a measurement function, which relates state to measurements, and in the case of an EKF, also provide a linearization of the measurement function. The measurement functions are also used in the simulator to generate measurements. In this section, accelerometer, gyroscope, magnetometer, barometric pressure sensor and GPS are treated.

4.2.1 Accelerometer

An accelerometer is a three-axis sensor. It does not measure acceleration though, but specific force (Farrell, 2008). The specific force is the acceleration minus the gravity. If both the gravity vector \mathbf{g} and the acceleration \mathbf{a} are defined in the same frames, then the specific force \mathbf{f} is given as:

$$\mathbf{f} = \mathbf{a} - \mathbf{g} \quad (4.8)$$

The acceleration is measured at a point which is not necessarily equal to the center of mass. Therefore, the acceleration component in the specific force is not equal to the acceleration of the body if the body rotates. The acceleration of this point is related to the acceleration of the center of mass by the following relation, where \mathbf{a}_P is the acceleration of the accelerometer at point P, \mathbf{a}_C the acceleration of the center of mass, \mathbf{r}_P the vector from the center of mass to P and $\boldsymbol{\omega}$ and $\dot{\boldsymbol{\omega}}$ the angular velocity and angular acceleration respectively of the vehicle (Török, 2000):

$$\mathbf{a}_P = \mathbf{a}_C + \dot{\boldsymbol{\omega}} \times \mathbf{r}_P + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}_P) \quad (4.9)$$

Collecting these results, along with equation (2.90) for the gravity, the specific force can be expressed in the sensor frame. If \mathbf{C}_{BI} is the transformation matrix from the inertial to body frame, \mathbf{C}_{BC} the transformation matrix from the Earth fixed frame to the body frame, \mathbf{C}_{SB} the transformation matrix from body to sensor frame, \mathbf{a}_C^I the acceleration of the center of mass in the inertial frame, $\boldsymbol{\omega}$ the angular velocity in the body frame, \mathbf{r}_a the vector from the center of mass to the acceleration sensor and \mathbf{g}^C the gravitational acceleration given by (2.90), then the specific force measured by the acceleration sensor is:

$$\mathbf{f} = \mathbf{C}_{SB} (\mathbf{C}_{BI} (\mathbf{a}_C^I + \dot{\boldsymbol{\omega}} \times \mathbf{r}_a + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}_a)) + \mathbf{C}_{BC} \mathbf{g}^C) \quad (4.10)$$

The output of the sensor is not directly equal to the measurement of the specific force, but diluted with noise sources. The most significant noise sources are bias, scale factor errors, non-orthogonality, white noise, and higher-order effects (Farrell, 2008). It is assumed that the higher-order effects are negligible compared to the other effects. White noise is a random, Gaussian distributed difference of the output from the actual value and designated \mathbf{w}_a . Bias is a constant offset from the actual specific force and designated \mathbf{b}_a . Three biases can be distinguished: the constant bias, the run-to-run bias and the in-run bias (Aggarwal et al., 2010). The constant bias can be removed by calibration, while the in-run bias can be considered small. The scale factor error is an error that is linear to the actual specific force, and is given by a diagonal matrix $\boldsymbol{\delta S}_a$ with entries s_x , s_y and s_z . Non-orthogonality quantifies the cross-axes sensitivity: to what extent the x -component of the specific force is sensed on the y - and z -axes, etc. For small non-orthogonalities, this can be quantified by the non-orthogonality matrix $\boldsymbol{\delta a}$. The diagonal elements of the matrix are zero, and the other elements δa_{ij} , with for example $i = x; j = y$ for the element on the first row and second column. The resulting scale factor error matrix, non-orthogonality matrix, and bias vector are respectively:

$$\boldsymbol{\delta S}_a = \begin{bmatrix} \delta s_{ax} & 0 & 0 \\ 0 & \delta s_{ay} & 0 \\ 0 & 0 & \delta s_{az} \end{bmatrix} \quad \boldsymbol{\delta a} = \begin{bmatrix} 0 & -\delta a_{xy} & \delta a_{xz} \\ \delta a_{yx} & 0 & -\delta a_{yz} \\ -\delta a_{zx} & \delta a_{zy} & 0 \end{bmatrix} \quad \mathbf{b}_a = \begin{bmatrix} b_{ax} \\ b_{ay} \\ b_{az} \end{bmatrix} \quad (4.11)$$

These errors yield for the sensor output \mathbf{y}_a the following equation:

$$\mathbf{y}_a = (\mathbf{I}_3 - \boldsymbol{\delta}\mathbf{a} - \boldsymbol{\delta}\mathbf{S}_a) \mathbf{f} + \mathbf{b}_a + \mathbf{w}_a \quad (4.12)$$

For the purpose of flight estimation, partial derivatives of equation (4.12) with respect to the state variables must be determined. Furthermore, it may be required to estimate some of the noise sources. If this is the case, these noise sources will need to be included in the state vector, and the partial derivatives with respect to them should be determined as well. For the determination of partial derivatives, the rules of differentiation established in appendix B.1 are used.

First, the partial derivatives with respect to position will be determined. The only dependence of equation (4.12) with respect to position is gravity. In appendix B.3.1, the partial derivative of equation (2.90) is worked out. This results in:

$$\frac{\partial \mathbf{y}_a}{\partial \mathbf{r}} = (\mathbf{I}_3 - \boldsymbol{\delta}\mathbf{a} - \boldsymbol{\delta}\mathbf{S}_a) \mathbf{C}_{SB} \mathbf{C}_{BC} \frac{\partial \mathbf{g}^C}{\partial \mathbf{r}} \quad (4.13)$$

The partial derivative with respect to attitude depends on $\frac{\partial \mathbf{C}_{BI}}{\partial \sigma_i}$, which is derived in appendix B.1.2. Since this is the only dependence of equation (4.13) on attitude, the partial derivative with respect to attitude becomes:

$$\frac{\partial \mathbf{y}_a}{\partial \sigma_i} = (\mathbf{I}_3 - \boldsymbol{\delta}\mathbf{a} - \boldsymbol{\delta}\mathbf{S}_a) \mathbf{C}_{SB} \left(\frac{\partial \mathbf{C}_{BI}}{\partial \sigma_i} (\mathbf{a}_C^I + \dot{\omega} \times \mathbf{r}_a + \omega \times (\omega \times \mathbf{r}_a)) + \frac{\partial \mathbf{C}_{BI}}{\partial \sigma_i} \mathbf{C}_{IC} \mathbf{g}^C \right) \quad (4.14)$$

The partial derivative with respect to the noise sources is trivial, considering that they are linear in equation (4.12). The partial derivative with respect to bias is given by the identity matrix. The partial derivative with respect to a non-orthogonality or scale factor error is a matrix with zeros on all elements, except with a one on the position of the corresponding non-orthogonality. For example, for the partial derivative with respect to δa_{xy} this gives:

$$\frac{\partial \mathbf{y}_a}{\partial a_{xy}} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{f} \quad (4.15)$$

4.2.2 Gyroscope

A gyroscope is a three-axis sensor like an accelerometer. It directly measures the angular velocity in the body frame ω . Therefore, the measurement equation only contains this angular velocity and the error sources that also applied to the accelerometer as described in section 4.2.1. These are the bias \mathbf{b}_ω , scale factor errors $\boldsymbol{\delta}\mathbf{S}_\omega$, non-orthogonality $\boldsymbol{\delta}\omega$ and white noise \mathbf{w}_ω , defined by:

$$\boldsymbol{\delta}\mathbf{S}_\omega = \begin{bmatrix} \delta s_{\omega_x} & 0 & 0 \\ 0 & \delta s_{\omega_y} & 0 \\ 0 & 0 & \delta s_{\omega_z} \end{bmatrix} \quad \boldsymbol{\delta}\omega = \begin{bmatrix} 0 & -\delta\omega_{xy} & \delta\omega_{xz} \\ \delta\omega_{yx} & 0 & -\delta\omega_{yz} \\ -\delta\omega_{zx} & \delta\omega_{zy} & 0 \end{bmatrix} \quad \mathbf{b}_\omega = \begin{bmatrix} b_{\omega_x} \\ b_{\omega_y} \\ b_{\omega_z} \end{bmatrix} \quad (4.16)$$

This yields the following measurement equation:

$$\mathbf{y}_\omega = (\mathbf{I}_3 - \boldsymbol{\delta}\omega - \boldsymbol{\delta}\mathbf{S}_\omega) \omega + \mathbf{b}_\omega + \mathbf{w}_\omega \quad (4.17)$$

Also for the the partial derivatives with respect to the noise sources, the same equations can be used as for the accelerometer.

In addition to the mentioned noise sources, it is possible that there is a slight sensitivity of the gyroscope towards acceleration, the so-called *g-sensitivity*. However, this effect is minor, and even commonly absent from literature, and is therefore assumed to be insignificant.

4.2.3 Magnetometer

Similar to the accelerometer and gyroscope, the magnetometer is a three axis sensor. It measures the components of the local magnetic field in the sensor frame, and with the sensor frame being fixed with respect to the body frame, provides information about the attitude of the vehicle. With the magnetic field vector in the inertial frame defined as \mathbf{m} , the measurement equation becomes:

$$\mathbf{y}_m = (\mathbf{I}_3 - \boldsymbol{\delta}\mathbf{m} - \boldsymbol{\delta}\mathbf{S}_m) \mathbf{m} + \mathbf{b}_m + \mathbf{w}_m \quad (4.18)$$

where respectively the scale factor error, non-orthogonality and bias are given by:

$$\boldsymbol{\delta}\mathbf{S}_m = \begin{bmatrix} \delta s_{m_x} & 0 & 0 \\ 0 & \delta s_{m_y} & 0 \\ 0 & 0 & \delta s_{m_z} \end{bmatrix} \quad \boldsymbol{\delta}\mathbf{m} = \begin{bmatrix} 0 & -\delta m_{xy} & \delta m_{xz} \\ \delta m_{yx} & 0 & -\delta m_{yz} \\ -\delta m_{zx} & \delta m_{zy} & 0 \end{bmatrix} \quad \mathbf{b}_m = \begin{bmatrix} b_{m_x} \\ b_{m_y} \\ b_{m_z} \end{bmatrix} \quad (4.19)$$

There are several models for the magnetic field. An example of this is the *World Magnetic Model 2010* (Maus et al., 2010), which provides an expansion in spherical harmonics for the magnetic field. However, it is not necessary to implement the full model in the estimator. The magnetic field vector stays relatively constant during flight. Therefore, it is possible to use software that implements WMM2010, such as MATLAB, to obtain the magnetic field vector at the launch location, and provide this information before flight to the estimator.

The assumption that the magnetic field vector from the model is equal to the real magnetic field vector is not entirely valid. Ferromagnetic materials in the presence of the sensor may cause these to differ. However, during flight any ferromagnetic material in the rocket is fixed with respect to the magnetometer, which will only result in a bias, which is already part of equation (4.18).

4.2.4 Barometric pressure sensor

The static pressure is a quantity that can be directly sensed using a pressure sensor. The atmospheric pressure is a function of the altitude and if standard atmospheric conditions are assumed, the altitude can be derived from pressure data. An important condition for using a pressure sensor to determine the altitude, is that the pressure sensor is located at a position where the static pressure is equal to the atmospheric pressure or where the static pressure can be related to the atmospheric pressure. This may not be the case if the compartment where the sensor is located is sealed, or if the the pressure inlet is behind a shock wave. In the latter case, it is possible to use shock relations to determine the pressure as a function of Mach number and geometry, but this requires accurate knowledge of both quantities. Therefore, pressure measurements will only be used when the estimated Mach number is well in the subsonic regime.

Since a pressure sensor output is a single scalar, there are no misalignments. The measurement equation therefore reduces to a scale factor error, bias and white noise:

$$y_p = (1 - s_p)p + b_p + w_p \quad (4.20)$$

Linking the pressure to an altitude requires an atmospheric model, which gives the altitude as function of the pressure. Two atmospheric models are commonly used: the exponential atmosphere and the international standard atmosphere. The exponential model is much easier to implement: only a single logarithm is needed to obtain the altitude from the pressure. Because of its simplifications, it is probably less accurate than the standard atmospheric model for large ranges of altitudes. However, because the vehicle will already fly transonic at relatively low altitudes, it will not have usable pressure measurements over a long range of altitudes, making the exponential atmosphere suitable for use.

With r the distance from the origin of the coordinate system to the vehicle, r_0 the distance from the origin of the coordinate system to the ground, p_0 the pressure at ground and H a constant scale height, the pressure is given by:

$$p = p_0 e^{-\frac{r-r_0}{H}} \quad (4.21)$$

The advantage of an exponential atmosphere is that the partial derivative with respect to position \mathbf{r} can be determined analytically. The partial derivative with respect to all other state variables is zero. The partial derivative of pressure with respect to position can be derived as:

$$\frac{\partial p}{\partial \mathbf{r}} = p_0 e^{-\frac{r-r_0}{H}} \frac{1}{H} \frac{\partial r}{\mathbf{r}} = \frac{p_0}{H} e^{-\frac{r-r_0}{H}} \frac{\mathbf{r}^T}{r} \quad (4.22)$$

The use of the exponential atmosphere introduces errors in the calculation of altitude. For example, for a US1976 atmosphere, a pressure of 900 hPa corresponds to an altitude of 989 m, whereas for an exponential atmosphere with scale height $H = 7800$ m, the corresponding altitude is 925 m. To deal with this effect, in addition to the RMS noise of that will be obtained from the calibration process, the Kalman measurement noise matrix \mathbf{R} should be increased with altitude to allow the filter to deal with the inaccuracy of the pressure measurement model.

4.2.5 GPS

The Global Positioning System (GPS) is a system of satellites orbiting Earth, each sending information about their own position and the local time. Using this information, the range to the satellite can be determined. This is called the pseudorange. With known pseudoranges to four satellites of known position, an estimate of position and the current time can be made, using least-squares estimation.

The GPS system should in principle be useful for sounding rockets, as it was developed with missile guidance for the US Armed Forces in mind. However, the US government imposes restrictions on GPS receivers, forcing them to only output information when the *COCOM* limits are not exceeded. These limits are a maximum altitude of 60,000 feet (18.29 km) and/or a maximum velocity of 1000 knots (514.44 m/s) (Betts and Rutherford, 2009). Whether these limits should be satisfied both or individually is left ambiguous and indeed open to the receiver manufacturer.

All GPS receivers output position information once at least four satellites are available. This information is directly part of the state and the measurement equation can be derived trivially, with \mathbf{y}_{GPS} the GPS measurement, and \mathbf{w}_{GPS} random, zero-mean Gaussian distributed noise:

$$\mathbf{y}_{GPS} = \mathbf{r} + \mathbf{w}_{GPS} \quad (4.23)$$

The partial derivative is given by:

$$\frac{\partial \mathbf{y}_{GPS}}{\partial \mathbf{r}} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \end{bmatrix} \quad (4.24)$$

If GPS measurements are used according to these equations, care should be taken that the noise vector is indeed zero-mean Gaussian distributed, or reasonably close to this. Internally, the GPS receiver contains tracking loops and possibly a Kalman filter, which makes the error time-dependent (Farrell, 2008).

Some receivers also allow for output of the raw pseudorange data. The pseudorange y_r is given by the following equation, with \mathbf{r}_{GPS} the position of the GPS satellite:

$$y_r = \|\mathbf{r}_{GPS} - \mathbf{r}\| + w_r \quad (4.25)$$

The partial derivative is given by:

$$\frac{\partial y_r}{\partial \mathbf{r}} = \left(\frac{\mathbf{r}_{GPS} - \mathbf{r}}{\|\mathbf{r}_{GPS} - \mathbf{r}\|} \right)^T \quad (4.26)$$

There are several effects which degrade the GPS position estimation. These errors together form the user equivalent range error (UERE). Typically four common mode errors and two non common mode errors are distinguished. The common-mode errors are ionospheric delay, tropospheric delay, clock errors and ephemeris errors, and they are common to all nearby receivers. Non-common-mode errors are receiver noise and multipath errors (Farrell, 2008).

Ionospheric delay and tropospheric delay are as implied by the name, deviations in propagation velocities of the GPS signals caused by the respective atmospheric layers. The amount of ionospheric delay is determined by the level of ionization in the ionosphere. Tropospheric delay is related to temperature, pressure and humidity in the troposphere. Both effects are time dependent. Obviously, as the vehicle travels through the respective layers the error associated with that layer decreases and eventually becomes zero as the edge of the layer is reached.

The range between a satellite and the receiver is determined by calculating the time it took for the signal to reach the receiver. Thus, a clock error propagates in the position determination. Most of the clock error can be removed when at least four satellites are available. However, there will always be a small residual error left, which is modeled as a random error.

Ephemeris error is the error in position of the satellite. Receiver noise is the collective random error originating at the receiver. Multipath is the error due to reflections of the signal, giving rise to interference with the directly received signal.

For each of the six errors, the typical standard deviation is given in table 4.1. The total UERE standard deviation is obtained by taking the root-sum-square of the individual errors.

Table 4.1: UERE standard deviation components. Source: Farrell (2008)

Error source	σ [m]
Ionospheric delay	7 - 10
Tropospheric delay	1
Residual clock error	2
Ephemeris error	2
Receiver noise	0.1 - 0.7
Multipath error	0.1 - 3.0
UERE	8 - 11

Dilution of precision

Dilution of precision is a measure of the influence of the geometry of the satellites visible to the user on the total error budget. The ideal situation is when satellites are at orthogonal angles from a user perspective. In case the satellites are close together from a user point of view, the UERE propagates much stronger in the total error budget.

To calculate the dilution of precision, the matrix \mathbf{H} is defined, which is a matrix in which each of the rows is a vector containing the direction cosines from the user to a satellite in the first three entries, and 1 in the fourth entry. There are as many rows as there are satellites in view. Then the dilution of precision matrix is defined as (Langley, 1999):

$$\mathbf{D} = (\mathbf{H}^T \mathbf{H})^{-1} \quad (4.27)$$

The covariance matrix \mathbf{P} of the position estimate is equal to the dilution of precision times the UERE variance:

$$\mathbf{P} = \mathbf{D}\sigma_{UERE}^2 \quad (4.28)$$

4.3 Kinematic estimator model

Previously, an estimator model was derived that include a model of the flight trajectory. However, if an accelerometer and gyroscope are present, then an estimator can be derived that does not require a model of the flight trajectory, if acceleration and gyroscope measurements are assumed to be state derivatives. With this approach, the system model is purely kinematic, as mass, inertia and all other vehicle characteristics disappear from the equations. This approach has as advantage that the estimator is more robust against deviations from nominal flight, and because some measurements are not considered measurements in the normal sense of a Kalman filter, the Kalman filter equations have to be performed less often, yielding a computationally less expensive filter. The major disadvantage is that the filter is less resilient to measurement errors, and that a (temporary) lack of measurement data will yield a fully unfunctional filter.

To derive the flight model-free estimator, first the relation between measurements and state derivative must be obtained. The output of an accelerometer is given by equation (4.13). The specific force in terms of the accelerometer output is to the first order (Farrell, 2008):

$$\mathbf{f} = (\mathbf{I}_3 + \boldsymbol{\delta}\mathbf{a} + \boldsymbol{\delta}\mathbf{S}_a) \mathbf{y}_a - \mathbf{b}_a \quad (4.29)$$

Then, neglecting the rotational terms in equation (4.10), the acceleration becomes:

$$\mathbf{a}_C^I = \mathbf{C}_{BI}^T (\mathbf{C}_{SB}^T \mathbf{f} - \mathbf{C}_{BC} \mathbf{g}^C) \quad (4.30)$$

Equation (4.17) gives the gyroscope measurement equation. Then, from equation (2.40), it follows that the time derivative $\dot{\boldsymbol{\sigma}}$ of the MRP $\boldsymbol{\sigma}$ as a function of the gyroscope measurement is:

$$\boldsymbol{\omega} = (\mathbf{I}_3 + \boldsymbol{\delta}\boldsymbol{\omega} + \boldsymbol{\delta}\mathbf{S}_\omega) \mathbf{y}_\omega - \mathbf{b}_\omega \quad (4.31)$$

$$\dot{\boldsymbol{\sigma}} = \frac{1}{2}\boldsymbol{\sigma}\boldsymbol{\sigma}^T \mathbf{y}_\omega + \frac{1}{4}(1 - \sigma^2) \mathbf{y}_\omega - \frac{1}{2}\boldsymbol{\sigma} \times \mathbf{y}_\omega \quad (4.32)$$

From the above equation, it follows that contrarily to the other estimator models, the angular velocity is not part of the state derivative, making the MRP $\boldsymbol{\sigma}$ the only rotational part of the state derivative. The translational part stays unchanged, with \mathbf{r} and \mathbf{V} being part of the state derivative. Then, even though the major part of the bias can be removed by calibration, a run-to-run bias will be present, which may need estimation during flight (Aggarwal et al., 2010). The bias of the accelerometer and gyroscope are designated as \mathbf{b}_a and \mathbf{b}_ω respectively. This results in the following preliminary state vector:

$$\mathbf{x} = \begin{bmatrix} \mathbf{r} \\ \mathbf{V} \\ \boldsymbol{\sigma} \\ \mathbf{b}_a \\ \mathbf{b}_\omega \end{bmatrix} \quad (4.33)$$

The next step in developing the estimator model, is obtaining the state transition function which transitions the state from time t_i to t_{i+1} . It is assumed that at both time instances measurements from the accelerometer and gyroscope are taken. The state derivative function needs to be integrated to obtain the state at t_{i+1} . This is done via the Runge-Kutta 4 numerical integrator. As measurements are only available at discrete intervals, during an integration step, the measurement part of the state derivative will be constant.

Finally, the process noise matrix is required. It was found out that unlike the simplified estimator model from section 4.1, the process noise matrix could be held constant for the entire flight, except for the three entries corresponding to the MRP. It was found that the higher the angular velocity, the larger the MRP process noise was required. Therefore, it was decided to scale the attitude part of the process noise with the squared norm of the angular velocity, which was found to yield good results. The resulting process noise matrix for the entire flight is:

$$\text{diag } \mathbf{Q}_k = (t_k - t_{k-1}) \cdot \begin{bmatrix} 0 & 0 & 0 & 10^{-2} & 10^{-2} & 10^{-2} & 10^{-6}\|\boldsymbol{\omega}\|^2 & 10^{-6}\|\boldsymbol{\omega}\|^2 & 10^{-6}\|\boldsymbol{\omega}\|^2 \end{bmatrix} \quad (4.34)$$

The initial error covariance is given in table C.3.

4.4 Flight computer model

It is important to not only simulate the flight mechanics and the measurements for an on-board navigation solution, but also the logic in the flight computer. The navigation system will need to select appropriate process noises and flight parameters. Such flight parameters can be based on flight events like the motor start time and the stage separation time. During in-flight navigation it is only possible to detect these events based on measurements. For example, with acceleration sensors the motor start in itself cannot necessarily be detected; only lift-off can be detected. When liftoff has been detected the rocket will have already gained a little altitude, which is not yet reflected in the estimated state.

The flight computer is a state machine, consisting of different states for each flight phase. The state flow consists of the following consecutive states, which is also given in a flowchart in figure 4.1:

Tower This is the phase while the rocket is still in the tower. During this phase all necessary calibration can be performed.

Powered-constrained This is the phase while the first stage motor is on, but the rocket is still subject to the launch tower constraints. This phase is entered when it is detected that the acceleration sensor output has reached a certain threshold value.

Powered This is the phase while the first stage motor is on, and the rocket is not subject anymore to tower constrained. This phase is entered when the estimated altitude is above the tower length. The effect of the launch tower being not exactly vertical is neglected.

Coast 1A This is the phase after the first stage motor has burnt out, and the first stage has not yet separated. It is entered when the output of the acceleration sensor is below a certain threshold.

Coast 1B This is the phase after the first stage has separated, but the second stage has not yet ignited. It is entered when stage separation time has passed.

Powered 2 This is the phase after the second stage has been ignited. It is entered when it is detected that the acceleration sensor output has reached a certain threshold value.

Coast 2 This is the phase after the second stage motor has burnt out and before the rocket has reached apogee. It is entered when the output of the acceleration sensor is below a certain threshold.

Descent This is the phase after the rocket has reached apogee. It is entered when the estimated attitude indicates that the rocket is pointing downwards.

Separated descent This is the phase after the rocket has separated the nose cone from the second stage, and is entered once the nose cone separation timer has expired.

Parachuting descent This is the phase after the rocket has ejected the parachute and is entered once the parachuting timer has expired.

Landed This is the phase when the rocket has impacted ground. This can be detected when the only output of the acceleration sensor is the gravity.

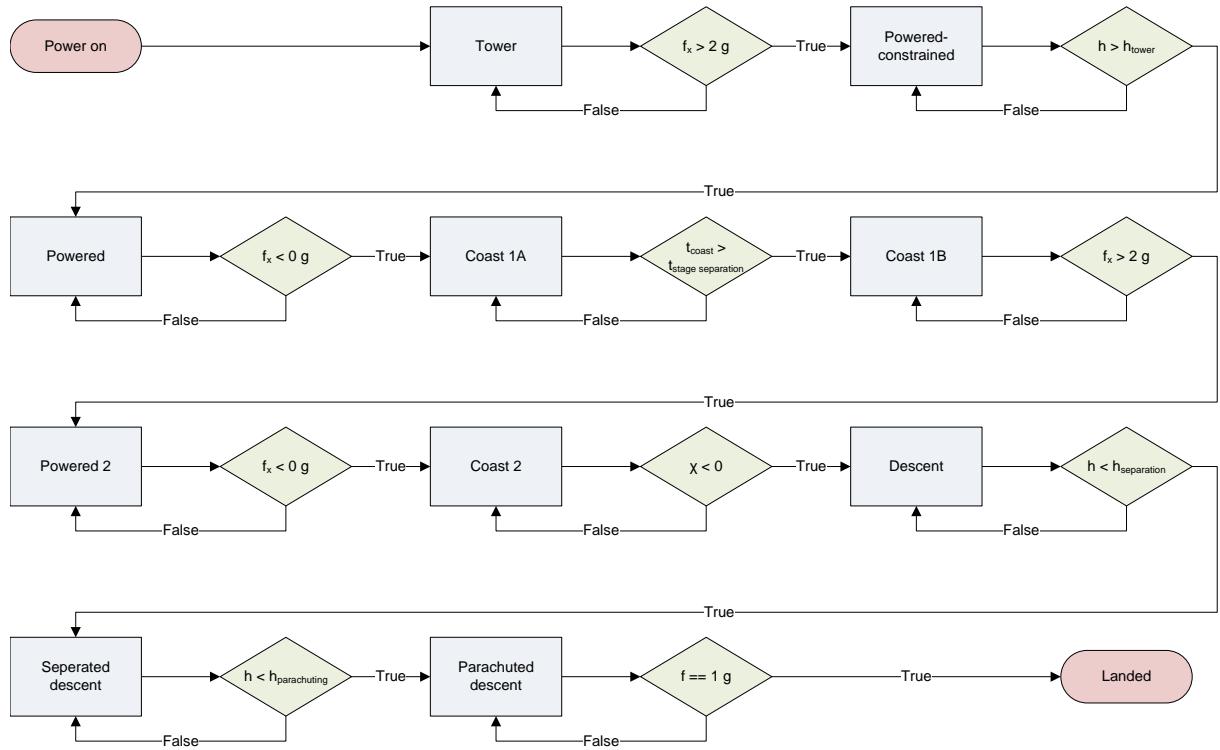


Figure 4.1: State flow of the flight computer. Threshold values are indicative.

Chapter 5

Measurement system design

In order to test several sensors for the final Stratos II measurement system, and to obtain actual flight data, a preliminary measurement system was designed and built, and flown on the Stratos II concept launcher. This chapter elaborates on the development of the measurement system in section 5.1 and the calibration in section 5.2.

5.1 Measurement system development

At the time when the measurement system design was started, there was only very little time left until the concept launcher flight. This meant that during the design of the measurement system the flight simulations were not yet fully performed and that the flight regime was still unknown. Therefore, it proved to be very hard to pose good requirements, and as a consequence, only a few rough requirements were posed:

1. The system should be able to measure the translational and rotational motion of the rocket during flight.
2. The system should store the measured data for post-flight analysis.
3. The system should be low-cost.
4. The system should be developed within one month.

As the flight regime was not known, to fulfill requirement 1, estimates of the flight conditions were required. The initial guess for the required accelerometer range was obtained by dividing the maximum thrust (1.5 kN) by the empty rocket mass (22 kg), yielding a maximum expected acceleration of 68 m/s^2 or 7.0 g. For the required rotational rate range, it was suspected that the dominating angular motion would be roll. The magnitude of the roll rate was not known. Literature research suggested that sounding rockets with canted fins would reach roll rates up to 4 rotations per second (NASA Goddard Space Flight Center, 2001). The budget requirement 3 was never fully quantified, but it was understood that the approximate cost of the measurement system would be in the order of 300 Euro. Requirement 4 was posed because it was only one month to the launch of the rocket on which the measurement system would fly, when the design of the system started.

5.1.1 Component selection and system level design

The choice was made to observe as many variables that were potentially useful for post-flight analysis as reasonably possible within the cost and time constraints. It was therefore determined that an accelerometer, an angular rate gyroscope, a pressure sensor, a magnetometer and a GPS receiver would be present. Then, sensors were selected that specified the lowest noise levels as possible, resulting in the following sensors:

- A 3-axis MEMS accelerometer, Bosch Sensortech BMA180, with range +/- 16 g (Bosch Sensortech, 2009)
 - A 3-axis MEMS gyroscope, Invensense IMU-3000, with range +/- 2000 degrees per second (InvenSense, 2011)
 - An absolute pressure sensor, Measurement Specialities Model 1451, with range 0 Pa to 1.03×10^5 Pa (Measurement Specialties, 2011)
 - A 3-axis magnetometer, PNI Corp MicroMag 3, with range +/- 1100 uT (PNI Corporation)
 - A GS407 GPS receiver with Sarantel SL1206 active helical antenna and u-Blox 5H chipset (S.P.K. Electronics Co., Ltd., 2010)

The choice was made to include the 3-axis accelerometer three times, due to its small price, and so that more data would be available. To enable easy storage and readout of the data, it was chosen to store the sensor data on a micro SD card. To readout the sensors and write them to the SD card, it was chosen to use a microcontroller, as that is the most straightforward way to perform these tasks. The selected microcontroller was an NXP LPC2148 ARM7. This microcontroller was selected because there was already a lot of experience with this microcontroller, the microcontroller is reasonably powerful, and because NXP could provide the microcontroller for free.

In figure 5.1 the block diagram of the system is shown. Here is clearly shown how the various components are connected to the microcontroller: the micro SD card on a dedicated SPI bus, the accelerometers on a shared SPI bus, and the gyroscope and the pressure sensor via an A/D converter on a shared I²C bus.

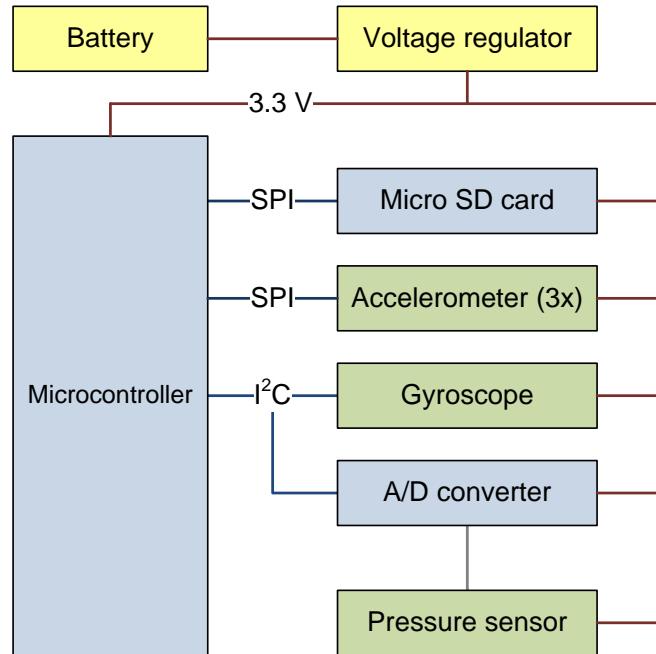


Figure 5.1: Block diagram of the preliminary measurement system

5.1.2 Circuit and PCB design

A printed circuit board (PCB) to mount the sensors, microcontroller and SD card on was designed in Altium Designer. First, a schematic of the circuit was made, where all the electrical connections between the components were designed. In such a schematic, the components from

the block diagram are placed, and the supporting components are added. Then, they are connected to each other appropriately. An example of this is shown in figure 5.2a, where the part of the schematic with the gyroscope is drawn.

Next, footprints are drawn. Footprints provide the link between the schematic representation of the component, and the what the component looks like in reality. The footprint defines where each of the numbered pins are located with respect to each other. The footprint of the LGA24 package of the gyroscope is shown in figure 5.2b.

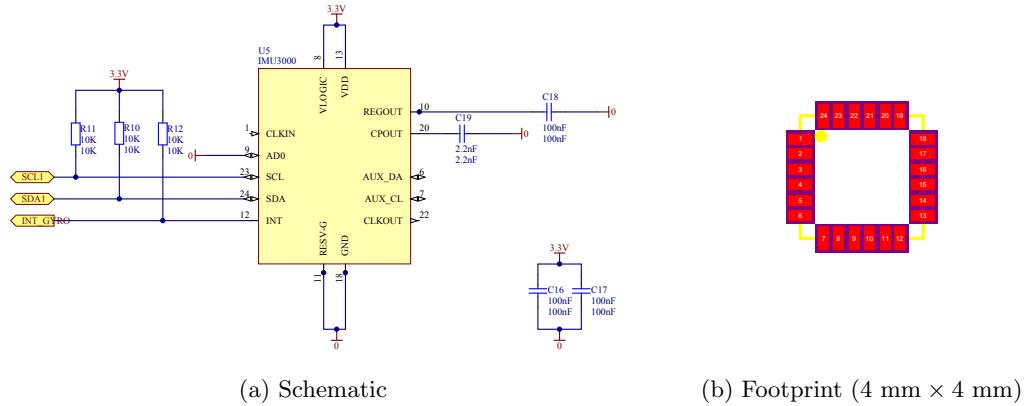


Figure 5.2: Schematic and footprint of gyroscope in Altium Designer

Finally, with the full schematic and all footprints drawn, the design of the PCB can be started. Altium Designer will place all footprints on a sheet, and indicate which pins should be connected to each other. The components can then be positioned, taking into account clearance and manufacturing constraints. Then, one-by-one tracks are drawn to connect the pins. This process is called routing. A part of the board, with some partial routing can be seen in figure 5.3, with the gray lines between track and pad indicating unrouted nets.

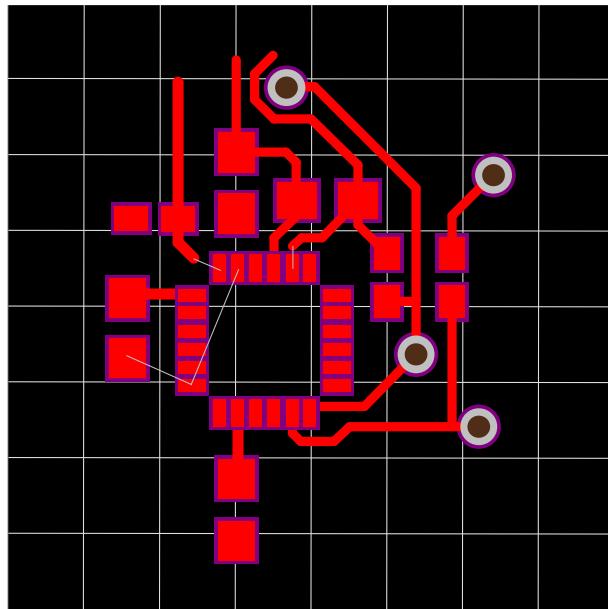


Figure 5.3: Partially routed measurement board

The top and bottom view of the fully routed PCB are shown in figure 5.4. There are several manufacturing and mounting constraints taken into account in the positioning of the components. First, to limit deflections and vibrations of the PCB and sensors, four mounting

points have been provided. Second, connectors for the GPS (P1), battery (Bat1), micro SD card (SD1) are mounted at the edge of the board for easy connection. Furthermore, several components, such as the gyroscope (U5) and accelerometers (U1, U2, U3) are in land grid array (LGA) package. These components can only be soldered in a reflow oven, and should therefore be all located on the same side of the board. Finally, due to constraints posed by the manufacturer of the PCB, the minimum spacing between tracks is limited to 8 mil (0.2 mm). To minimize the cost of the PCB, the area of the PCB was minimized, resulting in a rectangular PCB of 45 mm by 55 mm.

5.1.3 Manufacturing

Once the design in Altium Designer was finished, the design files were sent to a PCB manufacturer. They produce an empty PCB from the PCB design. This empty PCB can then be populated with components. First, the LGA components on the top layer were soldered in a reflow oven. Then, the other components were soldered manually piece-by-piece.

There were several challenges in the manufacturing process. First, the accelerometers initially appeared to not arrive on time. A first board was manufactured without the accelerometers. This board is shown on top of figure 5.5.

Nevertheless, the accelerometers did arrive just in time. This meant though that an entire new board had to be manufactured. When this was done, the board was not working. It appeared that one of the accelerometers was misplaced by 0.5 mm, while another accelerometer had residual solder underneath it. As there was no non-destructive way of solving this problem, it was decided that one of the two misbehaving accelerometers would be cut off from the board. This fix allowed the remaining accelerometer to function properly.

After this, another problem arose due to a surge in the power supply, which destroyed the remaining accelerometer. This was solved by mounting a new accelerometer on a separate board, and connecting this board with wires to the original measurement system. This finally resulted in a fully functioning system, so that the measurement board could be readied for software development, calibration and flight.

5.1.4 Software development

Software was written directly for the microcontroller, without an intermediate operating system, in C. The software is written such that measurements were sampled with the highest possible frequencies. The flowchart of this is shown in figure 5.6. The determination of data availability is left to the sensors, each which is connected with an interrupt line to the microcontroller. The software will serially check the status of these interrupt lines, and if necessary communicate with the sensor to request the data. This data is then copied to an intermediate buffer and timestamped with an accuracy of 3.1×10^{-5} s. If sufficient data for a sector of an SD card, 512 bytes, is available in the buffer, these bytes are written to the SD card. This means that at power loss at most 512 bytes are lost, which represents only a few milliseconds of data.

5.2 Sensor calibration

The sensors on the measurement system were not calibrated in the factory, so calibration had to be performed. Sensor calibration aims to obtain the actual relation between the sensor output and the variable. This relation can be defined as the transfer function that maps the value of the sensed quantity to the output of the sensor. For almost all sensors this transfer function is a linear function. Let y be the output of the sensor, f be the value of the sensed quantity, s the scale factor and b the offset. Then for an ideal sensor the transfer function becomes:

$$y = sf + b \quad (5.1)$$

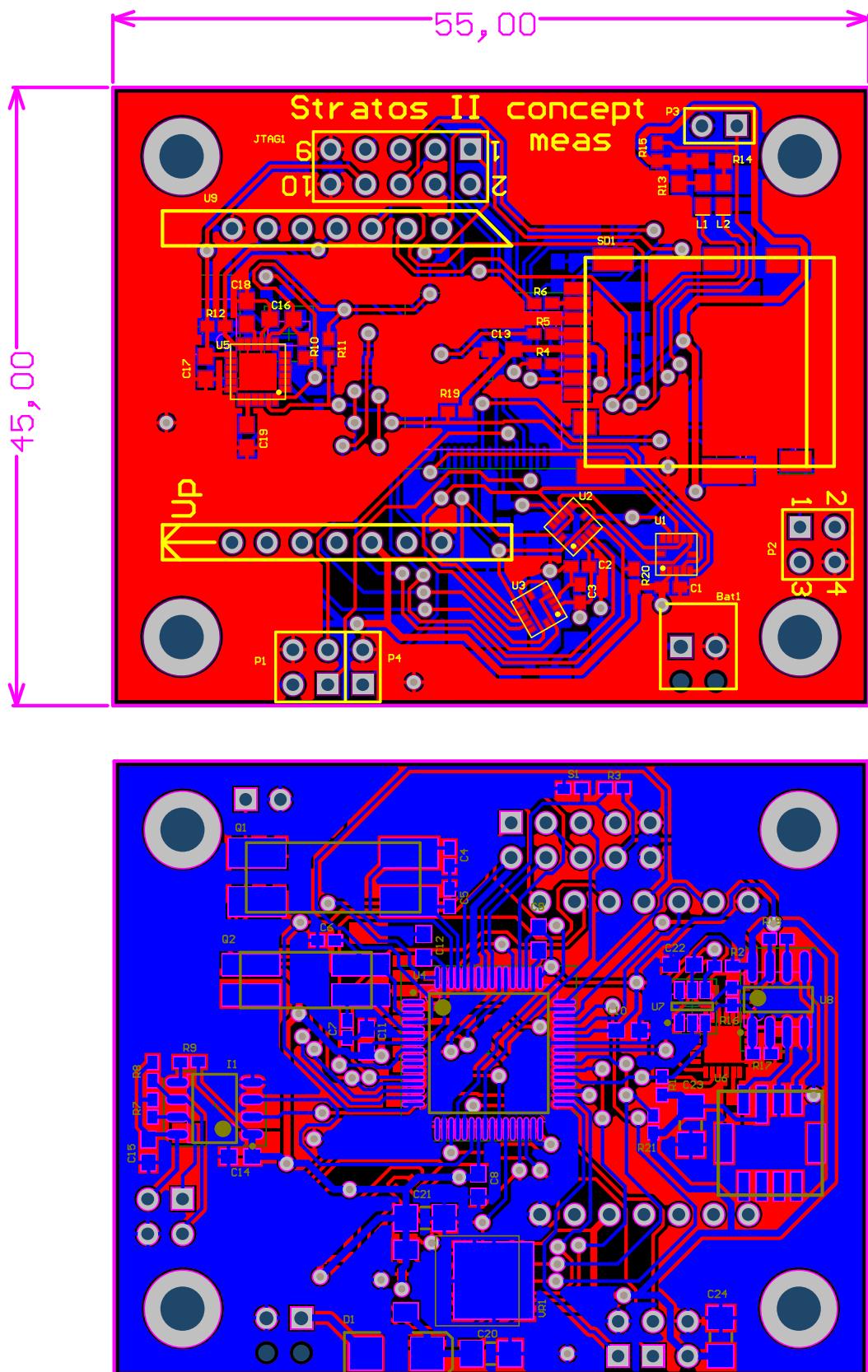


Figure 5.4: Respectively the top and bottom view of the designed PCB. Dimensions in mm.

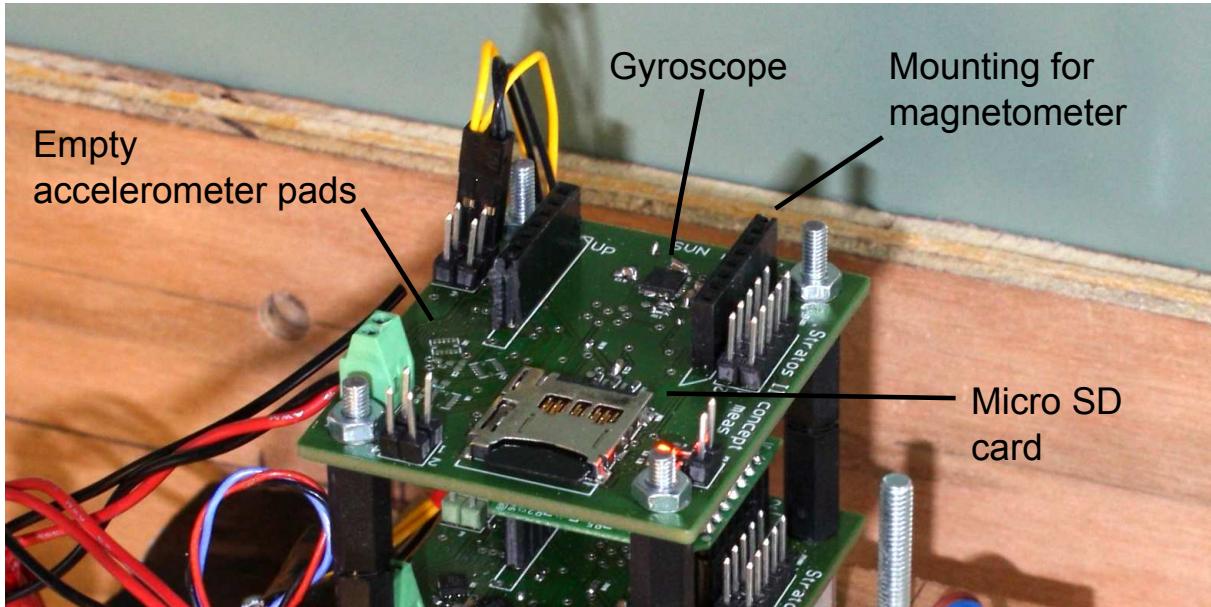


Figure 5.5: A measurement board without accelerometers

To determine the values of the unknown scale factor s and offset b , the sensor is subjected to multiple known values of f , while recording the output y . Least-squares estimation can then be used to determine the scale factor and offset, when at least two independent measurements are available. The output variable y is an integer whose range depends on resolution of the analog-to-digital converter, and the unit is commonly called a *least-significant-bit* (LSB).

Often a sensor measures quantities along three orthogonal axes. An example of this is a rate gyroscope, which measures angular velocity along its x-, y- and z-axes. The transfer function can then be written in vector-matrix form:

$$\mathbf{y} = \mathbf{S}\mathbf{f} + \mathbf{b} \quad (5.2)$$

The matrix \mathbf{S} is the scale factor matrix. If the sensor axes are orthogonal, and no misalignment is present, this is a diagonal matrix with on its diagonal the three scale factors s_x , s_y and s_z respectively. The vectors \mathbf{f} and \mathbf{b} are the sensed quantity and offset respectively.

An ideal sensor exactly follows the measurement equation of equation (5.2). In practice however a sensor is non-ideal, and there are several ways causing it to not exactly follow the ideal measurement equation. Non-orthogonality, misalignment and noise are such non-idealities that can be present. It will be assumed that non-orthogonality is negligible. However, if it is not negligible, then the off-diagonal elements of the scale factor matrix must be estimated as well. Misalignment will be discussed in section 5.2.1 and noise in section 5.2.2.

5.2.1 Misalignment

The output of the sensor \mathbf{y} is in a reference frame fixed to the sensor, while the values of \mathbf{f} are known in a different reference frame. The two reference frames are designated the *sensor-fixed reference frame* F_S and the *calibration-fixed reference frame* F_C , respectively. If these frames are not equal, then a misalignment is present, with the transformation from F_C to F_S given by the DCM \mathbf{C} , representing a full 3D-rotation. This gives as measurement equation:

$$\mathbf{y} = \mathbf{S}\mathbf{C}\mathbf{f} + \mathbf{b} \quad (5.3)$$

Because the misalignment is small, the DCM can be parametrized by the MRP as defined in section 2.2.3, without having to take into account the possibility of singularities. The DCM in

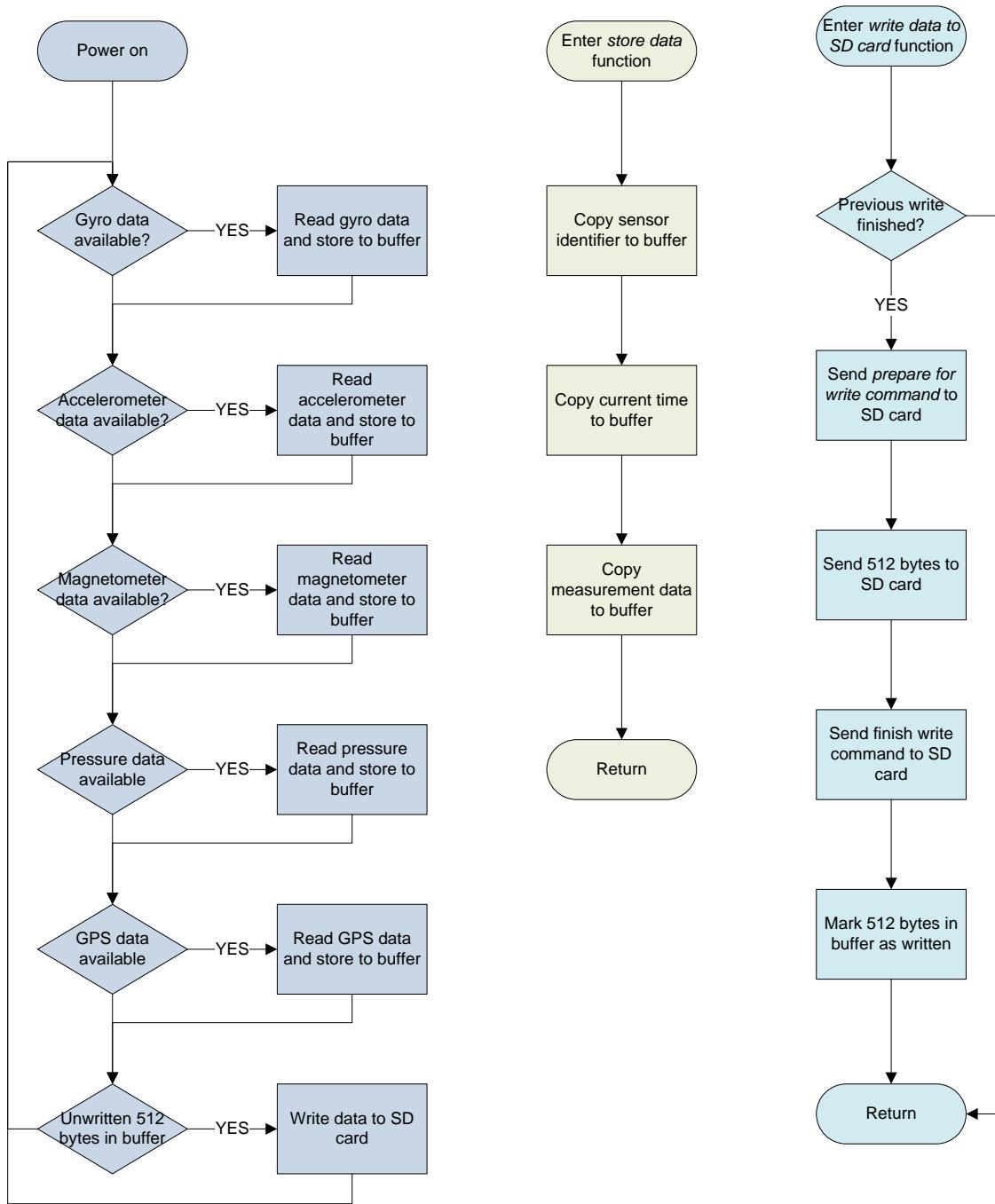


Figure 5.6: Flowchart of the measurement board software

terms of the MRP is given by equation (2.36). It is possible to combine the matrix product \mathbf{SC} in a single matrix, however, the advantage of separating them is that the mounting misalignment can be quantified in terms of an MRP, and converted to an angle.

The vector of unknowns for a misaligned sensor and the measurement vector are respectively:

$$\mathbf{x} = \begin{bmatrix} s_x & s_y & s_z & \sigma_1 & \sigma_2 & \sigma_3 & b_1 & b_2 & b_3 \end{bmatrix}^T \quad \mathbf{y} = \begin{bmatrix} y_x & y_y & y_z \end{bmatrix}^T \quad (5.4)$$

To estimate \mathbf{x} using a least-squares estimator, \mathbf{y} must be linearized in terms of \mathbf{x} :

$$\mathbf{H} = \frac{\partial \mathbf{y}}{\partial \mathbf{x}} \Big|_{\hat{\mathbf{x}}_0} = \frac{\partial}{\partial \mathbf{x}} (\mathbf{SC}\mathbf{f} + \mathbf{b})_{\hat{\mathbf{x}}_0} \quad (5.5)$$

The only non-trivial part of this equation is the partial derivative of the DCM with respect to the MRP. The derivation of this partial derivative can be found in appendix B.2.2. The resulting partial derivative with respect to σ_i is:

$$\frac{\partial \mathbf{C}}{\partial \sigma_i} = \frac{1}{Q} \left(\frac{\partial \mathbf{R}}{\partial \sigma_i} - \mathbf{C} \frac{\partial \mathbf{Q}}{\partial \sigma_i} \right) \quad (5.6)$$

The linearized measurement equation can then be used in a least-squares estimator as defined in section 3.1, to obtain the best estimate for the scale factors, misalignment and biases.

5.2.2 Noise

Noise is generally used as a way to describe random variation of the output signal. Noise is quantified as the root-mean-square (RMS) of the deviation from the mean of the output signal. Among other parameters, the amount of noise is dependent on the sampling frequency; often it is proportional to the square root of frequency (Jongkind, 2003). In this thesis, all measurements of a specific sensor are performed at fixed frequencies, namely 250 Hz for the accelerometer and gyroscope, and 50 Hz for the pressure sensor and magnetometer, and therefore it is assumed that the noise level is constant.

5.2.3 Pressure sensor

A pressure sensor is a simple sensor in the sense that it only measures one variable and is independent of orientation. The pressure sensor is calibrated in a pressure chamber, which is a sealed chamber in which the atmospheric pressure can be brought down to prescribed levels. The measurement model of the pressure sensor is given as follows, with pressure p , output y_p and unknown calibration parameters s_p and b_p :

$$y_p = s_p p + b_p \quad (5.7)$$

For the calibration the pressure was held constant at 23 different levels. For each pressure level the noise, quantified as the standard deviation of the output signal, was calculated. Figure 5.7 shows the noise per pressure level. From levels 1 - 18, it can be seen that there is a clear decrease in noise with pressure level. However, this would mean that levels 19 - 23 should have an increasing noise level. There is a certain increase with respect to level 18, and the final level 23 is almost equal to the noise of level 1. However, the intermediate pressure levels do not show an increase in noise. So, while there appears to be some correlation between noise and pressure, it cannot be stated for certain that noise decreases with pressure. The value of the RMS noise is at its maximum at 1000 hPa, around 0.7 hPa, and this value is used for all pressure measurements.

Least-squares estimation with the pressure data, gives the following estimate for s_p and b_p , with y_p in hPa:

$$s_p = 1.43195603 \quad (5.8)$$

$$b_p = 252.964339 \quad (5.9)$$

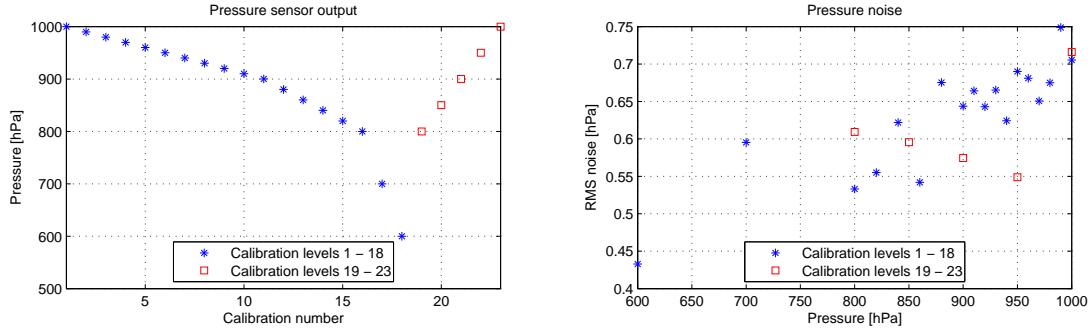


Figure 5.7: Pressure noise

5.2.4 Gyroscope

A rate gyroscope is a three-axis sensor. To calibrate it, it is mounted to a spin table, a table which can rotate at specified angular velocities, shown in figure 5.8. During mounting of the sensor on the table it is possible that misalignments are present. Therefore, the measurement equation takes the form of equation (5.3). Because the spin table can only rotate along a single axis, in order to calibrate all axes, the gyroscope should be mounted in three different orthogonal orientations to the table. This obviously means that three different, independent misalignments are present. The scale factor and offset are orientation independent.

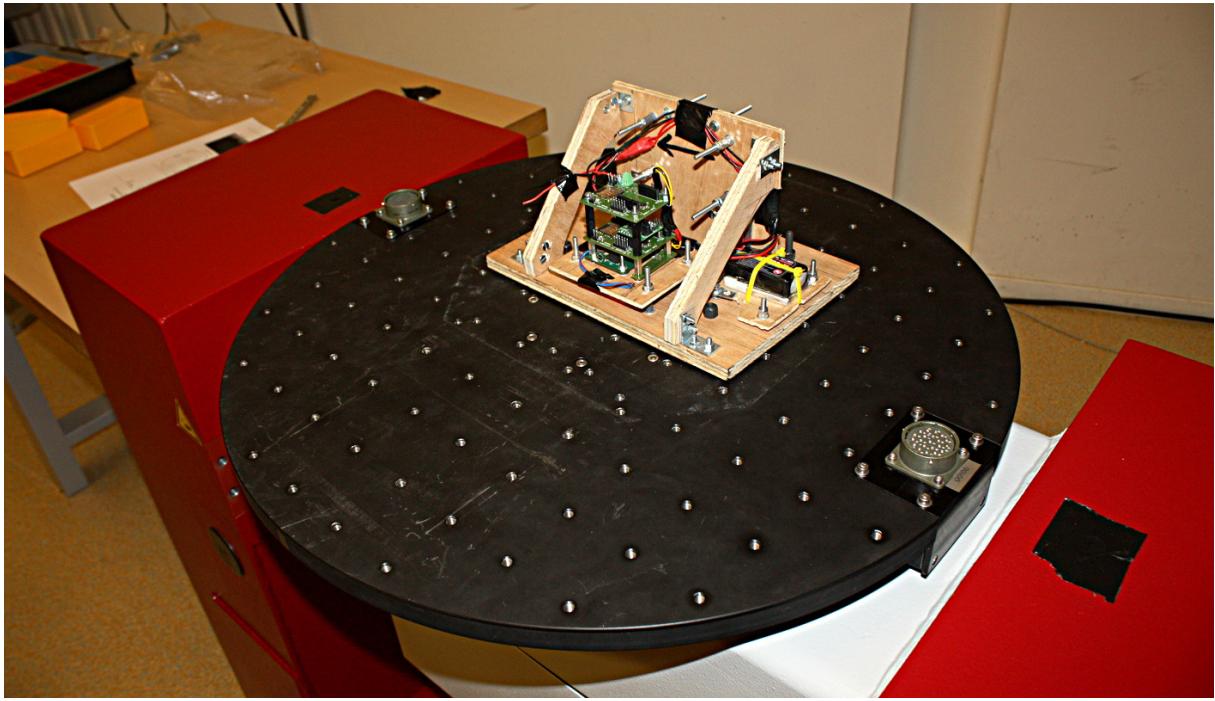


Figure 5.8: Measurement boards mounted on the spin table

Two rate gyroscopes were calibrated: gyroscope A and gyroscope B. Figures 5.9a and 5.9b show the angular rates at which the calibration was performed and the associated calculated RMS noise. It can be seen that there is an approximate linear relation between the noise and the magnitude of the angular rate. In table 5.1 the RMS noise per axis is given while the gyroscope is at rest. Furthermore, the slope of the noise with respect to the angular rate is given. Finally, this is extrapolated to the maximum angular rate of the sensor ($2000 \text{ } ^\circ/\text{s}$) to obtain the maximum RMS noise. It is specified by the manufacturer that the noise spectral density is $0.01 \text{ } ^\circ/\text{s}/\sqrt{\text{Hz}}$. With a sampling frequency of 250 Hz, this gives a specified RMS noise

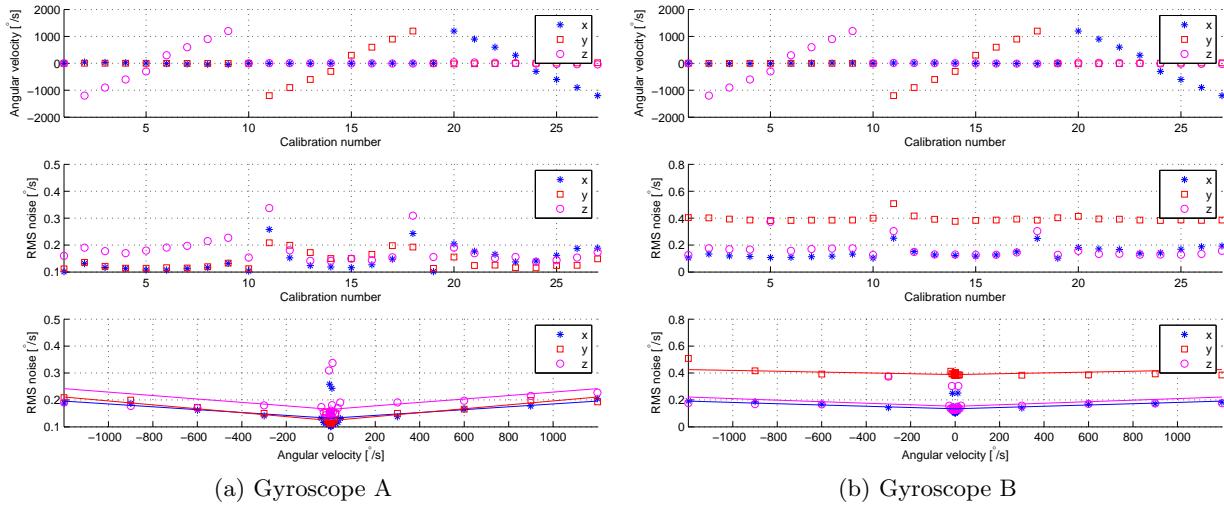


Figure 5.9: Gyroscope noise

of 0.16 °/s. In most cases the actual zero-rate noise is below this value, but it does exceed it with increasing angular rates.

Table 5.1: Gyroscope noise results

Gyroscope	Axis	Zero-rate noise [$^{\circ}/s$]	Noise slope [$10^{-5} \, ^{\circ}/s/(^{\circ}/s)$]	Maximum noise [$^{\circ}/s$]
A	x	0.131899	5.31757	0.2383
A	y	0.123477	7.26549	0.2688
A	z	0.165149	6.40807	0.2933
B	x	0.134023	4.75758	0.2292
B	y	0.388526	3.05786	0.4497
B	z	0.152494	5.74589	0.2674

Least-squares estimation is performed on the measurements with a 15-element vector of unknowns, containing the unknown three scale factors, three MRP sets for the three misalignments and three offsets. The resulting estimations are shown in table 5.2. The estimated scale factors

Table 5.2: Gyroscope calibration results

Gyroscope	s_x [LSB/(°/s)]	s_y [LSB/(°/s)]	s_z [LSB/(°/s)]	b_x [°/s]	b_y [°/s]	b_z [°/s]
A	16.1143559	16.2446792	16.3409995	-88.6253599	-54.3481712	107.253512
B	16.2816967	16.3578908	16.2007940	-30.5898529	40.2244492	-16.4907013
Specified	16.4	16.4	16.4	0	0	0

Table 5.3: Gyroscope mounting misalignment

Gyroscope	Misalignment 1 [°]	Misalignment 2 [°]	Misalignment 3 [°]
A	2.144	22.34	2.682
B	1.721	20.12	1.709

and offset are reasonably close to the specified ones. It can immediately be seen that the misalignment is small for the first and last orientation, but is rather large for the second. There is

also some mutual misalignment between the two gyroscopes. Despite this mutual misalignment, independent estimation of the misalignment gives same order of magnitude for the misalignment.

5.2.5 Accelerometer

Similar to a gyroscope, an accelerometer is a three-axis sensor. However, as described in section 4.2.1, an accelerometer measures specific force, not acceleration directly. The specific force is the acceleration minus the gravity. If both the gravity vector \mathbf{g}_C and the acceleration vector are defined in the calibration-fixed reference frame, then the measurement equation for a misaligned accelerometer is:

$$\mathbf{y}_a = \mathbf{SC}(\mathbf{a} - \mathbf{g}_C) + \mathbf{b}_a \quad (5.10)$$

The presence of the gravity in the equation can be used to calibrate the accelerometer without actually subjecting it to accelerations. Let \mathbf{C}_{CV} be a transformation matrix from a vertical reference frame to a calibration-fixed reference frame, and be fully known. Then calibration can be performed by subjecting the sensor to different orientations, giving the following measurement equation:

$$\mathbf{y}_a = \mathbf{SCC}_{CV}\mathbf{g}_V + \mathbf{b}_a \quad (5.11)$$

The calibration is performed such that one of the axes is perpendicular to the gravity vector, and the accelerometer is subjected to different orientations by rotating around this axis. Then, to be able to also calibrate this rotation axis, a different rotation axis is chosen about which the accelerometer is rotated. This gives rise to two possible misalignments in the estimation vector.

The measured RMS noise of the accelerometer is given in Figure 5.10. The noise is normally constant around 0.15 m/s^2 , but there are some peaks up to 0.31 m/s^2 . There is no clear observable relation between the peaks and the specific force on a specific axis. In any case, the noise is much larger than specified by the manufacturer. The specified spectral noise density is $150 \mu\text{g}/\sqrt{\text{Hz}}$, which leads to an RMS noise of 0.0510 m/s^2 .

The observed noise is at least three times higher than this specified value. There could be several reasons for this. As stated in section 5.1.3, the accelerometer had to be connected with wires to the measurement system, due to manufacturing problems. This may have caused noise on the power lines, which may propagate to the measurement output. Other reasons could be that the specific accelerometer is simply a badly manufactured sample, or that the manufacturer overstates the noise performance. To determine which of these is the real reason for the bad noise performance, the measurement system should be rebuilt, with new sensors. Then measurement of the noise of this new system can confirm or reject any of these hypotheses.

The estimated scale factor and offset and misalignment can be found in table 5.4 and 5.5 respectively. As can be found in the table, the obtained value for the scale factor and offset are close to the specified values.

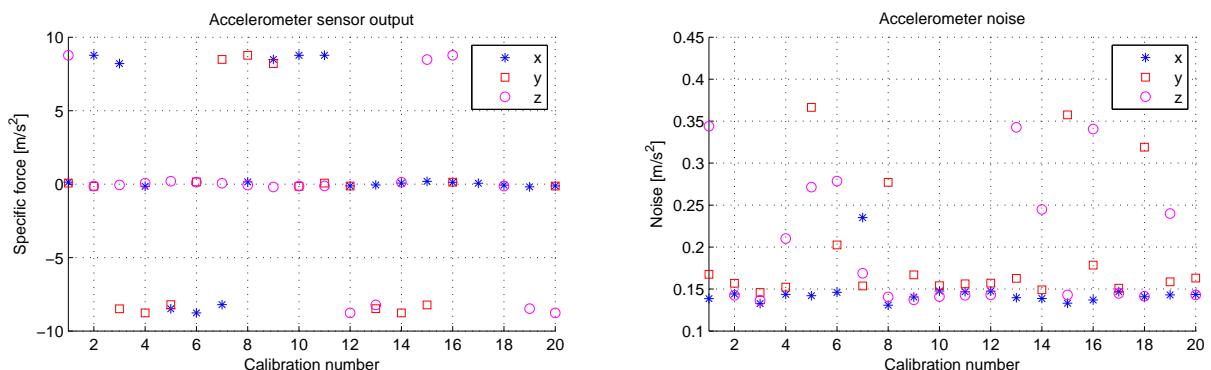


Figure 5.10: Accelerometer noise

Table 5.4: Accelerometer calibration results

	s_x [LSB/(m/s ²)]	s_y [LSB/(m/s ²)]	s_z [LSB/(m/s ²)]	b_x [m/s ²]	b_y [m/s ²]	b_z [m/s ²]
Actual	211.136949	199.110221	213.491777	-54.8747397	-82.840354	-52.5523516
Specified	208.77	208.77	208.77	0	0	0

Table 5.5: Accelerometer mounting misalignment

Orientation	Misalignment [°]
1	1.379
2	1.275

Chapter 6

Verification and validation

In this thesis, three systems are developed: simulator, measurement system and estimator. For all these, designs have been made, after which they have been implemented. Once they are implemented, it must be *verified* that the implementation corresponds to the design. Furthermore, it must be *validated* that the design and implementation correspond to the reality that they must represent. The goal of verification and validation is to gain confidence that the obtained result is a valid result. Therefore, this chapter deals with verification and validation of these three elements.

Verification is primarily performed on two levels: on unit level and on system level. With *unit testing*, the smallest possible element of the system is subjected to an input, and the output is compared with the desired output. With *system testing*, the entire system or program is subjected to an input, and the output is compared with the desired output. Obviously, testing is not limited to these two extremes; any subsystem or group of units can be tested this way as well. Validation is performed on the entire system. This should typically be done against existing systems, for which for a given input it is known that the output is valid.

6.1 Simulator

As described in chapter 2, two simulators are in use: ROSIE and the simplified simulator model. ROSIE has been extensively verified in Engelen, 2012. However, as described there, validation of it is lacking. It was only validated against a small amateur rocket, with noisy sensors. The conclusion was that this validation was insufficient to declare the simulator fully validated. Fortunately, for the purpose of this thesis, a very accurate correspondence of the simulation with a real flight is not required. In the evaluation of the navigation system, the response of the navigation system to various flight trajectories will be determined. A proper navigation system is able to obtain a position and state estimate for a wide variety of flight trajectories, as it is certain that an actual flight will not correspond to a nominally simulated flight. For these reasons, it is of less importance that the simulated flight exactly corresponds with the real flight, than that the flight characteristics are indicative for the flight.

For the purpose of measurement generation, functionality has been added to ROSIE. These measurement generation routines have been verified in the following way:

- Measurements with, and without noise have been generated and differenced. From this difference the mean and standard deviation have been determined. The mean should be zero, and the standard deviation equal to the specified noise level. Furthermore, distribution plots have been created and compared with those of a normal distribution, to validate that the noise is indeed normally distributed. Figure 6.1 shows the distribution of the accelerometer noise.
- The frequency domain of the measurements has been inspected using a discrete Fourier

transform. This has been performed with MATLAB's `fft` function. For the gyroscope the measurement output in the frequency domain has been plotted in figure 6.2. For all frequencies, the amplitude of frequency component is approximately equal, as is expected for white noise.

- For the gyroscope measurements, the sign of the y- and z-components of the measurements have been determined. The required sign of these measurements can be determined intuitively for a gravity turn, and it has been verified that they do indeed equal each other.
- For the pressure measurements, the measurement equation has been inverted to obtain a relation of altitude according to pressure versus time, and this have been verified to show good correspondence with the altitude according to the simulator.

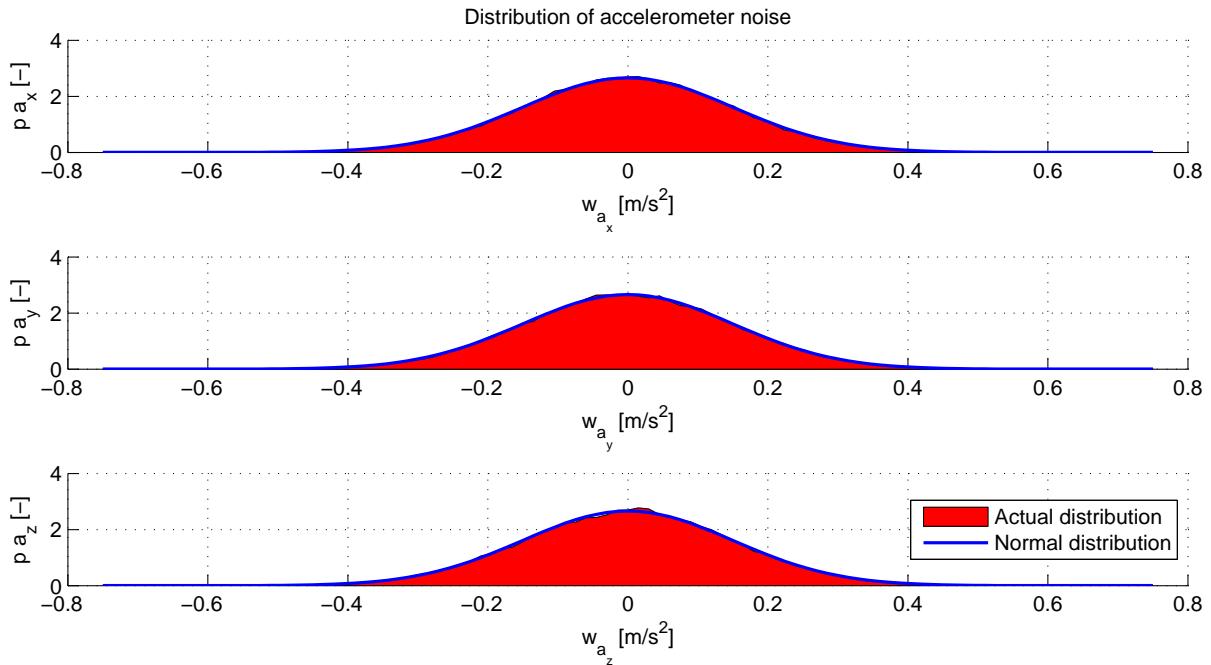


Figure 6.1: Distribution of accelerometer noise

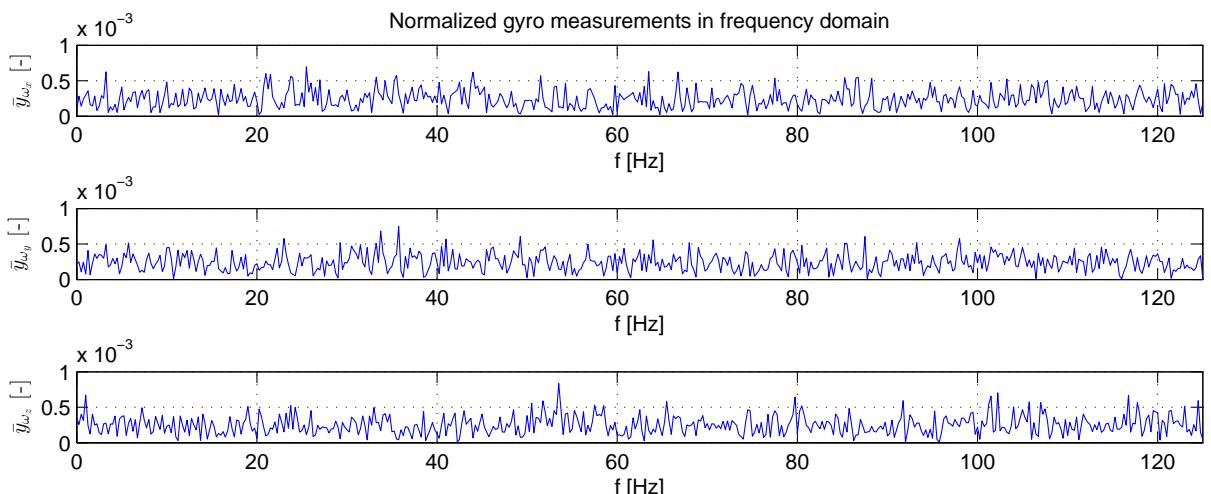


Figure 6.2: Gyroscope measurements in the frequency domain

The simplified simulator implementation has been verified against ROSIE. The following verifications on state derivatives against ROSIE have been performed:

- Gravitational force has been compared against ROSIE gravitational force. As both simulators use the same gravity model, the gravitational force before flight and after motor burn should be equal. During motor burn the gravitational forces are not equal, due to differing mass models. It has been verified by inspection that the gravitational force during motor burn is always between that of an empty and full rocket.
- Thrust force has been verified by inspection of its magnitude versus time.
- Aerodynamic forces have been verified by comparing them with aerodynamic forces generated by ROSIE. The aerodynamic forces are not equal due to the simplified model used for the aerodynamics, but do correspond relatively closely.

Furthermore, attention has been paid to MRP verification and validation. The MRP propagation has been validated by converting the MRP before and after propagation to a quaternion, and comparing with quaternion propagation by ROSIE. MRP direction cosine matrix generation has been verified by converting the MRP to a quaternion, and using ROSIE to generate a DCM of the quaternion. Finally, the shadow transform has been validated by generating DCMs both from the regular MRP and the shadow MRP, and comparing them.

System level checks have not been performed extensively. The output of the simplified simulator has been compared to the ROSIE output for a non-rotating Earth, and no-wind environment, and the results correspond reasonably, except on points where differences can be expected. These differences are, among others: differing apogee altitudes and velocity development during motor burn due to simplifications in the thrust profile; differing angular velocities due to simplified moment models and moments of inertia.

6.2 Measurement system

Verification and validation of the measurement system can be performed both on hardware and software level. During the design of the hardware, it can already be partially verified that the design corresponds to the requirements. The design program *Altium Designer* aids with this via *design rules*. Essentially, in during the schematic design, it should be specified which pins of various components should be connected together. Then, during PCB design, a *design rule check* (DRC) can be performed, which verifies that on the designed PCB the pins are actually connected.

The first verification step on the physical hardware is electrical testing of the PCB. This is done by the PCB manufacturer, who checks that all traces of the bare PCB are connected appropriately (Eurocircuits N.V., 2012). After mounting the components, the correct placement of the components can only be verified after the entire system is finished. Individual solder connections can be verified by measuring the resistance between the two points of the connection. This verification step has not been performed on all components, only on those for which visual inspection gave rise to doubts about the soldering quality.

The next step in verification is the testing of the onboard software. Units of the software, can be grouped into three groups:

Peripheral drivers provide an easy to use interface to the peripherals of the microcontroller.

These peripherals are I²C, RTC, SPI, timers, and UART. All these drivers have been extensively used in electronics for previous rockets of DARE, including Stratos I, and further verification in the scope of this thesis was not deemed necessary.

Sensor drivers provide a software interface to the functions and registers of attached sensors.

Register write and read routines have been tested for each sensor by writing data to several

registers, reading them out and comparing them. The following example shows this for one of the registers of the BMA180 accelerometer.

```
// Set bandwidth to 1200 Hz
data = bma180_i2c_read_reg(i, BMA180_bw_tcs, 1);
*data |= /* bw */ 0x07 << 4;
bma180_i2c_write_reg(i, BMA180_bw_tcs, *data);
if ( bma180_i2c_read_reg(i, BMA180_bw_tcs, 1) != data )
{
    // Reading register does not return the written data
    LED_R_ON();
}
```

The entire functionality of the sensor drivers has been tested by using the sensor drivers to obtain measurements from the sensor.

Main is the routine that implements the software flow from figure 5.6. The units of this diagram that do not fall in either of the two categories above have not been tested individually, rather, extensive testing has been performed by running the implemented software, and checking the data on the SD card.

The purpose of the measurement system was to measure the flight characteristics that the system was subjected to. The primary validation for this has been the calibration. During the calibration, it was validated that the system indeed measures the accelerations, angular velocities and atmospheric pressure it is subjected to. However, two important parts were omitted from validation: the magnetometer was not validated at all, and the data storage was validated only partially. After flight, it appeared that the output of the magnetometer did not have a constant norm, rendering it unusable. This is obviously a defect that could have been detected with validation, showing the importance of pre-flight validation. For the data storage, it was verified that any data written to the SD card could be read by the computer and was equal to the written data. It was however not validated that the frequency with which measurement data was written to the SD card was equal to the measurement frequency. After flight, it proved to be the case that the SD card was too slow to keep up with the measurement frequency, occasionally leading to several milliseconds of lost measurement samples.

6.3 Estimator

The verification and validation of the estimators can be divided into two parts: verification of the estimator algorithm, i.e., the generic EKF, and verification and validation of the models used. To verify the estimator algorithm, the *falling body test* was performed, which is described in section 6.3.1. Beside this system level test, several unit tests were performed for **Measurement** objects, to check if the input, i.e., measurement data, would be obtained via the get-functions. Since at this point already successful system level tests were performed, it was deemed that no further unit level testing would be required. Verification of the models is described by section 6.3.2. Finally, testing of the entire system is performed in section 6.3.3.

6.3.1 Falling body estimation

Verification of the EKF and UKF has been performed using examples provided by Simon (2006). Here, an example problem where a falling body is tracked by a radar is posed, for which the results have been reproduced as verification. The falling body is described by a system of three

differential equations, with x_1 the altitude, x_2 the velocity and x_3 ballistic coefficient as follows:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{1}{2}\rho_0 e^{-\frac{x_1}{k}} x_2^2 x_3 - g \\ 0 \end{bmatrix} \quad (6.1)$$

where the coefficients ρ_0 , k and g have the following values, in imperial units:

$$\begin{aligned} \rho_0 &= 2 \text{ lb} \cdot \text{s}^2/\text{ft}^4 \\ g &= 32.2 \text{ ft/s}^2 \\ k &= 20000 \text{ ft} \end{aligned} \quad (6.2)$$

The initial conditions are:

$$\mathbf{x}_0 = [300000 \quad -20000 \quad 0.001]^T \quad (6.3)$$

This model represents a body of constant drag coefficient, subject to a constant gravity field, falling in an exponential atmosphere. This model has been simulated using a forward Euler numerical integrator, with a step size of 0.001 s. The altitude and velocity were given by Simon (2006) as in figure 6.3a, while the results from the simulation are given in figure 6.3b. Visibly, there is no difference between the two plots.

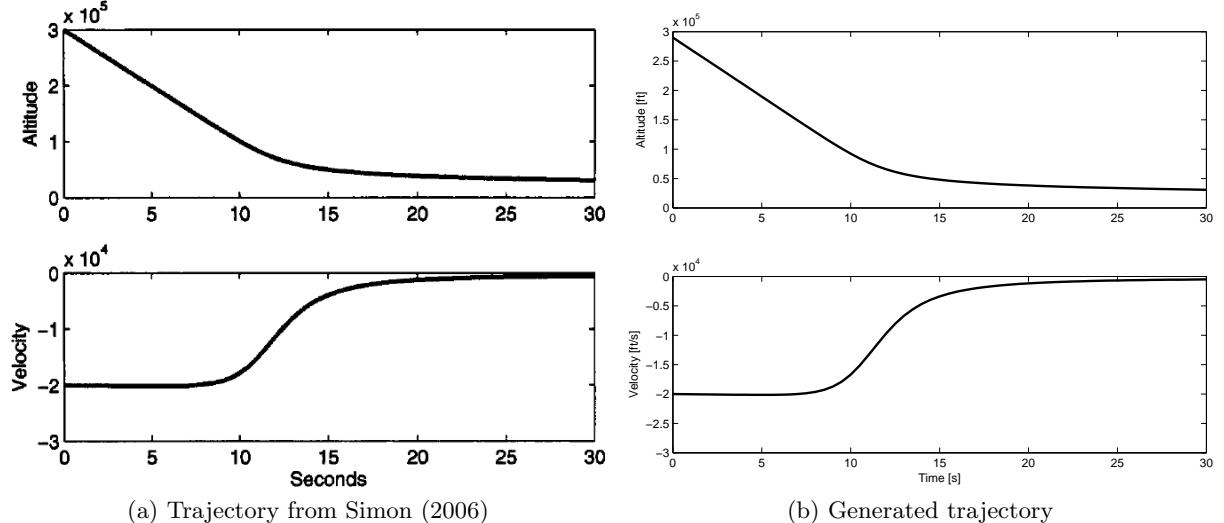


Figure 6.3: Simulated trajectory of a falling body

The simulated trajectory is used to generate measurements. The measurements are distance measurements obtained from radar tracking, given by the following equation:

$$y = \sqrt{M^2 + (x_1 - a)^2} \quad (6.4)$$

where M is the horizontal distance to the falling object, and a the altitude of the radar station. Both quantities are constant, and equal to 10000 ft. A zero-mean, Gaussian distributed noise with variance equal to 10000 ft² is added to the measurements. These measurements are used as input for both the EKF and the UKF. As described in section 3.2, for the EKF, the partial derivatives of the state transition function, and the measurement equation are required. They

are given by:

$$\mathbf{A} = \frac{\partial \dot{\mathbf{x}}}{\partial \mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 \\ \frac{1}{2} - \frac{\rho_0}{k} e^{-\frac{x_1}{k}} x_2^2 x_3 & \rho_0 e^{-\frac{x_1}{k}} x_2 x_3 & \frac{1}{2} \rho_0 e^{-\frac{x_1}{k}} x_2^2 \\ 0 & 0 & 0 \end{bmatrix} \quad (6.5)$$

$$\mathbf{F}_{k-1} = \mathbf{I} + \mathbf{A}(t_k - t_{k-1}) \quad (6.6)$$

$$\mathbf{H} = \frac{\partial y}{\partial \mathbf{x}} = \begin{bmatrix} \frac{x_1 - a}{\sqrt{M^2 + (x_1 - a)^2}} & 0 & 0 \end{bmatrix} \quad (6.7)$$

The initial conditions of the estimator are given by:

$$\hat{\mathbf{x}}_0 = \mathbf{x}_0 \quad \mathbf{P}_0 = \begin{bmatrix} 1000000 & 0 & 0 \\ 0 & 4000000 & 0 \\ 0 & 0 & 10 \end{bmatrix} \quad (6.8)$$

The measurements are generated every 0.5 s. Then, finally the estimation can be performed. Figure 6.4a shows the estimation errors according to Simon (2006), while figure 6.4b shows the estimation errors according to the implemented EKF and UKF. The estimation errors are not exactly the same, but this is not expected, as there is a large stochastic effect involved, by the addition of noise to the measurements. In particular, the implemented EKF performs slightly better than the example version, but its performance is still worse compared to the UKF, as expected. Also, it appears that Simon (2006) did not generate measurements at $t = 0$ s, which may explain the difference between the two plots for the ballistic coefficient. The plots do show the same orders of magnitude for the estimation errors, and roughly the same evolution of the errors in time, and thus it can be concluded that the implementation of the EKF and UKF are performed properly.

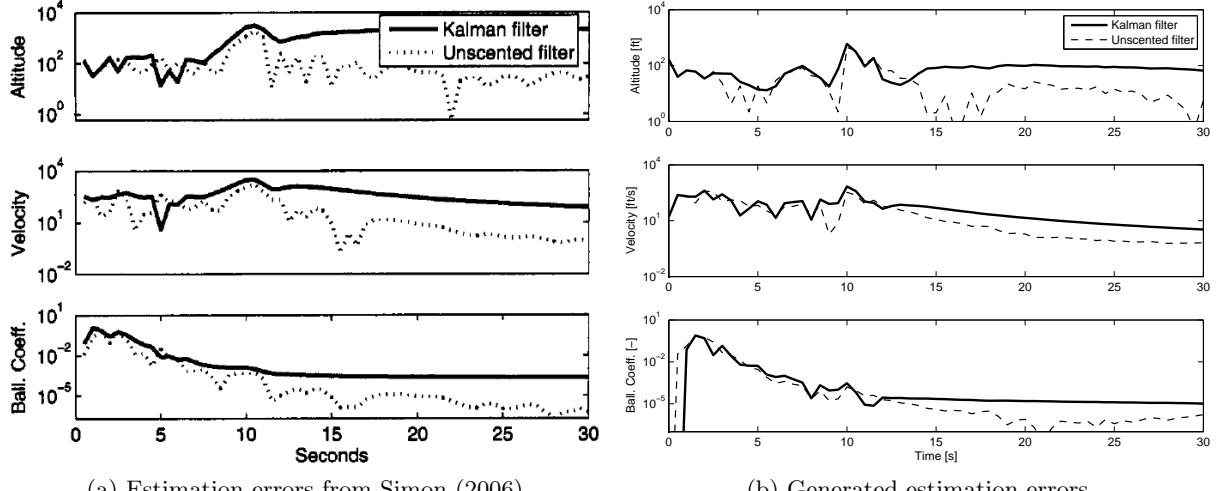


Figure 6.4: Estimation errors of the trajectory of a falling body

6.3.2 Model testing

For the models, the primary testing was done on the partial derivatives. Validation of the partial derivatives is performed by comparing them with numerical partial derivatives. A central difference method of 8th order is used. In scalar form, the derivative of $f(x)$ with respect to x ,

with a small value of h , is given by (Fornberg, 1988):

$$\frac{df}{dx} = \frac{1}{h} \left[\frac{1}{280} f(x - 4h) - \frac{4}{105} f(x - 3h) + \frac{1}{5} f(x - 2h) - \frac{4}{5} f(x - h) + \frac{4}{5} f(x + h) - \frac{1}{5} f(x + 2h) + \frac{4}{105} f(x + 3h) - \frac{1}{280} f(x + 4h) \right] \quad (6.9)$$

The analytical partial derivatives given in chapter 4 are verified against their numerical variants given by this equation. Figure 6.5 shows these differences for the partial derivatives of the MRP time derivative, for which the analytical expressions are given in appendix B.3.2. Clearly, the differences in the order of 10^{-15} or smaller show that the analytical and numerical match.

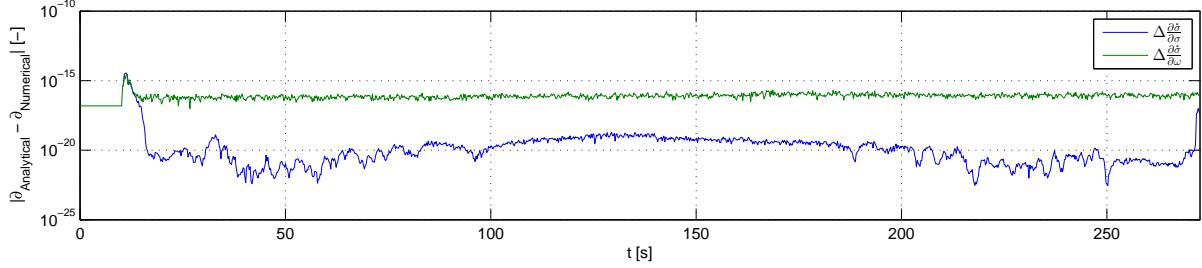


Figure 6.5: Difference between analytical and numerical partial derivatives of MRP time derivative

6.3.3 System testing

For the model-free estimator, an estimation with accelerometer and gyroscope with no noise is performed. Theoretically, this would yield an estimation that is exactly equal to the simulation. In practice, numerical noise such as round-off and integration errors yield a minimum value for the estimation error. Still, it is found that the estimation errors always stay below 0.1 %. As example of this behavior, in figure 6.6, the attitude estimation error with noise free measurements has been plotted. Here, beside a small peak during the motor burn phase, the attitude error stays below 0.01° , an extremely small value.

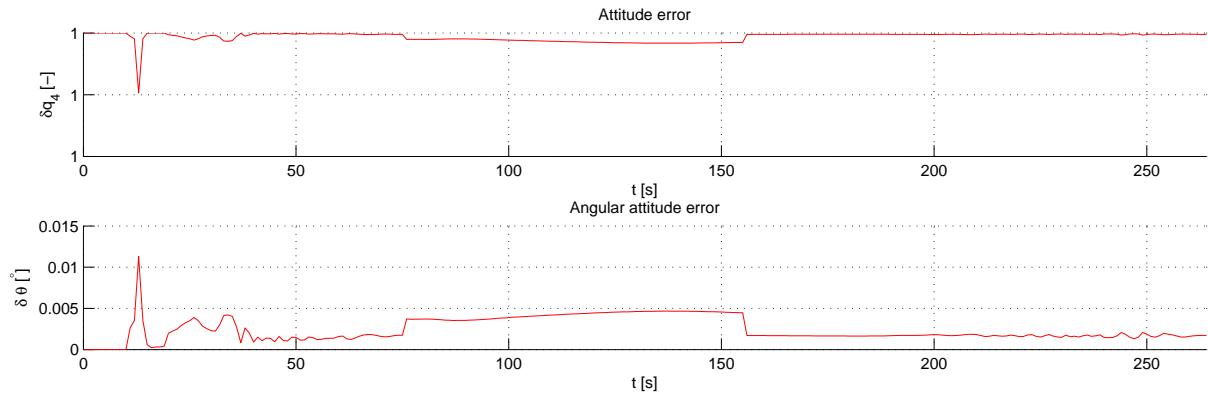


Figure 6.6: Attitude estimation error with noise-free accelerometer and gyroscope measurements

Chapter 7

Stratos II simulation and estimation results

In previous chapters the mathematical background of simulation and estimation, the actual implementation of simulator and estimator model, and how to obtain measurements in real-life have been covered. These topics come together in this chapter, where the results of flight simulation and the estimator results based on simulation based on flight measurements will be discussed.

In the development of the flight estimator, there are several factors that influence the estimation result. Not only the estimator model and the filter type affect the result, but also factors such as the trajectory, and the sensor characteristics – both controllable and noise factors. To be able to evaluate the performance of the estimators, performance measures are established. First however, simulation of the nominal flight is performed, to which the estimated trajectory can be compared. Then, using measurement data generated from this nominal flight, the estimators covered in previous chapters are evaluated. A single estimator type is selected for which further analysis is performed. This analysis includes the effect of varying sensor factors, sensor failure and non-nominal flight.

7.1 Estimator performance evaluation

In this chapter, for many combinations of sensor characteristics and flight profiles, the estimated trajectory will be compared with the actual (simulated) trajectory. To perform this comparison, it must be quantified how well the estimator performs. For this purpose, several performance measures are established, and characteristic points in the flight are chosen where these performance measures are evaluated.

It is possible to directly compare the estimated and actual state vector. However, it is much more insightful to segment or transform the state vector into quantities, which have a direction physical meaning. For all of the characteristic points the following performance measures are evaluated:

Time error is the difference between the time the specified characteristic flight point occurred, and the time the point occurred according to the estimator.

Position error is the magnitude of the vector difference of the actual position and the estimated position.

Altitude error is the difference between actual and estimated altitude at the specified event.

Velocity error is the magnitude of the vector difference of the actual velocity and the estimated velocity.

Attitude error is the rotation angle between the actual and estimated body frame, as defined in section 2.2.4.

Elevation error is the difference in elevation angle.

Azimuth error is the difference in azimuth angle.

Angular velocity error is the magnitude of the vector difference of the actual angular velocity and estimated angular velocity. This quantity is only evaluated for estimators that estimate the angular velocity.

Furthermore, for all these quantities, except the time error, the maximum and mean error are found, and the standard deviation around this mean.

The characteristic flight points are selected based on sharp or discrete changes in the flight profile, or on important decision points. There are six characteristic points, of which three are decision points. The decision points are as follows:

Stage separation At stage separation, or very shortly afterwards, the decision will be made to ignite the second stage. The second stage will only be ignited when the altitude and elevation angle are within acceptable range. Therefore, it is of utmost importance that those quantities are estimated properly at this point. This applies to nominal flight, but equally to a non-nominal flight which would cause the rejection of the second stage ignition.

Nose cone separation The nose cone should be separated from the motor during the descent, when it has crossed a certain altitude.

Parachuting After nose cone separation, again when the nose cone has crossed a certain altitude, a decision point exists to deploy the parachute. Due to constraints on the parachute shock (which increases with dynamic pressure), there could be tighter velocity or altitude estimation error requirements for the parachute separation than for nose cone separation.

The other characteristic points are:

Lift-off The point just before lift-off is selected, to verify the reference state. If the estimated state already differs significantly from the actual state at lift-off, chances are great that the entire estimation is worthless.

Second stage burn-out The point of second stage burn-out is when the rocket will have the highest velocity, and thus when the dynamics of the flight are most severe.

Apogee In contrast, apogee is the point in flight where a local minimum in dynamics will be reached. The apogee altitude is mostly of interest for post-flight operations, and public relation purposes.

7.2 Experimental design

The design of the measurement system and estimator are based on experiments performed by computer simulation of the flight. Therefore, this design process is called *experimental design*. In the description of experiments, several definitions are used, which are defined in this section. Furthermore, section 7.2.1 describes experimental design using orthogonal design, a method to limit the number of experiments required. Section 7.2.2 describes the response surface, a way to analyze the results of the experiments.

During experimental design, several *factors* are varied, and their influence on the *response* of the system is studied. Examples of factors could be sensor noise or measurement frequency. The factors can be subdivided into two categories: firstly, the *control factors*, which are factors which

can be changed by the designer, and *noise factors*, which cannot be changed by the designer. In the experimental design performed in this thesis, the control factors are some of the sensor characteristics. The noise factors consist of the other sensor characteristics, and actual flight characteristics. The performance measures established in section 7.1 are the response.

7.2.1 Orthogonal arrays

If the number of factors is small, it is possible to perform experiments with all possible combinations of factors. This is called a *full factorial design*. The number of experiments is given by the following equation, with n_e the number of experiments, n_l the number of levels and n_f the number of factors:

$$n_e = n_l^{n_f} \quad (7.1)$$

As the number of experiments is exponential with the factor, full factorial design is only possible with a small number of factors. Tachugi proposed an alternative, where experiments would be performed according to an orthogonal array, a $k \times n_f$ -matrix, with k much smaller than n_e . Each row of the matrix corresponds to an experiment, with columnwise the factor level for corresponding factors. For example, in table 7.1, an orthogonal array of four experiments and three factors is shown. For the first experiment, factor A is set to -1, factor B to -1 and factor C to -1. For the second experiment, factor A is set to -1, factor B to 1 and factor C to 1. The orthogonal array has as property that for any pair of columns, all combinations of factor levels occur an equal number of times (Mooij, 1998). This so-called *balancing property* has as a consequence that the response to the orthogonal array can be used to construct a reasonable approximation to response over the entire region that the orthogonal array spans.

Table 7.1: Orthogonal array of 4 runs and 3 factors

Experiment	A	B	C
1	-1	-1	-1
2	-1	1	1
3	1	-1	1
4	1	1	-1

7.2.2 Response surface

To analyze the result of a number of experiments, it is possible to obtain a response surface of the experiments. In essence, a response surface is a multi-dimensional polynomial fit through the responses, with the factors as independent variables. For the experiments performed in this thesis, a first order polynomial will be fitted through the result. For a response r to n factors $f_1 \dots f_n$, the response surface has m multiplicative coefficients $\beta_1 \dots \beta_n$, and a coefficient β_0 , yielding the following equation for the response (Mooij, 1998):

$$r = \beta_0 + \beta_1 f_1 + \dots + \beta_n f_n \quad (7.2)$$

If k experiments are performed, then m responses from r_1 to r_m are available, with for each r_i known factors f_{ij} . In matrix-vector form this gives:

$$\begin{bmatrix} r_1 \\ \vdots \\ r_m \end{bmatrix} = \begin{bmatrix} 1 & f_{11} & \cdots & f_{1n} \\ \vdots & & & \vdots \\ 1 & f_{m1} & \cdots & f_{mn} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_n \end{bmatrix} \quad (7.3)$$

or, equivalently:

$$\mathbf{r} = \mathbf{F}\mathbf{b} \quad (7.4)$$

Estimating the unknown \mathbf{b} is then a least-squares problem, as described in section 3.1. The estimate $\hat{\mathbf{b}}$ is given by:

$$\hat{\mathbf{b}} = (\mathbf{F}^T \mathbf{F})^{-1} \mathbf{F}^T \mathbf{r} \quad (7.5)$$

The corresponding residual $\boldsymbol{\varepsilon}$, and RMS residual ε_{RMS} , and normalized residual $\bar{\varepsilon}_{RMS}$ are given by:

$$\boldsymbol{\varepsilon} = \mathbf{r} - \mathbf{F}\hat{\mathbf{b}} \quad (7.6)$$

$$\varepsilon_{RMS} = \sqrt{\frac{\|\boldsymbol{\varepsilon}\|^2}{m}} \quad (7.7)$$

$$\bar{\varepsilon}_{RMS} = \frac{\varepsilon_{RMS}}{r_{nom}} \quad (7.8)$$

Each of the coefficients β_j is then an indication of the contribution of factor f_j to the response, with β_0 the mean response. The process described above, is nothing more than a linearization of the response around the mean response. In practice, it might be more useful to obtain the linearization around some known response, such as the nominal response r_{nom} . This can be done modifying equation (7.3) to:

$$\begin{bmatrix} r_1 - r_{nom} \\ \vdots \\ r_m - r_{nom} \end{bmatrix} = \begin{bmatrix} f_{11} & \cdots & f_{1n} \\ \vdots & & \vdots \\ f_{m1} & \cdots & f_{mn} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix} \quad (7.9)$$

The coefficients β_j then provide the linearization around r_{nom} , and β_0 is absent from the equation.

7.3 Nominal trajectory simulation

In this section, the flight trajectory of the Stratos II using ROSIE and the model described in section 2.5.1 is described. The development and evaluation of the estimators is in principle performed with the nominal flight trajectory. However, the nominal model does not cause the rocket to roll, whereas in practice it has been observed that due to imperfections in the rocket, the rocket will almost certainly spin up during flight. Therefore, two nominal trajectories are generated: the nominal trajectory for an ideal rocket, which is a rocket that does not exhibit spin, and a nominal non-ideal rocket flight, where the rocket does exhibit spin.

Figure 7.1 shows the altitude and ground track of both nominal flights. The rocket will reach an altitude of 58 km in approximately 130 s, and then descents until 260 s after launch. It will have traveled 100 km northwards and 50 km westwards by then. The roll has almost no influence on the position of the rocket. The velocity of the rocket is shown in figure 7.2, along with the Mach number in figure 7.3. The stage burnout can easily be identified as the velocity peaks around 500 m/s and 1400 m/s respectively. During first stage burn, the rocket will cross into supersonic flight, while during its first coasting flight it will get below the sonic boundary again, and then go supersonic during second stage burn, reaching a peak Mach number of almost 5 at second stage burnout. During the entire coasting flight, it will continue to stay in the supersonic region, and near the end of flight it will get into the subsonic regime again.

The attitude of the rocket and the angular velocity are plotted in respectively figure 7.4 and 7.5. The rocket flies a mostly stable flight, as the low angular velocity components along the y- and z-body axes indicate. A small angular motion can be observed near the beginning of the flight. This coincides with the rocket leaving the launch tower, and physically corresponds to it

weathercocking into the wind. If a roll coefficient is present, then the rocket will reach a spin rate of about 1.5 rotations per second (9 rad/s), otherwise, there will be virtually no angular velocity around the x-axis.

Finally, in figure 7.6, the measurements that have been generated based on the simulation of the ideal rocket has been plotted. In the accelerometer plots it is clear that the dominating acceleration is the x-component, where the two stage burns can be clearly distinguished. The accelerometer measurement is negative when drag is dominating, and it can be seen that drag starts to become less and less significant during the second coast phase, until well into the descent. The gyroscope repeats the angular velocity plots, only with the noise visible. In the magnetometer measurements, it can be observed that the x-component varies during the flight from 5×10^4 G to -5×10^4 G, and that the other two components start at almost zero value. This behavior will become important in later sections, when the gyroscope bias will be estimated. The last plot shows the pressure measurements. Pressure measurements are only generated when the rocket is in subsonic flight, which can be observed from the gaps in measurement data. From this data, already some conclusions on the previously designed measurement system can be made. First, the accelerations the rocket reaches, are outside the range of the accelerometer, as they reach magnitudes of almost 200 m/s^2 , or 20 g, whereas the range of the accelerometer was $-/+ 16 \text{ g}$. For the gyroscope the angular velocities of the rocket clearly lie well within the range of the measurement system, although it should be noted again that an ideal rocket, such as simulated, does not exhibit roll motion, whereas an imperfectly manufactured rocket does.

7.4 Model and estimator type

In this thesis, several models and estimators have been covered. Those were the numerical models derived from ROSIE, the analytical model derived from the simplified simulator model and the kinematic model. They can be used in conjunction with both the extended Kalman filter (EKF) and the unscented Kalman filter (UKF) and the Rauch-Tung-Striebel (RTS) smoother.

First, the computational requirements of these estimators will be analyzed in section 7.4.1. Then, the estimator performance of the simplified and kinematic model will be analyzed in sections 7.4.2 and 7.4.3 respectively. The estimation processes will be performed against the nominal flight. As was stated in section 7.3, both the nominal flight for an ideal and non-ideal

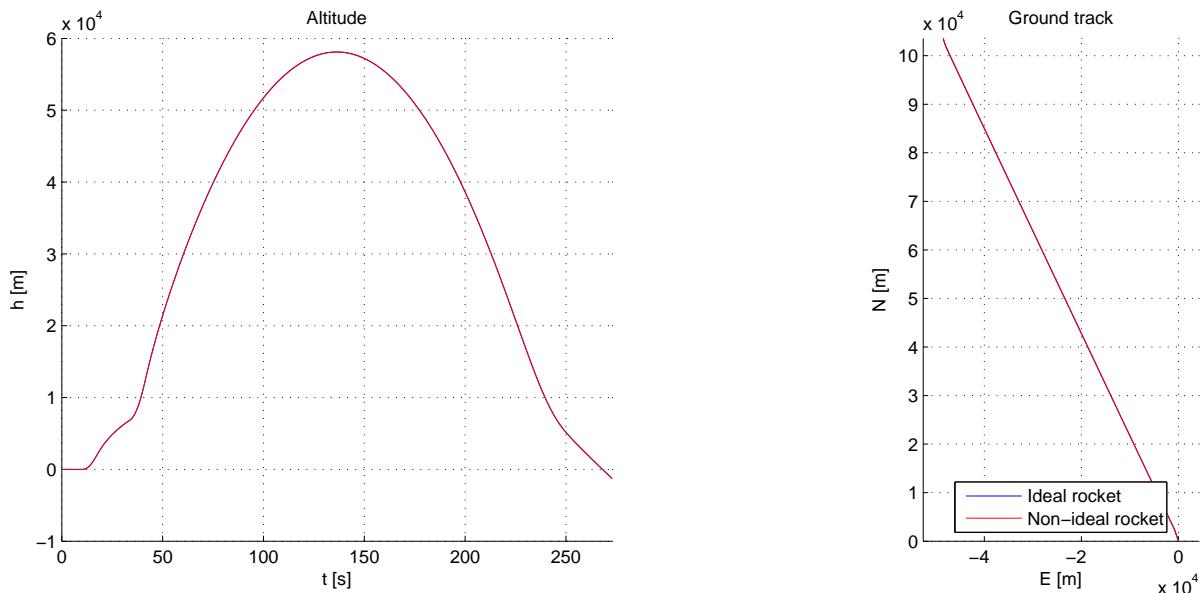


Figure 7.1: Nominal flight trajectory

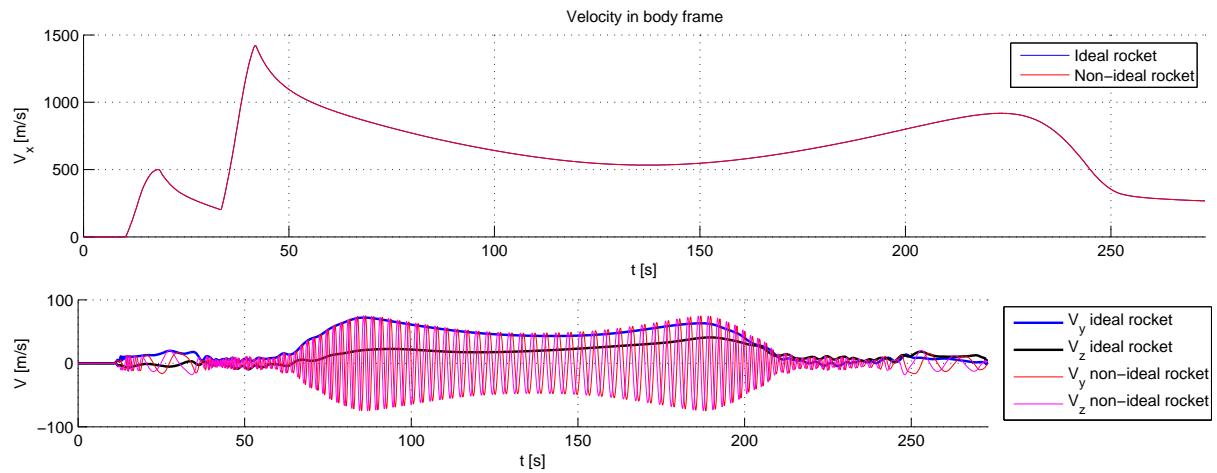


Figure 7.2: Nominal velocity

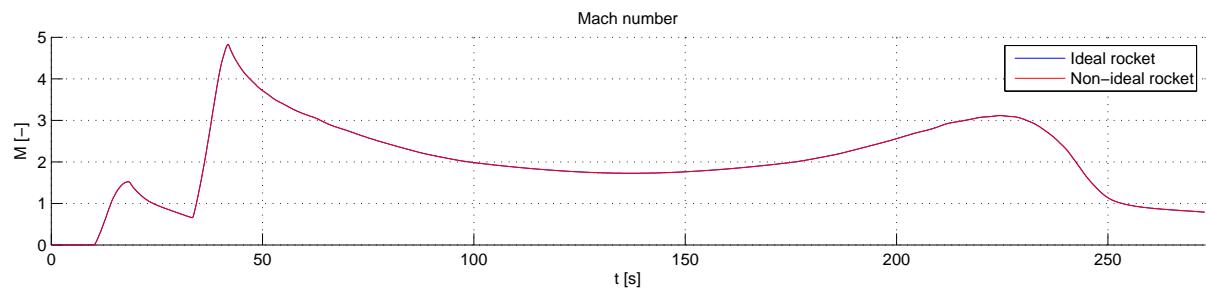


Figure 7.3: Nominal Mach number

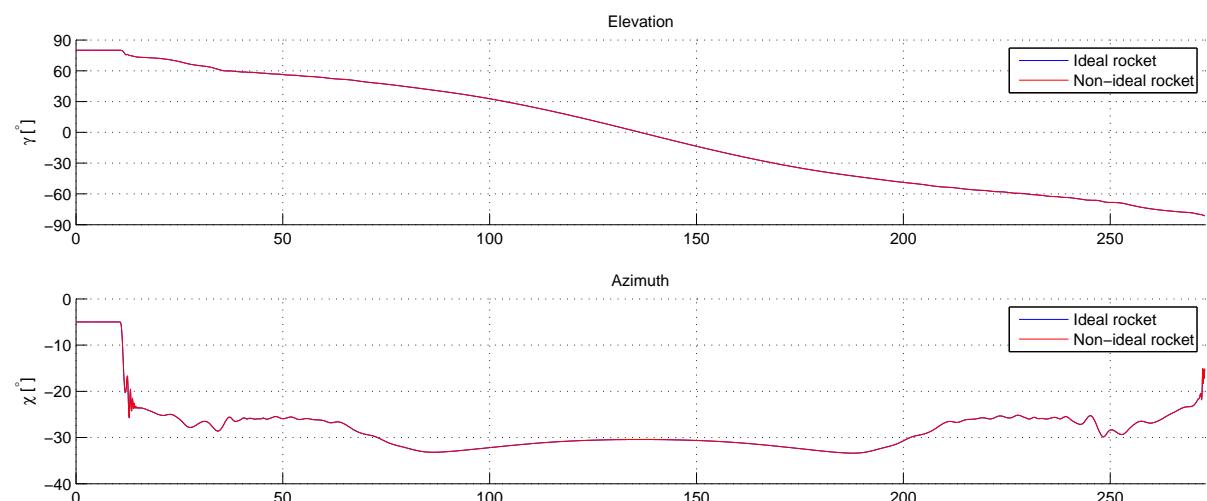


Figure 7.4: Nominal attitude

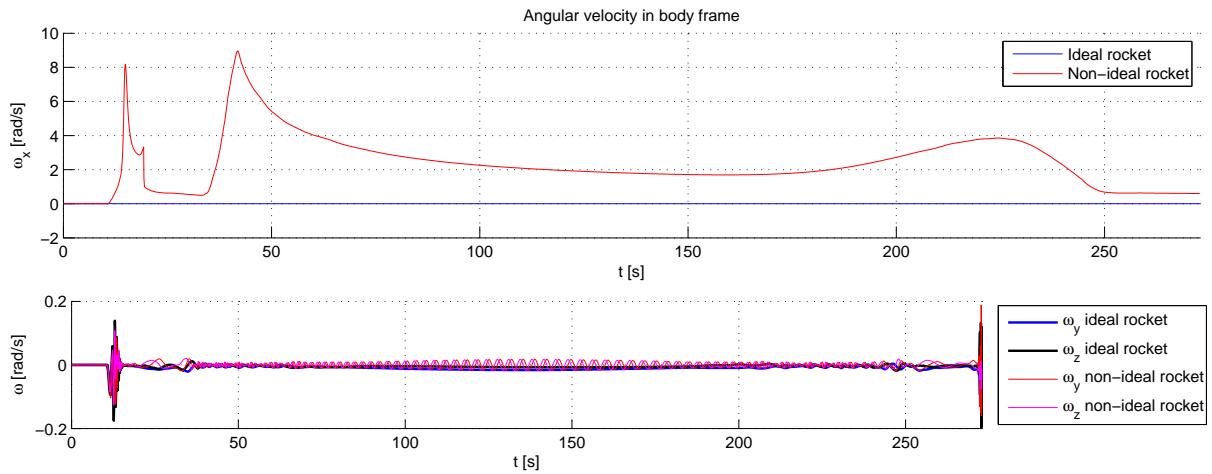


Figure 7.5: Nominal angular velocity

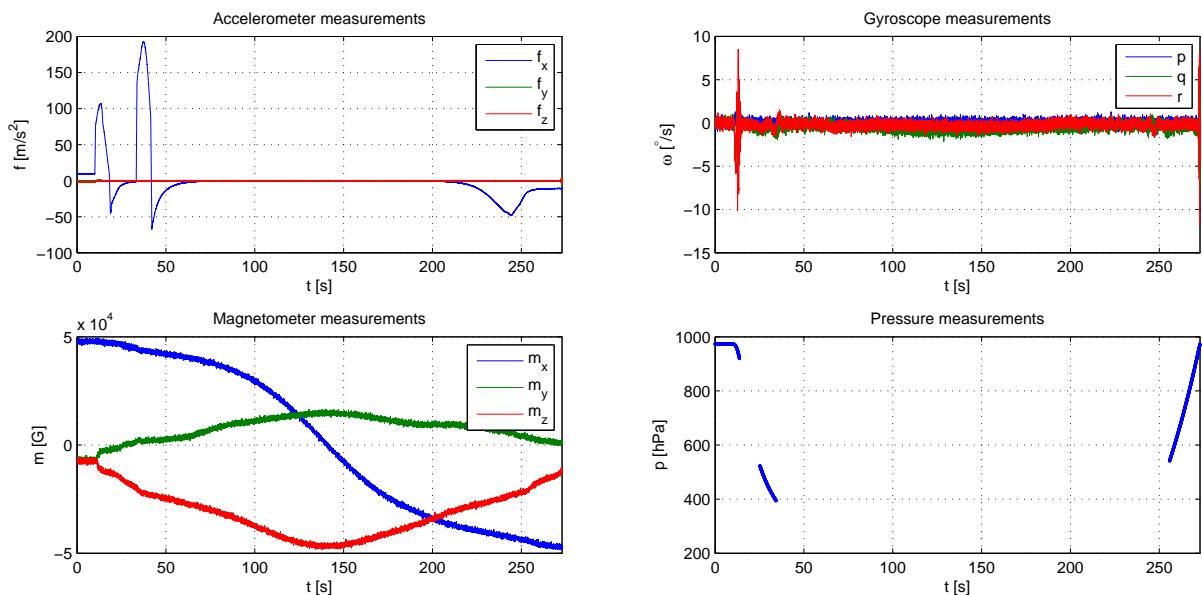


Figure 7.6: Nominal simulated measurements

rocket will be studied. The nominal case only includes white noise on the measurement data. In section 7.5, the other sensor noise factors will be studied. In 7.4.4, first a single estimator will be selected for which this study is performed. However, as it is expected that different estimators behave differently towards them, bias is already included in the analysis in this section.

7.4.1 Computational requirements

First, it will be determined which model and estimator are not suitable for real-time and post-flight estimation, respectively, in order to avoid analyzing results for estimators that cannot be implemented in practice anyway.

Originally, the ROSIE model along with the calculation of numerical derivatives was used. This had as advantage that only little effort would be required to perform the estimation. The resulting estimation error, required computational power, and lack of robustness against modeling errors, proved to be unacceptable. Therefore, further analysis using this model was halted, and the development was refocused on the simplified and kinematic model. The obtained results for the ROSIE estimator are presented in appendix D.

The second estimator type that can be discarded for real-time estimation is the RTS smoother. The RTS smoother can only be used to estimate the state at a certain instance in time, with known measurements from both before, and after that time instance. Therefore, it can only be used for post-flight estimation.

To determine the suitability with respect to computational constraints of the remaining estimators, for a nominal flight of the Stratos II, the time for a full estimation has been recorded. This has been done on a laptop with an Intel Core i7-2630QM processor, running at 2 GHz clock speed. It should be noted that this processor is at least an order of magnitude faster than typical on-board processors. Therefore, in order for the estimator to be suitable for on-board, real-time estimation, the time for the full flight estimation on the laptop should be much lower than the 273 seconds that the flight takes. The results of this evaluation can be found in table 7.2.

Table 7.2: Estimator computational time comparison

Model	Estimator	Average run time (3 runs) [s]	Times faster than real-time [-]
Simplified	EKF	36.38	7.50
Simplified	UKF	101.7	2.68
Kinematic	EKF	0.831	328
Kinematic	UKF	10.15	26.9

It can immediately be seen that using the simplified UKF based estimator for real-time estimation is unlikely to be feasible with the current state of technology: even a modern laptop is barely able to perform real-time estimation. The simplified EKF is three times faster than this. The kinematic models are even faster, with the kinematic model UKF being ten times as fast as the simplified EKF. Even faster is the kinematic model EKF, which is the fastest of all benchmarked estimators. The reasons for the speed increase can be easily identified. Since the accelerometer and gyroscope are not used as measurements in the sense of a Kalman filter, the computation of the Kalman gain and innovations have to be performed much less often. It is also interesting to note that the speed loss of the UKF compared to the EKF is much larger for the kinematic model estimator than for the simplified estimator. This can be explained by the fact that the state transition of the simplified model is computationally inexpensive compared to the Kalman filter equations and, while for the kinematic model the state transitioning and the Kalman filter equations are of equal order of magnitude regarding computational expense. It can be found that for the simplified model only 1.7 % of the time is spent transitioning the state, while for the kinematic model this is 37 %. Ultimately, the difference can be traced to the

state vector, since the expense of the Kalman filter equations are only dependent on the length of the state vector, and the base state vector of the simplified model has 15 elements, whereas the kinematic model only has 9 elements in its state vector.

7.4.2 Simplified model estimator

First, the nominal Stratos II flight has been estimated with the simplified model, using the EKF, in UD-filter form. It was found that using the regular EKF would sometimes lead to extremely strong filter divergence, which was solved by using the UD-form. This indicates that this estimation is strongly influence by the maximum numerical precision. The results have been plotted in figures 7.7 to 7.15, while the estimation errors at the characteristic points have been tabulated in table 7.3. All state variables have been plotted along with the estimation error and the estimated error covariance. In all cases the estimated error covariance gives a good measure of the estimation error, meaning that the estimator has appropriate knowledge of the estimation error.

Figures 7.7 and 7.8 show the simulated and estimated position and position estimation error in time. In the error plot, also the 1-standard deviation and 3-standard deviation according to the estimated error covariance matrix have been plotted. The position error does slowly diverge during flight, but mostly stays within the estimated error covariance bounds. The two different simulations compare quite good with each other in altitude, but less so in ground track. This can mainly be attributed to wind, which is not modeled in the simplified simulation. The estimator – which is in fact the simplified simulation, corrected with measurements – shows a better correspondence, both in altitude and ground track. Still, the position errors up to 5 km and altitude errors up to 2 km are large, and improvements to decrease the error, such as improving the system model or improving the sensor frequency or noise characteristics must be performed for this estimator to be useful.

The reason for the divergence of the position is the slight estimation error in the velocity. As during major part of the flight there are no measurements that directly provide position information, estimation errors in velocity propagate directly into the position. During the entire flight, the velocity estimation error is non-zero, as can be found in figure 7.10. It can be found from this figure that during motor burn the velocity error strongly diverges, but converges in the coast phases. Also in descent the velocity error and error covariance still converge, until it has reached a local maximum at 223 s. After this, the velocity decreases and the estimator error increases. This local maximum occurs when the deceleration due to drag equals the acceleration due to gravity. Before this point, gravity is the dominating force, while after it drag is dominant. The dominance of drag corresponds to an increase in error, because gravity can be modeled much better than aerodynamics.

Figure 7.11 shows the simulated and estimated MRP, the estimation error, and the error covariance. Around 175 s, a discrete jump can be seen in the MRP. This corresponds to the time that the MRP norm becomes 1, and the shadow transform is performed keep the MRP bounded. Due to interpolation, this also corresponds to a very short peak in the estimation error, which can be found in each of the individual MRP estimation error plots. It can also be seen that the shadow transform does not increase the magnitude of the estimation error, but only the sign of it. To better analysis of the attitude estimation error, the estimation error has also been quantified as an error quaternion and angle in figure 7.12. Here, no error covariance has been plotted, since the convergence or divergence of the attitude estimation can already be found from the MRP estimation errors. Finally, in figure 7.13, the attitude has been converted to elevation and attitude. There, it can be found that the attitude estimation error is mostly due to azimuth estimation error, as the elevation estimation closely follows the actual elevation. For example, at apogee the elevation error is only 1.25° , while the azimuth error is -8.66° . For the attitude error, the same trends can be found as for the velocity error, namely that the estimation error and error covariance decreases until the local maximum velocity has been reached. Then,

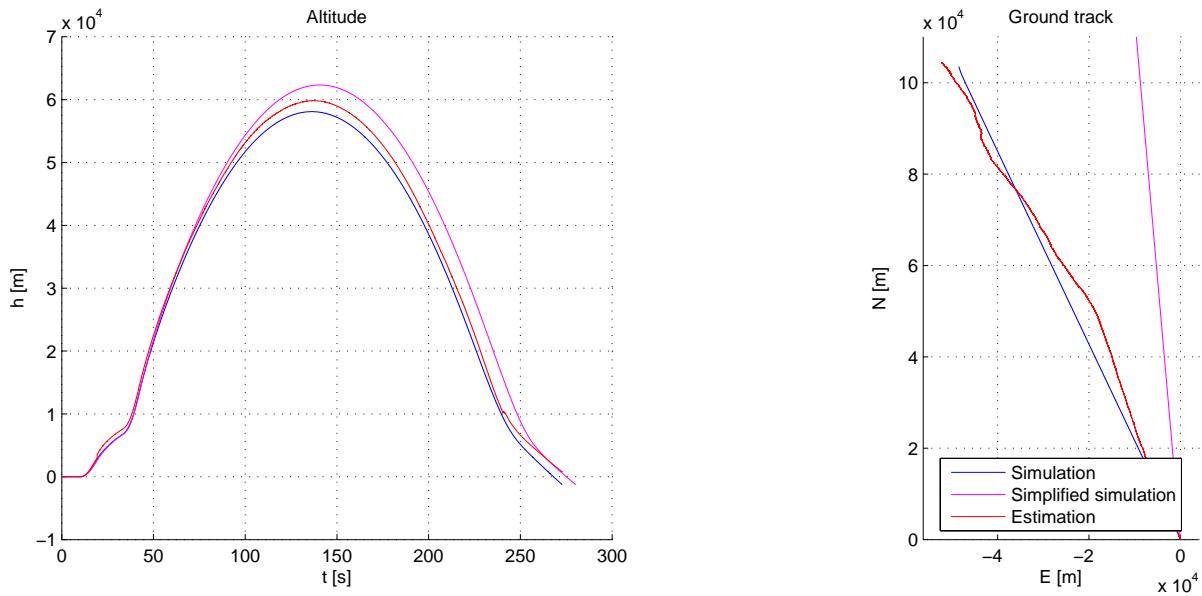


Figure 7.7: Estimated altitude and ground track with the simplified model EKF

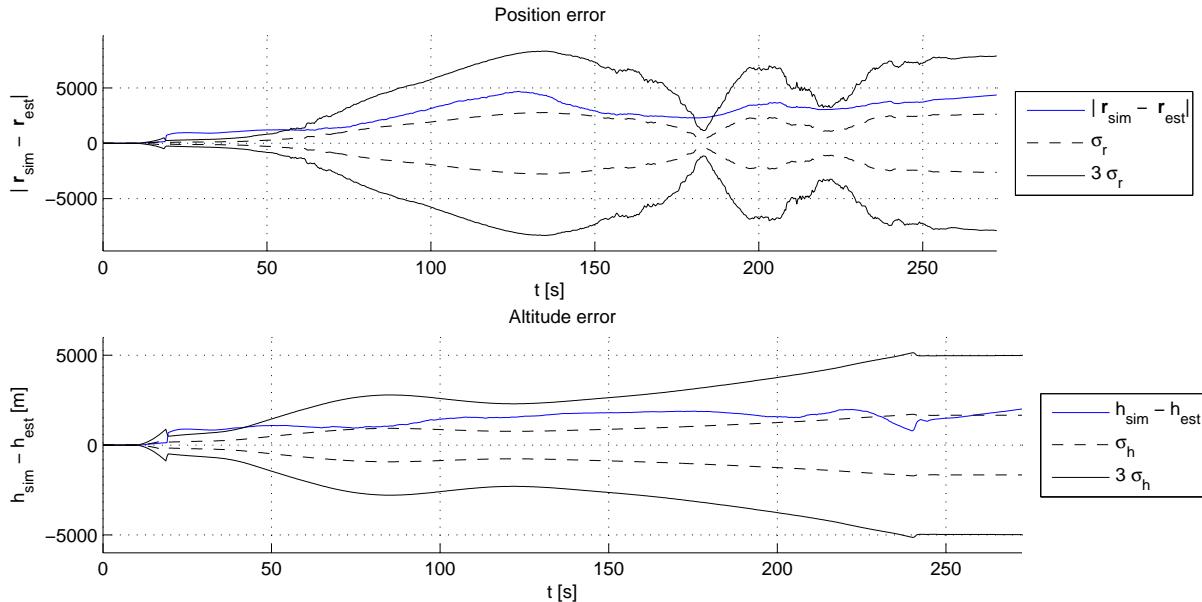


Figure 7.8: Position estimation error with the simplified model EKF

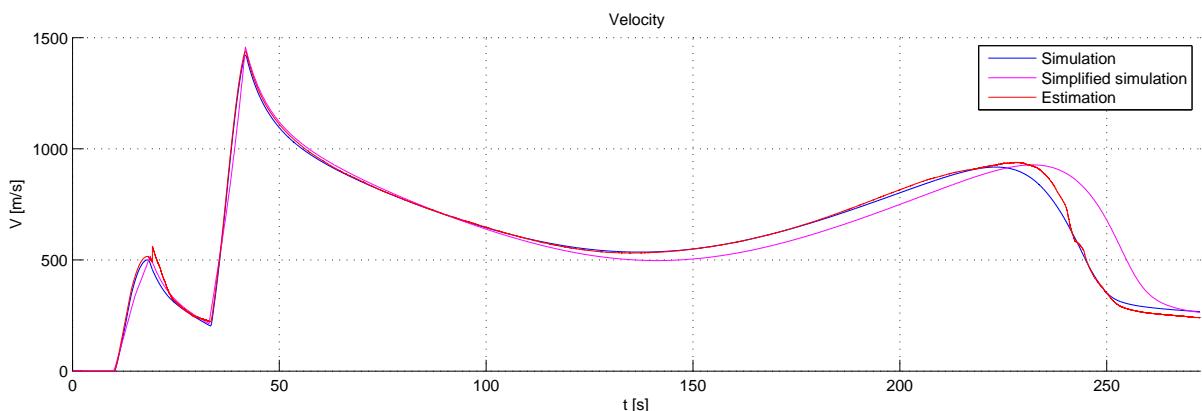


Figure 7.9: Estimated velocity with the simplified model EKF

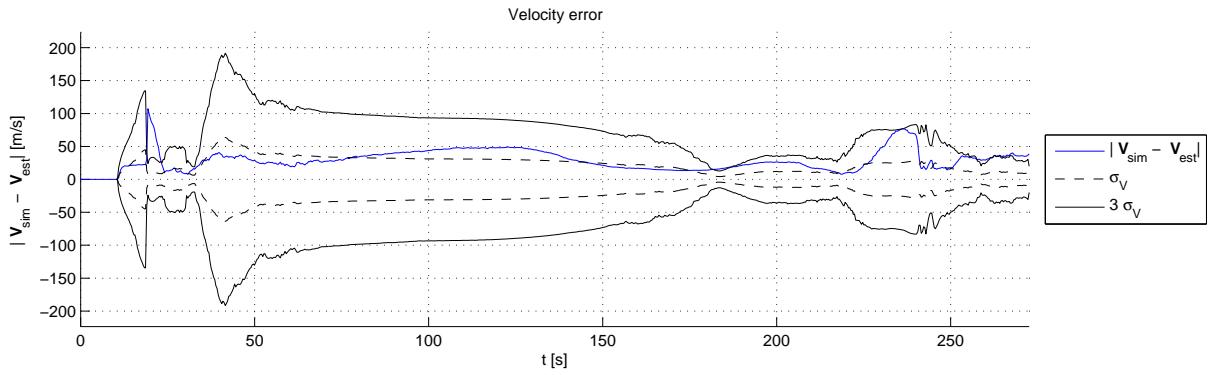


Figure 7.10: Velocity estimation error with the simplified model EKF

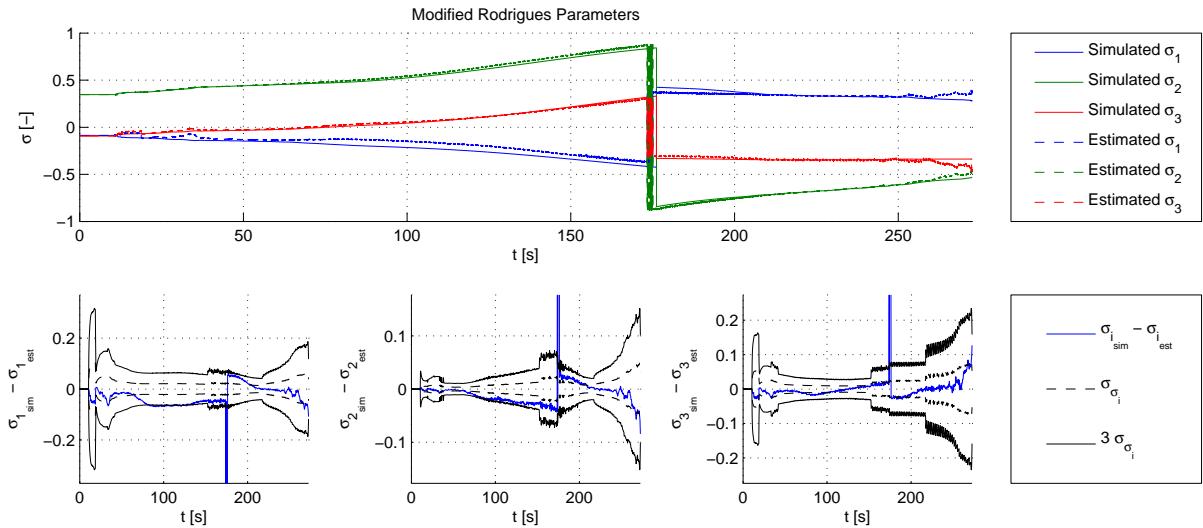


Figure 7.11: Estimated attitude with the simplified model EKF

in figure 7.14 the angular velocity is shown. The angular velocity is estimated very well, which is not very surprising due to the fact that the angular velocity is measured directly.

Finally, in figure 7.15, the augmented state variables have been plotted. The error covariance of the two aerodynamic state variables is small during the initial and final phase of the flight, but large in between. This can be explained by the fact that to estimate aerodynamic coefficients and density, some aerodynamic force must be present. At large altitudes around apogee, there is almost no atmosphere, making the variables only barely observable.

Unscented Kalman filter

The previous plots and tables have been generated using the EKF. This has also been done using the UKF. It was expected that the UKF would perform superior to the EKF. For example, in Julier and Uhlmann (1997), a falling body was estimated both using the EKF and UKF, and it was found that the UKF estimation error was an order of magnitude lower than that of the EKF. However, as the position plots in figure 7.16 show, this is not the case for the studied case. The altitude estimation follows the actual altitude closely, but the ground track diverges, and around 220 s, the position estimation makes a jump. Further inspection of the results shows that this may likely be due to the density estimation, as shown in figure 7.17: in ascent, the estimated density follows the actual density reasonably, but in descent, it keeps the density almost zero.

Further tuning the process noise matrix, it might be possible to obtain a better estimate using the UKF. However, because it was already previously determined that the UKF would

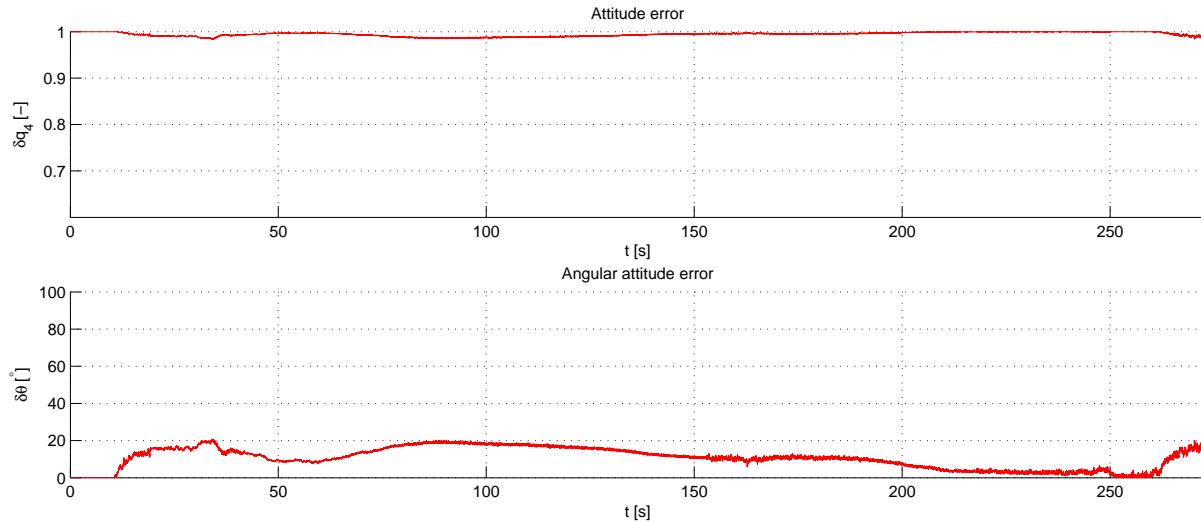


Figure 7.12: Attitude estimation error with the simplified model EKF

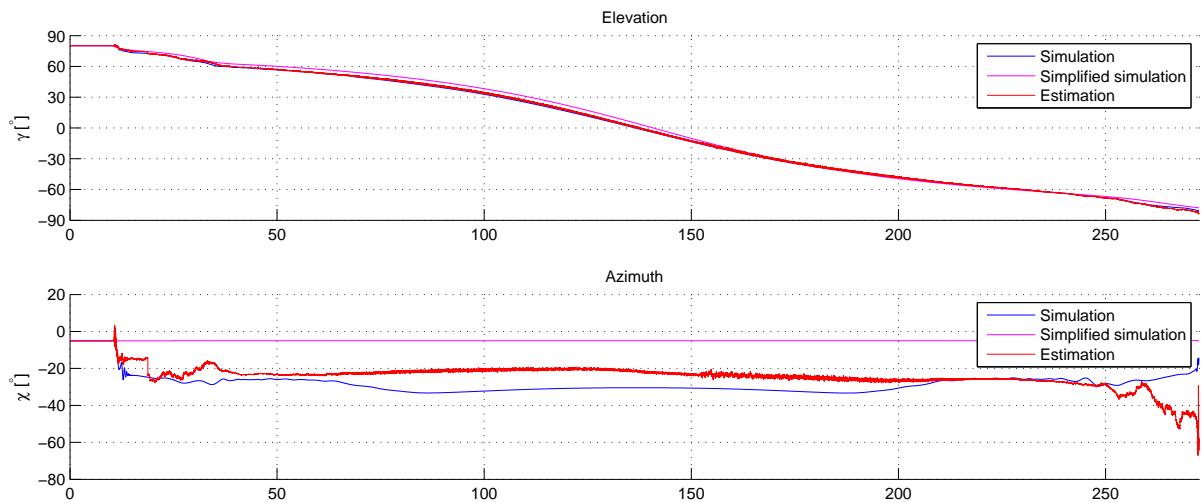


Figure 7.13: Estimated elevation and azimuth with the simplified model EKF

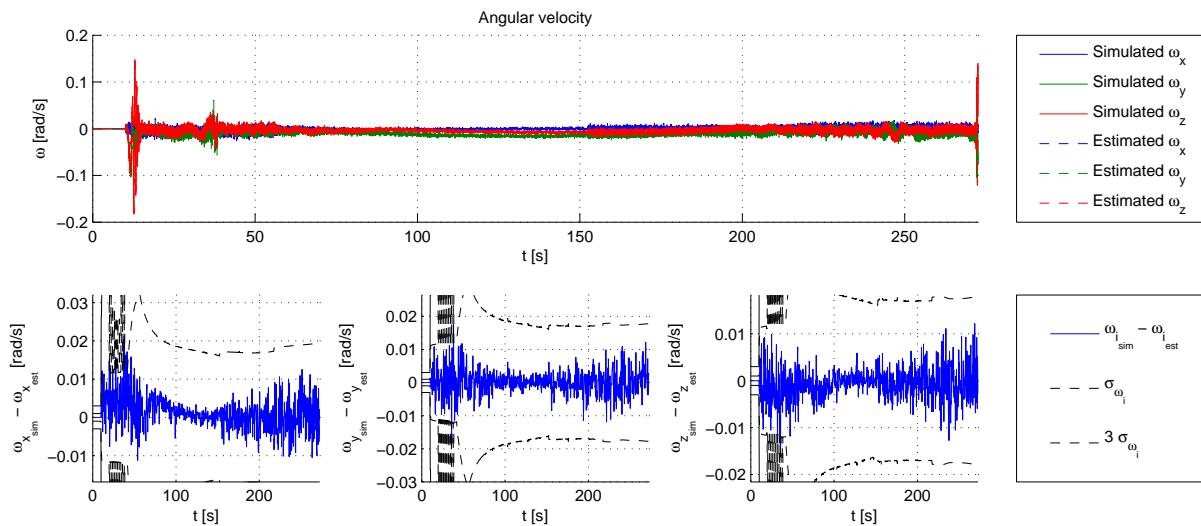


Figure 7.14: Estimated angular velocity with the simplified model EKF

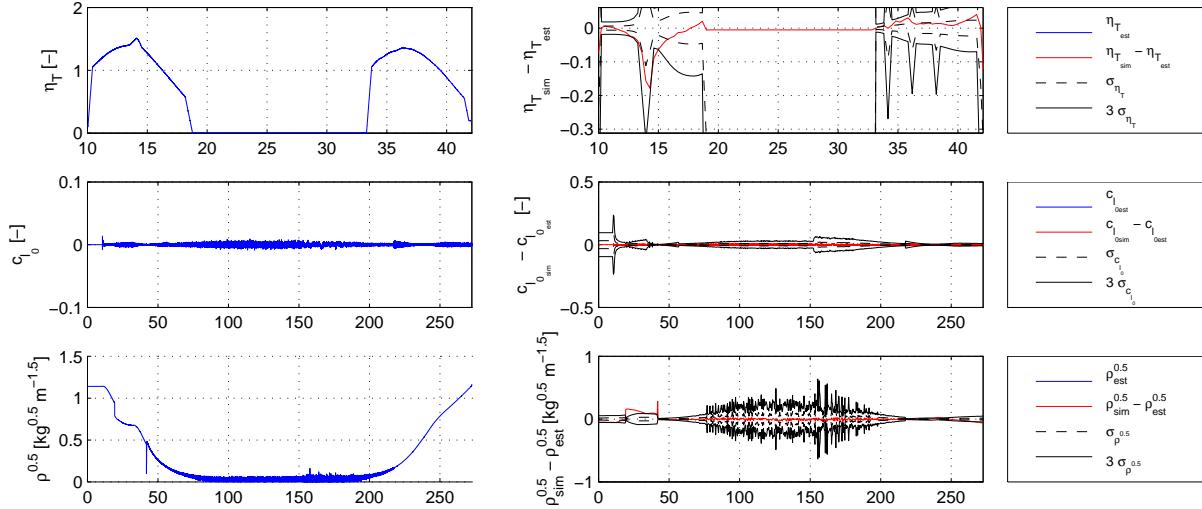


Figure 7.15: Estimated augmented state variables with the simplified model EKF

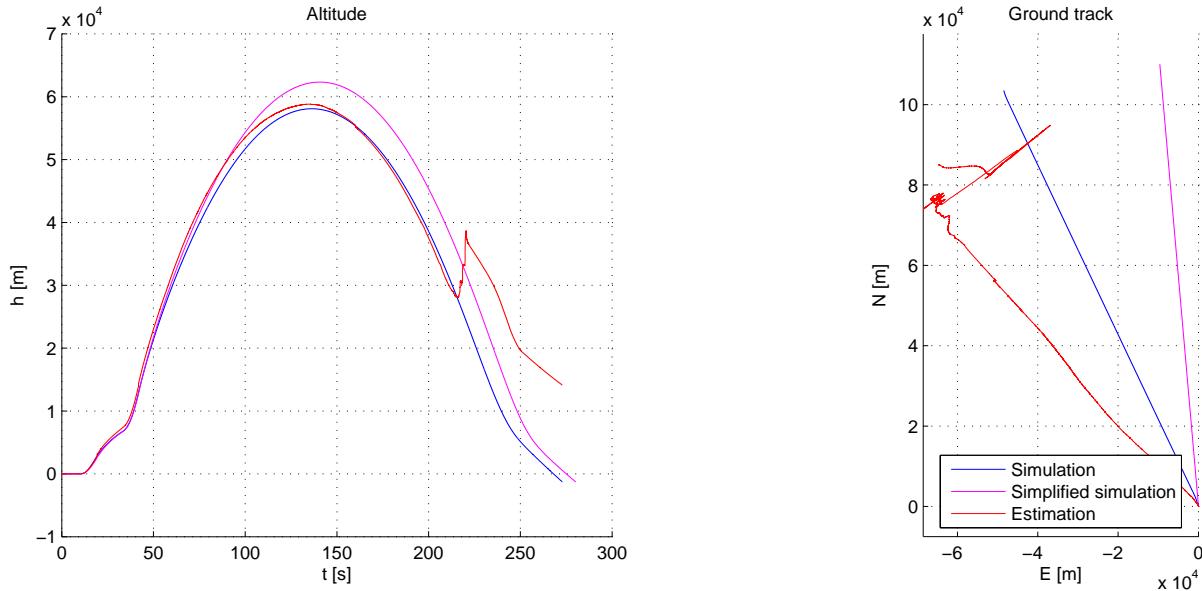


Figure 7.16: Estimated altitude and ground track with the simplified model UKF

be too resource intensive for real-time estimation, and the fact that even at points where the density is estimated properly, such as second stage burn, the estimate is already worse than the EKF estimate, it was decided not to further investigate the UKF with the simplified model.

Sensor bias

In figure 7.18 the estimated position of an estimator with biased sensors is plotted. It can immediately be seen that the estimation is worse than for a sensor without bias. The estimated position is much closer to that of the simplified simulation in the biased case, than in the non-biased case. This may be explained by the fact that the estimator will use more information from the model in order to deal with the sensor uncertainties.

Contrary to the position estimate, the attitude estimate does not worsen considerably, as shown in figure 7.19. This can be explained by figure 7.20, where the estimated bias is plotted. Here, it can be found that the gyroscope bias converges much better to the actual bias than the accelerometer bias. This is due to the fact that much more attitude information is available,

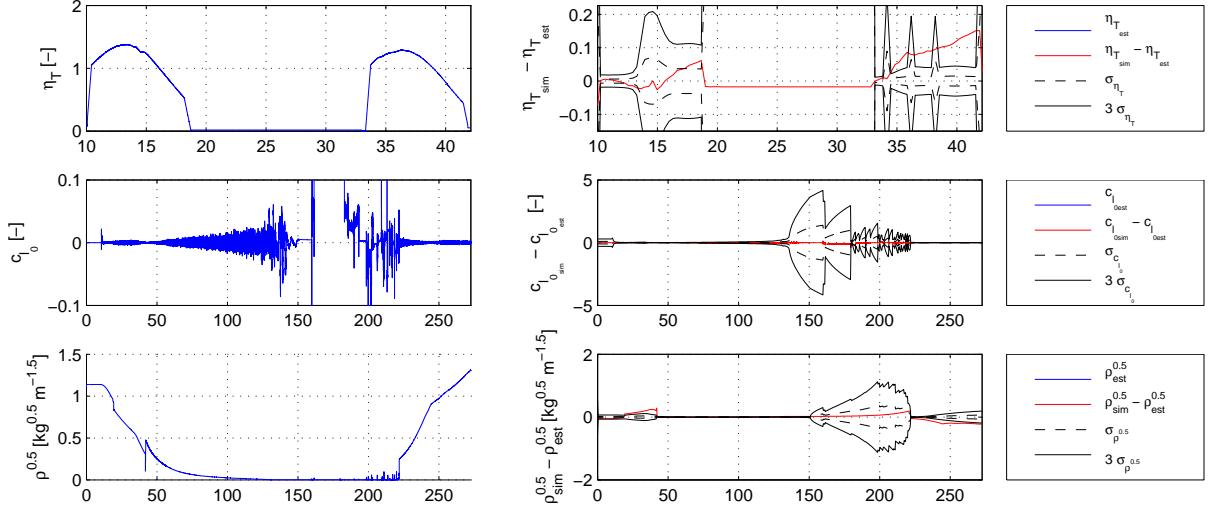


Figure 7.17: Estimated augmented state variables with the simplified model UKF

thanks to the presence of a magnetometer.

In the bias figure it can also be found that only the accelerometer bias in y- and z-direction can be properly estimated. The reason for this is that, in principle, both in practice and in the model, the two lateral accelerations are about zero. The acceleration in x-direction though strongly depends on the thrust and aerodynamic coefficients, which are parameters where the estimator model does not necessarily closely match reality. Therefore, the estimator will use the bias in x-direction to absorb any errors these parameters, which is visible by the bad convergence of this bias. For the gyroscope, a rather quick convergence can be found, with the y- and z-bias converging quicker than the x-bias.

Nominal non-ideal flight

As previously stated, beside the nominal flight for an ideal rocket, there is also a nominal flight for a non-ideal rocket, where a roll coefficient is introduced. The position estimate for this case is shown in figure 7.21. It can be immediately seen that the position estimate is considerably worse than for the ideal rocket. Even worse is figure 7.22, where the velocity error is shown. The velocity error along the x-body axis corresponds rather well with the velocity error for the ideal flight, but the two lateral error components do not. The oscillatory trend in the velocity error leads to the conclusion that the velocity error is most likely due to an attitude error. Figure 7.23 shows that indeed the attitude error is slightly larger than the ideal flight. Still, as shown by figure 7.24, for most of the flight the estimated attitude corresponds well to the actual attitude, and the elevation for the entire flight. Finally, in figure 7.25, the augmented state variables are shown. Here, it is clear that the estimator is properly able to estimate the added roll coefficient.

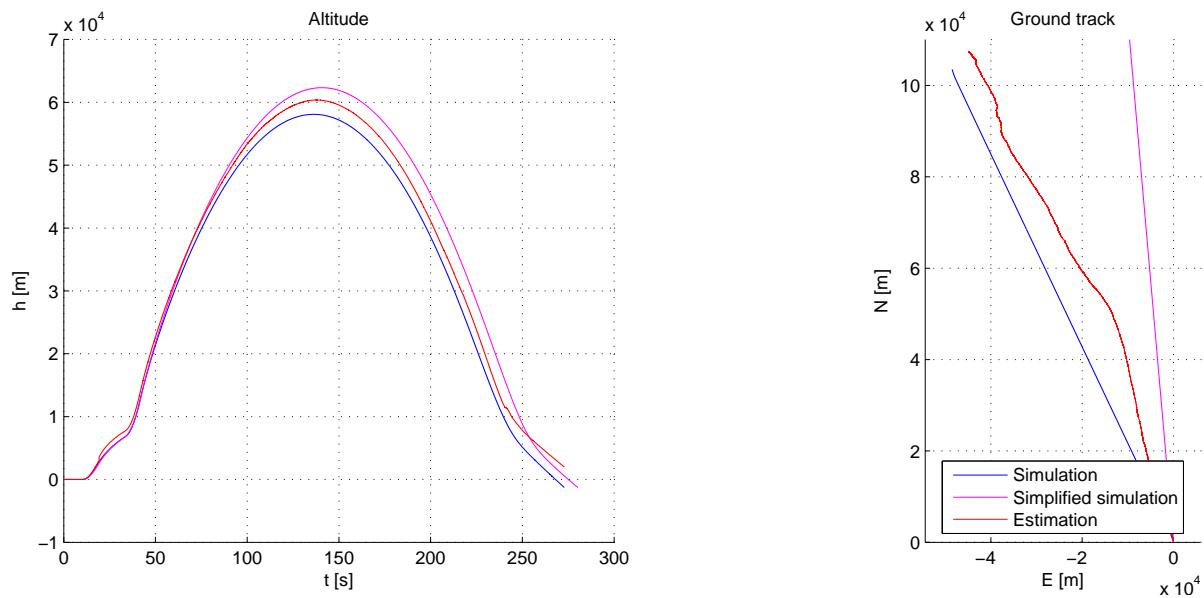


Figure 7.18: Estimated altitude and ground track with biased sensors with the simplified model EKF

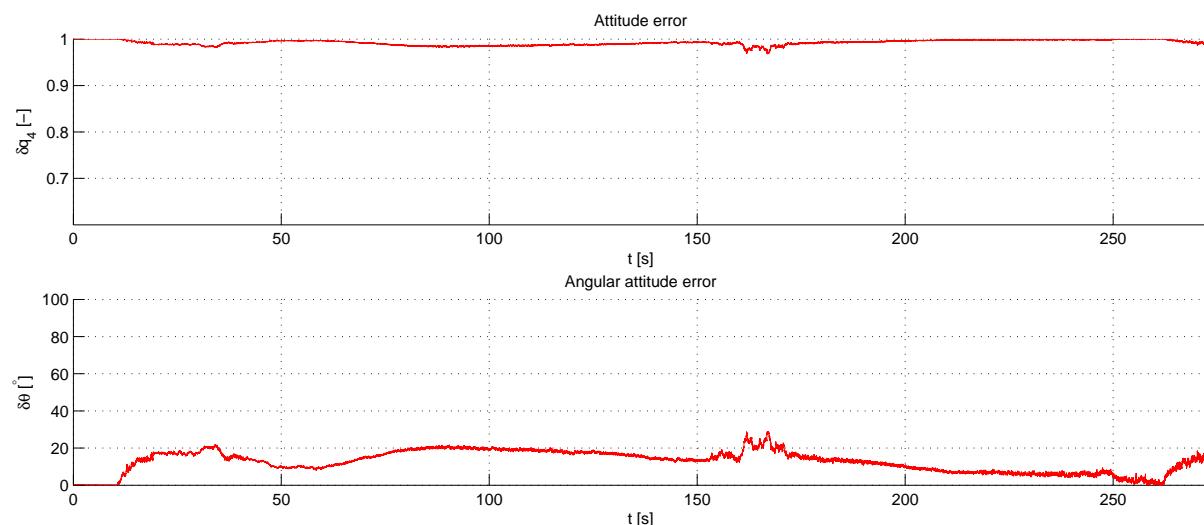


Figure 7.19: Attitude estimation error with biased sensors with the simplified model EKF

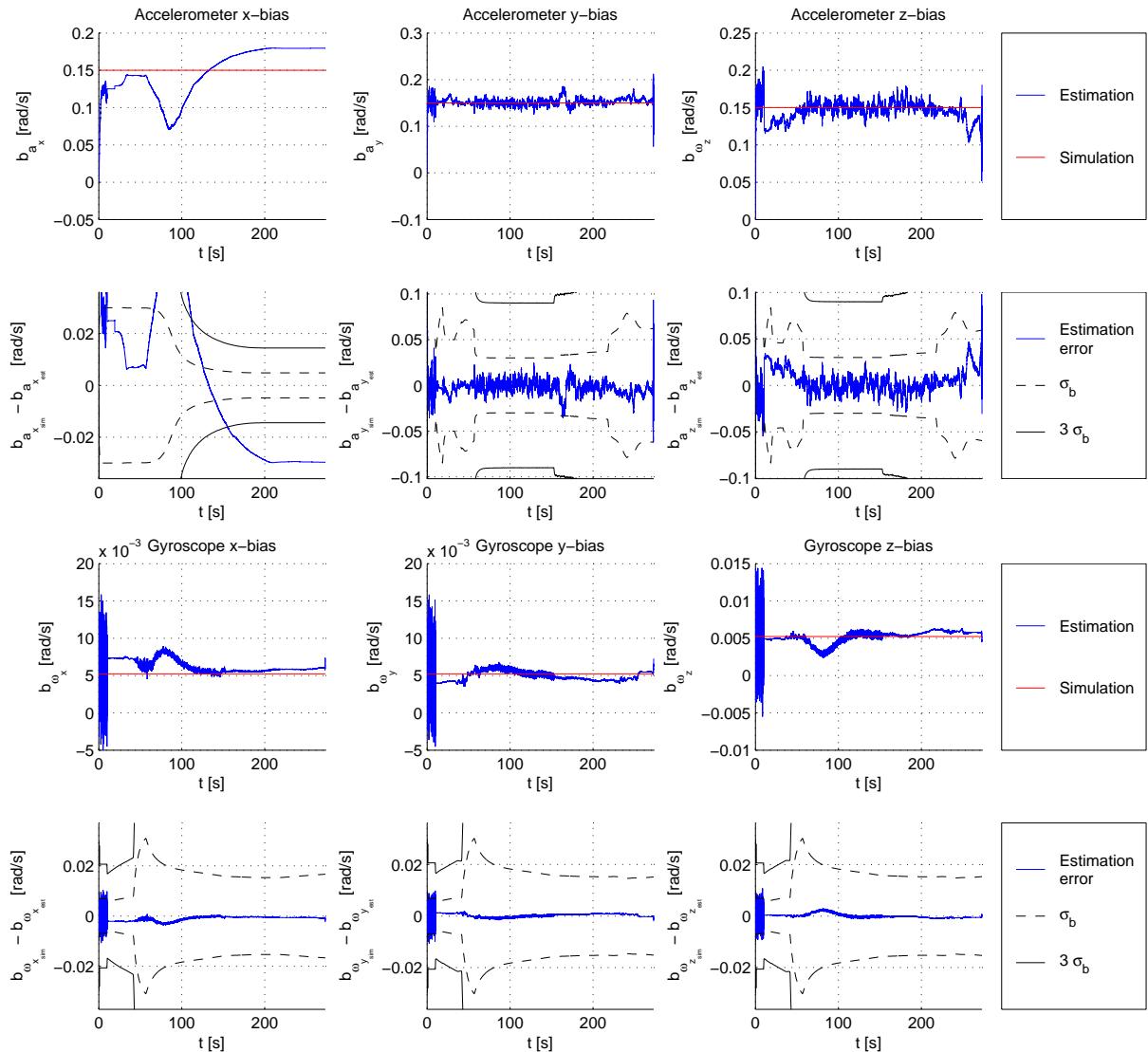


Figure 7.20: Estimated bias with the simplified model EKF

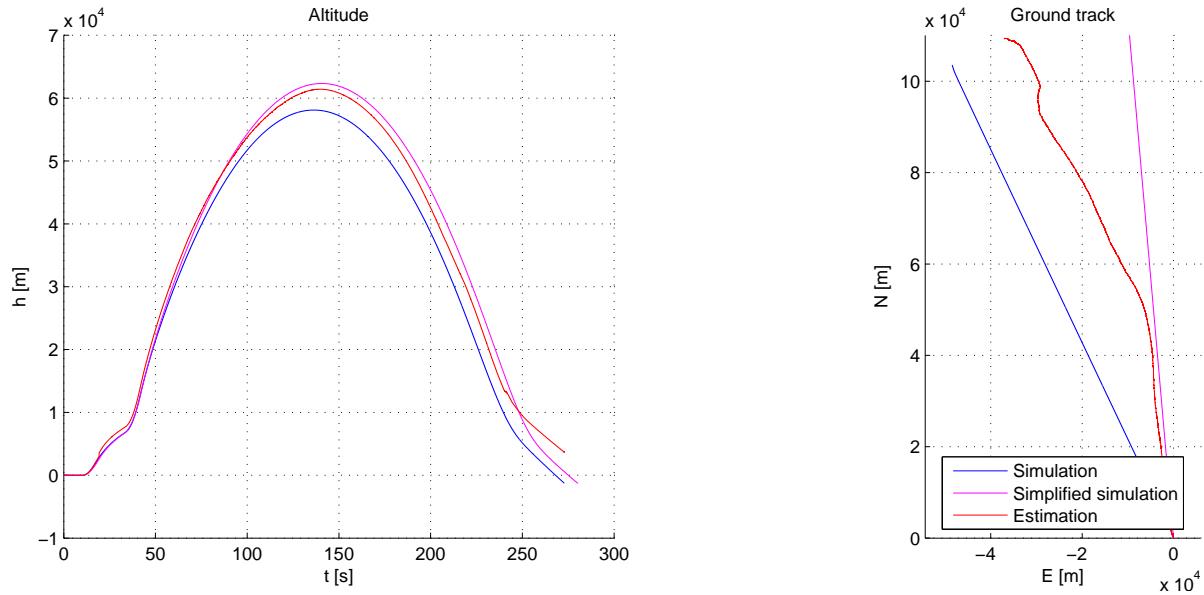


Figure 7.21: Non-ideal rocket estimated altitude and ground track with the simplified model EKF

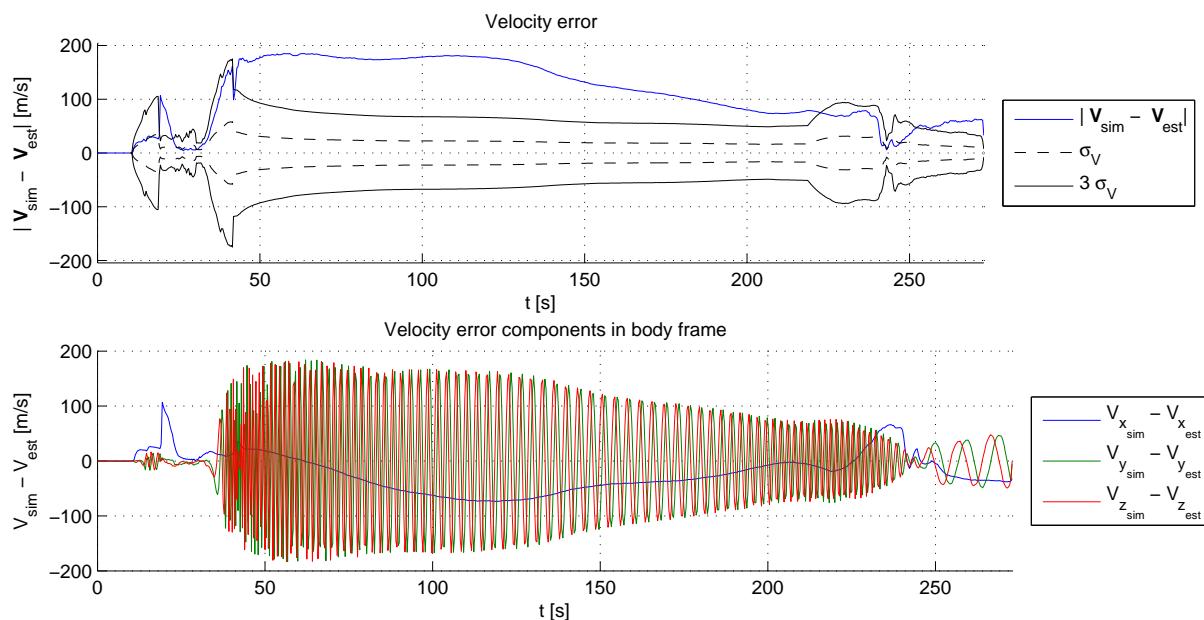


Figure 7.22: Non-ideal rocket velocity estimation error with the simplified model EKF

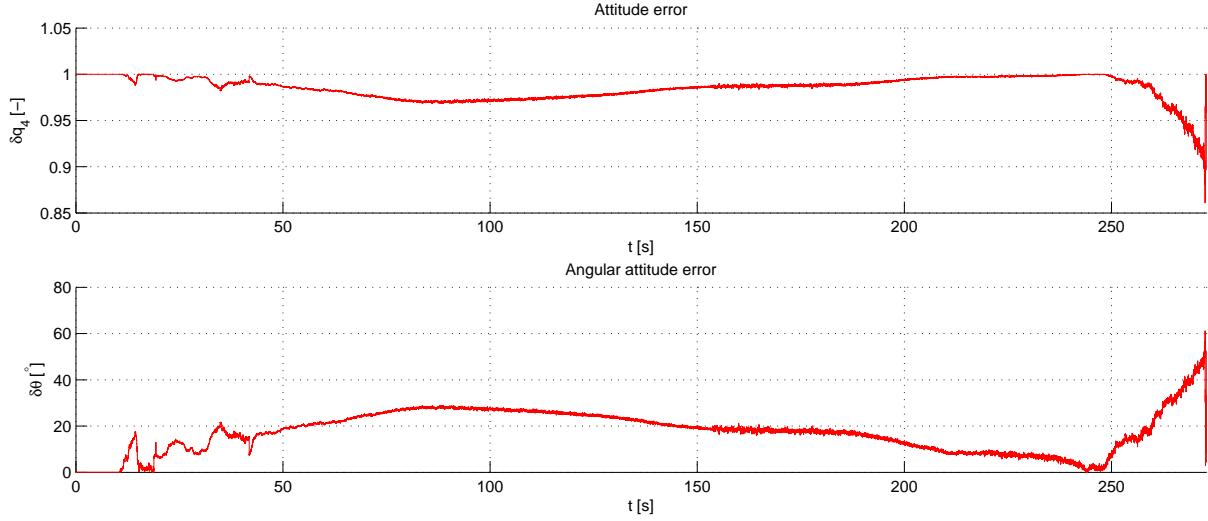


Figure 7.23: Non-ideal rocket attitude estimation error with the simplified model EKF

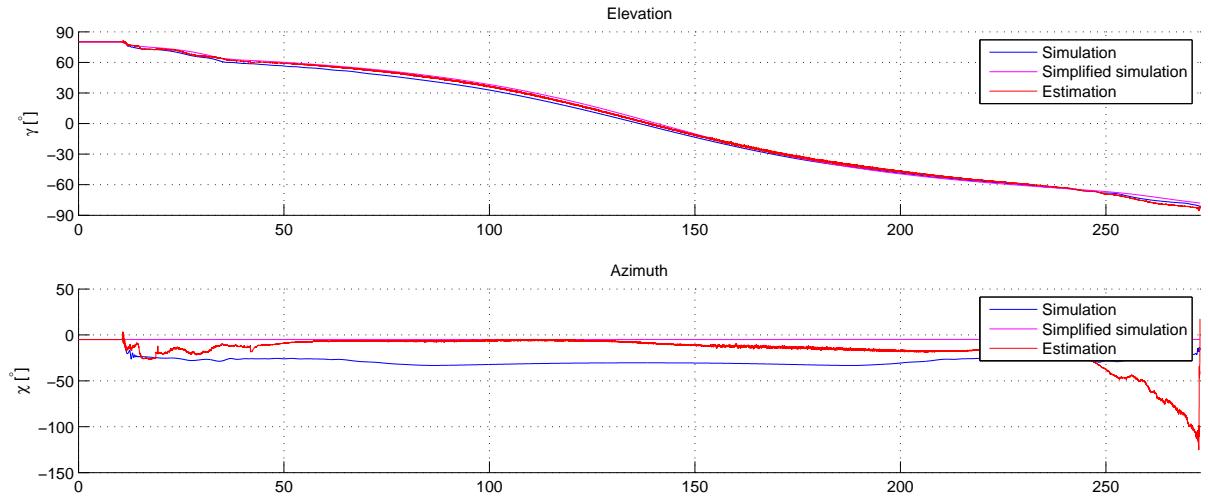


Figure 7.24: Non-ideal rocket estimated elevation and azimuth with the simplified model EKF

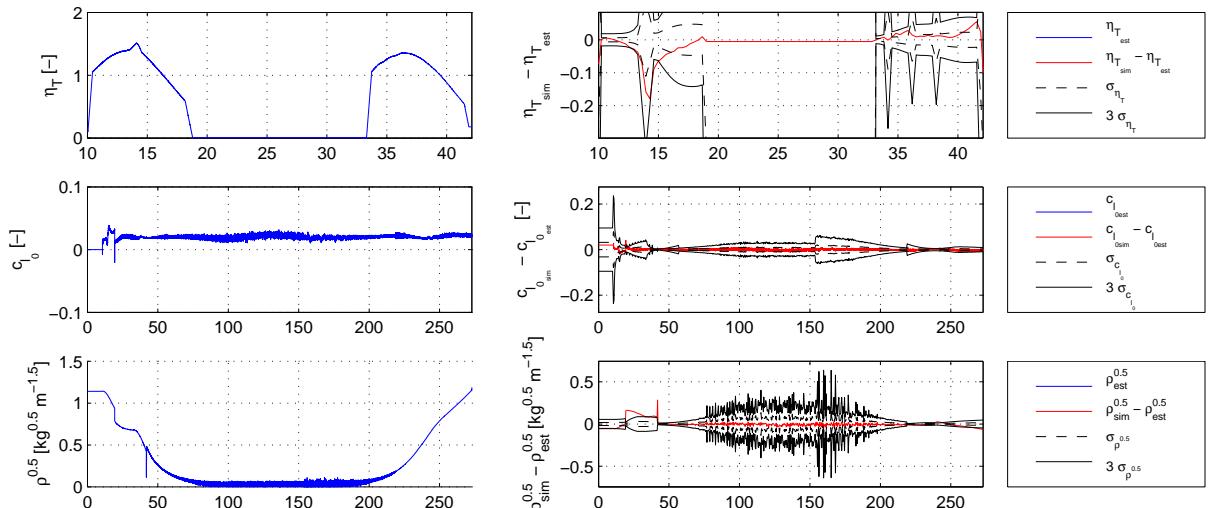


Figure 7.25: Non-ideal rocket estimated augmented state variables with the simplified model EKF

7.4.3 Kinematic model estimator

The nominal Stratos II flight has been estimated with the kinematic model estimator, both using the regular EKF, and the EKF in UD-form. It was found that the estimation results did not differ between the regular EKF and the UD-filter, therefore, any further estimations will be performed with the regular EKF. The results have been plotted in figures 7.26 to 7.32, while the estimation errors at the characteristic points have been tabulated in table 7.7. All state variables have been plotted along with the estimation error and the estimated error covariance. In all cases, the estimated error covariance gives a good measure of the estimation error, meaning that the estimator has appropriate knowledge of the estimation error.

Figures 7.26 shows simulated and estimated position. Both in altitude and ground track there is excellent correspondence between the estimator and the simulation. Still, as seen from figure 7.27, the error does not converge, but slowly increases in time. This can be explained by the lack of absolute position information: during major part of the flight only accelerometer data is available for the position. Only when pressure data becomes available again, after 250 s, the estimation error converges again. From the velocity error in figure 7.29, it can be found that the estimation error in general converges, except during motor burn, when the strong accelerations appear to cause an increase in estimation error covariance.

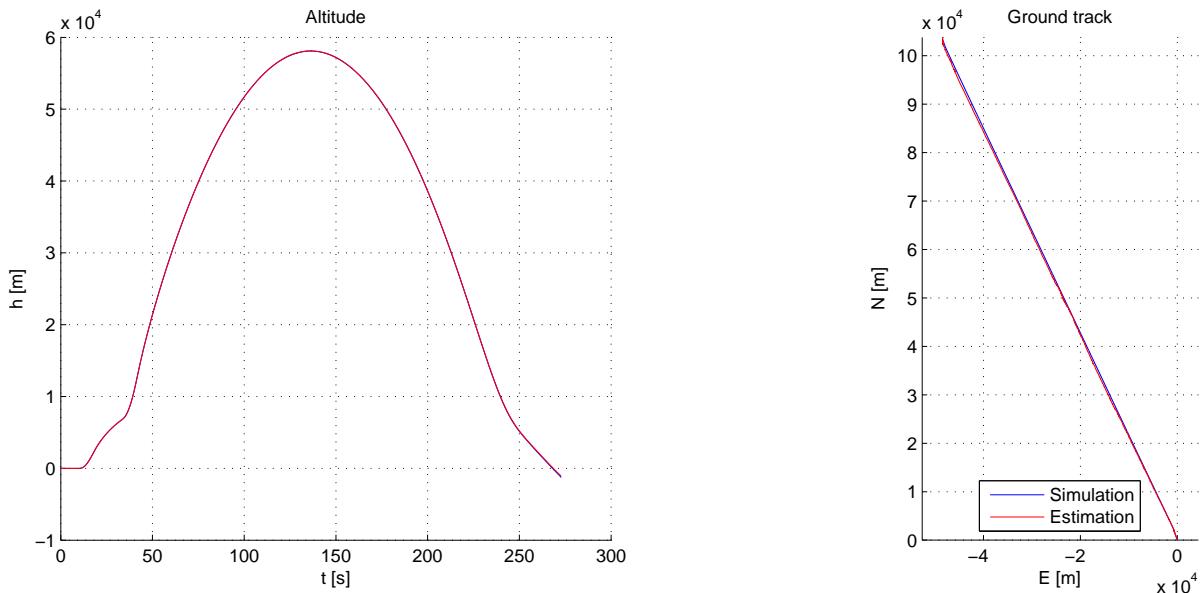


Figure 7.26: Estimated altitude and ground track with the kinematic model EKF

Figure 7.30 and 7.31 show the simulated and estimated attitude. Also the estimated attitude corresponds very well with the actual attitude: the attitude error is not larger than 2° . The exception to this is the from lift-off to second-stage burnout, when the attitude error reaches peaks of 6° . From figure 7.32, it can be found that these error correspond almost fully to azimuth errors, as the elevation error never exceeds 1° . From the error covariance it can be seen that the estimator overestimates the estimation error. This could be solved by tuning the process noise matrix, however, in later sections, when other non-nominal flights are analyzed, the used process noise will be necessary to cope with large angular velocities. As with the simplified model, around 175 s, a discrete jump can be seen in the MRP, corresponding to the time the shadow transform is performed to keep the MRP bounded. Due to interpolation, this also corresponds to a very short peak in the estimation error, which can be found in each of the individual MRP estimation error plots.

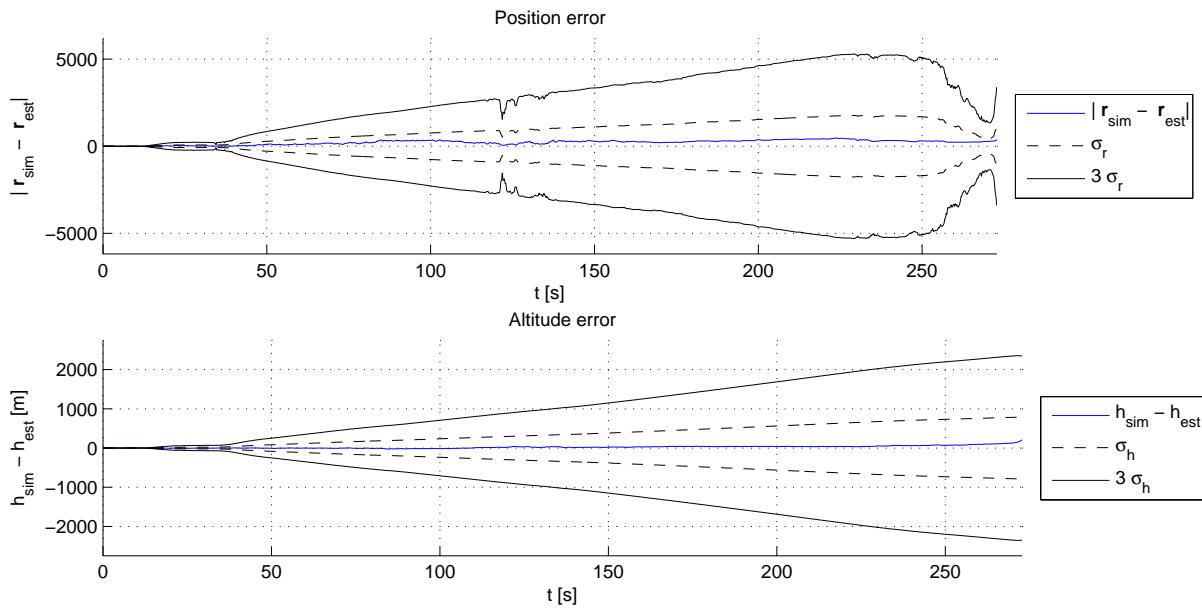


Figure 7.27: Position estimation error with the kinematic model EKF

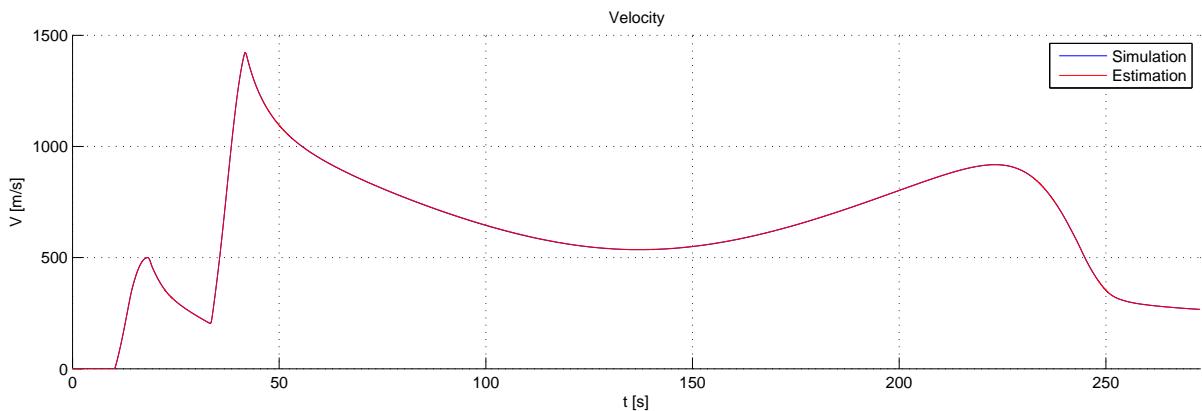


Figure 7.28: Estimated velocity with the kinematic model EKF

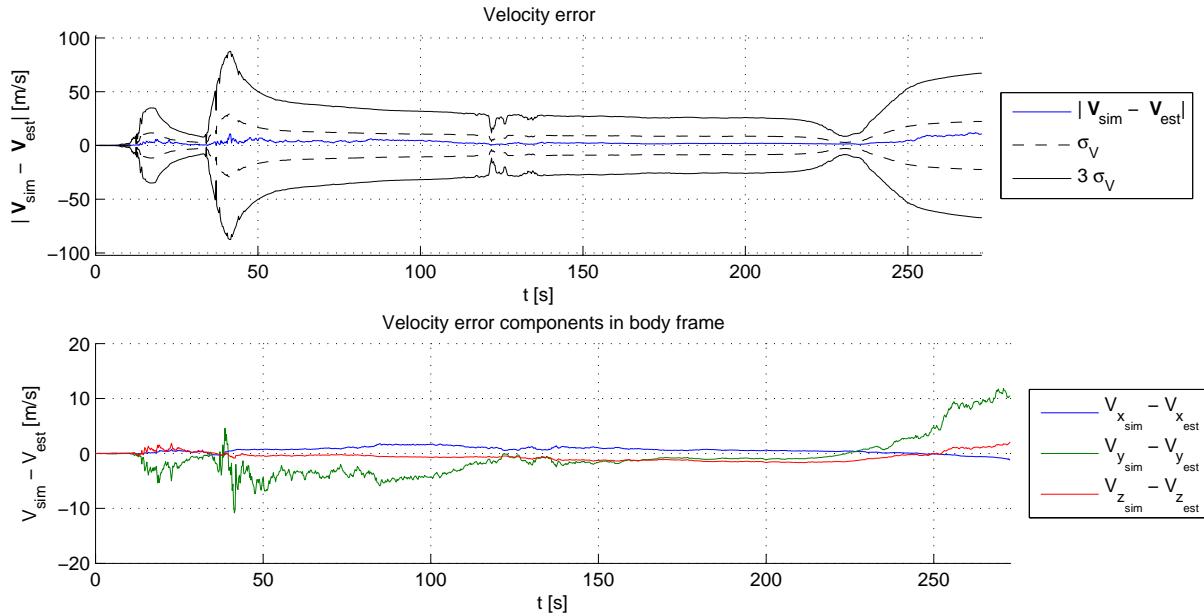


Figure 7.29: Velocity estimation error with the kinematic model EKF

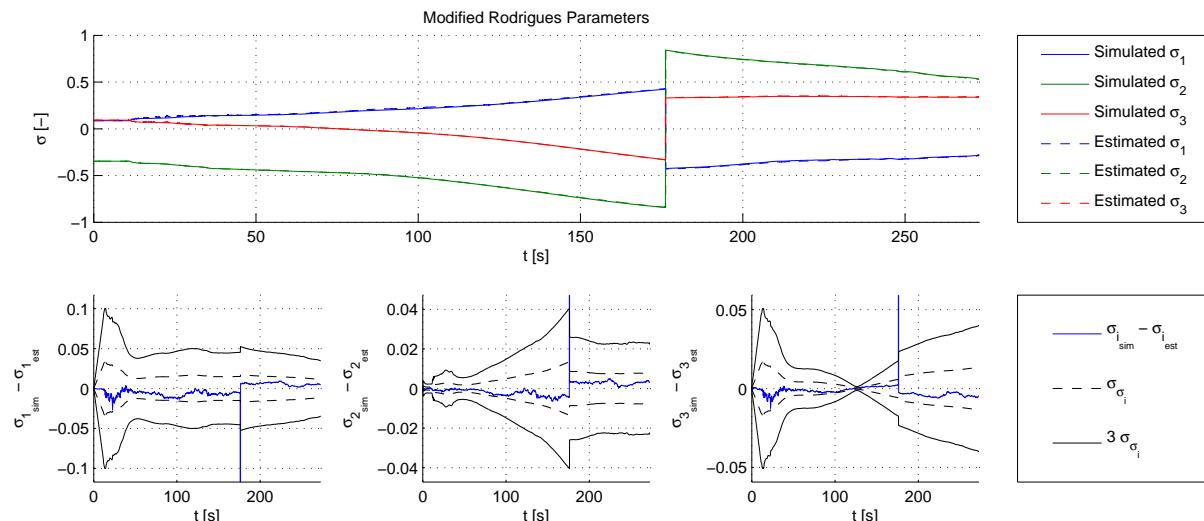


Figure 7.30: Estimated attitude with the kinematic model EKF

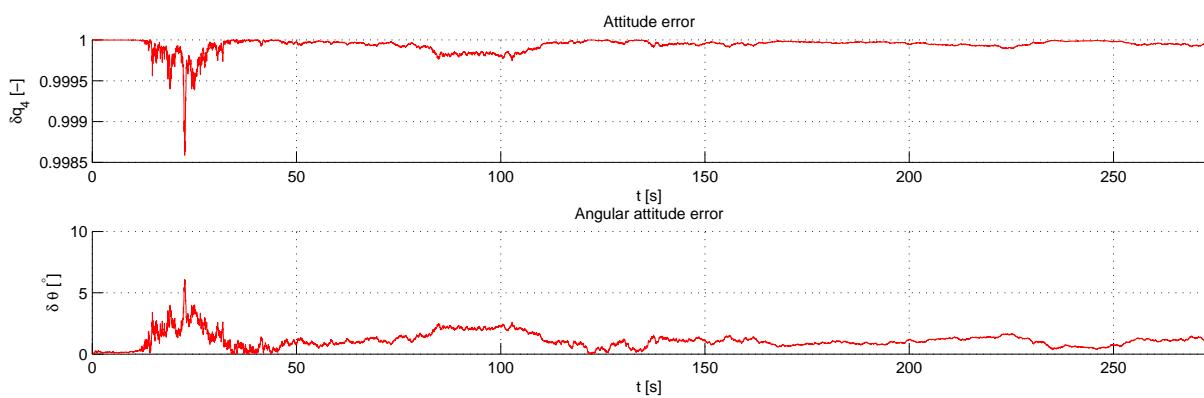


Figure 7.31: Attitude estimation error with the kinematic model EKF

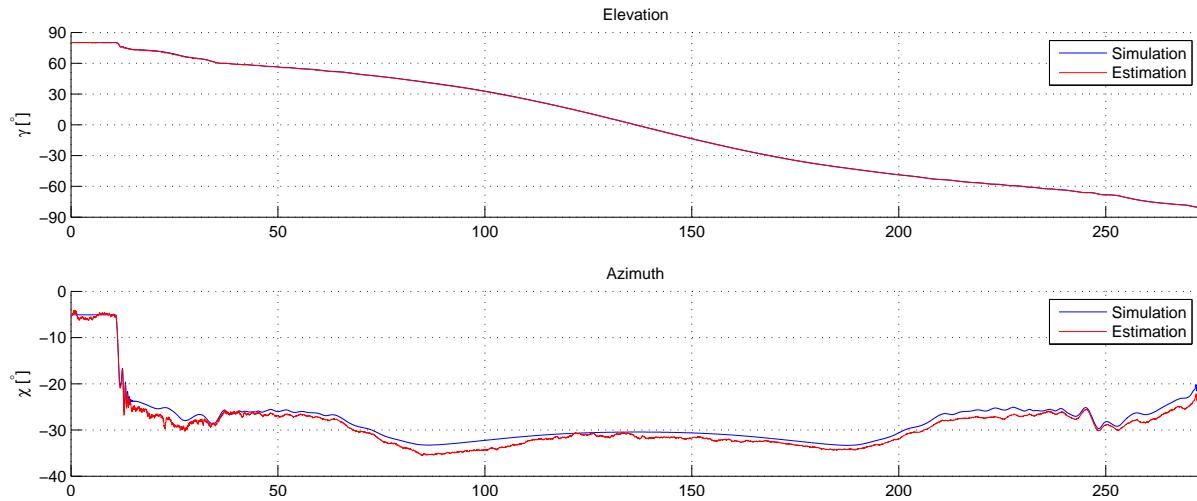


Figure 7.32: Estimated elevation and azimuth with the kinematic model EKF

Unscented Kalman filter

The previous plots and analysis applied to the model-free estimator using the EKF. As stated before, the estimation has also been performed using the UKF. The results are in table 7.8. The plots look almost the same as the EKF plots, only with differences in the behavior of the error covariance. Therefore, only the position, velocity and MRP error plots are shown.

From the table 7.8, it can already be seen that the UKF does not universally improve the estimation error. In general, the maximum errors are worse for the UKF, but the mean error is smaller. The notable exception to this is position and altitude, for which the estimation errors are always smaller for the EKF.

In figure 7.33, the position estimation error for the UKF is shown, in figure 7.34, and in figure 7.35. The main difference with the EKF is the behavior of the error covariance: whereas the error covariance is mostly smooth for the EKF, in the case of the UKF the error covariance shows much more random variations, and is also smaller in magnitude. This smoothing of the covariance by the EKF is caused by the linearization in the propagation of the covariance, while the UKF is able to propagate the covariance in much higher orders (Julier and Uhlmann, 1997).

Sensor bias

In figure 7.36 the estimated position of an estimator with biased sensors is plotted. It can immediately be seen that the estimation is worse than for a sensor without bias. The attitude estimate does worsen, but not as much as the position estimate, as shown in figure 7.37. This can be explained by the fact that the bias on the accelerometer is not estimated. The reason for this can be found in figure 7.38, which shows the estimated bias without accelerometer-bias estimation, and figure 7.39 which shows the estimated biases with accelerometer bias estimation. It is clear that the accelerometer bias does not converge to the actual bias. From the covariance lines it can be seen that the error covariance does not converge, a clear indication that there is no actual information available to correct for the accelerometer bias. Therefore, only the gyroscope bias is estimated.

It can also be found that the gyroscope bias in x-direction takes much longer to converge than in the other two directions. This is the case because the estimator uses the information from the magnetometer to estimate the gyroscope bias. The simulations are performed with as initial position the launch site Esrange, Kiruna ($\delta = 67.89^\circ$, $\tau = 21.11^\circ$). Here the magnetic field lines are almost perpendicular to the surface, which means that they are initially almost fully parallel to the x-body axis. During flight, the rocket will make a gravity turn, during which the

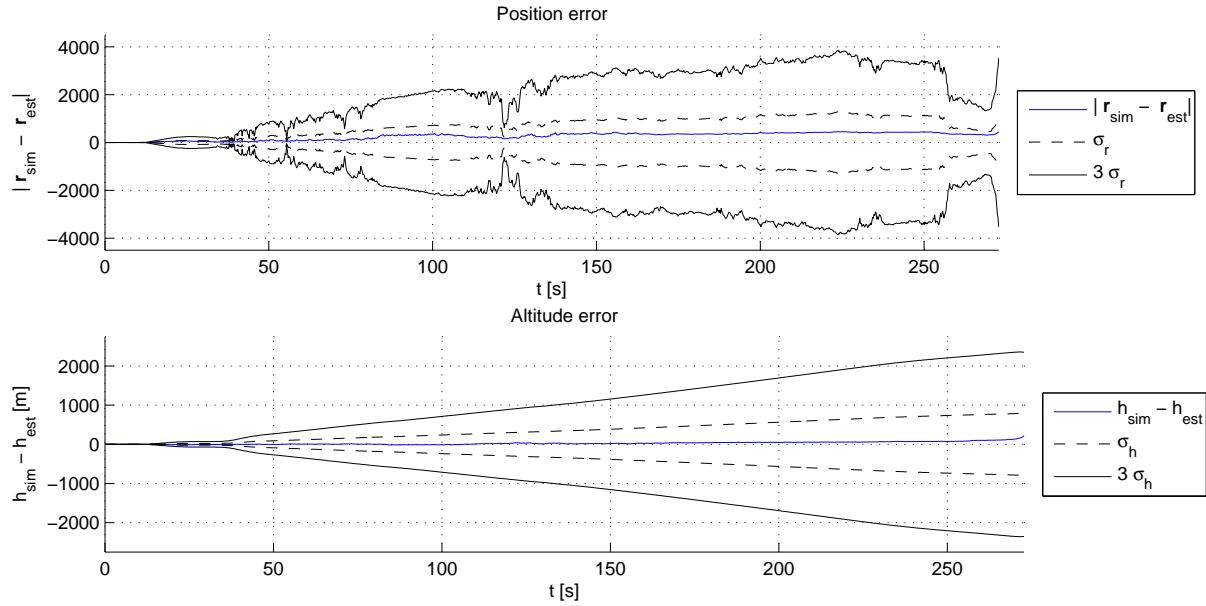


Figure 7.33: Position estimation error with the kinematic model UKF

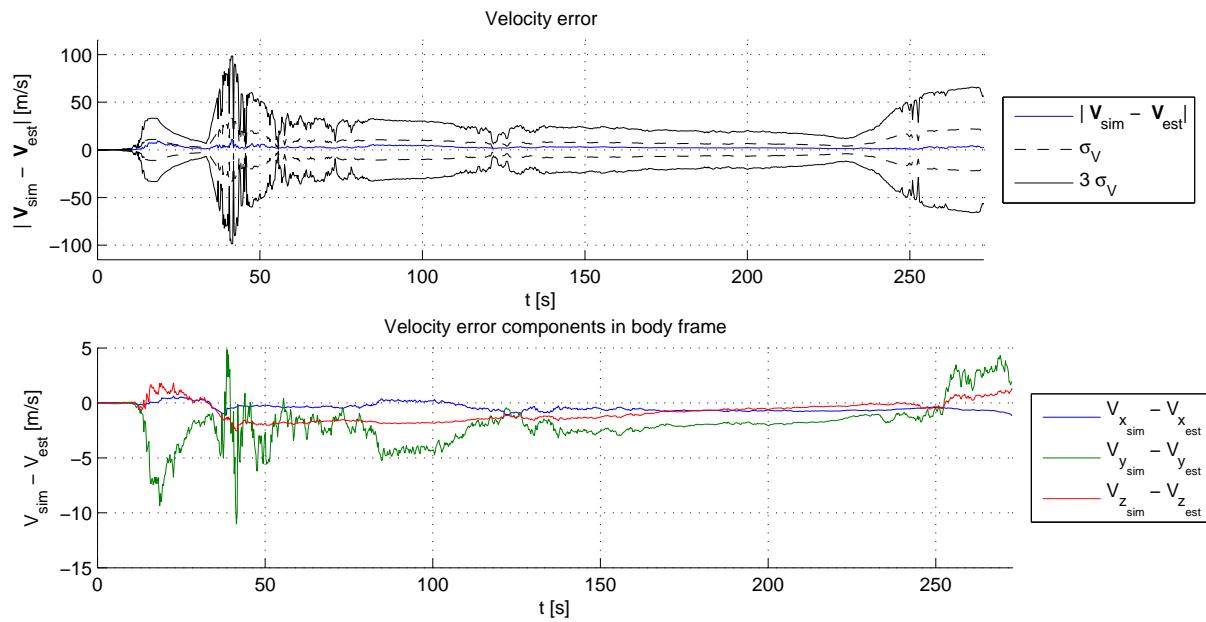


Figure 7.34: Velocity estimation error with the kinematic model UKF

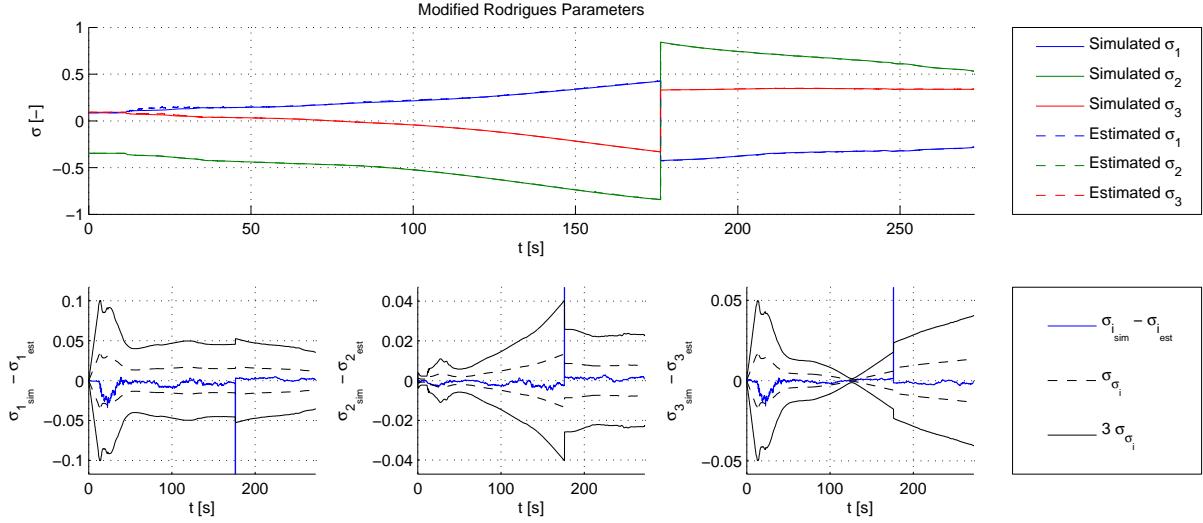


Figure 7.35: Estimated attitude with the kinematic model UKF

angle between the magnetic field lines and the x-body axis increases, providing the estimator with more information about the gyroscope bias in x-direction.

Nominal non-ideal flight

Also for the model-free estimator the nominal non-ideal flight has been estimated. Since the estimator does not contain the flight model, there is only little way in which the extra rolling motion can affect the estimation result. The only reason for it to be affected, would be because the dynamic circumstances are more severe. Figures 7.40, 7.41 and 7.42 show respectively the position, velocity and attitude error. Indeed, the errors are larger than for the ideal flight. The mean position error doubles to 400 m, while the maximum increases even further to almost 2 km. Also the altitude error increases to a mean of 120 m, and a maximum of 500 m. Contrary to the position estimate, the velocity error is not affected significantly. The estimation errors in attitude even decrease.

7.4.4 Model comparison and selection

In the previous sections the estimation result for a nominal flight using the simplified model and the kinematic estimators have been analyzed. For both estimators this has been done using the nominal flight trajectory for an ideal rocket, using the EKF and UKF, using the EKF with sensor bias, and using the non-ideal rocket flight. The results of these estimation processes can be found in the tables on the following tables, which have been discussed in the sections above.

The development of the position and attitude determination system requires further analysis on the response of the estimator to various other factors, such as varying levels of noise, sensor noise factors such as non-orthogonality, sensor failure, and varying flight conditions. This could be done for each of the estimators treated above, but doing so would be a time consuming task. Therefore, based on the performed performance evaluations, one estimator type is selected for which further analysis will be performed.

In general, the kinematic model estimator is an order of magnitude better than the simplified model estimator. This applies both to translational and rotational motion, and both to the mean and the maximum error. In addition to this, the model-free estimator is much faster. Also, during the development it has been found that the simplified model estimator is very sensitive to the selection of process noise, while for the model-free estimator this is much less the case. An important downside of the kinematic model estimator is the fact that it is not able to properly

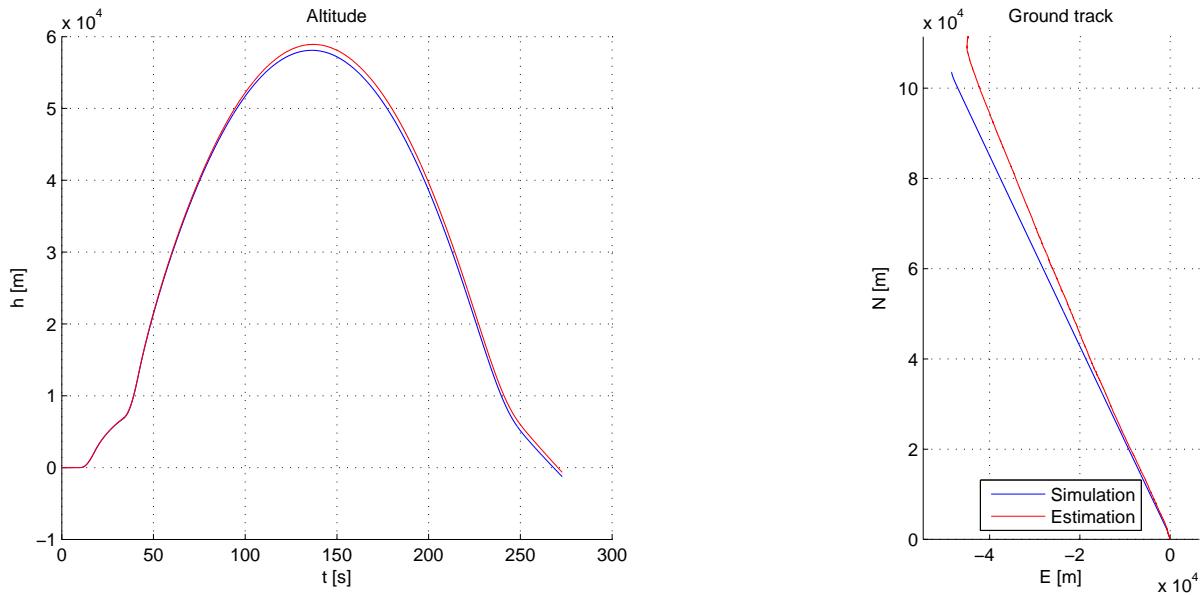


Figure 7.36: Estimated altitude and ground track with biased sensors with the kinematic model EKF

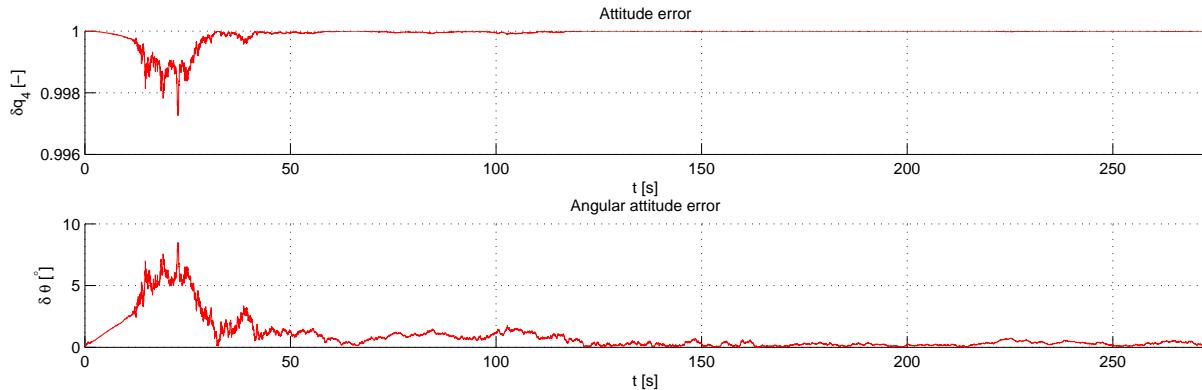


Figure 7.37: Attitude estimation error with biased sensors with the kinematic model EKF

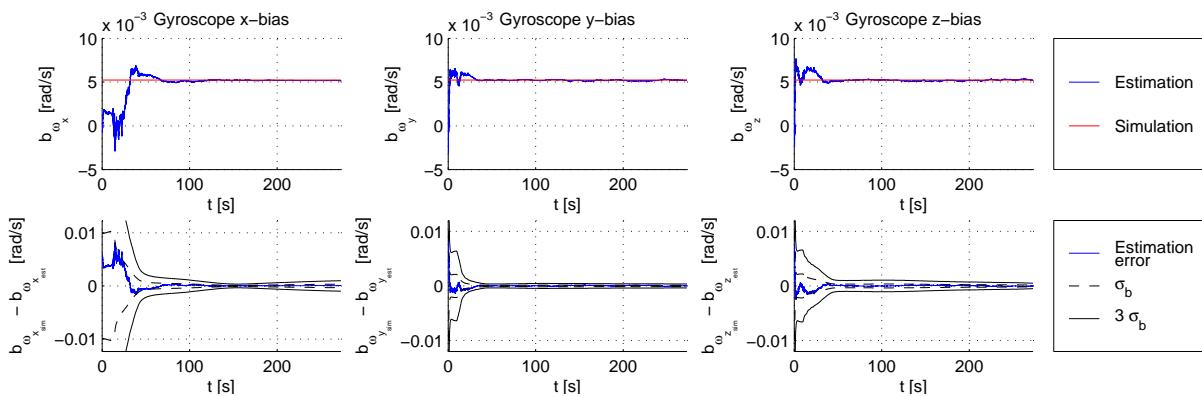


Figure 7.38: Estimated bias with the kinematic model EKF

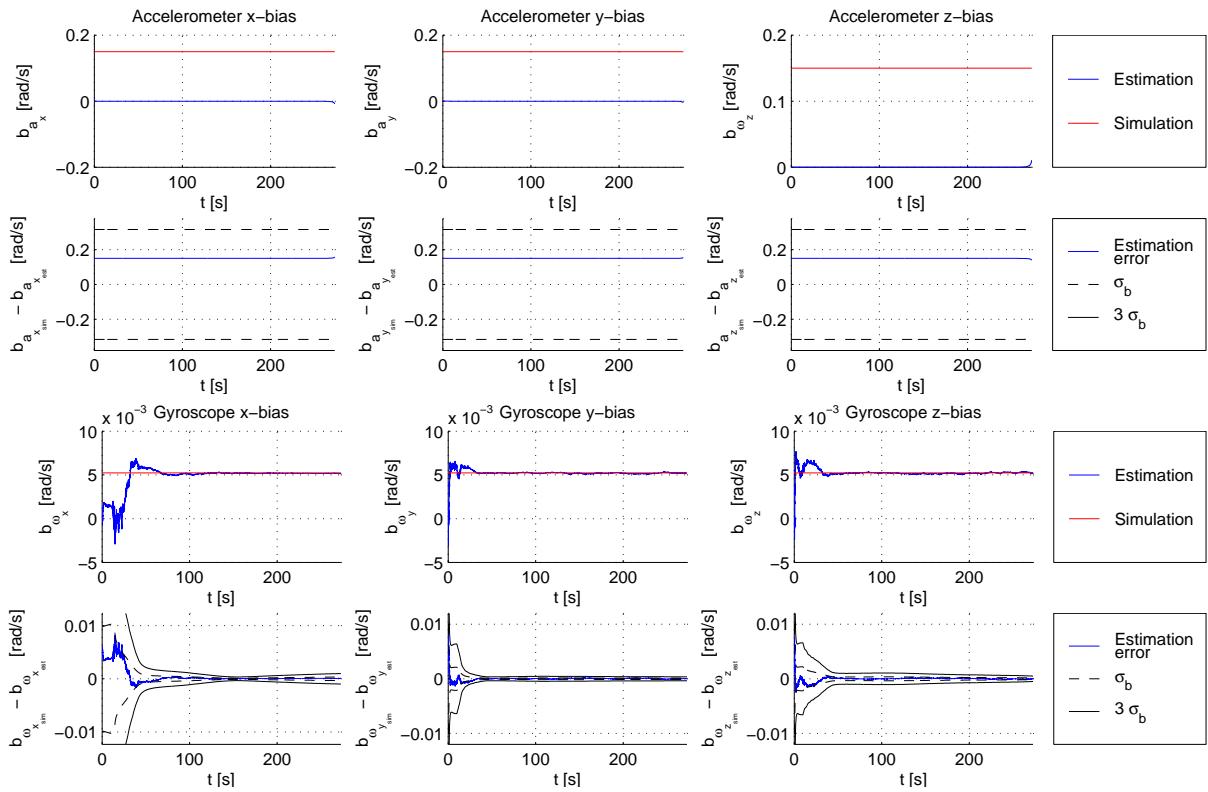


Figure 7.39: Estimated bias with accelerometer bias estimation with the kinematic model EKF

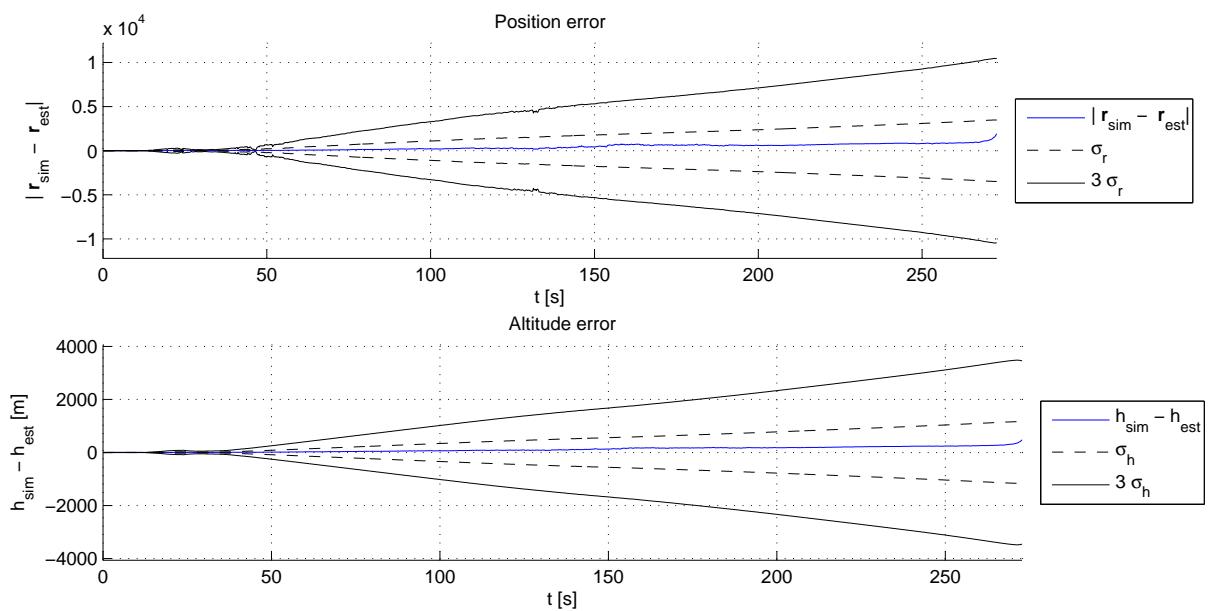


Figure 7.40: Non-ideal rocket position estimation error with the kinematic model EKF

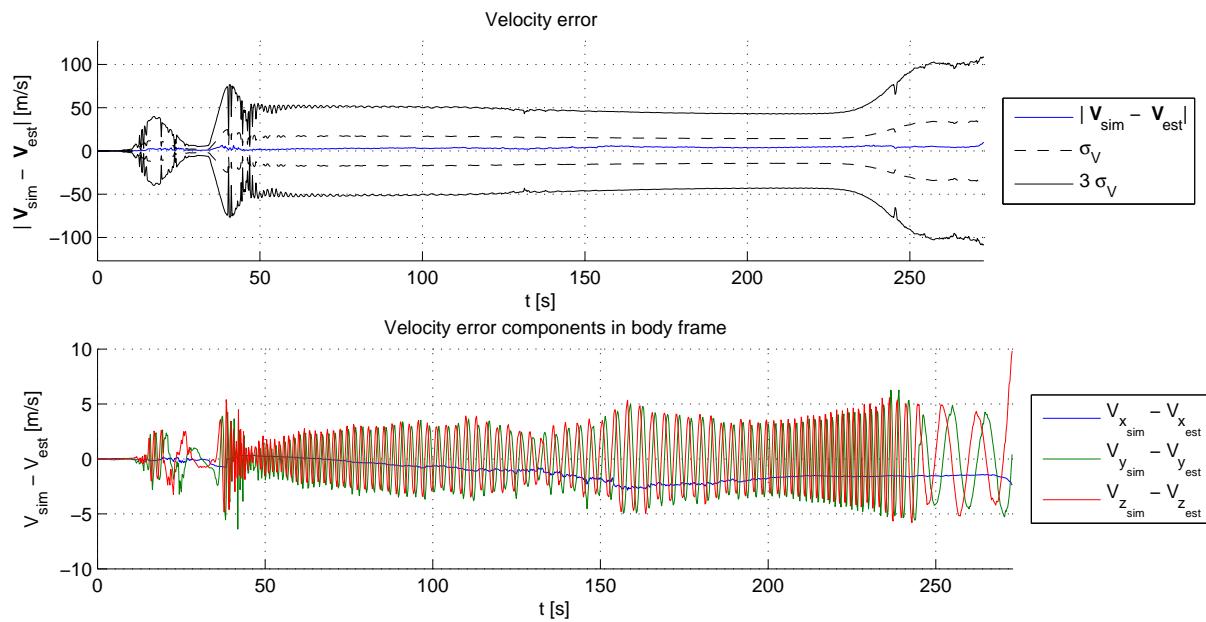


Figure 7.41: Non-ideal rocket velocity estimation error with the kinematic model EKF

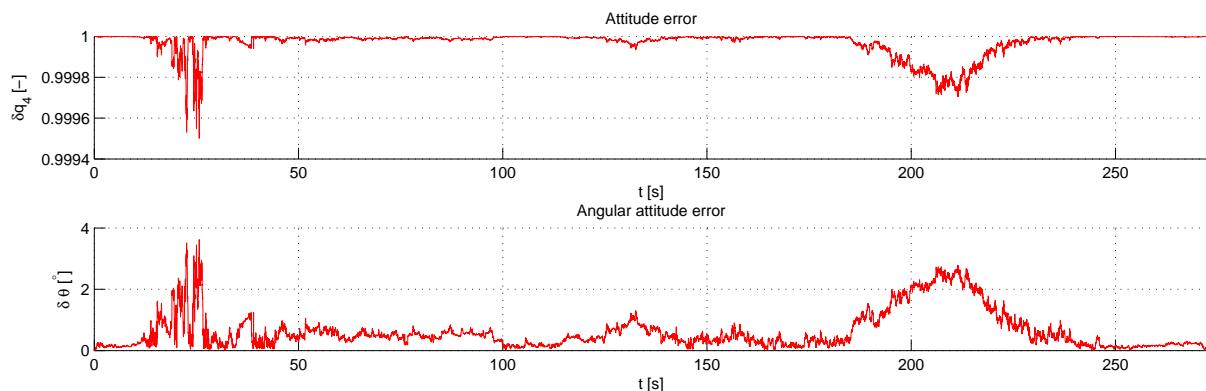


Figure 7.42: Non-ideal rocket attitude estimation error with the kinematic model EKF

estimate accelerometer bias. The estimation errors also show this: with the assumed biases, the position estimation error increases more than tenfold. With the simplified model estimator the bias only doubles the estimation error. Still, the estimation error is larger than the kinematic model estimator. The dynamic circumstances of the non-ideal rocket prove to be much more difficult for the estimator than the ideal rocket. Both estimators are able to estimate this flight, although yielding an increased estimation error.

For the simplified model estimator, the UKF does not improve the estimation result; it even renders the estimation result totally unusable. For the model-free estimator, the UKF does improve the estimation result, but only marginally, and only for certain variables. The maximum estimation error does decrease in general though. Given the tenfold increase in computation time with respect to the EKF though, it is unlikely that it is worth to use the UKF.

Summing up these results, there clearly is a preferred estimator: the kinematic model EKF. The performance of this estimator is vastly superior over the other estimators. The only downside is the performance with a biased accelerometer. Therefore, further analysis, such as the effect of sensor noise, failure, measurement frequency, and various flight trajectories will be performed in the following sections.

Table 7.3: Estimation errors for simplified model EKF

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\Delta\gamma$ [$^\circ$]	$\Delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	-0.00
Stage separation	19.31	-0.00	374.53	-341.76	51.40	0.46	-0.02	0.62
Second stage burnout	41.81	-0.04	1061.77	-976.27	34.44	3.97	0.23	-2.64
Apogee	136.36	-0.60	4195.29	-1752.01	38.23	8.77	1.25	-8.66
Nose cone separation	243.12	-3.06	3599.74	-1390.14	16.09	0.62	-0.05	0.94
Parachuting	257.20	-6.46	4007.86	-1634.12	31.88	5.32	-0.22	6.92
Landing	268.11	-4.74	4255.05	-1898.14	37.05	18.73	-0.32	24.13
Mean error			2614.70	-1367.61	29.11	6.55	0.48	-4.12
Maximum error			4686.61	-2045.94	107.27	125.95	2.35	134.35

Table 7.4: Estimation errors for simplified model UKF

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\Delta\gamma$ [$^\circ$]	$\Delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	-0.00
Stage separation	19.31	-0.00	251.23	-10.69	56.36	16.50	-0.63	12.20
Second stage burnout	41.81	-0.05	2573.45	-1015.10	267.45	20.96	-1.26	18.71
Apogee	136.36	1.91	18497.02	-703.00	168.34	12.43	-1.23	12.31
Nose cone separation	243.12	-29.73	26363.04	-16137.59	195.10	18.32	-0.87	22.21
Parachuting	257.20	-15.65	28120.46	-14810.67	66.29	20.00	-0.60	26.96
Landing	268.11	-4.74	28753.34	-15191.96	61.41	26.84	-0.49	37.78
Mean error			16549.10	-3551.76	165.04	16.93	-0.36	8.85
Maximum error			29809.51	-17244.21	367.74	-83.52	62.57	-87.22

Table 7.5: Estimation errors for simplified model EKF with biased sensors

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\Delta\gamma$ [$^\circ$]	$\Delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.03
Stage separation	19.31	-0.00	237.21	-206.10	42.12	17.62	0.63	-14.52
Second stage burnout	41.81	-0.04	1537.08	-1160.96	157.36	14.66	0.96	-12.29
Apogee	136.38	-1.65	10574.93	-2130.19	96.52	16.05	1.77	-15.91
Nose cone separation	243.16	-12.08	13949.69	-4185.86	23.41	6.99	0.37	-7.48
Parachuting	257.24	-15.65	13776.63	-4495.69	31.66	3.77	0.02	-3.28
Landing	268.15	-4.74	13641.94	-4823.36	38.02	13.49	-0.24	17.58
Mean error			8174.77	-2239.44	70.79	12.79	1.06	-11.16
Maximum error			14094.01	-4985.38	158.72	114.97	4.32	133.16

Table 7.6: Estimation errors for simplified model EKF of non-ideal flight

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\Delta\gamma$ [$^\circ$]	$\Delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	-0.00
Stage separation	19.31	-0.00	207.23	-192.79	34.43	7.41	0.26	-5.76
Second stage burnout	41.81	-0.06	1541.89	-1267.27	183.39	15.35	1.10	-14.41
Apogee	136.37	-3.52	16711.17	-3247.84	157.84	22.08	2.45	-21.90
Nose cone separation	243.15	-12.11	15733.86	-4311.68	6.72	2.73	0.12	-2.04
Parachuting	257.23	-15.65	14342.90	-4491.48	48.19	15.83	-0.46	21.01
Landing	268.14	-4.74	13886.83	-4802.17	59.68	38.27	-0.60	62.20
Mean error			11264.12	-2913.21	109.45	17.37	1.37	-11.25
Maximum error			16810.18	-5045.37	185.28	110.87	3.83	110.87

Table 7.7: Estimation errors for kinematic model EKF

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\Delta\gamma$ [$^\circ$]	$\Delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Stage separation	19.31	-0.01	30.87	1.26	5.09	3.49	0.39	2.41
Second stage burnout	41.81	-0.01	31.70	-12.77	4.22	0.40	0.03	0.38
Apogee	136.36	0.48	255.31	-18.55	2.21	1.04	0.19	1.03
Nose cone separation	243.12	-0.14	318.76	-65.05	2.84	0.54	0.06	0.63
Parachuting	257.20	-0.32	255.22	-86.44	8.96	1.23	0.12	1.58
Landing	268.11	-0.45	250.89	-114.23	10.03	1.18	0.19	1.76
Mean error			222.84	-25.21	3.03	1.11	0.13	1.11
Maximum error			474.38	-221.29	11.98	6.09	0.83	4.55

Table 7.8: Estimation errors for kinematic UKF

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\Delta\gamma$ [$^\circ$]	$\Delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Stage separation	19.31	-0.01	52.00	4.28	8.66	5.76	0.62	4.26
Second stage burnout	41.81	-0.01	35.80	-11.72	3.48	0.21	-0.01	0.23
Apogee	136.36	0.48	326.93	-16.19	2.91	0.61	0.12	0.60
Nose cone separation	243.12	-0.14	430.65	-72.42	1.41	0.06	0.01	-0.05
Parachuting	257.20	-0.36	362.56	-93.50	2.75	0.57	0.07	0.78
Landing	268.11	-0.47	327.98	-117.17	3.09	0.47	0.13	0.83
Mean error			264.15	-29.47	2.65	0.78	0.09	0.72
Maximum error			455.99	-228.17	11.31	7.62	0.95	5.87

Table 7.9: Estimation errors for kinematic model EKF with biased sensors

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\Delta\gamma$ [$^\circ$]	$\Delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Stage separation	19.31	-0.01	67.54	-16.94	9.88	7.09	0.86	5.18
Second stage burnout	41.81	-0.01	291.20	-125.94	23.18	1.56	-0.21	-1.24
Apogee	136.36	-1.64	2410.60	-816.17	34.42	0.22	0.13	0.19
Nose cone separation	243.12	-1.80	7066.85	-887.70	56.06	0.37	-0.00	0.53
Parachuting	257.20	-2.68	7858.54	-775.48	58.69	0.19	0.04	0.30
Landing	268.11	-2.37	8479.57	-657.89	60.98	0.31	0.06	0.82
Mean error			3046.76	-624.10	33.46	0.87	0.07	0.51
Maximum error			8707.23	-1078.08	62.29	8.49	1.11	6.34

Table 7.10: Estimation errors for kinematic model EKF of non-ideal flight

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\Delta\gamma$ [$^\circ$]	$\Delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Stage separation	19.31	-0.01	15.34	-5.14	2.61	1.82	-0.32	-1.13
Second stage burnout	41.81	-0.01	34.95	-22.47	3.67	0.42	-0.39	-0.19
Apogee	136.37	0.55	264.36	-95.27	2.61	0.77	0.05	0.76
Nose cone separation	243.15	-0.47	839.26	-233.77	5.88	0.15	-0.01	0.26
Parachuting	257.23	-0.91	849.13	-247.48	5.44	0.16	-0.12	0.16
Landing	268.14	-1.16	1008.42	-290.23	4.25	0.28	-0.02	-0.53
Mean error			406.04	-118.10	3.44	0.62	0.03	0.34
Maximum error			1935.85	-504.51	10.13	3.62	-0.57	3.42

7.5 Sensor factors

In the previous section, it has been concluded that the model-free EKF is the most suited estimator for real-time estimation of the Stratos II flight. In this section, the performance of this estimator under varying sensor factors will be evaluated. First, the performance in the presence of GPS pseudorange measurements is evaluated, and the possibility of bias estimation with these measurements. Then, the sensor noise factors are considered, followed by the sensor control factors.

7.5.1 GPS measurements and bias estimation

Previous estimation results were obtained without any GPS measurements, in fact, aside from the accelerometer, there were no sensors that could provide position information. It was shown that this also had as effect that no accelerometer bias estimation was possible. As described in section 4.2.5, a major problem with using GPS in a sounding rocket, is that commercially available GPS receivers are limited by the COCOM limits of 60,000 feet altitude, or 1000 knots velocity. For Stratos II these limits are crossed respectively just after second stage burnout and during second stage burn.

There are two ways of processing GPS data (Aggarwal et al., 2010). The first possibility is using the position estimate from the GPS receiver. This is called *loose coupling*. This requires a full position fix from the GPS receiver. The second way is *tight coupling*, where the pseudorange to the satellites are directly processed as measurements. This is already possible when a fix to only one satellite is available. Only some commercial GPS receivers do output the pseudoranges; most of them only output the position estimate. All GPS receivers output this data at a frequency of at least 1 Hz, although some of receivers output at higher frequencies such as 5 Hz or even 10 Hz. All estimations in this section are done with a frequency of 1 Hz. Using measurements at higher frequencies would obviously improve the estimation error.

Loose coupling

First, a loose GPS coupling is performed for the nominal configuration without sensor bias. The GPS measurement equations are given by equations (4.23) and (4.24). In table 7.11, the results are presented. In general, compared to the nominal case in table 7.7, the presence of GPS data improves the estimation results, without any noticeable downside. The mean position error decreases by a factor two, and also the other performance measures significantly improve.

Then, the loose coupling is combined with a bias on the sensors. Figure 7.43 shows the resulting position estimate. During the initial phase where a GPS fix is available, the estimate stays within bounds, but after the rocket passes through the COCOM limits, the position estimate quickly diverges. Then, around 250 s, a jump can be seen both in altitude and ground track. This corresponds to the time the rocket comes below the COCOM limits again.

At first instance, it looks like the addition of GPS is not much helpful in the presence of biased sensors, because GPS measurements are not available in the major part of the flight. The power of an added GPS however, is that it provides extra position information, which means that the bias on the accelerometer can be estimated, something that originally was not possible. This is done, with the results in table 7.12, and the position in figure 7.44. In the altitude plot, the jump when the rocket passes through the COCOM limits is still visible, but in the ground track it is clear that the bias estimation causes the estimated flight track to match the real flight track much more accurately. Figure 7.45 shows estimated bias, and it can be seen that in the short time that there is GPS data available, the bias converges to its actual value. The GPS-less part of the flight is also easily identified, as the part where the covariance does no longer converge. At the end of flight, the passing through the COCOM limits can be found

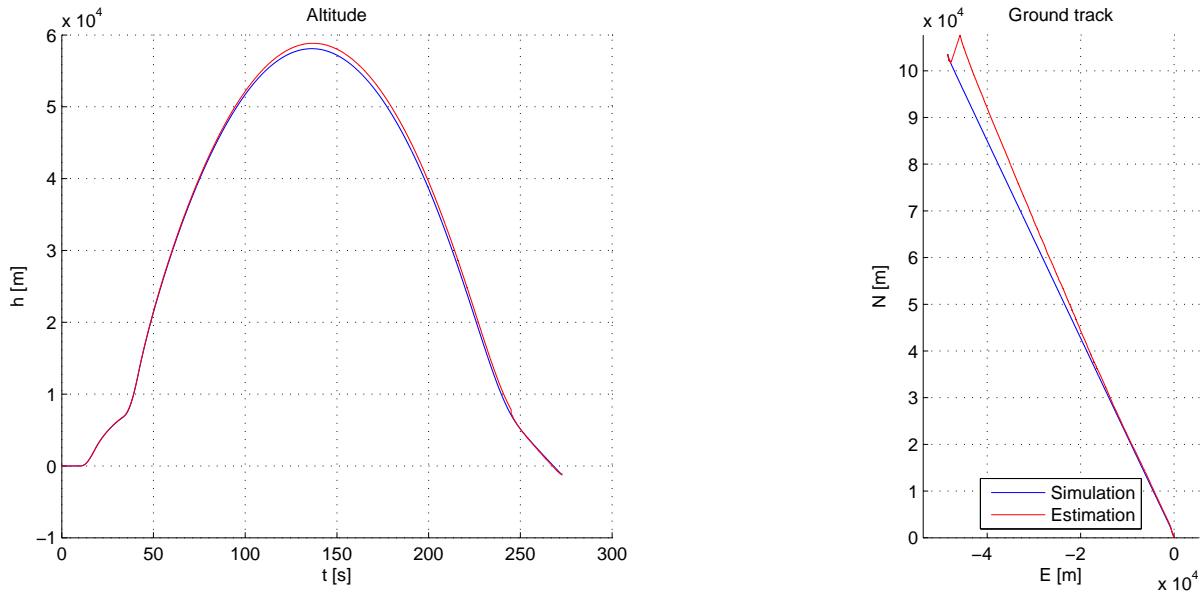


Figure 7.43: Estimated altitude and ground track with biased sensors and loose GPS coupling

when the bias converges again. With the bias estimation, the estimation results are an order of magnitude better than when bias is not estimated at all.

Tight coupling

Next, tight GPS coupling is performed. For all estimations with tight coupling, it is simulated that the satellites are available at orthogonal positions, with at most five satellites available. One satellite would be located at zenith, directly above the launch tower, with the others at the horizon at in northwards, westwards, southwards and eastwards location. It is well established in the GPS field, that the geometry of available satellites has a strong influence on the accuracy of the position estimate. The used configuration is the most optimal configuration with respect to geometry, as it yields the lowest dilution of precision. However, as not only configurations with five satellites will be evaluated, but also with four or less, the estimation results are still useful.

The estimated bias with 5 satellites is presented in figure 7.46. Compared to the bias for loose coupling from figure 7.45, the convergence to the real bias is slightly quicker, and the result is a lower estimation error: in the tightly coupled case, the estimation error is about twice as low as for the loosely coupled case.

In figures 7.47a and 7.47b the position and attitude error for both loose and tight coupling for various numbers of GPS satellites have been plotted. The figures show that tight coupling with 5 satellites, and for some configurations also with 4 satellites, tight coupling is better than loose coupling. However, there are also configurations where four satellites is worse than loose coupling. What this figure does not show is the fact that if only 3 or even less satellites are in view, loose coupling is not possible at all.

Clearly, introducing or adding more GPS pseudorange measurements does not always improve the result. For example, there is a case where a single satellite is worse than no satellites. To study these cases, in appendix F.1 the accelerometer bias estimation for some cases has been plotted, numbered according to maximum position error per satellite, as shown in figure F.1. The difference between no satellites and the worst case with a single satellite is explained by the convergence of the accelerometer bias. Without satellites (case 1, figure F.2), y- and z-component of the bias stay zero. However, with the specific single satellite (case 2, figure F.3), the y-bias slightly diverges, causing the stronger position divergence. In contrast, for case 4,

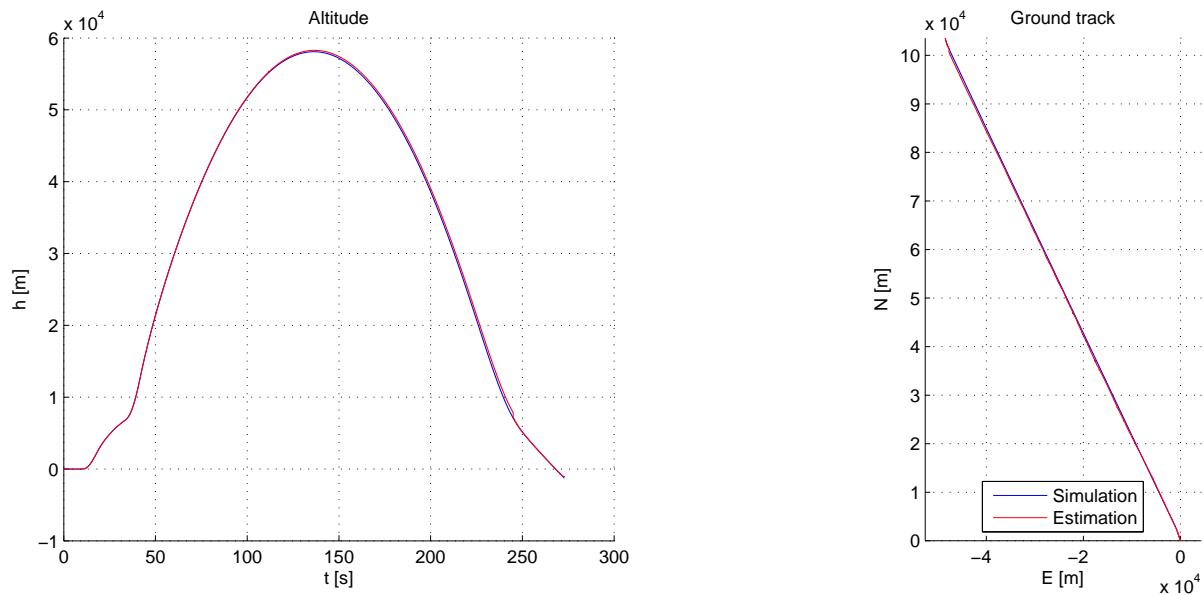


Figure 7.44: Estimated altitude and ground track with bias estimation and loose GPS coupling

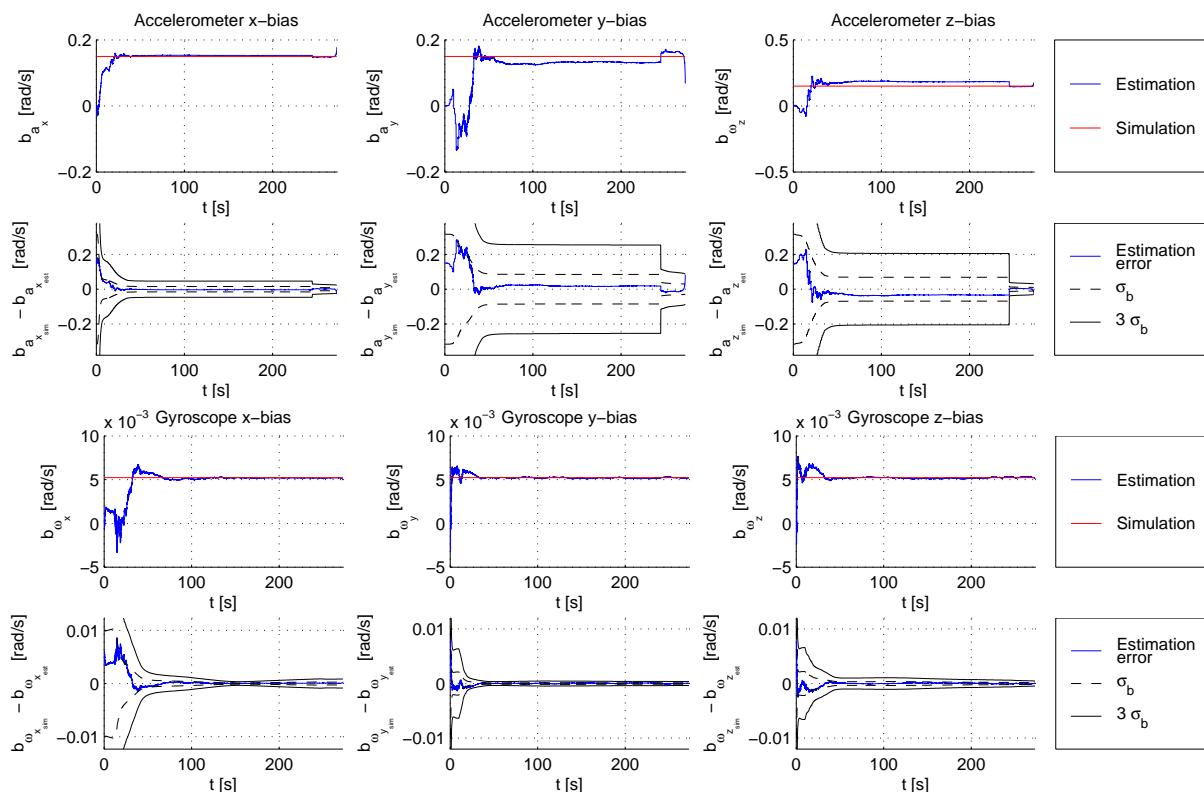


Figure 7.45: Estimated bias with loose GPS coupling

Table 7.11: Kinematic model EKF estimation errors with loose GPS coupling

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\Delta\gamma$ [$^\circ$]	$\Delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Stage separation	19.31	-0.01	32.36	1.50	5.35	3.67	0.41	2.54
Second stage burnout	41.81	-0.01	19.52	-15.00	1.76	0.15	0.00	0.17
Apogee	136.36	0.48	125.66	-38.83	1.01	0.84	0.17	0.82
Nose cone separation	243.12	-0.20	141.28	-99.10	2.53	0.36	0.03	0.42
Parachuting	257.20	-0.04	16.10	-10.56	1.56	0.24	0.00	0.36
Landing	268.11	-0.05	15.41	-11.06	0.91	0.08	0.05	0.28
Mean error			104.06	-33.20	1.50	0.81	0.09	0.78
Maximum error			253.66	-220.58	7.52	4.66	0.54	3.25

Table 7.12: Kinematic model EKF estimation errors with bias estimation and loose GPS coupling

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\Delta\gamma$ [$^\circ$]	$\Delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Stage separation	19.31	-0.01	37.75	4.73	7.41	7.37	0.90	5.38
Second stage burnout	41.81	-0.01	54.86	-26.45	11.79	1.33	-0.18	-1.04
Apogee	136.36	0.40	300.02	-189.32	4.52	0.38	0.15	0.36
Nose cone separation	243.12	-1.56	900.58	-757.67	7.04	0.53	0.02	0.72
Parachuting	257.20	-0.02	20.51	-5.34	1.08	0.20	0.04	0.32
Landing	268.11	-0.05	9.03	-7.21	0.89	0.33	0.07	0.86
Mean error			309.63	-206.62	4.73	0.96	0.09	0.64
Maximum error			910.92	-770.82	17.16	8.14	1.02	5.98

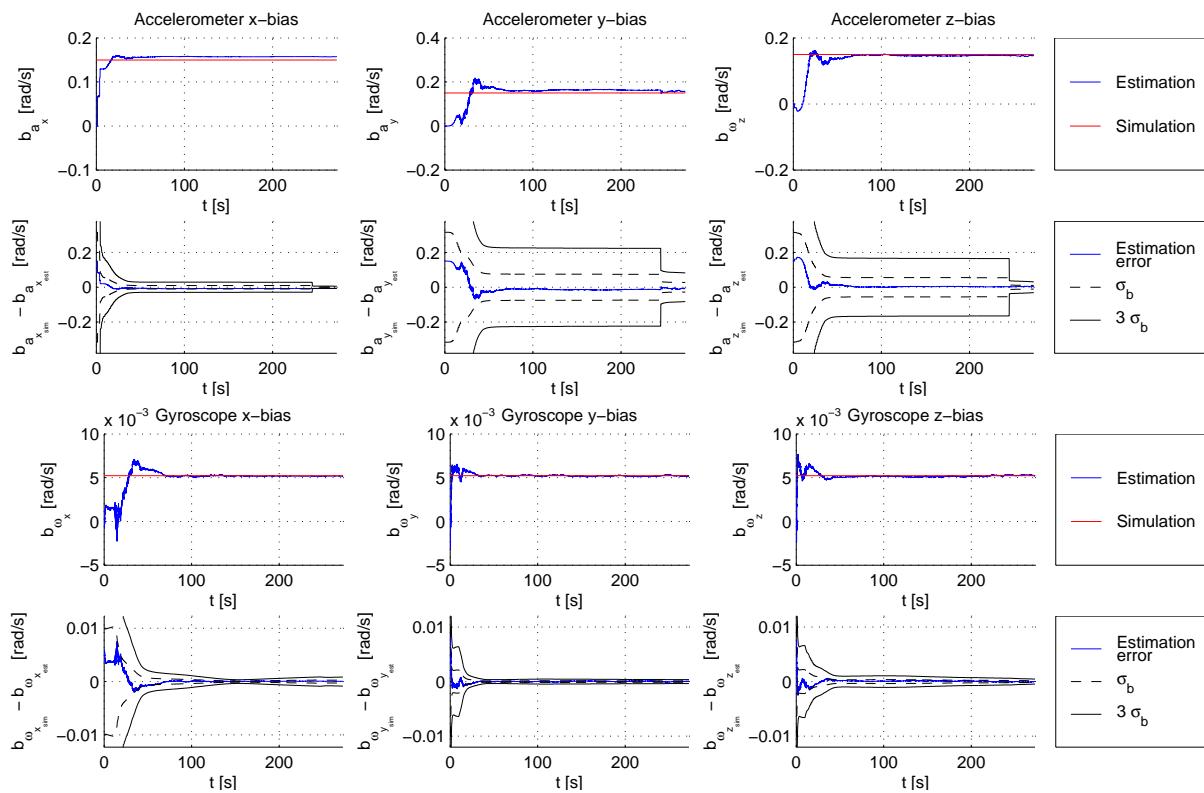


Figure 7.46: Estimated bias with tight GPS coupling

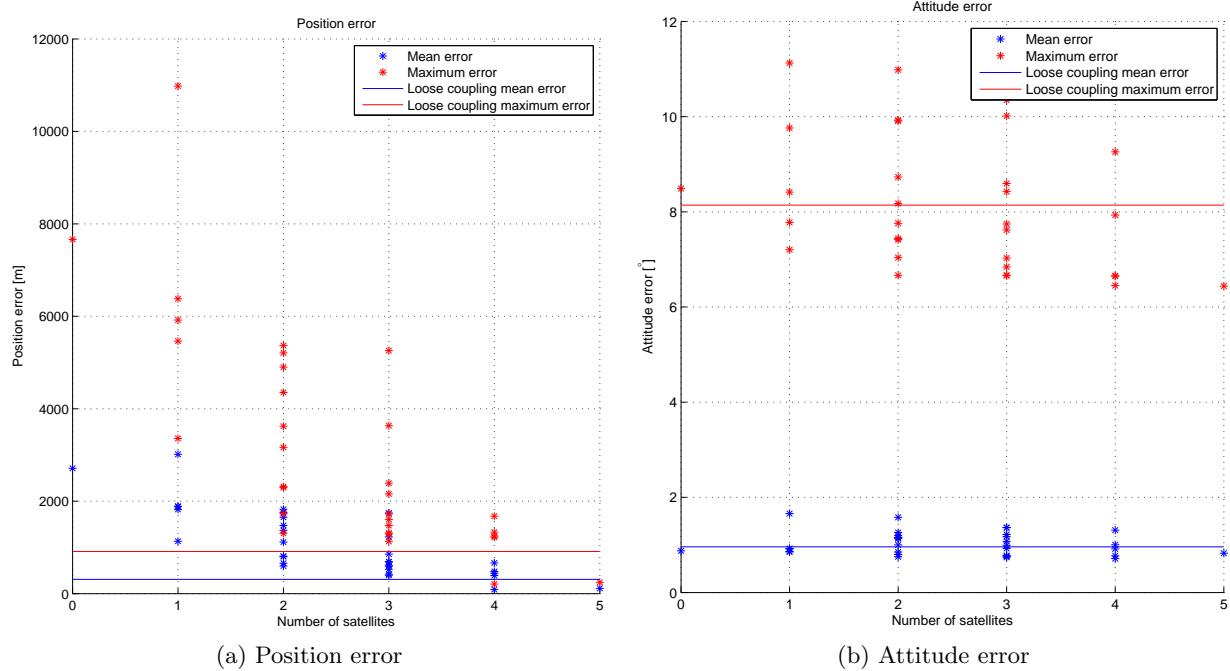


Figure 7.47: Position and attitude error for various levels of GPS coupling

still with one satellite, it can be seen in figure F.5, that all biases partially converge. Because of this convergence, and despite the overshoot on the x- and z-axes, the estimation result is slightly better than without GPS. For case 3 in figure F.4, the results are in between case 2 and case 4, as the y-bias almost fully converges, while the z-bias slightly diverges.

Also case 31, the best 4-satellite case, and case 32, the 5-satellite case are shown in figures F.6 and F.7, respectively. The position estimation error for the 4-satellite case is actually lower than the 5-satellite case, by approximately 30 m. The bias figures clearly shows that the biases in case 31 matches the actual bias slightly better. This effect is due to the stochastic nature of the process. Even after convergence, the estimated bias naturally shows random deviations from the true bias. The loss of the GPS signal happens coincidentally at a more favorable time for case 31 than for case 32. Slight deviations in flight may alter this time such that it is more favorable to case 32. Therefore, for all practical purposes, case 31 and case 32 can be considered equivalent.

For the attitude, the behavior is truly stochastic. For all number of satellites, there are cases that yield both improvements and worsen the estimate compared to the nominal case. Also the attitude error appears to be fully uncorrelated with the position error. Therefore, the only conclusion that can be drawn is that GPS measurements do affect the attitude estimation error, but that the effect is random, and that differences up to 3° , in either direction can be expected.

Table 7.13: Kinematic model EKF estimation errors with bias estimation and tight GPS coupling

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\Delta\gamma$ [$^\circ$]	$\Delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Stage separation	19.31	-0.01	21.37	3.81	4.65	4.83	0.59	3.51
Second stage burnout	41.81	-0.01	82.03	-24.32	19.16	2.18	-0.28	-1.78
Apogee	136.36	0.48	175.80	-38.96	1.08	0.56	0.04	-0.55
Nose cone separation	243.12	-0.02	168.03	-3.46	3.58	0.33	-0.09	-0.25
Parachuting	257.20	-0.00	8.77	1.07	1.49	0.07	0.03	0.16
Landing	268.11	-0.07	17.96	-15.19	0.93	0.22	0.05	0.71
Mean error			115.23	-22.71	2.44	0.83	-0.03	-0.25
Maximum error			243.71	-123.41	22.52	6.44	0.75	4.76

7.5.2 Sensor control factors

Previous estimation results were obtained by simulating sensors with the characteristics of the developed measurement system. However, it may be possible that with different sensor characteristics, much better estimation results can be obtained. This topic is the subject of this section.

The three most influential sensors are the accelerometer, gyroscope and magnetometer. For each of these sensors, measurement frequency and noise are control factors, as sensors can be selected based on these factors. Although each of these sensors actually contains three sensors themselves, in general they cannot be selected individually. This leaves six parameters: a noise and a frequency parameter for each of the sensors.

First, the general influence of the noise and frequency are determined by varying the noise and frequency for all of the sensors with the same ratio. Then, an orthogonal array is used to obtain the response surface for varying the noise and frequency of the sensors individually.

Noise and frequency

To study the dependence of noise and measurement frequency on the accuracy of the estimator, these two parameters have been varied for the nominal flight. The measurement frequency for the gyroscope and accelerometer have been set equal to each other and varied between 0.25 and 4.0 times the nominal value. Equal multiplicative factors, varying between 0.5 and 2.0, have been applied to the measurement noise. Then, the most important performance parameters, the position error, altitude error, attitude error and elevation error have been plotted in figures 7.48 and 7.49. These data has also been tabulated in appendix F.2

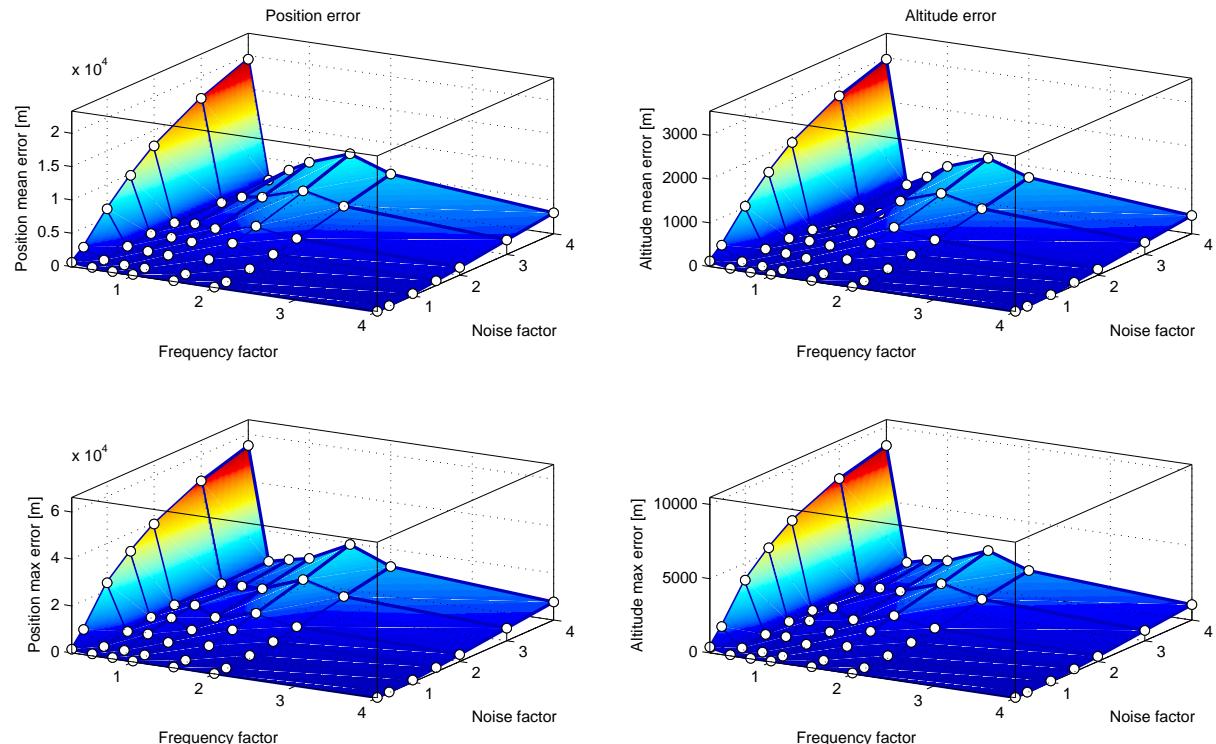


Figure 7.48: Translational error dependence on noise and frequency

In general the estimation error decreases with increasing frequency and decreasing noise. However, this applies to a limited fashion to the maximum attitude error: the maximum attitude error only improves little with decreasing noise or increasing frequency, much less than the other performance measures. Conversely, decreasing the frequency and increasing the noise has as a

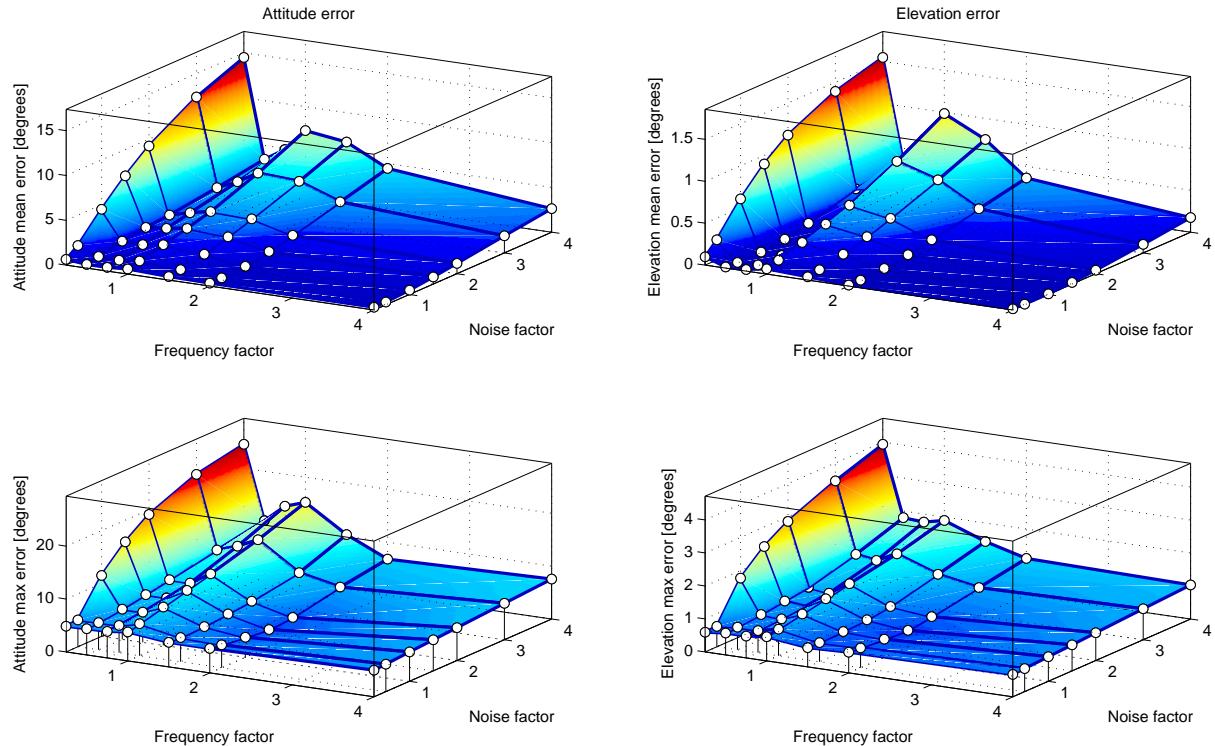


Figure 7.49: Attitudinal error dependence on noise and frequency

consequence that the estimation error increases. And, although this is also true for the elevation angle, for the studied range of noises and frequencies the elevation angle estimation error always stays very small, under 4 degrees.

The nominal value for the position mean error is 734 m. By decreasing the noise to 0.25 times the nominal RMS noise, the error can be reduced a factor 5 to 170 m. Increasing the measurement frequency fourfold would half the estimation error. Performing both improvements, the mean position error can be reduced a factor 20 in total, to only 39 m. The behavior in the other direction is also notable. While decreasing the measurement frequency a factor 2, only makes the position error grow by approximately 50 % to 1180 m, decreasing it a factor 4 yields an almost tenfold increase of estimation error to 6333 m. It appears that 125 Hz, half the nominal frequency, is the lower limit for a reasonable estimation result. For the noise factor the effects are much less pronounced, and in the studied range there appears to be no clear bound where the estimation result starts to grow unbounded.

It is surprising to see that the estimation error is not fully monotonic: there is a local maximum around the frequency factor of 1 to 1.5. This effect is also more pronounced in attitude than in position. The exact reason for these local maximums is not known, but in general the estimation error is stochastic and non-linear, so small changes in input parameters may have unpredictable effects on the output. This effect has as consequence, that once the maximum measurement frequency for the navigation system has been determined, simulations must be performed with slightly lower measurement frequencies, to determine if the estimation error becomes better at these frequencies.

The obtained results show that already the nominal sensor specifications are reasonable for use in the Stratos II. For separation and parachuting, it may though be required that the altitude error decreases, by increasing the measurement frequency or decreasing the noise. The elevation angle estimation error stays small for the full range of noise and frequency, and if only this measure were concerned, much worse sensors could be used.

Individual sensor characteristics

In the previous section, the influence of noise and frequency on the estimation accuracy was studied, where both the accelerometer and gyroscope had their respective noise and frequency changed with equal factors for either sensor. However, this does not show the influence of individual sensors. To study the effect of varying noise and frequency of individual sensors, a three-level full factorial design has been performed, where the noise and frequency of the accelerometer, gyroscope and magnetometer have been varied between 0.5, 1.0 and 1.5 times their nominal value. From these results, a response surface has been created, with the normalized coefficients shown in figure 7.50, and the unnormalized coefficients in table 7.14. The response conforms to the following equation:

$$\begin{aligned} r = r_{nom} &+ (1 + 0.5f_{N_a})\beta_{N_a} + (1 + 0.5f_{f_a})\beta_{f_a} \\ &+ (1 + 0.5f_{N_\omega})\beta_{N_\omega} + (1 + 0.5f_{f_\omega})\beta_{f_\omega} \\ &+ (1 + 0.5f_{N_m})\beta_{N_m} + (1 + 0.5f_{f_m})\beta_{f_m} \end{aligned} \quad (7.10)$$

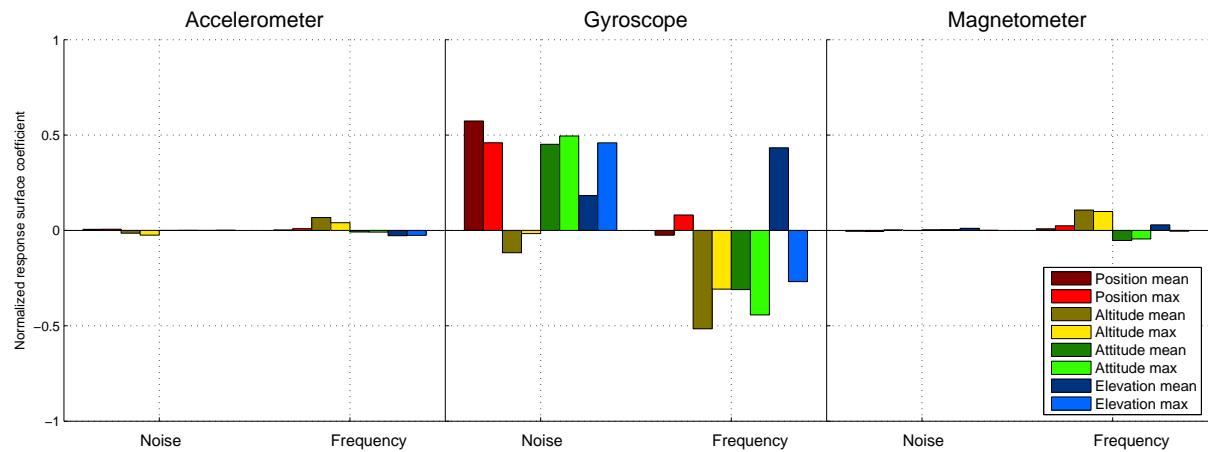


Figure 7.50: Normalized response surface coefficients of sensor control factors

From the RMS residual it can be found that the quality of fit is good for most responses. For altitude, the fit is slightly worse than for other responses, but still reasonable. The same appears to apply for elevation mean, however, in this case the nominal elevation is so small, that a residual that is small in magnitude is still relatively large.

The response surface clearly shows that neither the accelerometer, nor the magnetometer affects the results significantly. The only sensor for which the noise and frequency are significant in the studied range is the gyroscope. For this sensor the results are as expected: in general, increasing the noise increases the estimation error, while increasing the frequency decreases the estimation error. The noise characteristics are dominant over the frequency characteristics for the position, while for the other responses the importance is for the same order of magnitude. The effect that only the gyroscope is significant greatly simplifies design, as only attention must be paid to an improved gyroscope, if the performance characteristics of the selected sensors are insufficient. Also, it shows that the issues with the noise of the accelerometer that were found in the calibration in section 5.2.5, do not significantly alter the estimation result.

7.5.3 Sensor noise factors

Previous estimation results were obtained with only white noise and bias as noise sources. However, as described in section 4.2, there are more noise factors that apply to the accelerometer, gyroscope and magnetometer. In total, there are up to 36 sensor noise factors; for each sensor 3 scale factor errors, 6 non-orthogonalities and 3 biases. Since there are too many factors for full

Table 7.14: Response surface coefficients for sensor control factors

	r_{nom}	β_{N_a}	β_{f_a}	β_{N_ω}	β_{f_ω}	β_{N_m}	β_{f_m}	ε_{RMS}	$\bar{\varepsilon}_{RMS}$
Position mean	291.45	1.67	0.65	167.18	-7.32	-1.38	2.59	79.56	0.27
Position max	930.73	6.01	8.76	428.25	75.39	-4.80	22.84	91.10	0.10
Altitude mean	-156.09	-2.33	10.51	-18.24	-80.48	0.45	16.71	118.38	0.76
Altitude max	-500.90	-12.40	20.23	-8.27	-154.30	-0.24	49.78	322.71	0.64
Attitude mean	0.31	-0.00	-0.00	0.14	-0.10	0.00	-0.02	0.05	0.16
Attitude max	0.57	0.00	-0.00	0.28	-0.25	0.00	-0.03	0.09	0.16
Elevation mean	0.10	-0.00	-0.00	0.02	0.04	0.00	0.00	0.08	0.74
Elevation max	0.35	0.00	-0.01	0.16	-0.09	0.00	-0.00	0.04	0.11

factorial experiments, estimation is performed with an orthogonal array. The orthogonal array uses 3 levels, and has in total 81 runs and is given in appendix E.1. The results are used to obtain a first order response surface. A single level for all scale factors and non-orthogonalities is taken as 0.02, which can be found in sensor datasheets as typical maximum values. A single level for bias is taken as 0.5 times the nominal noise level.

In figures 7.55, the position, altitude, attitude and elevation error have been plotted, respectively. The first experiment in all plots is the nominal case. It can immediately be found that some of the noise factors may have severe impact on the estimation error. Maximum position errors average to 12 km, but may exceed 25 km, while altitude errors average to 5 km, but may exceed 16 km. The attitude error is also impacted, as is the elevation error. However, for the elevation, even though the estimation error may grow an order of magnitude, in all cases it stays within the very reasonable bound of 4 degrees.

The response surface coefficients have been plotted in figure 7.51 and tabulated in table F.9. In appendix F.3, this has been plotted full-page as figure F.8. The plotted coefficients have been normalized with respect to the nominal response, i.e., $\bar{\beta}_i = \beta_i / r_{nom}$; the tabulated coefficients are unnormalized. The designators in the figure and table are: s are scale factor errors, d are the non-orthogonalities, and b are the biases.

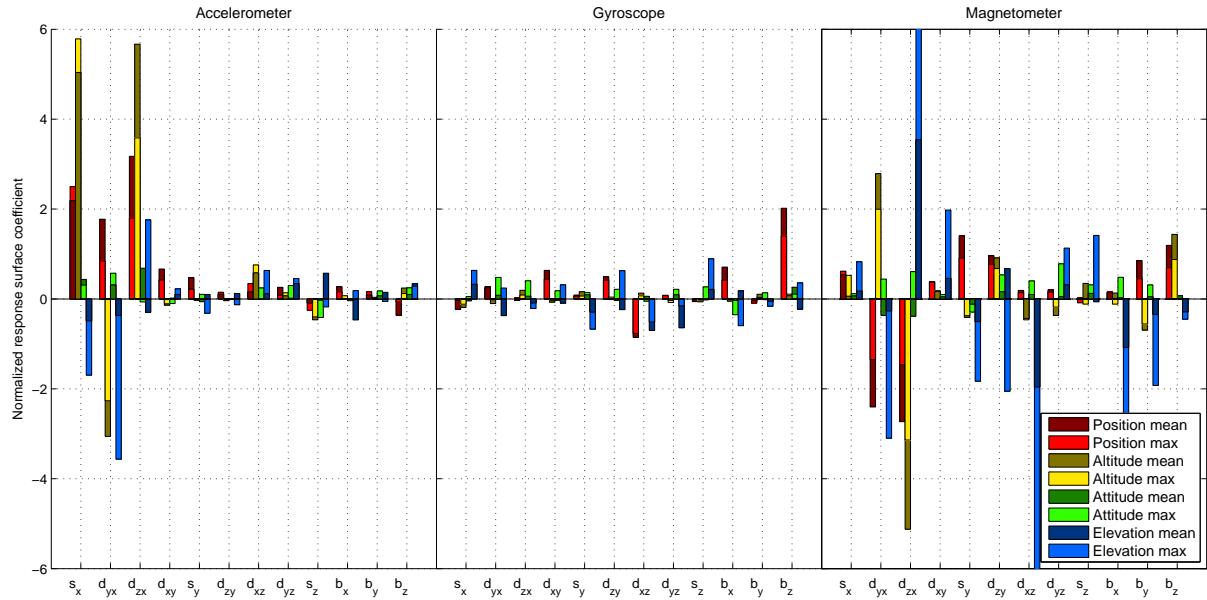


Figure 7.51: Normalized response surface coefficients with only bias estimation

From the coefficients, it appears that the main contributions to position error are the accelerometer scale factor error in x-direction, and the contribution of the x-component of the acceleration to the output on the z-axis. This is because the acceleration in x-direction is large

in magnitude, and therefore any scale factor error or non-orthogonality will contribute significantly. The biases are of much less influence, but that is expected as the state vector already includes the bias. Also the contributions of the magnetometer are significant. The contributions of the gyroscope are small compared to the other sensors. This is in particular interesting, since in section 7.5.2, for the control factors the gyroscope was found to be the dominating sensor.

Seeing these results, it looks that it may be worthwhile to add the scale factor error of the accelerometer in x-direction to the state vector. This way, it can be estimated and hopefully the errors can be reduced. This has been done and the normalized response surface coefficients have been plotted in figure F.9 and tabulated in table F.10. Figures 7.56 contain the estimation errors of all experiments.

The estimated scale factor error, for a run with only the scale factor error is set to 0.02, is given in figure 7.52. It can be seen that the estimated scale factor error quickly converges to its actual value. The error covariance has more difficulty converging, and after a short while stays constant. The reason for this is that due to the noise, too little information is available for the filter to converge the covariance further. If the noise on the pressure sensor is reduced, or GPS measurements are added, it can be seen that the covariance converges much better.

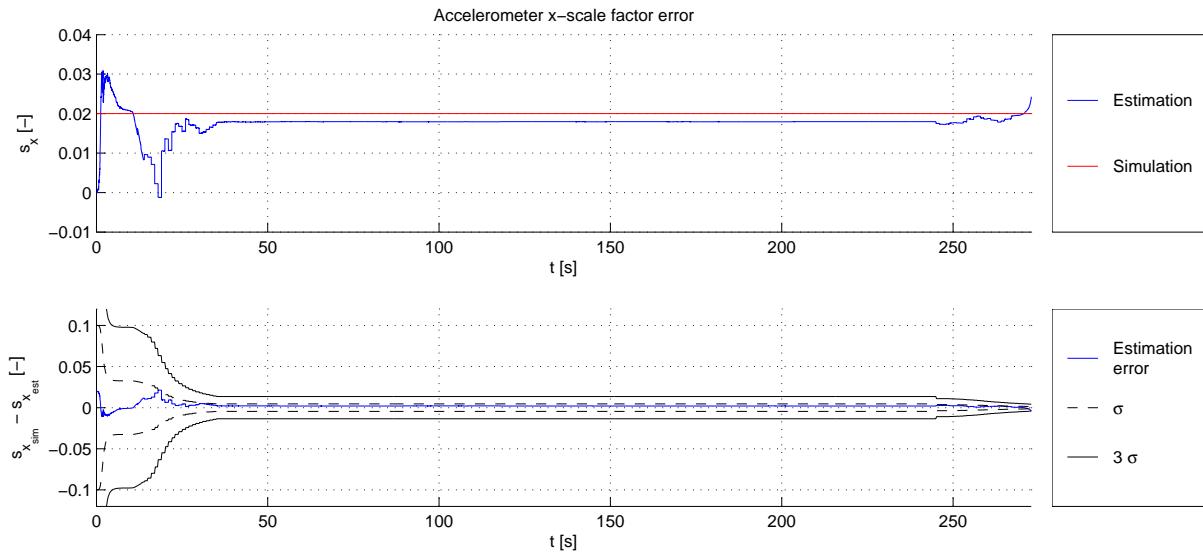


Figure 7.52: Estimated accelerometer x-scale factor error

The apparent next factors which should be estimated are apparently either the contribution of the x-component of the acceleration on the z-axis or the contribution of the x-component of the magnetic field on the z-axis. Since the respective response surface coefficients are of the same order of magnitude, both estimations have been performed. Figures 7.57 and F.10 and table F.11 give respectively the errors and response surface coefficients for the accelerometer non-orthogonality estimation, while figures 7.58 and F.11 and table F.12 give them for magnetometer non-orthogonalities.

Estimation of the accelerometer non-orthogonality can be found to be slightly more efficient than that of the magnetometer. The estimation errors are about 5 % better, both translational and rotational. Figure 7.53 shows the estimated non-orthogonality. It is clear that it has no difficulty converging. From the response surface coefficients it can be found that the next candidate for estimation is the other non-orthogonality, being the accelerometer yx-non-orthogonality.

The yx-non-orthogonality estimation is performed as well, with the results in figure 7.59, and the response surface coefficients in figure F.10 and table F.11. Figure 7.54 shows the estimated non-orthogonality. Also here the non-orthogonality still converges to its actual value. The estimation results have also improved. Position errors larger than 6 km and altitude errors larger than 3 km are no longer present, and the mean altitude mean estimation error has decreased to

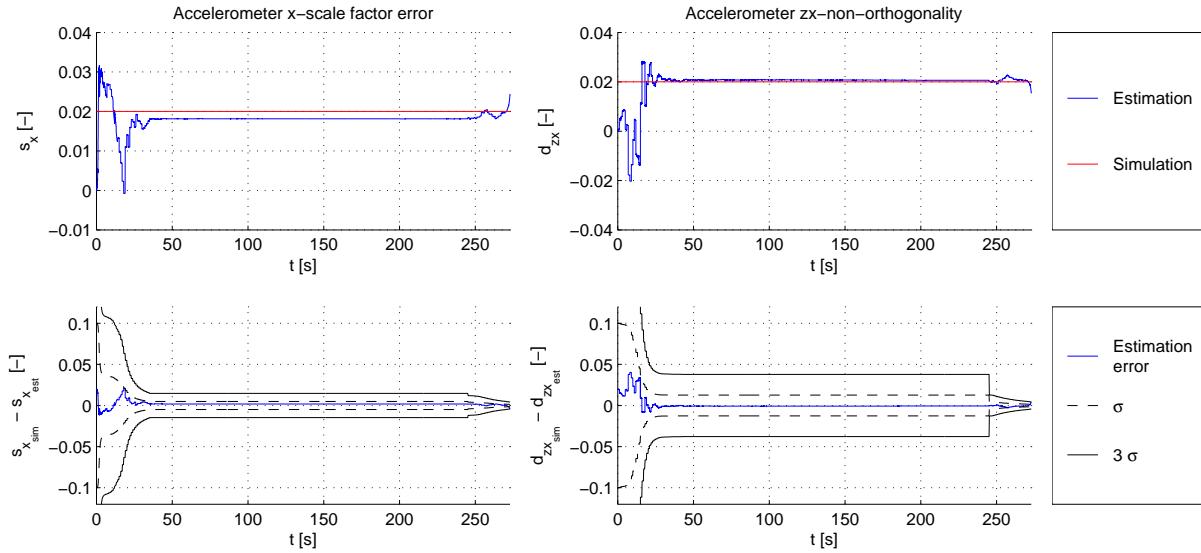


Figure 7.53: Estimated accelerometer x-scale factor and zx-non-orthogonality error

142 m. Attitude errors are still small.

In the response surface coefficient plot, suddenly many factors have started to become significant, even the biases which are actually already being estimated. This is a strong indication that the estimator has reached the limit of the variables that it is able to estimate with the available measurement data. It is thus expected that estimation of any further noise source estimation will not improve the estimation result. To test this expectation, the magnetometer zx-orthogonality has been estimated, and indeed it has been found that the estimation errors start to grow unbounded.

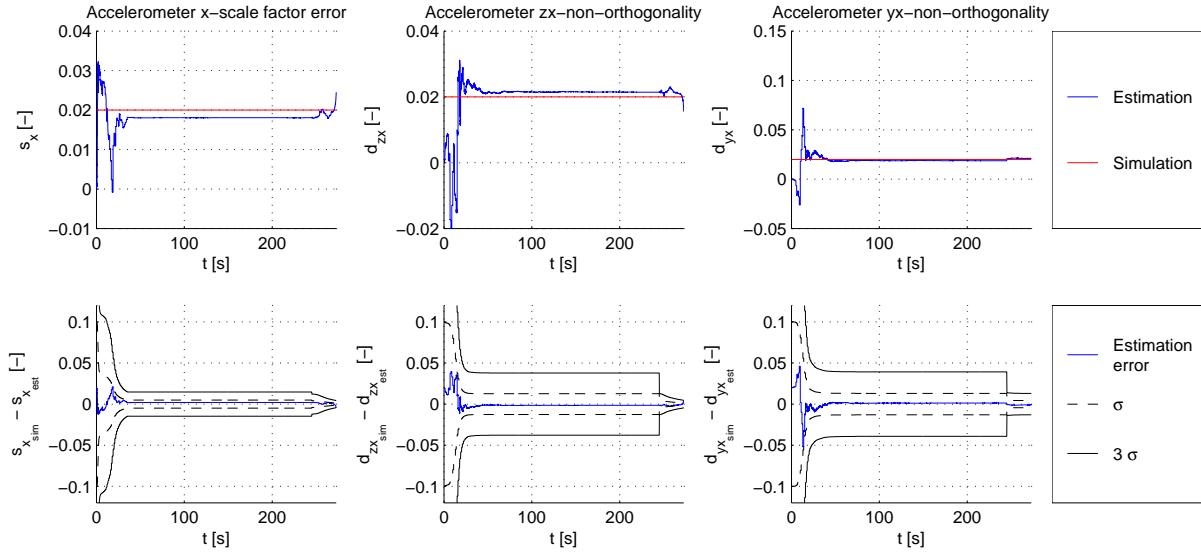


Figure 7.54: Estimated accelerometer x-scale factor and yx- and zx-non-orthogonality error

It can then be concluded that the best estimation results for noisy sensors are obtained when next to the state and the accelerometer and gyroscope biases, also the accelerometer x-scale factor error and yx- and zx-non-orthogonalities are estimated, and no further noise sources are estimated. Worst case errors are still large; in some cases over 3 km for the altitude. Therefore, in the development of a measurement system, care must be taken in removing the sensor noise factors when possible pre-flight. The main factors can be found from the response

surface coefficients to be the following, in order of effect on the response: x-scale factor error of the magnetometer, z-bias of the accelerometer, xz-non-orthogonality of the accelerometer, and most scale factor errors and non-orthogonalities of the magnetometer. It should be noted that this sensor particular configuration has been simulated at Esrange, Kiruna, Sweden, where the magnetic field lines are almost perpendicular to the surface. If the flight would be at another location, the effects of the magnetometer are different, and should be reevaluated. This location dependence is not present for the accelerometer, but the x-axis of the accelerometer should be roughly aligned with the direction of flight, for the results presented here to be valid.

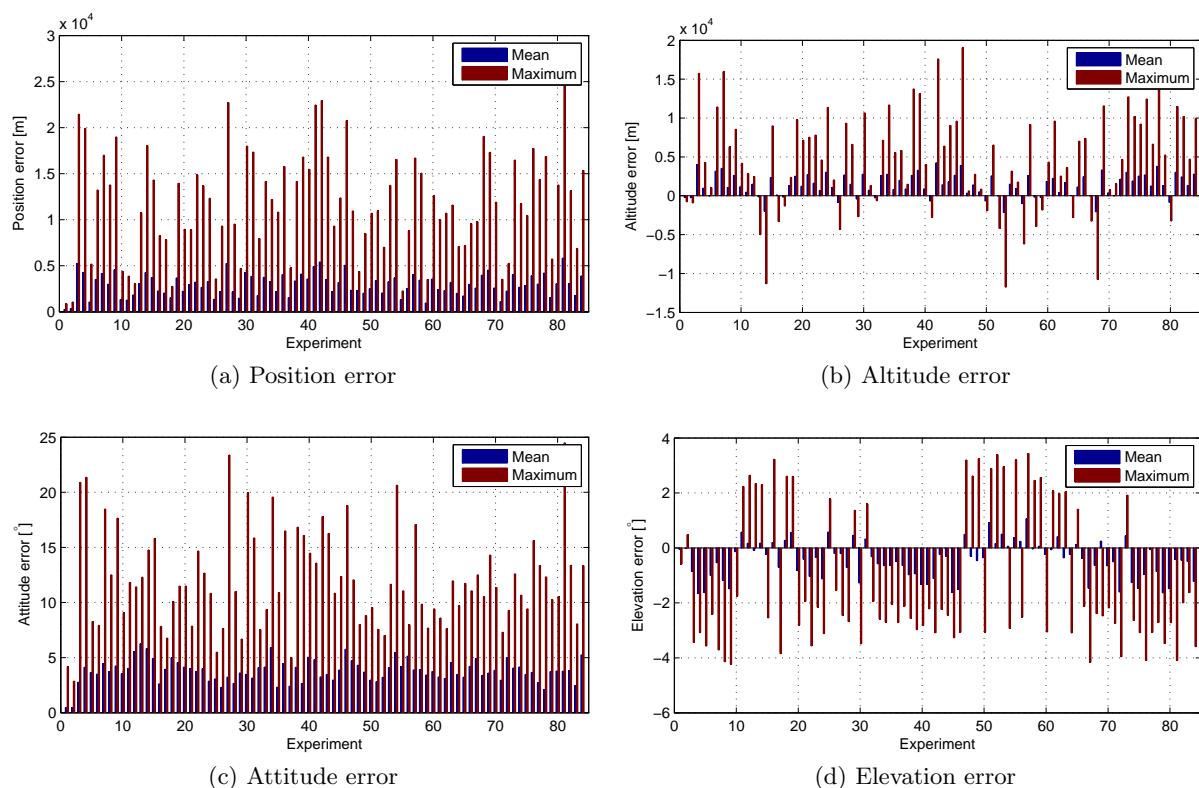


Figure 7.55: Estimation errors with bias estimation and sensor noise factors varied by orthogonal array

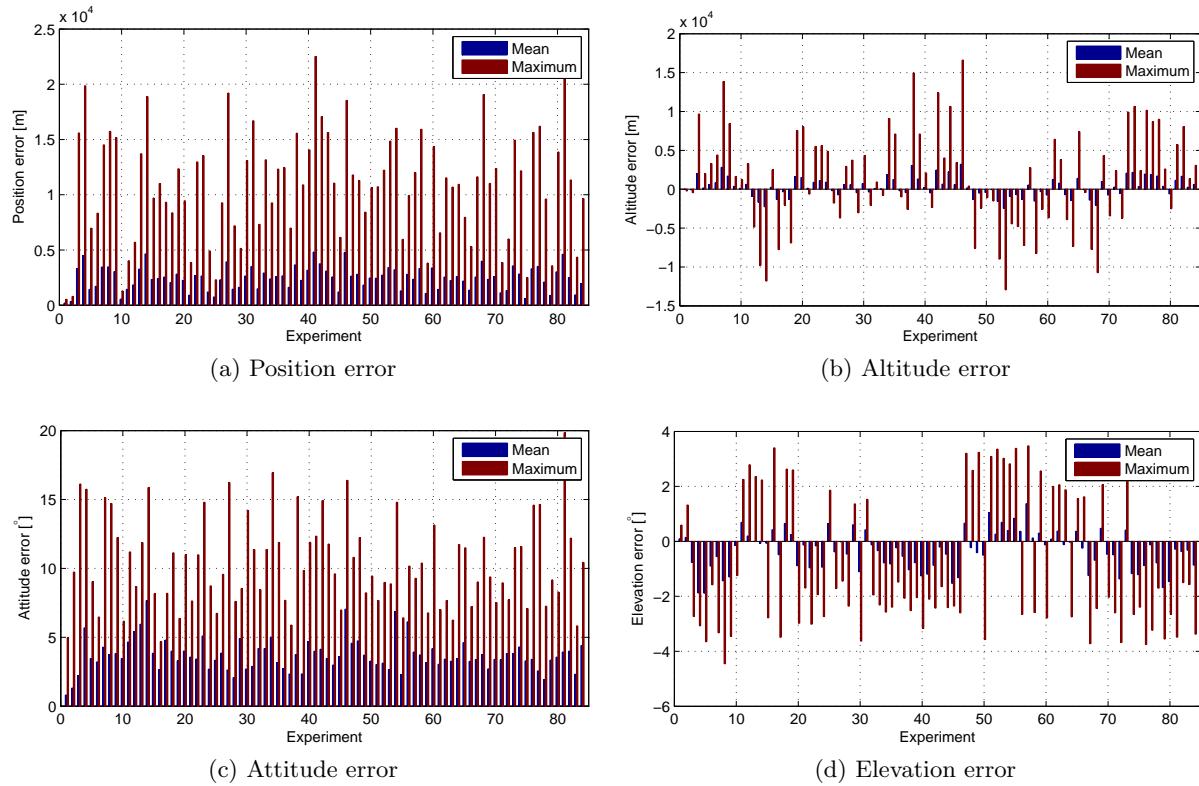


Figure 7.56: Estimation errors with accelerometer x-scale factor error estimation and sensor noise factors varied by orthogonal array

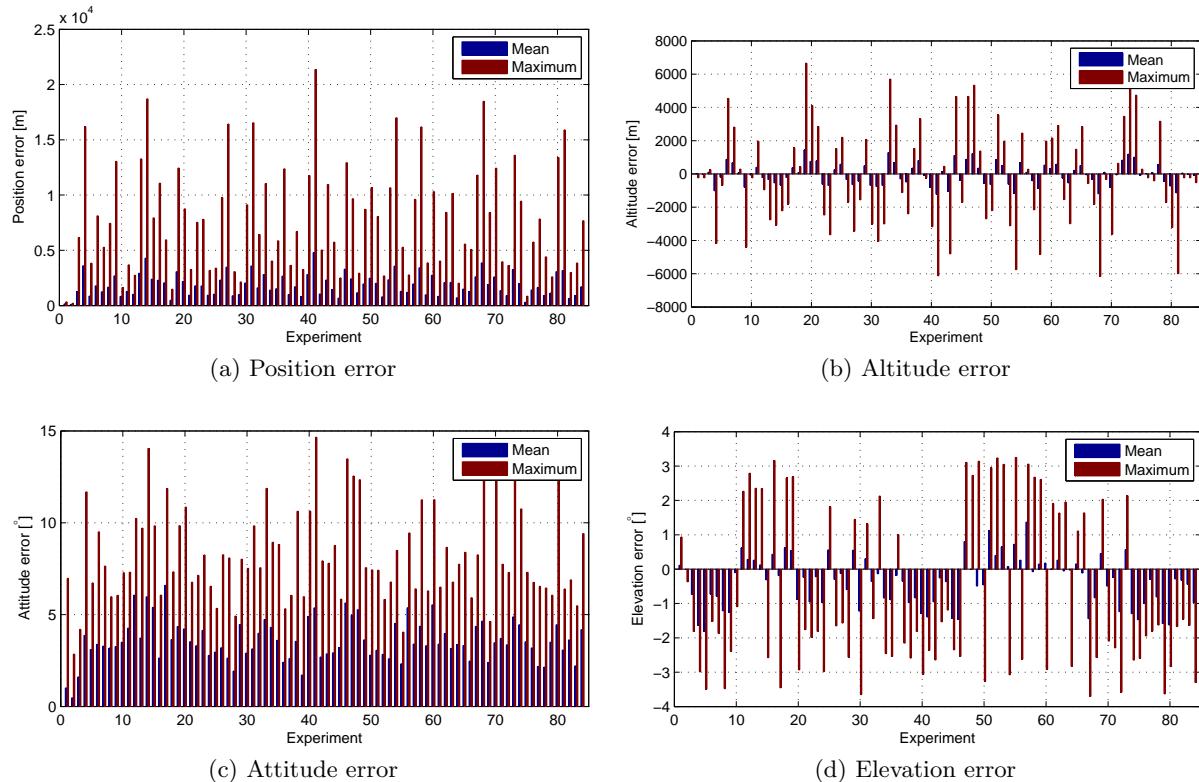


Figure 7.57: Estimation errors with accelerometer zx-non-orthogonality estimation and sensor noise factors varied by orthogonal array

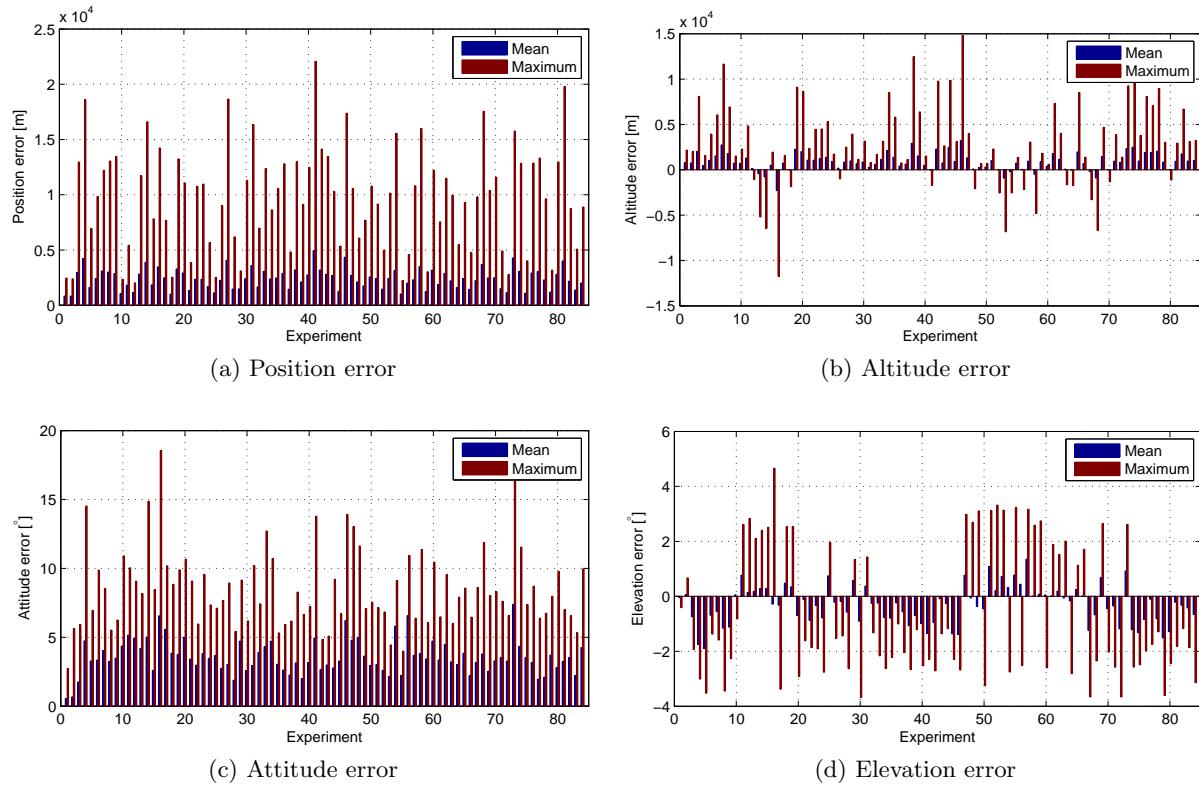


Figure 7.58: Estimation errors with accelerometer x-scale factor and magnetometer zx-non-orthogonality estimation and sensor noise factors varied by orthogonal array

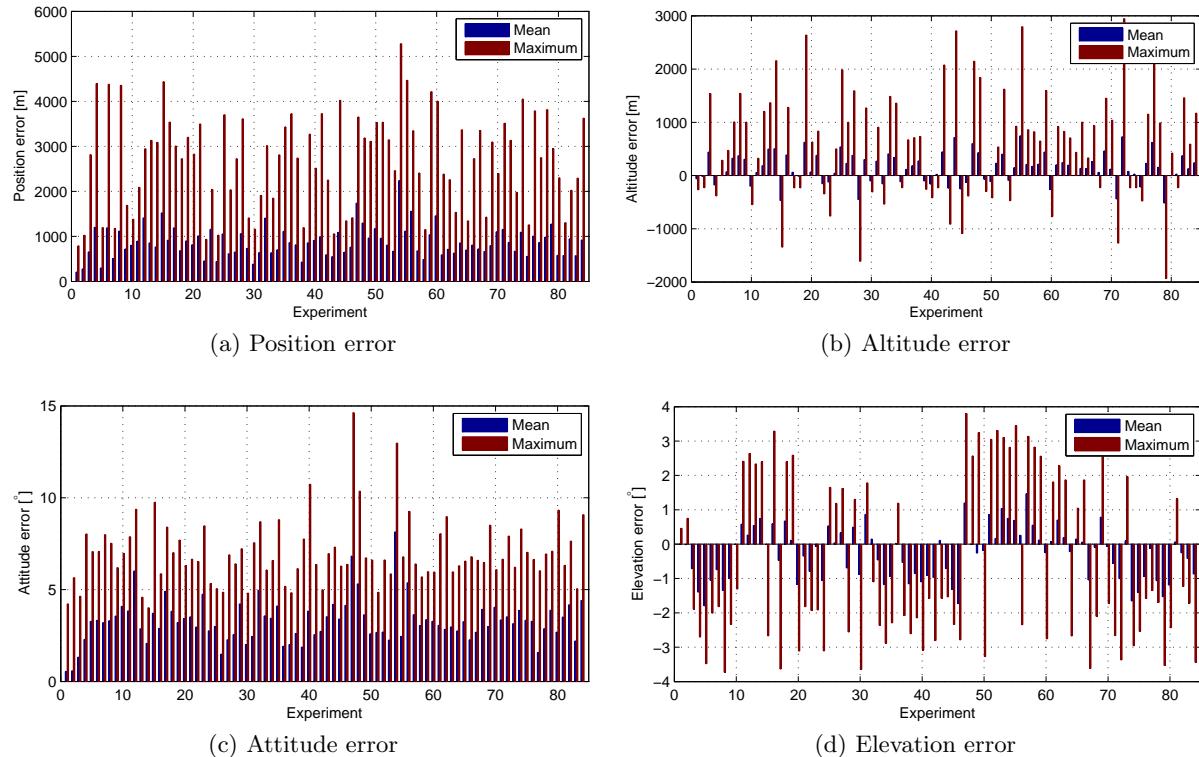


Figure 7.59: Estimation errors with accelerometer x-scale factor and yx- and zx-non-orthogonality estimation and sensor noise factors varied by orthogonal array

7.5.4 Sensor failure

Due to the severity of the flight conditions, it may be possible that one of the sensors stops functioning during the flight. Depending on the sensor, this may result in a catastrophic failure of the system, or a functional system with degraded performance. While during the entire flight a sensor may fail, the earlier the failure occurs, the more catastrophic it is. It is most likely that should a sensor fail mechanically, it is likely that this will happen early in flight, during motor burn, which is the phase when the largest accelerations and vibrations occur. To simplify the estimation, it is simply assumed that all failures occur before launch. In effect, this is just estimation with a missing sensor.

For the sensors that will be used to propagate the state, the accelerometer and gyroscope, it will be assumed that failure means that the state propagation will continue with the last known measurement, i.e., the measurement that would be obtained in the launch tower. This is done since without making this assumption, state propagation will not be possible at all. For the other sensors, failure will mean a simple absence of measurement data from these sensors.

First, failure from the accelerometer data will be analyzed. It was found that there is a significant difference between accelerometer failure when accelerometer bias is estimated, and when that is not the case. Figure 7.60 shows the position estimate when no accelerometer bias is estimated, and figure 7.61 shows this estimate with accelerometer bias estimation. Clearly, bias estimation yields a better position estimate, although in all cases the estimate is unusable, and the failure can be considered catastrophic. Since the sensor failed in the launch tower, the estimator will never detect translational motion. When no bias estimation is performed, the altitude is kept at zero, despite the presence of a GPS sensor. However, with bias estimation, the estimator will use the GPS information to estimate a bias, as seen in figure 7.63, and the estimator will detect some upwards translational motion. As seen in figure 7.62, estimation of the elevation error is mostly unaffected by the failure of the accelerometer. Then, since the attitude does change, but the specific force does not, according to (4.30), the resulting acceleration will be pointing downwards, which is the reason for the estimated downwards motion. Finally, near the end of flight, a large jump in altitude can be found, when the GPS signal becomes available again. Then, to a certain extend the estimator is able to estimate the position again. For the attitude, it was already found that the elevation is only little affected by the missing accelerometer. The azimuth estimate does get significantly worse. This is caused when the estimator tries to use the pressure and GPS measurements to correct the state. It can be found that when pressure and GPS measurements are removed, and the position and attitude estimation are essentially uncoupled, that the attitude estimate is entirely unaffected by the accelerometer failure. The responsibility of the GPS for the azimuth divergence can also be seen in the coupled case, by the fact that the azimuth estimate shows a discontinuous jump near 250 s, when GPS measurements become available again.

Next, failure of the gyroscope will be analyzed. In figure 7.64 the position estimate is presented, in figure 7.65 the attitude estimate and in figure 7.67 the attitude error. As shown, the loss of a gyroscope is much less catastrophic than the loss of an accelerometer. This is mostly due to the presence of the magnetometer, which allows estimation of the elevation, and partial estimation of the azimuth. The maximum elevation error is 3.5° , which is still reasonable, but an order of magnitude larger than the nominal case. The elevation error also propagates to the position error, due to the presence of the attitude in the form of a DCM in the translational equation of motion. The position errors are several orders of magnitude larger than the nominal case, up to 15 km. Also the azimuth error increases significantly, to a maximum of 30° . These errors, both in position and attitude, have as consequence that the position and attitude estimate become practically unusable; they only provide a coarse knowledge on the trajectory of the rocket.

Then, failure of respectively the pressure sensor and magnetometer will be analyzed. In principle, these sensors are redundant to the accelerometer and gyroscope, and not required to

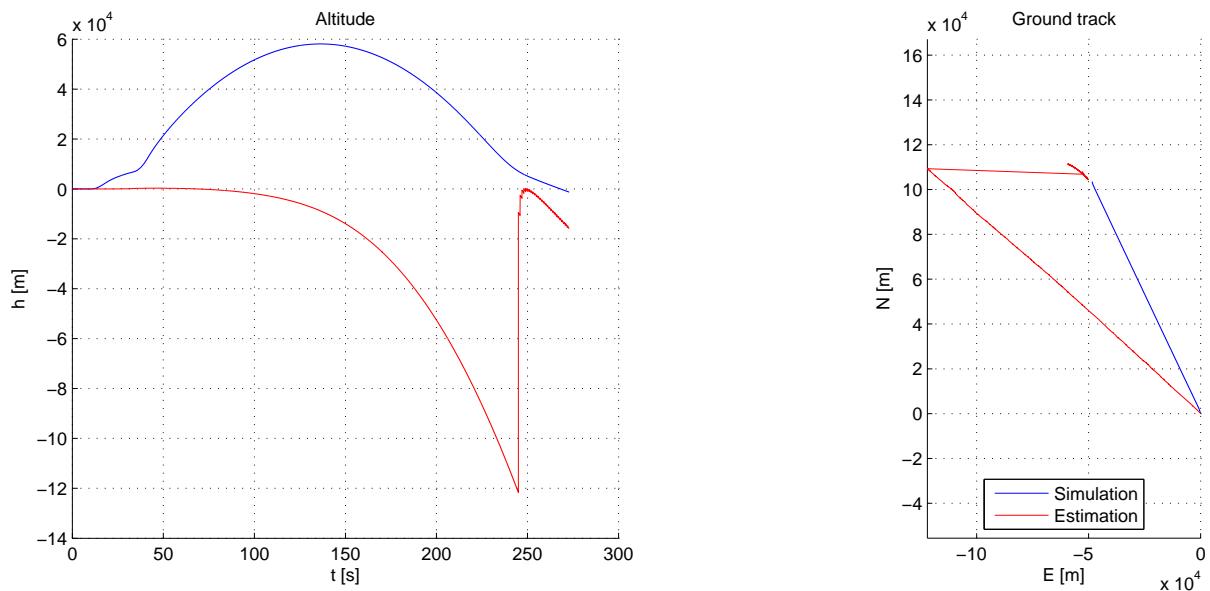


Figure 7.60: Position estimate with a failed accelerometer and without accelerometer bias estimation

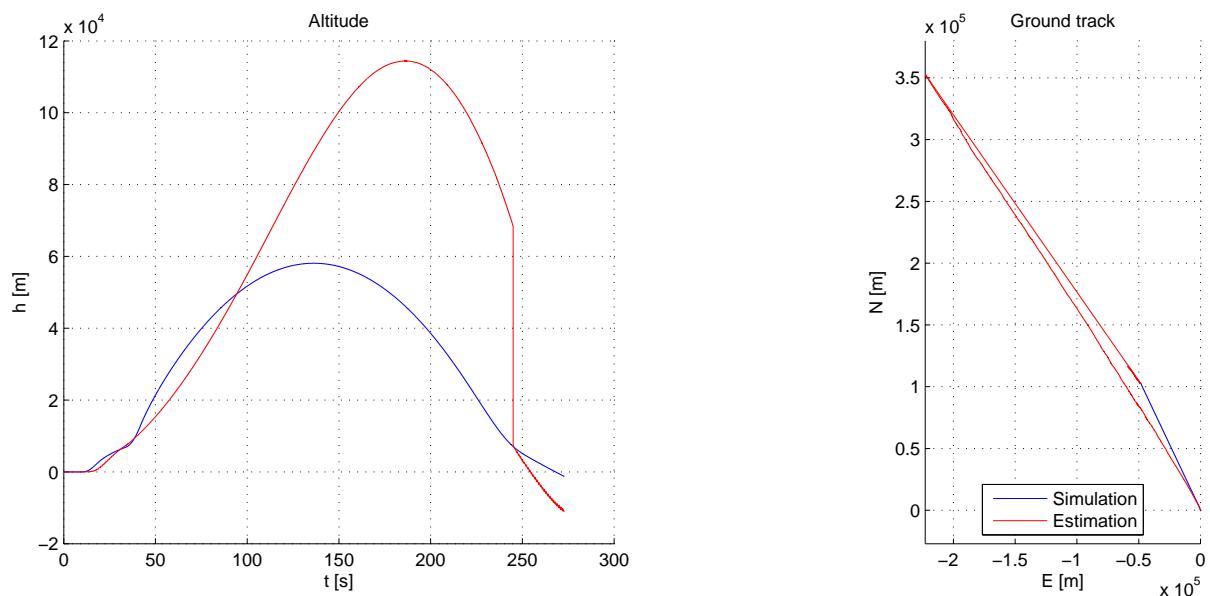


Figure 7.61: Position estimate with a failed accelerometer with accelerometer bias estimation

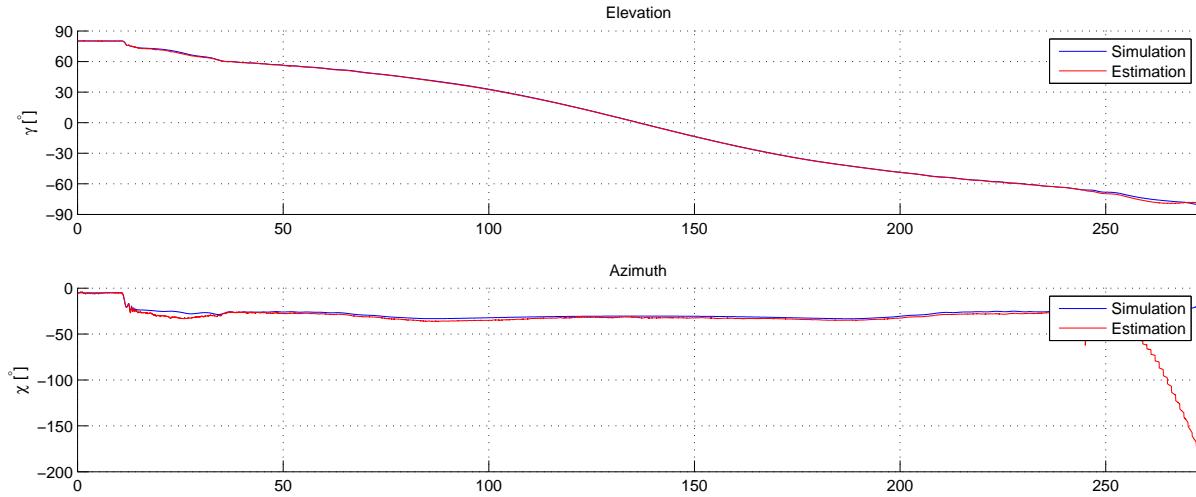


Figure 7.62: Attitude estimate with a failed accelerometer with accelerometer bias estimation

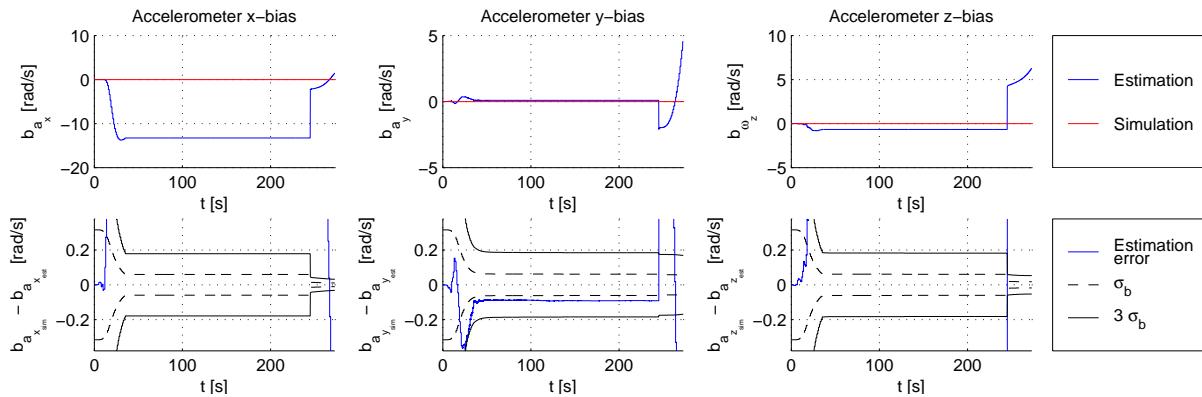


Figure 7.63: Accelerometer bias estimate with a failed accelerometer with accelerometer bias estimation

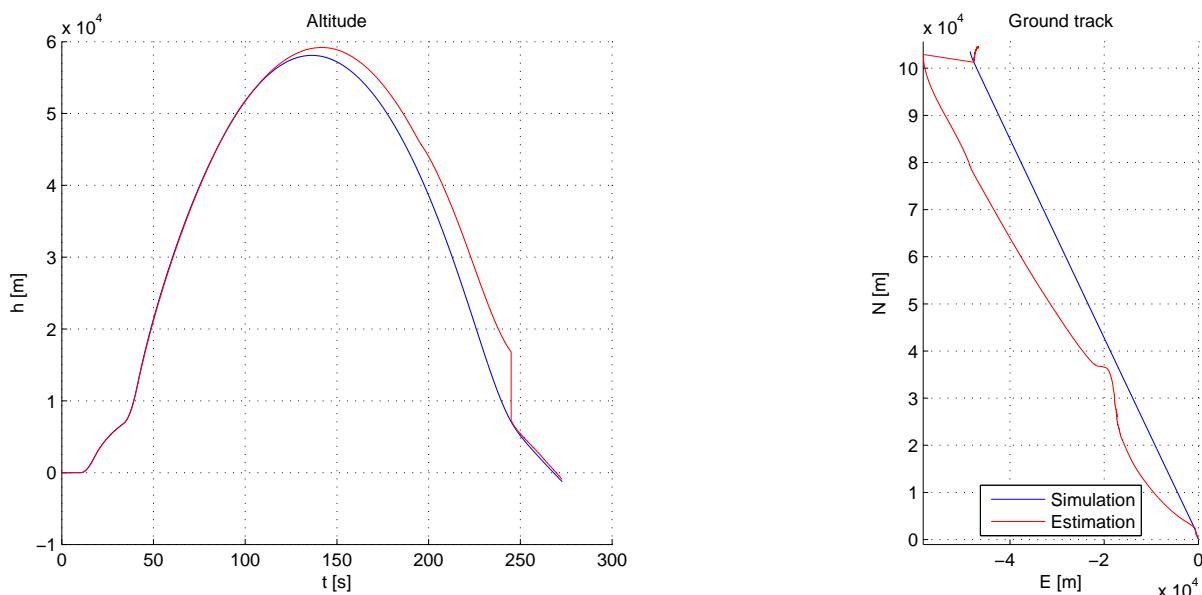


Figure 7.64: Position estimate with a failed gyroscope

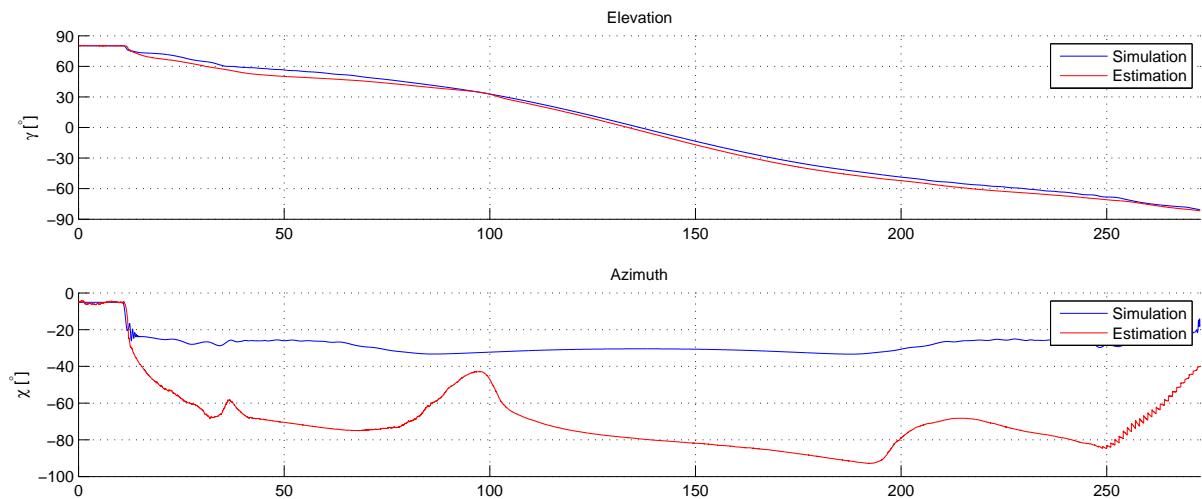


Figure 7.65: Attitude estimate with a failed gyroscope

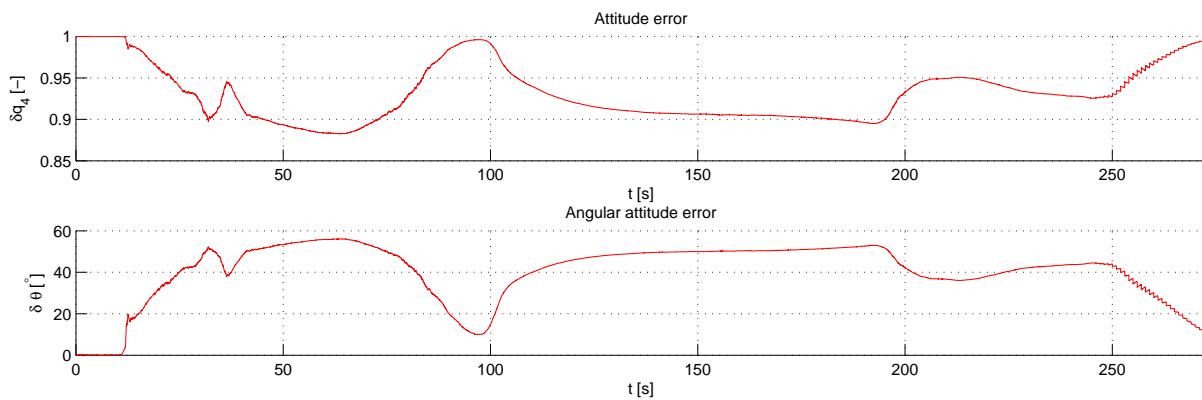


Figure 7.66: Attitude estimation error with a failed gyroscope

estimate flight. As can be found from table 7.17, lack of a pressure sensor only has very limited influence on the result. This is non unexpected, as the pressure sensor is only used when the flight is subsonic, which is only a very small part of the flight. Failure of the magnetometer is more significant, as can be seen from table 7.18 and figure 7.67. Indeed, the attitude is slowly diverging without magnetometer, to a value of 10° . Then, when GPS measurements become available, also the attitude is affected, as was also seen in previous estimations. It can also be found that the attitude error propagates to the position error, as the position errors reaches maximum values of 3 km, which is considerably worse than the nominal case. The data presented here is for an unbiased gyroscope. For a gyroscope with bias, the attitude error will be even worse, and was found to reach values up to 100° . From this it can be concluded that the redundancy of the magnetometer is only not required for a non-biased gyroscope, and that should the gyroscope be biased, the magnetometer is required for proper attitude estimation.

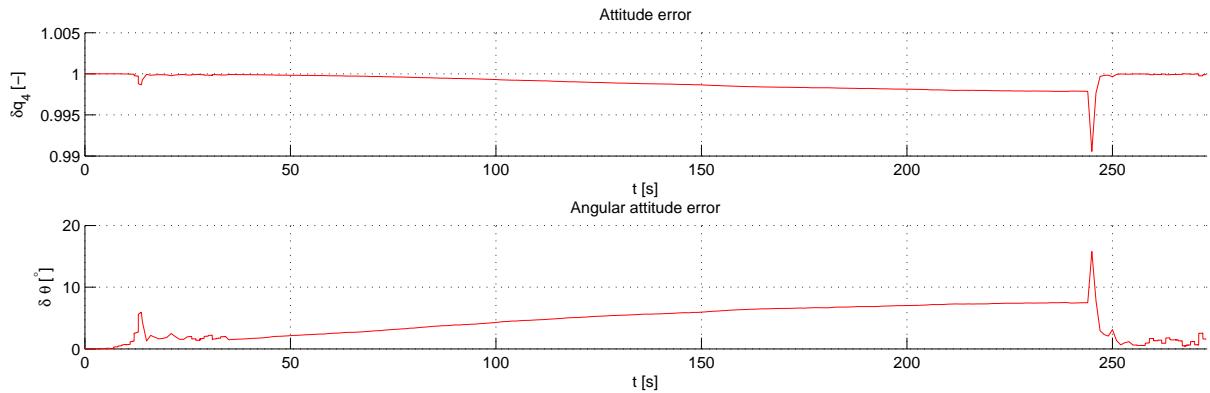


Figure 7.67: Attitude estimation error with a failed magnetometer

Concluding this section, it can be stated that all of the accelerometer, gyroscope and magnetometer are required for a proper estimation result. Not all failures are equally catastrophic, but in general it cannot be expected that good estimation results are obtained without any of these sensors. The lack of the accelerometer, and the jumps in attitude error when GPS become available, show the coupling between the position and attitude. It is clear that without a proper attitude estimate, no position estimate can be made. However, if a proper position estimate cannot be made, then trying to estimate this position will significantly affect the attitude estimation result. Therefore, if problem is deemed to be unacceptable, it might be worthwhile to investigate decoupling the position and attitude estimation.

Table 7.15: Kinematic model EKF estimation errors with a failed accelerometer

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\delta\gamma$ [$^\circ$]	$\delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Stage separation	19.31	-0.01	1791.94	1732.43	333.86	7.39	0.87	5.29
Second stage burnout	41.81	-203.17	2438.04	1950.18	848.73	0.58	0.05	0.56
Apogee	136.36	-50.44	50846.76	-31318.63	1626.03	1.71	0.27	1.69
Nose cone separation	243.12	-1.88	306642.53	-63395.00	3625.22	1.25	0.14	1.44
Parachuting	257.20	6.80	10659.23	4831.21	1833.27	19.85	2.05	26.68
Landing	268.11	13.65	17164.38	8526.76	1614.91	67.43	0.78	108.89
	Mean error		73248.03	-25412.29	1467.04	5.52	0.36	7.37
	Maximum error		312079.20	-75357.40	3705.07	92.80	2.68	163.70

Table 7.16: Kinematic model EKF estimation errors with a failed gyroscope

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\delta\gamma$ [$^\circ$]	$\delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Stage separation	19.31	-0.01	134.23	15.39	26.18	30.25	4.74	24.22
Second stage burnout	41.81	-0.01	1629.56	123.61	417.73	50.19	5.85	41.84
Apogee	136.36	-5.04	7609.16	-972.29	56.25	49.32	2.98	49.02
Nose cone separation	243.12	-1.88	14488.94	-9507.47	278.18	43.99	3.11	54.13
Parachuting	257.20	-1.80	1776.19	-458.60	310.65	31.86	0.97	43.91
Landing	268.11	-1.45	1879.95	-389.53	173.90	16.38	1.91	25.82
	Mean error		6397.93	-1968.38	114.70	40.29	3.26	41.04
	Maximum error		14576.01	-9720.74	420.02	56.16	6.48	60.09

Table 7.17: Kinematic model EKF estimation errors with a failed pressure sensor

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\delta\gamma$ [$^\circ$]	$\delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Stage separation	19.31	-0.01	23.67	13.94	4.69	4.40	0.51	3.08
Second stage burnout	41.81	-0.01	28.15	1.28	5.73	0.61	0.06	0.56
Apogee	136.36	0.40	281.96	31.25	3.55	1.17	0.21	1.15
Nose cone separation	243.12	-0.42	613.84	-209.72	6.48	0.65	0.07	0.75
Parachuting	257.20	-0.04	17.42	-9.07	1.59	0.33	0.01	0.48
Landing	268.11	-0.03	9.95	-4.87	0.89	0.22	0.07	0.49
	Mean error		258.65	-12.02	3.67	1.18	0.14	1.14
	Maximum error		617.88	-216.84	11.54	5.08	0.69	3.82

Table 7.18: Kinematic model EKF estimation errors with a failed magnetometer

	t [s]	Δt [s]	Δr [m]	Δh [m]	ΔV [m/s]	$\delta\theta$ [$^\circ$]	$\delta\gamma$ [$^\circ$]	$\delta\chi$ [$^\circ$]
Tower	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00
Stage separation	19.31	-0.69	297.98	-287.18	27.76	1.89	-0.13	4.18
Second stage burnout	41.81	-0.19	277.23	-229.90	28.71	1.74	-0.30	2.77
Apogee	136.36	0.36	1563.58	7.14	15.25	5.63	0.30	5.57
Nose cone separation	243.12	-0.88	2233.28	380.29	74.89	7.49	-0.44	13.77
Parachuting	257.20	-0.04	64.70	-8.89	9.44	0.55	-0.54	0.31
Landing	268.11	-0.07	19.15	-16.78	1.09	0.61	0.32	-0.60
	Mean error		163.16	-3.90	4.17	1.58	0.04	0.42
	Maximum error		2612.77	-95.51	41.87	15.76	2.13	38.46

7.6 Non-nominal trajectory

There are several factors that influence the flight. Those uncertainties can be divided into those due to the environment, and those due to an imperfect or mismatched vehicle model. For the purpose of evaluating estimator performance, it is not necessary to study the effect of each individual factor. Rather, it is more important to obtain a set of simulation results for which the state and dynamic characteristics are varied over a sufficient wide range to represent possible flights. Specifically, this means that flight factors are sought that vary the following quantities: altitude, (cross-)range, velocity, and angular velocity.

The primary factors that contribute to altitude are the thrust magnitude and the axial force coefficient, and they are largely related. Since the axial force coefficient applies during a longer flight time, it is selected as flight factor. The flight factors that influence range and cross-range have been extensively studied by Engelen (2012). There, it was found that major contributions can be attributed to thrust magnitude, axial force coefficient, center of mass offset y -body direction, and to a lesser extend thrust offset. The largest influence is given by the center of mass offset, and these are thus selected as flight factors. Finally, roll motion is almost fully caused by a static aerodynamic roll coefficient, and it therefore included as flight factor as well.

Table 7.19: Flight factors

Symbol	Description
y_{CoM}	Center of mass offset in y_B -direction
η_{C_X}	Axial force coefficient factor
C_{l_0}	Roll static moment coefficient

This results in the following list of flight factors (given in table 7.19): center of mass offset y -body direction, axial force coefficient, and static roll moment coefficient. This results in a total set of only 3 flight factors, which allows full factorial design at 3 levels (27 simulations in total). For each of the simulations, the resulting position, velocity, attitude and angular velocity have been plotted. In each of the plots of the flight characteristics, trajectories that are close to each other have been grouped by color, and it has been indicated which of the three flight factors is responsible for this grouping.

Figure 7.68 contains a plot of the altitude and ground track. As indicated in the plots, the driving factor for altitude and flight time is the aerodynamic axial force coefficient, whereas the factor that drives the cross-range and azimuth (figure 7.71) is the center of mass offset. From figures 7.69 and 7.70, it can be found that also the magnitude of velocity and Mach number are primarily driven by the axial force coefficient. How the velocity oscillates in the body frame is driven by the rolling moment coefficient, as well as the magnitude of angular velocity as found in figure 7.72.

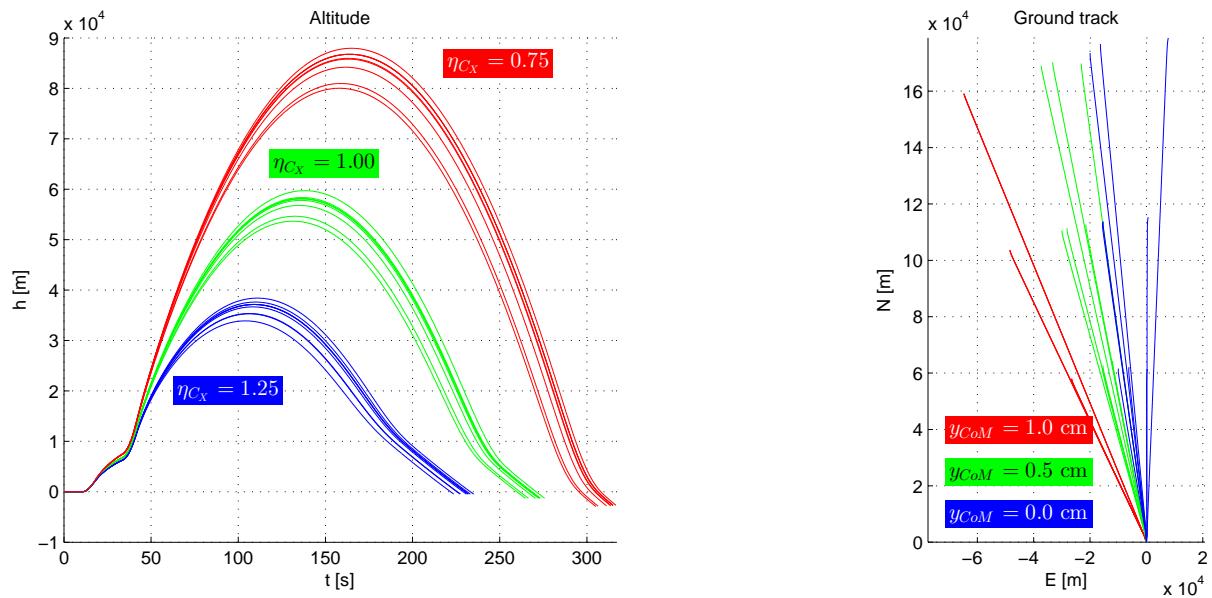


Figure 7.68: Non-nominal flight trajectory

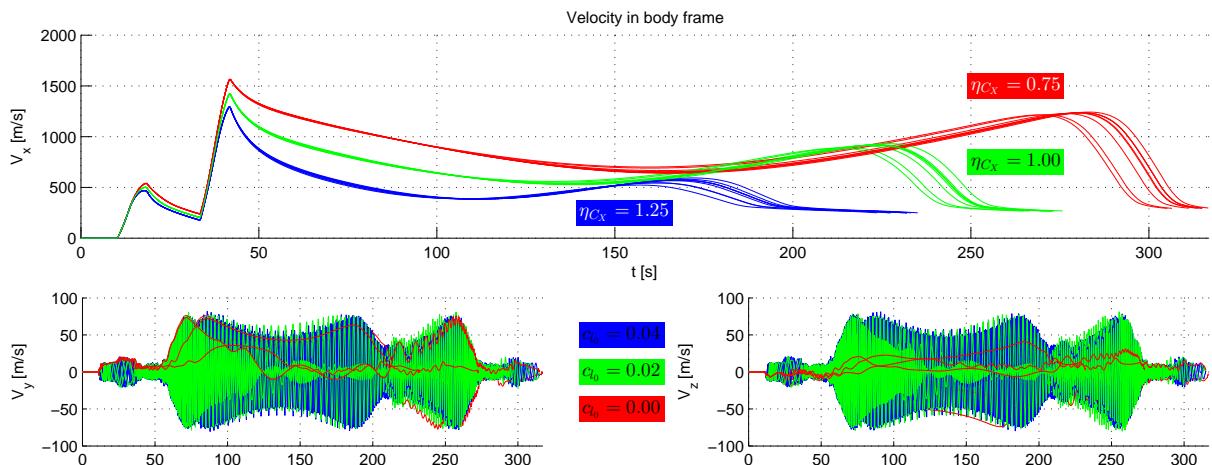


Figure 7.69: Non-nominal velocity

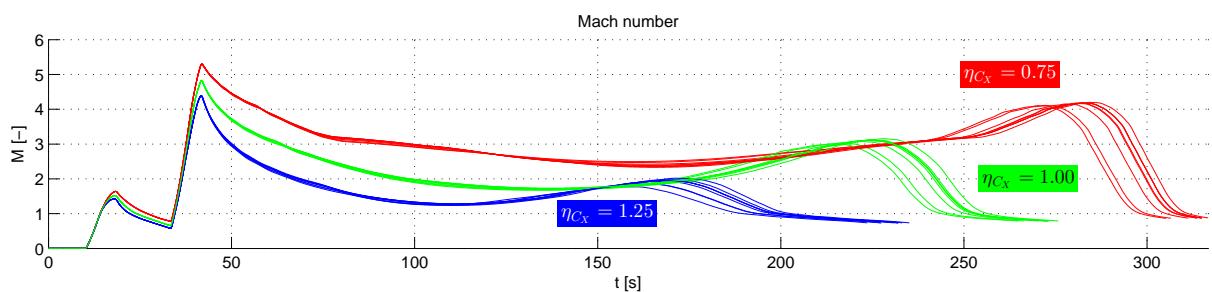


Figure 7.70: Non-nominal Mach number

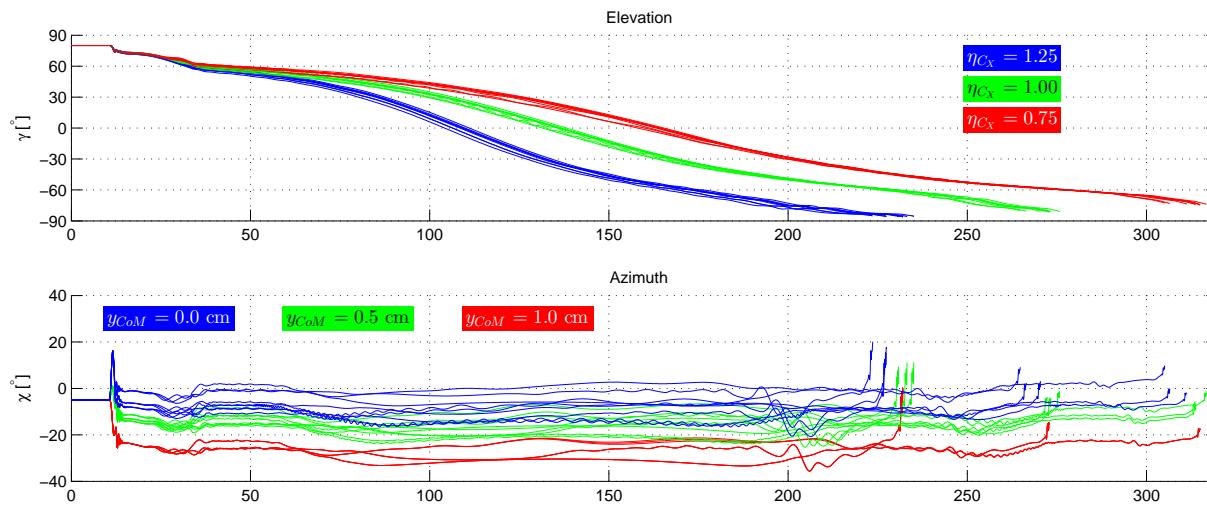


Figure 7.71: Non-nominal attitude

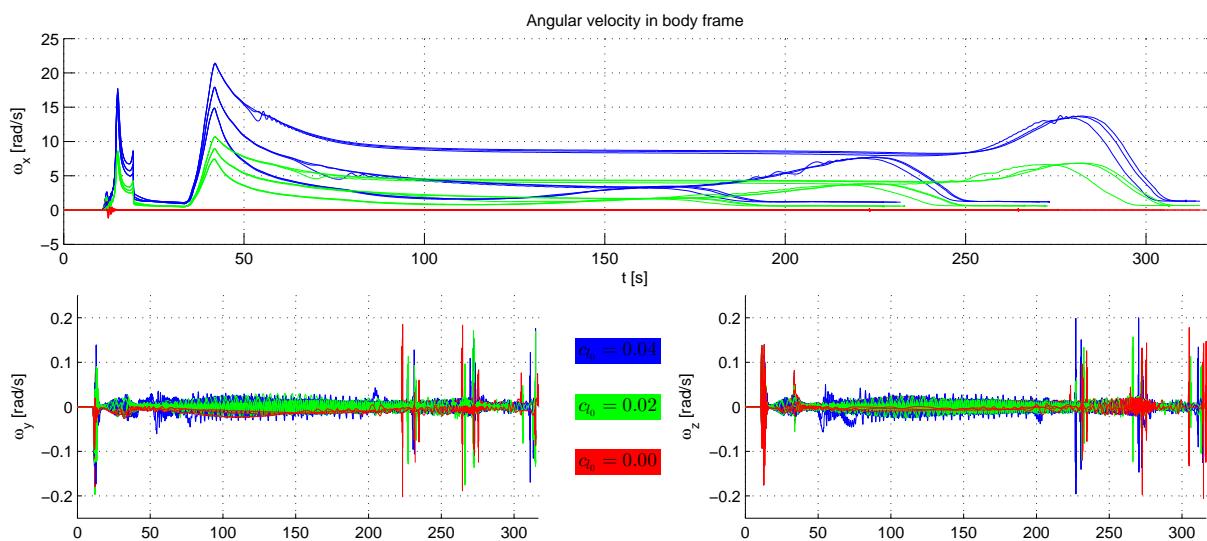


Figure 7.72: Non-nominal angular velocity

The maximums for static and dynamic flight characteristics are also presented in table 8.1. Also from this table it is shown that the selected flight factors represent a wide variety of time-of-flight t_f , apogee altitude h_A , peak velocity, peak Mach number and peak angular velocity.

In section 7.3, it was stated that no conclusions could yet be drawn for the suitability of the selected gyroscope, due to the fact that the nominal trajectory does not simulate the roll motion due to an imperfectly manufactured rocket. However, with these non-nominal simulations available, it is possible to draw some conclusions in this regard. For the simulated trajectories, the maximum angular velocity is approximately 20 rad/s. This corresponds to 1146 °/s, well within the -/+ 2000 °/s range of the measurement system.

Table 7.20: Flight characteristics non-nominal flight

Case	η_{Cx} [-]	C_{l_0} [-]	y_{CoM} [cm]	t_f [s]	h_A [km]	max V [m/s]	max M [-]	max ω [rad/s]
1	1.00	0.00	0.00	272.87	58.11	1420.61	4.82	0.20
2	1.25	0.00	0.00	232.07	37.14	1292.03	4.39	0.18
3	0.75	0.00	0.00	314.97	86.76	1563.96	5.30	0.21
4	1.00	0.02	0.00	272.87	58.11	1420.61	4.82	8.96
5	1.25	0.02	0.00	231.97	37.14	1292.03	4.39	8.56
6	0.75	0.02	0.00	314.97	86.75	1563.96	5.30	10.71
7	1.00	0.04	0.00	272.87	58.11	1420.61	4.82	17.91
8	1.25	0.04	0.00	231.97	37.14	1292.03	4.39	17.13
9	0.75	0.04	0.00	314.97	86.75	1563.96	5.30	21.42
10	1.00	0.00	0.50	275.77	59.71	1423.58	4.83	0.38
11	1.25	0.00	0.50	235.07	38.40	1295.53	4.40	0.38
12	0.75	0.00	0.50	316.67	87.94	1566.08	5.31	0.38
13	1.00	0.02	0.50	272.27	57.79	1421.33	4.82	8.96
14	1.25	0.02	0.50	233.17	37.62	1293.55	4.39	8.56
15	0.75	0.02	0.50	313.57	85.79	1564.49	5.30	10.71
16	1.00	0.04	0.50	273.27	58.35	1421.49	4.83	17.92
17	1.25	0.04	0.50	230.87	36.67	1292.34	4.39	17.23
18	0.75	0.04	0.50	313.77	85.95	1564.32	5.30	21.41
19	1.00	0.00	1.00	264.77	53.65	1418.93	4.81	1.32
20	1.25	0.00	1.00	223.57	33.89	1289.53	4.38	1.32
21	0.75	0.00	1.00	305.17	80.01	1562.53	5.29	1.32
22	1.00	0.02	1.00	266.47	54.64	1418.09	4.81	8.91
23	1.25	0.02	1.00	227.47	35.35	1290.02	4.38	8.64
24	0.75	0.02	1.00	306.47	80.96	1561.67	5.29	10.66
25	1.00	0.04	1.00	270.47	56.78	1420.26	4.82	17.87
26	1.25	0.04	1.00	227.27	35.29	1290.14	4.38	17.73
27	0.75	0.04	1.00	311.27	84.18	1563.27	5.30	21.36

7.7 Non-nominal flight estimation

Finally, estimation for varying flight characteristics has been performed. For each of the simulated flights, an estimation has been performed and the estimation errors have been evaluated. In figures 7.73, 7.74, 7.75 and 7.76 respectively the position error, altitude error, attitude error and elevation error have been plotted.

For position and altitude, for almost all cases a smaller axial force coefficient (corresponding to larger velocities), or a larger rolling moment coefficient cause a larger estimation error. For

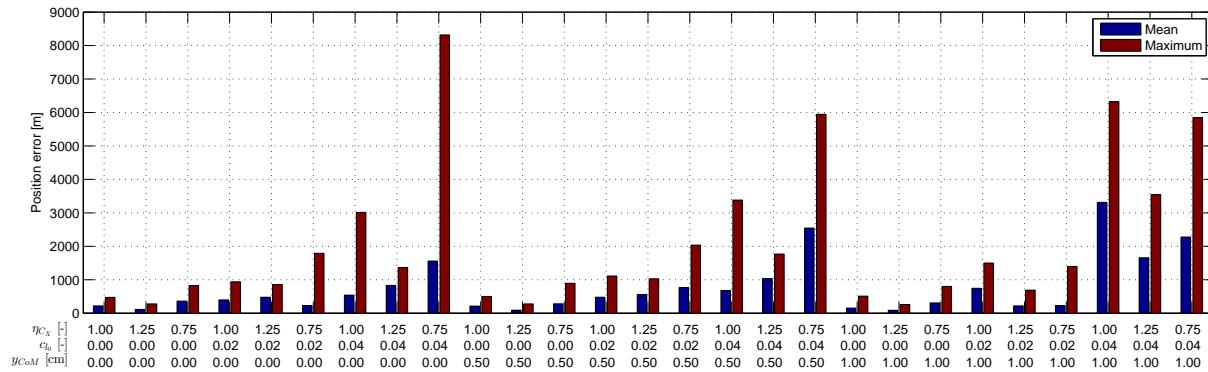


Figure 7.73: Position error for non-nominal flight

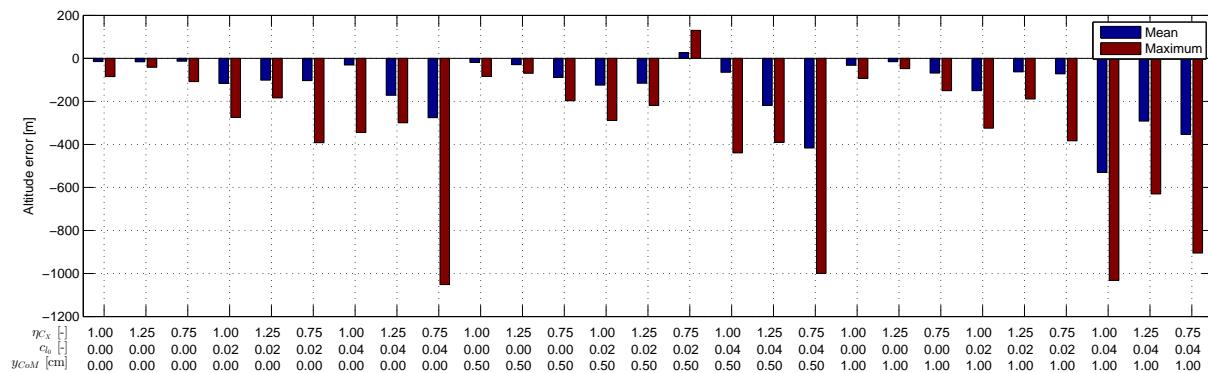


Figure 7.74: Altitude error for non-nominal flight

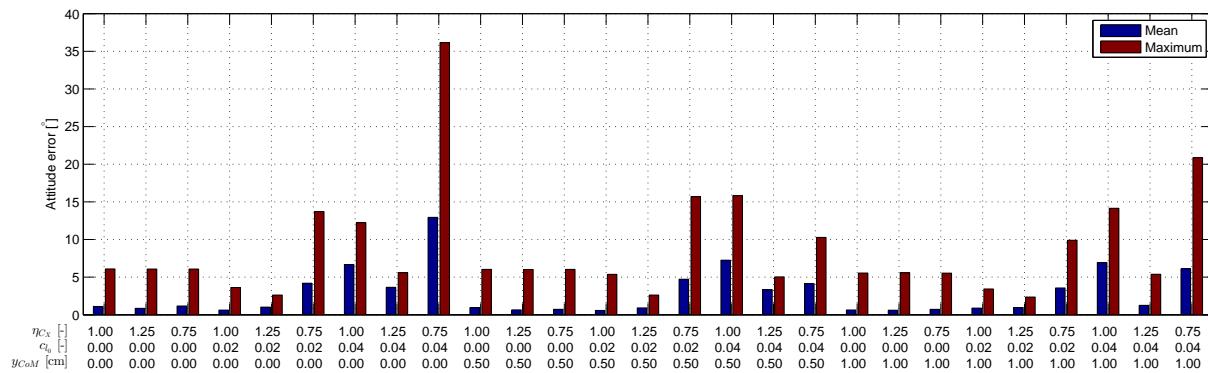


Figure 7.75: Attitude error for non-nominal flight

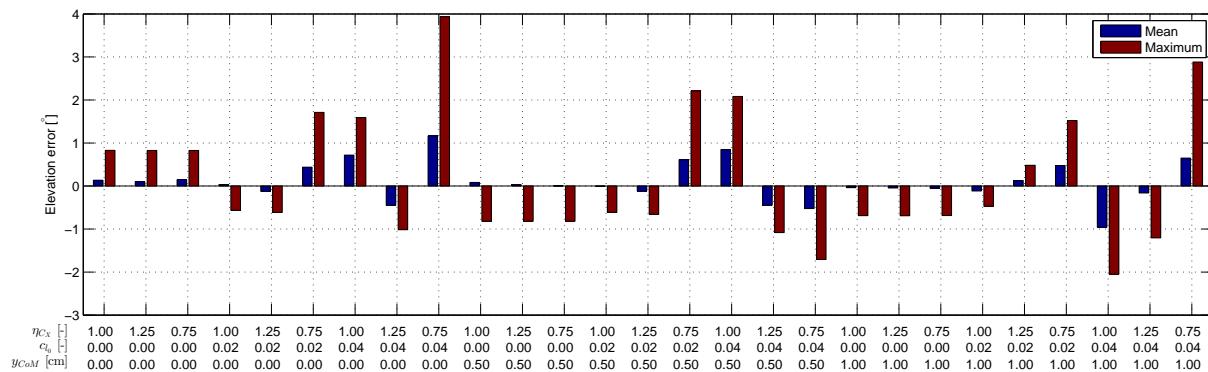


Figure 7.76: Elevation error for non-nominal flight

attitude estimation, the effect is axial force is much less pronounced. Finally, for the elevation error no clear correlations between flight factors and estimation error can be observed: for any flight factor, there are cases where the elevation errors either decrease or increases with respect to the nominal value.

As the most visible effect is that trials with a large rolling moment coefficient yield a large estimation error, in figure 7.77, the peak angular velocity has been set against the estimation error. This clearly shows that indeed the a larger angular velocity is correlated with a strong increase in estimation error. Earlier estimation in this chapter were performed for two flights, namely the ideal rocket flight, and the non-ideal rocket flight, with the difference being the presence of a static roll coefficient. The data obtained above clearly confirm that the assumption made previously, that the presence of roll would have a significant influence on the estimation, was correct.

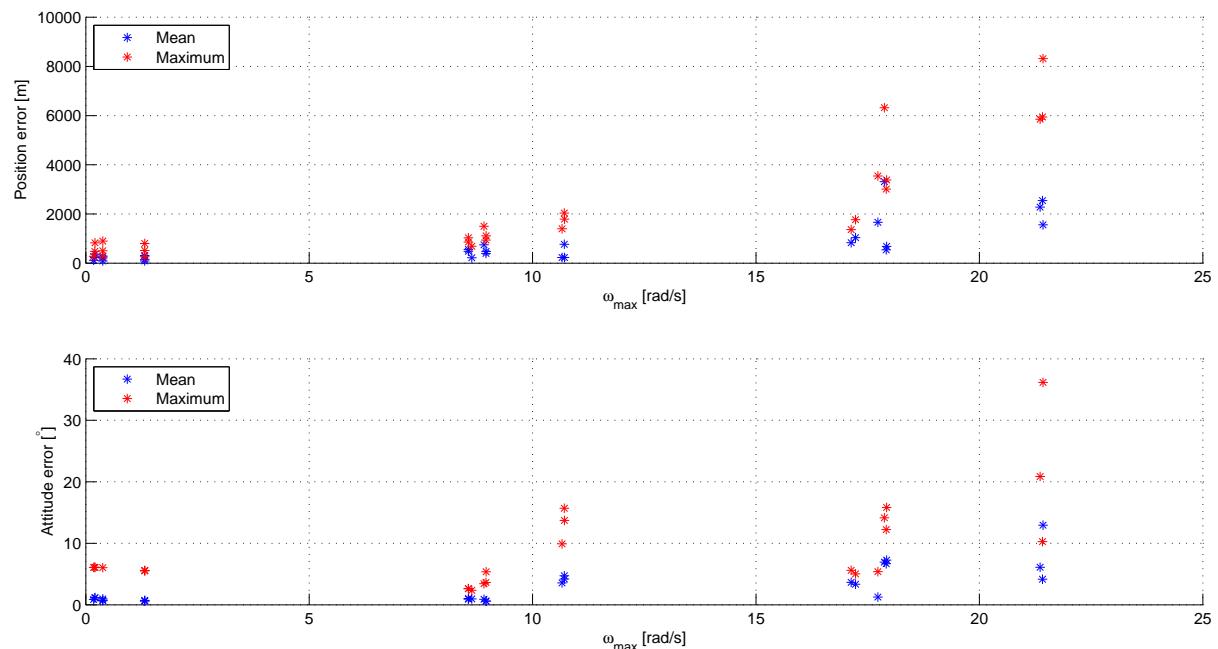


Figure 7.77: Estimation error versus maximum angular velocity

Less dominant, but still visible if the effect of the axial force coefficient. Looking at trials where only the axial force coefficient differs, it is clear that a larger axial force coefficient, corresponds to a lower estimation error. A large axial force coefficient corresponds to a low velocity and vice versa.

From this it can be concluded, that the larger the dynamic characteristics of the flight, i.e., the velocity and angular velocity, the larger the estimation error. To analyze the response further, a response surface can be made with the three flight factors as independent parameters. This gives the following equation for the response with normalized coefficients:

$$r = r_{nom} + \bar{\beta}_1(1 + \frac{1 - \eta C_X}{0.25}) + \bar{\beta}_2 \frac{c_{l_0}}{0.02} + \bar{\beta}_3 \frac{y_{CoM}}{0.005} \quad (7.11)$$

The coefficients for each of the responses is given in table 7.21, along with the residual and normalized residual.

The results from this table show that indeed the dominant factor to position, altitude and attitude error is the roll coefficient, followed by the axial force coefficient. Also the center of mass offset increases the estimation error, but this is minor compared to the other effects. The table also shows large to extremely large residuals for many responses. Therefore, strong doubts can be cast on the results of the response surfaces, and the response surface should certainly

Table 7.21: Non-nominal estimation response surface coefficients

	r_{nom}	$\bar{\beta}_1$	$\bar{\beta}_2$	$\bar{\beta}_3$	ε_{RMS}	$\bar{\varepsilon}_{RMS}$
Position mean	221.27	-193.41	700.23	236.71	740.65	3.35
Position max	475.38	-988.21	1926.28	167.18	1963.25	4.13
Altitude mean	-14.68	19.01	-114.25	-40.70	144.95	9.87
Altitude max	-84.75	110.39	-289.79	-54.18	311.06	3.67
Attitude mean	1.12	-1.39	2.49	-0.58	2.48	2.22
Attitude max	6.09	-4.61	4.03	-1.08	5.65	0.93
Elevation mean	0.13	-0.22	0.03	-0.13	0.39	2.93
Elevation max	0.83	-0.81	0.30	-0.47	1.40	1.69

not be used to interpolate the response given an arbitrary set of flight factors. However, as the conclusions that were drawn from the coefficients corresponds to the observed response from the plots, it can at least be stated that the conclusions are qualitatively correct.

Analyzing the magnitude of the estimation error, it is clear that for some flight cases the results are not very good. In particular altitude errors of 1 km are not very desirable, especially if they occur when the rocket is close to the ground, and when 1 km is a relatively large error. Similarly a position error of 8 km is undesirable when it occurs at landing. The consequences of this error are perhaps not so critical as the altitude error, since no flight events depend on it; rather, it makes retrieval of the rocket more difficult. The attitude error also gets very large in some cases, but again there is no real requirement on the total attitude estimation error. Rather, requirements are put on the elevation estimation error, for which it clearly can be seen that even though the estimation error increase is severe, the actual magnitude of the elevation estimation error stays small.

Further analysis of the estimation error on characteristic points can be performed by evaluating figures 7.78, 7.80 and 7.80, and table 7.22, where several estimation errors at characteristic points have been plotted. From these figures it follows that, while the maximum errors do not occur at the selected characteristic points, the estimation errors are still severe, especially for nose cone separation. At this event, maximum altitude errors up to 1 km may occur. Whether this value is acceptable, requires further analysis of the flight profile. A detailed analysis is outside the scope of this thesis, but it is possible to roughly determine the acceptability. The parachuting altitude is directly linked to the maximum parachuting shock, which is linked to the dynamic pressure, given by:

$$q_\infty = \frac{1}{2} \rho V^2 \quad (7.12)$$

The desired parachuting altitude is 3 km, but due to the estimation error, the actual altitude may be 2 km. For a nominal flight, the velocity is then 295 m/s. At 2 km altitude, the velocity is 285 m/s. The density is respectively 0.91 kg/m^3 and 1.00 kg/m^3 . It can then be calculated using equation (7.12), that the dynamic pressure at 2 km altitude is 2.6 % higher than at 3 km altitude, and consequently the forces during parachuting are higher by the same relative amount. This is obviously not a large difference, and well within any reasonable safety factor. Therefore, it can be tentatively concluded that, if the parachuting altitude is primarily determined by the maximum shock, then the obtained estimation errors are acceptable.

As for the other characteristic points, clearly the stage separation is well within limits: the altitude error is always less than 10 m, and the elevation error is well below 1 degree. Apogee altitude estimation errors are below 600 m. This is about 1 % error, and as no flight events depend on this altitude, it should be considered acceptable. The position error at landing is about 6 km at maximum. As stated before, this is undesirable for retrieval purposes, but not a critical error.

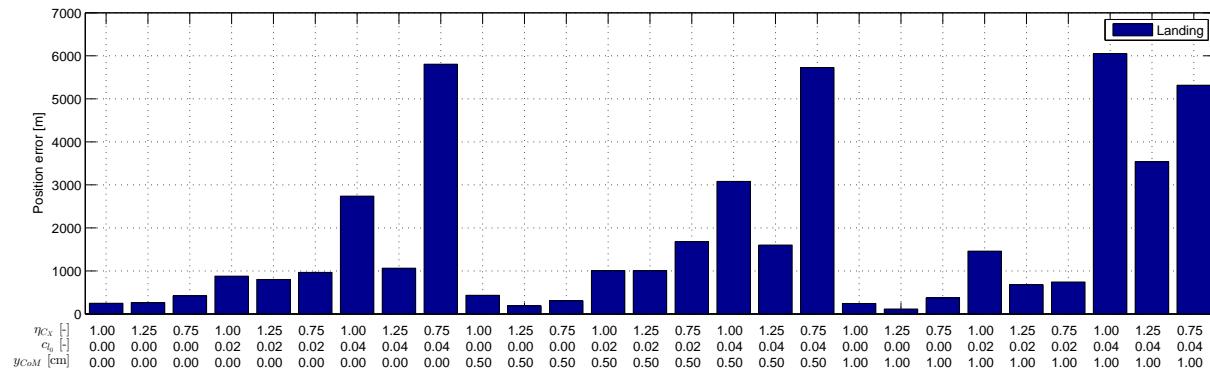


Figure 7.78: Position error at characteristic points for non-nominal flight

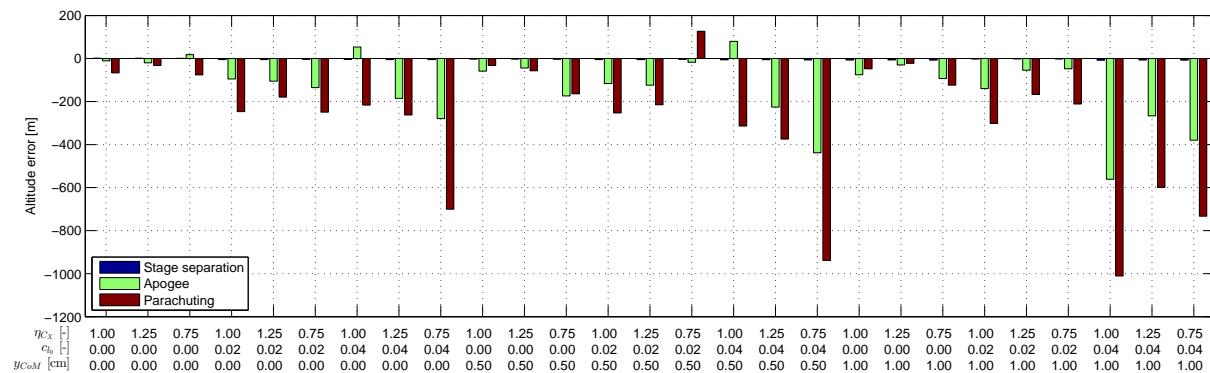


Figure 7.79: Altitude error at characteristic points for non-nominal flight

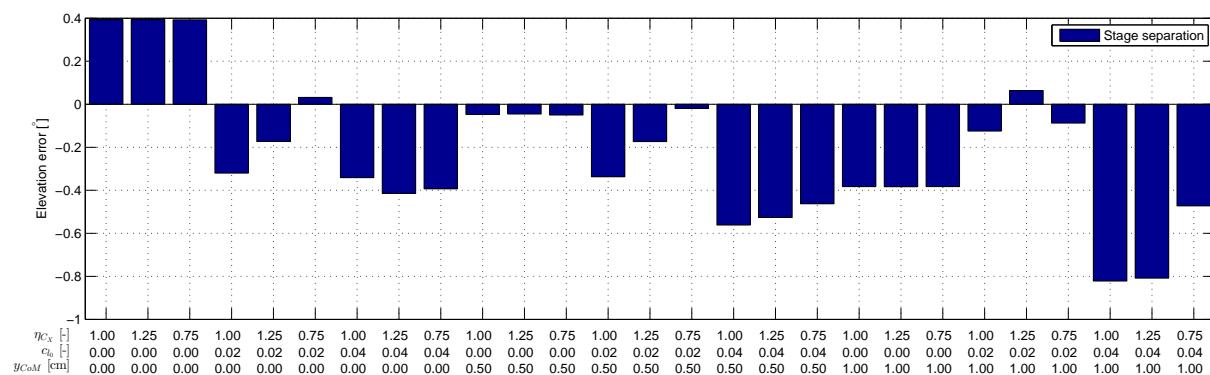


Figure 7.80: Elevation error at characteristic points for non-nominal flight

Table 7.22: Non-nominal estimation response surface coefficients characteristic points

	r_{nom}	$\bar{\beta}_1$	$\bar{\beta}_2$	$\bar{\beta}_3$	ε_{RMS}	$\bar{\varepsilon}_{RMS}$
Position landing	247.07	-671.23	1796.43	295.93	1789.74	7.24
Altitude stage separation	1.42	0.14	-1.63	-1.45	6.08	4.28
Altitude apogee	-10.61	27.23	-95.25	-49.35	166.26	15.68
Altitude parachuting	-66.86	64.58	-251.50	-65.99	270.89	4.05
Elevation stage separation	0.39	-0.03	-0.26	-0.16	0.64	1.64

7.8 Result summary

In this long chapter, several results were presented. Due to the large number of them, this section will provide a summary of these results. In section 7.4, several estimator types were compared, namely the simplified model and model-free estimator, paired with an extended Kalman filter (EKF) and unscented Kalman filter (UKF). There, it was concluded that the model-free EKF was the most suited estimator for further analysis. Also the UKF was found to be very suitable, but its gain in estimation accuracy compared to the performance penalty was deemed insufficient.

In the section about sensor factors, section 7.5, it was found that GPS measurements can have a very positive influence on the estimation result, especially in the presence of bias. Processing a position estimate or pseudorange measurement slightly improves the estimation result, provided that a sufficient number of satellites are in view. Furthermore, variations in sensor noise and measurement frequency were studied. There, it was found that in general increasing the noise or decreasing the frequency would increase the estimation error. However, the response is not fully monotonic, and indeed local minimums exist in the estimation error. The differences in response are more pronounced in the translational than in the rotational estimates; if only attitude and elevation are of interest, much worse sensors could be used. Looking at specific sensors, it can be found that the gyroscope is the dominant sensor for the estimation result: its influence is at least an order of magnitude larger than the accelerometer and magnetometer, both in noise and frequency. For sensor noise factors other than white noise, the accelerometer becomes dominant. In particular the x-scale factor error and the yx- and zx-non-orthogonalities were found to severely impact the estimation result, but their impact can be reduced by estimating them in-flight. In the end, sensor failure was studied. Here it was found that without accelerometer, no position estimation was possible, but attitude estimation was possible to a certain extend. Also gyroscope and magnetometer failure significantly impact the estimation result, but to a less catastrophic extend than failure of the accelerometer.

Finally, in section 7.7, it was concluded that non-nominal flight can have severe consequences for the estimation result. In this section it was confirmed that more severe dynamic characteristics, such as angular velocity and velocity, have adverse effect on the estimation results. Furthermore, it was tentatively concluded that the increased estimation errors are not critical to the flight, but further investigation would be required to confirm this.

Chapter 8

Post-flight Stratos II concept launcher results

The previous chapter presented the development of a flight estimator. This was performed with a simulated trajectory and simulated measurements. The measurement system developed in chapter 5 has actually been flown on a small rocket, the Stratos II concept launcher, shown in figure 8.1. This allows partial validation of the estimator against actual flight data. The validation is only limited, since it cannot be exactly known what the exact flight trajectory of the rocket was. Furthermore, the Stratos II concept launcher is a much smaller rocket than the actual Stratos II, and subject to less severe dynamic circumstances, whereas it was concluded in the previous chapter that dynamic characteristics have a significant influence on the estimation accuracy.

This chapter first starts with the description of the flight of the concept launcher on which the measurement system was flown. Next, in section 8.2 several trajectory simulations of nominal and non-nominal flights are performed, tailored to the observed flight. Then, these simulated trajectories are subject to flight estimation, to evaluate the performance of the estimator. Finally, estimation with the flight measurements will be performed.

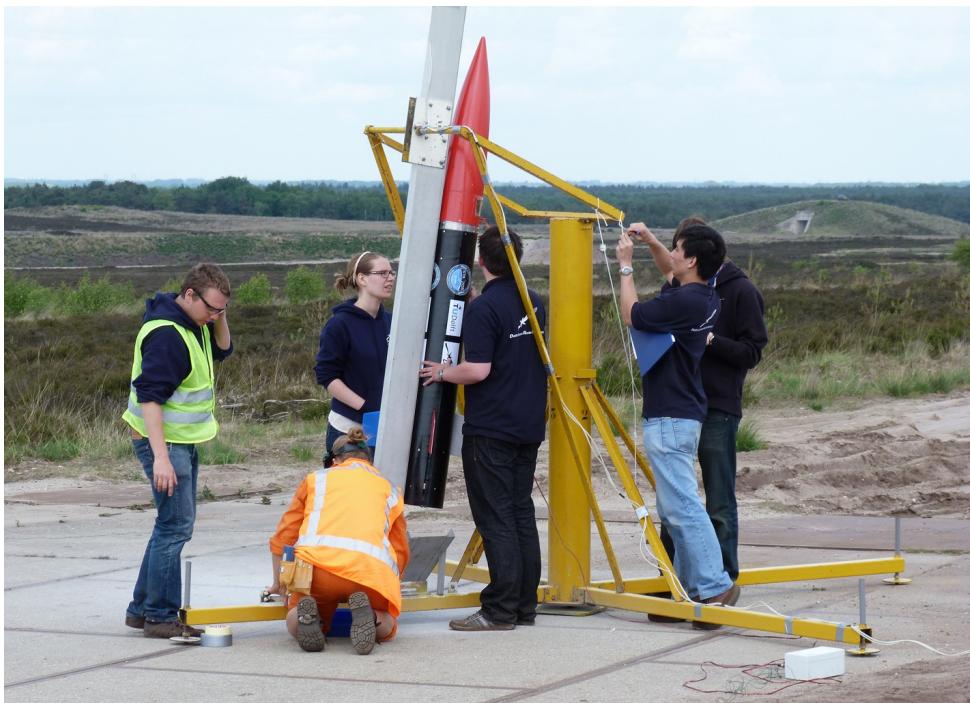


Figure 8.1: The Stratos II concept launcher in the launch tower. Source: B.A.C. Ambrosius

8.1 Flight results

The flight opportunity for the Stratos II concept launcher was on 9 September 2011 at ASK 't Harde in the Netherlands. The tower elevation was set to 80° from the horizontal plane, and the azimuth 30° east. At 13:13 CEST the rocket motor was ignited. Figure 8.2 shows the rocket in the launch tower just after ignition.



Figure 8.2: Rocket in launch tower just after ignition

After ignition, it was observed that it took more time for the motor to get to pressure than expected, before it lifted off. During its powered ascent no anomalies could be observed. After about 15 seconds the rocket reached apogee and separated the nose cone from the motor section. About 5 seconds later the parachute was ejected. The bag that was around the parachute to prevent it from getting entangled, however, stayed around the parachute, causing only a minimal deceleration, resulting in a high-velocity impact with the ground. During the last phase of the flight, violent tumbling could be observed.



Figure 8.3: Tumbling of the nose cone with undeployed parachute

After flight the motor module and nose cone were returned – the nose cone was cracked. The measurement system had mostly survived. The SD cards were ejected by the impact shock, but could be recovered. The measurement system showed only minor damage. Inspection of the motor module revealed a large hole in the side at the top of the motor, shown in Figure 8.4. Apparently the motor casing had burned near the end-cap and then through the motor mount. It can be expected that this had a significant effect on the motor performance.



Figure 8.4: Motor casing with burn hole

8.1.1 Raw measurement data

The pressure sensor, accelerometer and gyroscope worked flawlessly. The recorded measurement data are presented in Figure 8.5. Ignition is at $t = 0$. It is clear from the acceleration data that the motor takes some time to start up: the vehicle only starts accelerating after 3.4 seconds. After 17 seconds a small shock can be seen, which corresponds to nose cone separation. Then, a few seconds later the tumbling descent starts. This tumbling can also be seen in the gyroscope plots. From the gyroscope data it can also be seen that during ascent a roll motion of about $200 \text{ }^{\circ}/\text{s}$ is present. The third plot contains pressure data. It shows the expected parabolic curve for the pressure. Finally, the fourth plot shows the altitude according to the GPS receiver. Nominally, the GPS module would output a position estimate once per second. Unfortunately, this worked only during part of this flight. After ignition, there are only two points available, before regular position estimates can be made again after 9 seconds. Furthermore, after 13 seconds there is an unexplained jump in altitude. Finally during the phase which according to the accelerometer is the tumbling descent, the GPS receiver keeps the position estimate fixed. This is a measurement error, since it is clear both from the video footage as from the pressure measurements that the rocket is descending.

Finally, also a three-axis magnetometer was also present. Unfortunately the data from this sensor appeared to be hardly useful. The magnetic field can locally assumed to be constant, and therefore the norm of the output of the magnetometer should be relatively constant. However, there appeared to be no solution, even taking bias into account, where the norm of the output of the magnetometer stayed constant, and therefore no useful information could be extracted from this.

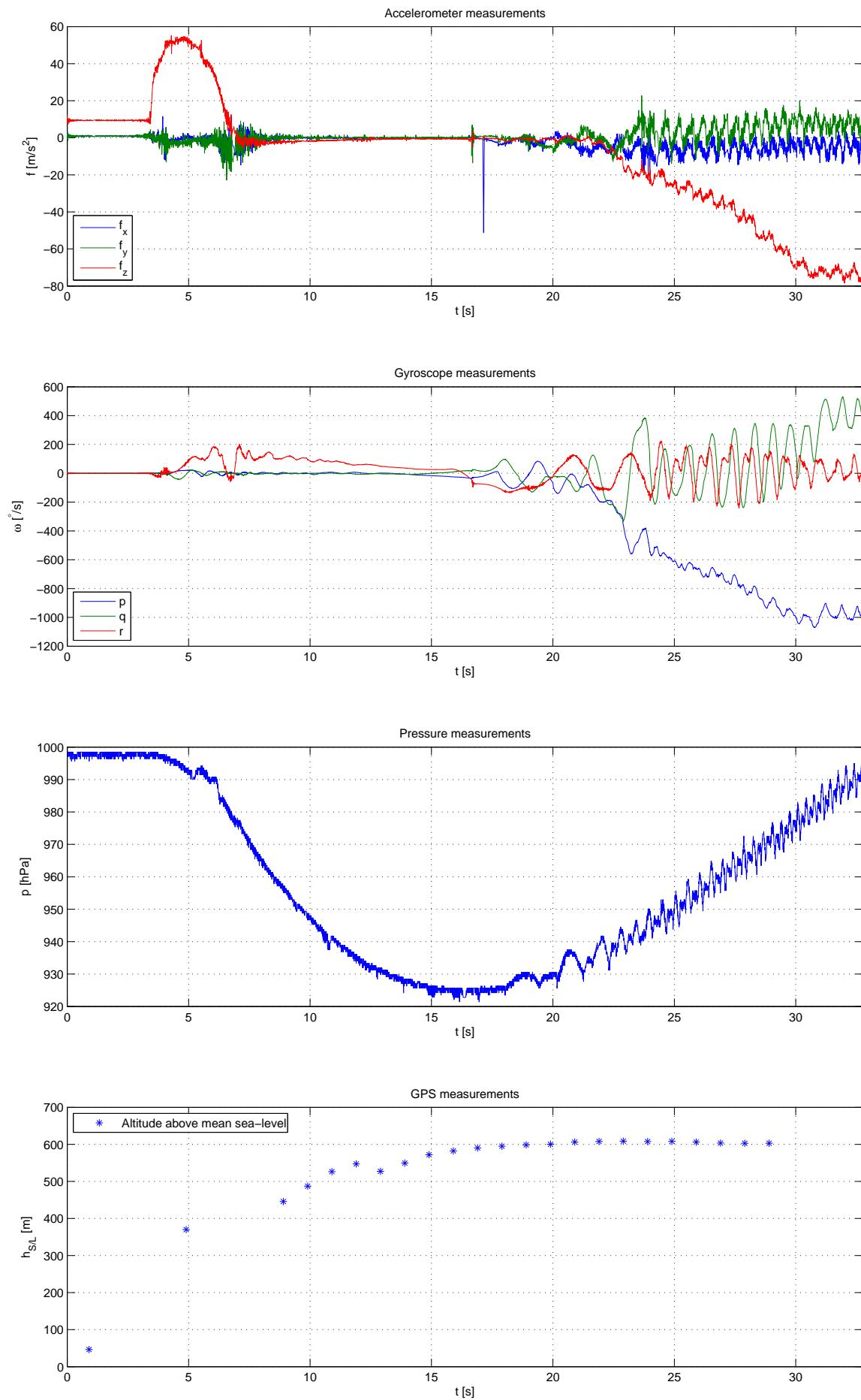


Figure 8.5: Raw measurement data

8.2 Trajectory simulation

For the concept launcher, simulations have been performed like those for the Stratos II. Also here three factors have been selected that are expected to have a major influence on the flight. These are the same as for Stratos II, except that the axial force coefficient has been replaced by the thrust magnitude. This has been done since it was observed in the previous section that there were problems with the motor, which most likely had as consequence that the thrust was lower than expected.

The resulting trajectories are shown in figure 8.6. The maximum altitude lies around 1160 m, while reducing the motor performance to 60 % yields apogee altitudes around 480 m. It is also clearly shown that center of mass offset yields a large spread of ground tracks. The flight time solely depends on thrust magnitude, and varies between 30 and 50 seconds. Figure 8.7 shows the velocity and figure 8.8 shows the Mach number. During the entire flight the rocket stays subsonic. Figures 8.9 and 8.10 show respectively the attitude and the angular velocity. Interestingly, in descent the angular velocity is relatively large, even for flights without rolling moment coefficient. Apparently, the used aerodynamic model possesses relatively little angular damping in y- and z-direction.

For a nominal, parachuted descent, the parachute could be modeled as an increased drag coefficient acting on the nose cone. It is clear that this model is insufficient for the actual observed descent motion, where the rocket was tumbling. A proper model of the tumbling descent would involve modeling both the folded parachute and the nose cone as individual six-degrees of freedom bodies, interconnected with a rope that constraints the maximum distance between the bodies. Because of the complexity of the model, only a non-parachuted descent has been simulated and in all simulations, the full rocket is in descent.

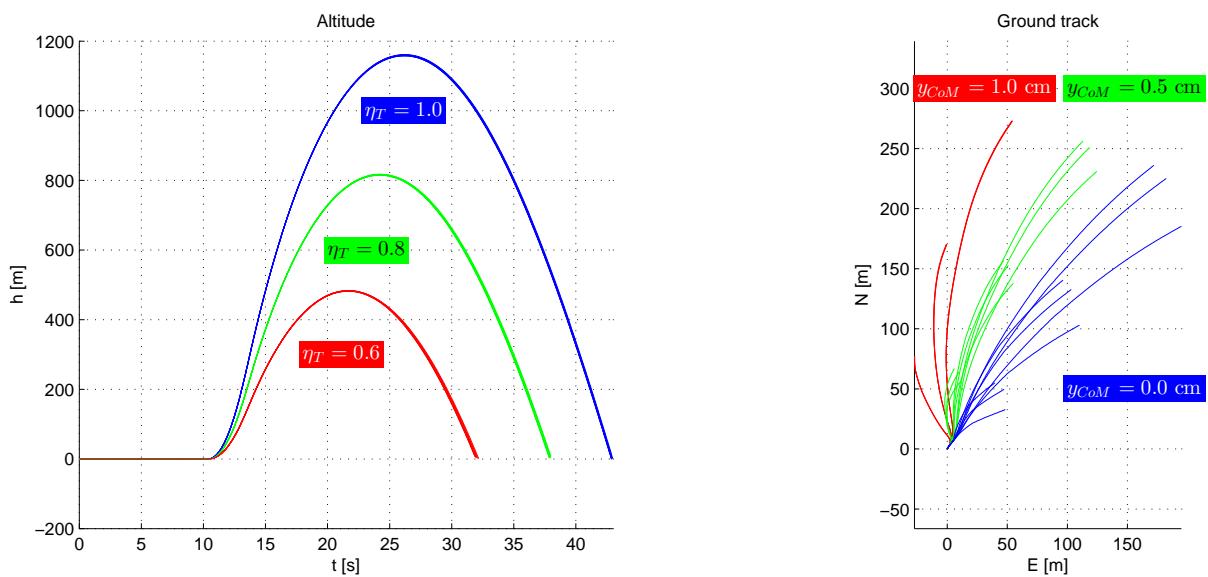


Figure 8.6: Non-nominal flight trajectory

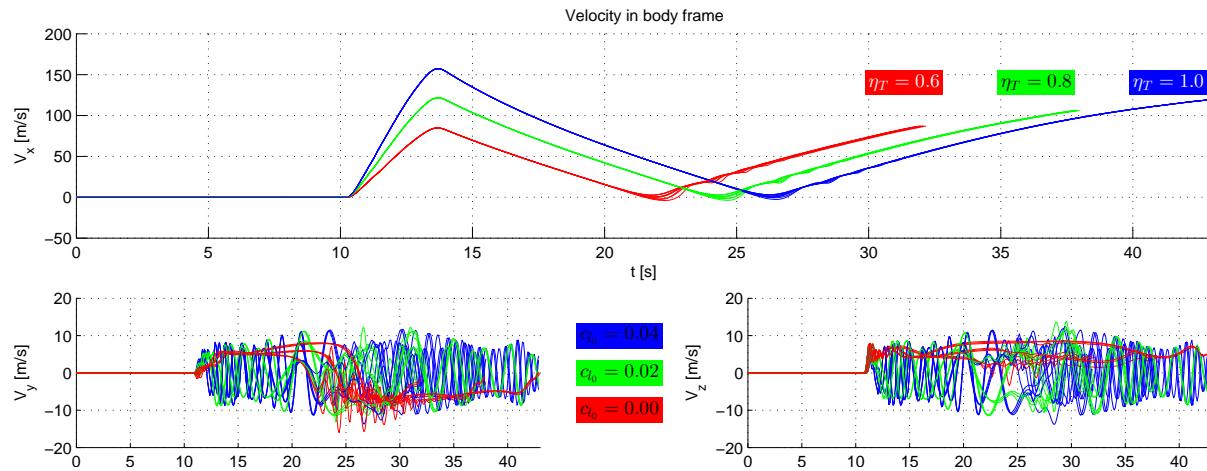


Figure 8.7: Non-nominal velocity

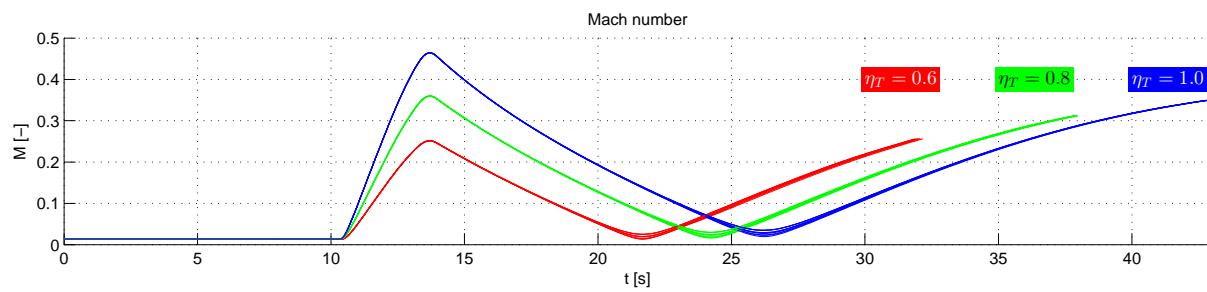


Figure 8.8: Non-nominal Mach number

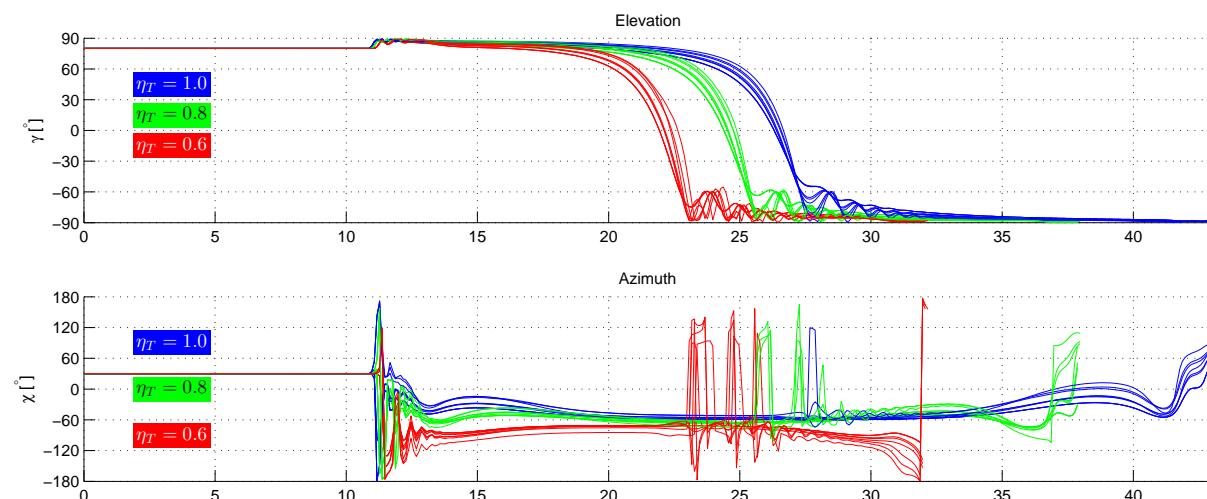


Figure 8.9: Non-nominal attitude

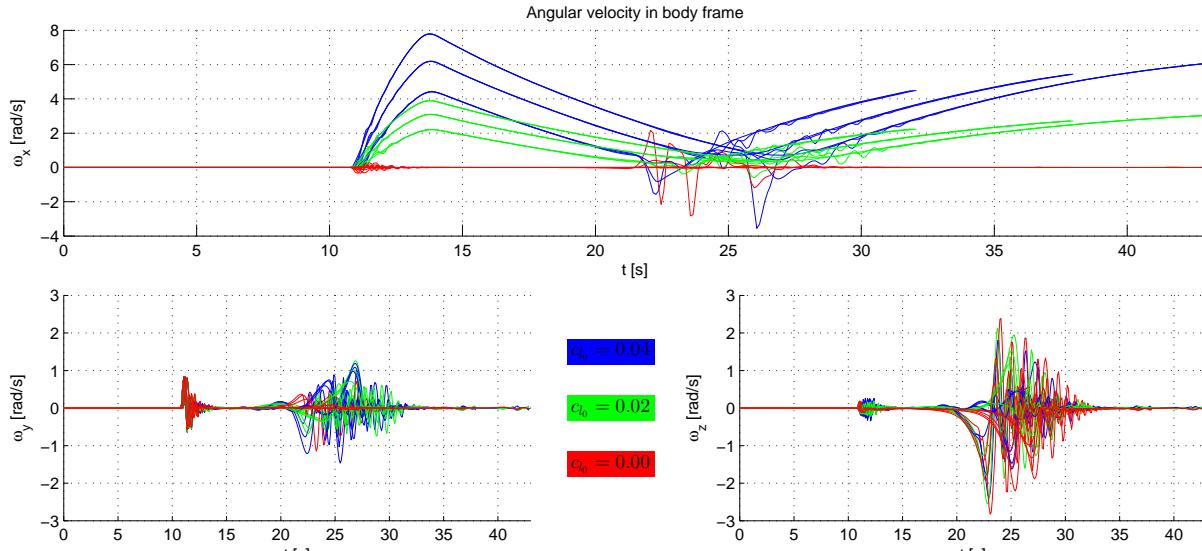


Figure 8.10: Non-nominal angular velocity

Table 8.1: Flight characteristics non-nominal flight

Case	η_T [-]	c_{l0} [-]	y_{CoM} [cm]	t_f [s]	h_A [m]	max V [m/s]	max M [-]	max ω [$^{\circ}/s$]
1	1.00	0.00	0.00	42.87	1157.98	157.40	0.46	56.14
2	0.80	0.00	0.00	37.87	815.17	121.88	0.36	67.58
3	0.60	0.00	0.00	31.87	481.92	86.95	0.26	82.13
4	1.00	0.02	0.00	42.87	1157.96	157.39	0.46	223.22
5	0.80	0.02	0.00	37.87	815.15	121.88	0.36	177.56
6	0.60	0.02	0.00	31.87	481.90	86.95	0.26	128.28
7	1.00	0.04	0.00	42.87	1157.93	157.39	0.46	446.45
8	0.80	0.04	0.00	37.87	815.13	121.88	0.36	355.10
9	0.60	0.04	0.00	31.87	481.88	86.95	0.26	256.46
10	1.00	0.00	0.50	42.87	1160.20	157.42	0.46	75.96
11	0.80	0.00	0.50	37.87	816.83	121.90	0.36	92.01
12	0.60	0.00	0.50	31.97	482.98	87.18	0.26	112.41
13	1.00	0.02	0.50	42.87	1159.57	157.42	0.46	223.20
14	0.80	0.02	0.50	37.87	816.47	121.90	0.36	177.38
15	0.60	0.02	0.50	31.97	482.95	87.23	0.26	128.29
16	1.00	0.04	0.50	42.87	1159.36	157.43	0.46	446.29
17	0.80	0.04	0.50	37.87	816.35	121.91	0.36	354.75
18	0.60	0.04	0.50	31.97	482.84	87.26	0.26	257.08
19	1.00	0.00	1.00	42.97	1161.32	157.43	0.46	108.26
20	0.80	0.00	1.00	37.97	817.59	121.90	0.36	133.54
21	0.60	0.00	1.00	32.17	483.33	87.23	0.26	177.44
22	1.00	0.02	1.00	42.87	1160.41	157.45	0.46	223.16
23	0.80	0.02	1.00	37.97	817.13	121.92	0.36	177.21
24	0.60	0.02	1.00	32.07	483.44	87.22	0.26	146.74
25	1.00	0.04	1.00	42.87	1160.14	157.45	0.46	446.10
26	0.80	0.04	1.00	37.97	817.01	121.93	0.36	354.38
27	0.60	0.04	1.00	32.07	483.33	87.36	0.26	257.47

8.3 Estimation results of simulated trajectory

Before estimation is performed with flight measurements, first the simulations from the previous section will be used to evaluate the performance of the estimator. As can be found in the previous section, during the entire flight the rocket stays subsonic, and as a consequence presumably valid pressure measurements are available during the entire flight. Secondly, as was described in section 8.1, the magnetometer did not function correctly during flight. Therefore, only estimations are performed without magnetometer.

In the previous chapter two estimators were used: the simplified model estimator, and the kinematic estimator. It was found there that for Stratos II the kinematic model estimator performed vastly superior to the simplified model estimator. Since the concept launcher flight is shorter, and less dynamic than the Stratos II flight, it is a most likely valid assumption to assume that this model is also superior for the concept launcher. Therefore, only the kinematic model estimator will be used. It will, however, not only be used in conjunction with the EKF, but also with the UKF, and also the RTS smoother.

8.3.1 Pressure measurements

Before further analysis is performed, first the effect of the pressure sensor is analyzed. In Figures 8.12 and 8.11, the position estimate without and with pressure measurements are shown, respectively. Clearly, the estimate is much better without pressure measurements, and the reason for this is the imprecision of the atmosphere model. It can be concluded, that pressure measurements will not necessarily improve the estimation result, and for it to be an improvement, a pressure model is required that is at least as accurate as the estimation error without using the pressure measurements. Still, even though the pressure measurements as they are available here may not improve the estimation error, it may improve the robustness of the estimator, in particular against sensor noise factors, and therefore they are still included in all further estimations.

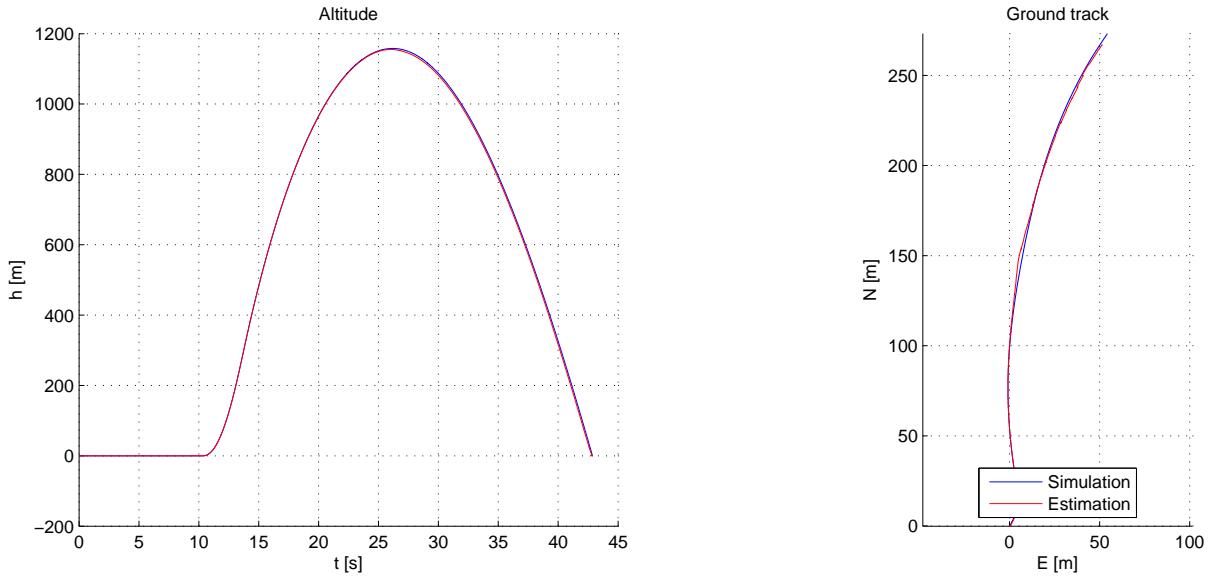


Figure 8.11: Position estimation with no pressure measurements

8.3.2 Estimation with nominal sensors

The nominal sensor set consists of an accelerometer and gyroscope, and optionally a pressure sensor. For all trajectories simulated in section 8.2, estimation has been performed. In figures

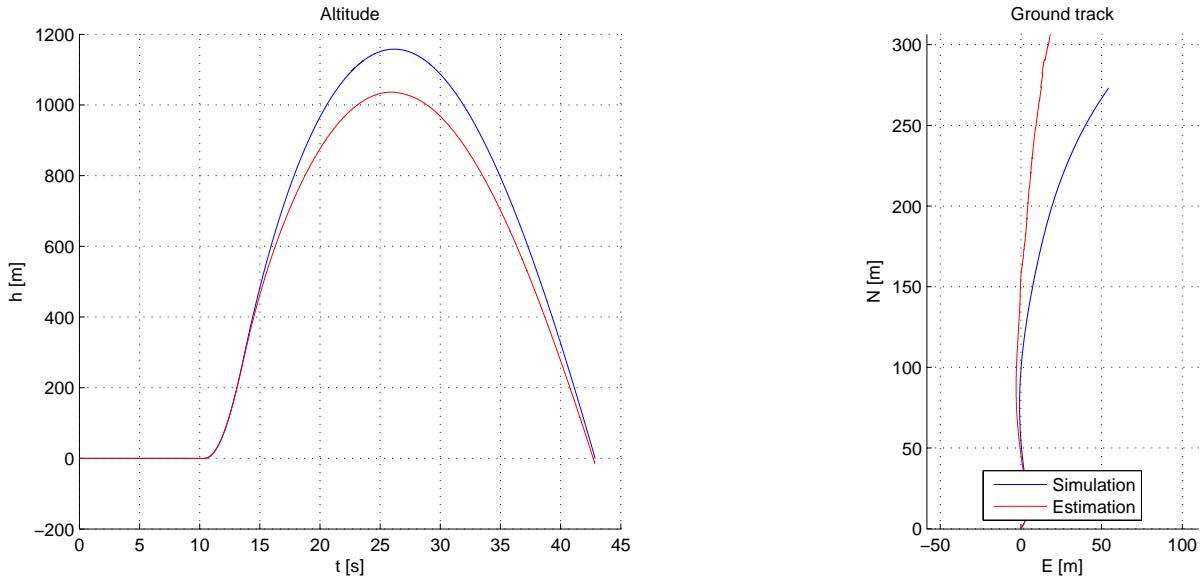


Figure 8.12: Position estimation with pressure measurements

8.13 to 8.16, the estimation errors without pressure sensor, and in figures 8.17 to 8.20 the estimation errors with pressure sensor are presented. The observation from the previous section is immediately clear: the addition of a pressure sensor actually increases the estimation error. For the case without pressure sensor, the errors are extremely small: less than 18 m in position, and less than 5 m in altitude. Attitude errors barely reach 1 degree. In the presence of a pressure sensor, the estimation errors are still very reasonable: position errors stay below 100 m, and altitude errors stay below 70 m. Attitude errors stay below 7 degrees.

To determine which flight factors influence the estimation result, a response surface has been made from the estimation errors. The response surface coefficients can be found in table 8.2 and table 8.3 for the cases without and with pressure measurements respectively. The response surface correspond to the following equation:

$$r = r_{mean} + \bar{\beta}_1 \frac{1 - \eta_T}{0.2} + \bar{\beta}_2 \frac{c_{l_0}}{0.02} + \bar{\beta}_3 \frac{y_{CoM}}{0.005} \quad (8.1)$$

In the tables, also the RMS residual and the normalized RMS residual can be found. Small values for these residuals indicate a good fit of the response surface. For the case without pressure measurements, the fit is good, as the residuals are small compared to the nominal estimation error. In the case of pressure measurements, the fit is good for position and altitude, but much worse for attitude and elevation. The consequence of this is that only the response surface coefficients for the case without pressure measurements, and for the case with pressure measurements, those corresponding to position and altitude, can be used to draw conclusions on the effect of flight factors on these responses.

The primary factors in estimation error can be immediately seen: the main error in position is due to thrust magnitude. This can also be found in the response surface coefficients, since $\bar{\beta}_1$ is by far the largest coefficient. In particular for position and altitude mean error, the thrust magnitude response surface coefficient is at least one order of magnitude larger than the others. For the attitude estimation errors thrust magnitude and rolling moment coefficient play an approximately equal role, as their coefficients are of the same order of magnitude. Both from the plots and the coefficients, it is clear that the attitude and elevation are much less sensitive to flight factors than position, some exceptions for the case with pressure measurements.

For the four responses that did not yet yield a good quality of fit, a second-order response surface, including interactions is made. This response surface corresponds to the following

equation:

$$\begin{aligned}
 r = r_{mean} + \bar{\beta}_1 \frac{1 - \eta_T}{0.2} + \bar{\beta}_2 \frac{c_{l_0}}{0.02} + \bar{\beta}_3 \frac{y_{CoM}}{0.005} \\
 + \bar{\beta}_{12} \left(\frac{1 - \eta_T}{0.2} \right) \left(\frac{c_{l_0}}{0.02} \right) + \bar{\beta}_{23} \left(\frac{c_{l_0}}{0.02} \right) \left(\frac{y_{CoM}}{0.005} \right) + \bar{\beta}_{31} \left(\frac{y_{CoM}}{0.005} \right) \left(\frac{1 - \eta_T}{0.2} \right) \quad (8.2) \\
 + \bar{\beta}_{11} \left(\frac{1 - \eta_T}{0.2} \right)^2 + \bar{\beta}_{22} \left(\frac{c_{l_0}}{0.02} \right)^2 + \bar{\beta}_{33} \left(\frac{y_{CoM}}{0.005} \right)^2
 \end{aligned}$$

The coefficients of this response surface can be found in table 8.4. With this response surface, the quality of fit improves significantly. It is still not as good as the other response surfaces, but it is clear that the cause of the bad initial fit is the effect of interactions and higher-order effects. The conclusions drawn previously do not change, however: the thrust magnitude and rolling moment coefficient, but in this case also the interaction between them, are primarily responsible for the attitude and elevation estimation error. The center-of-mass offset has almost no effect.

Table 8.2: Concept launcher estimation error response surface coefficients

	r_{nom}	r_{mean}	$\bar{\beta}_1$	$\bar{\beta}_2$	$\bar{\beta}_3$	ε_{RMS}	$\bar{\varepsilon}_{RMS}$
Position mean	5.25	4.00	1.47	0.05	0.07	0.16	0.03
Position max	14.62	11.60	3.74	0.13	-0.00	0.54	0.04
Altitude mean	1.23	0.97	0.20	-0.03	0.03	0.05	0.04
Altitude max	3.88	3.42	0.18	-0.12	0.12	0.15	0.04
Attitude mean	0.18	0.22	0.04	0.06	0.00	0.03	0.15
Attitude max	0.34	0.57	0.06	0.17	0.03	0.12	0.36
Elevation mean	-0.11	-0.07	-0.01	0.02	0.01	0.04	0.36
Elevation max	-0.32	-0.37	0.02	0.04	-0.06	0.07	0.21

Table 8.3: Concept launcher with pressure measurement estimation error response surface coefficients

	r_{nom}	r_{mean}	$\bar{\beta}_1$	$\bar{\beta}_2$	$\bar{\beta}_3$	ε_{RMS}	$\bar{\varepsilon}_{RMS}$
Position mean	17.56	12.55	8.07	1.61	-0.14	2.45	0.14
Position max	56.10	41.90	24.81	5.58	-0.63	8.16	0.15
Altitude mean	16.50	10.61	6.38	0.28	-0.00	0.44	0.03
Altitude max	54.85	37.25	19.54	0.95	0.01	1.26	0.02
Attitude mean	0.18	0.44	0.33	0.35	-0.02	0.34	1.92
Attitude max	0.33	1.26	1.03	1.09	-0.06	1.19	3.63
Elevation mean	-0.10	-0.29	-0.30	-0.27	0.03	0.31	3.16
Elevation max	-0.31	-1.15	-1.01	-1.01	0.04	1.17	3.81

Table 8.4: Concept launcher with pressure measurement estimation error second order response surface coefficients

	r_{nom}	r_{mean}	$\bar{\beta}_1$	$\bar{\beta}_2$	$\bar{\beta}_3$	$\bar{\beta}_{12}$	$\bar{\beta}_{23}$	$\bar{\beta}_{31}$	$\bar{\beta}_{11}$	$\bar{\beta}_{22}$	$\bar{\beta}_{33}$	ε_{RMS}	$\bar{\varepsilon}_{RMS}$
Attitude mean	0.18	0.14	0.33	0.35	-0.02	0.40	-0.03	-0.04	0.21	0.24	0.00	0.15	0.82
Attitude max	0.33	0.18	1.03	1.09	-0.06	1.36	-0.16	-0.14	0.79	0.82	0.01	0.53	1.61
Elevation mean	-0.10	0.01	-0.30	-0.27	0.03	-0.36	0.03	0.04	-0.22	-0.24	0.00	0.13	1.28
Elevation max	-0.31	-0.06	-1.01	-1.01	0.04	-1.34	0.15	0.19	-0.76	-0.85	-0.02	0.50	1.64

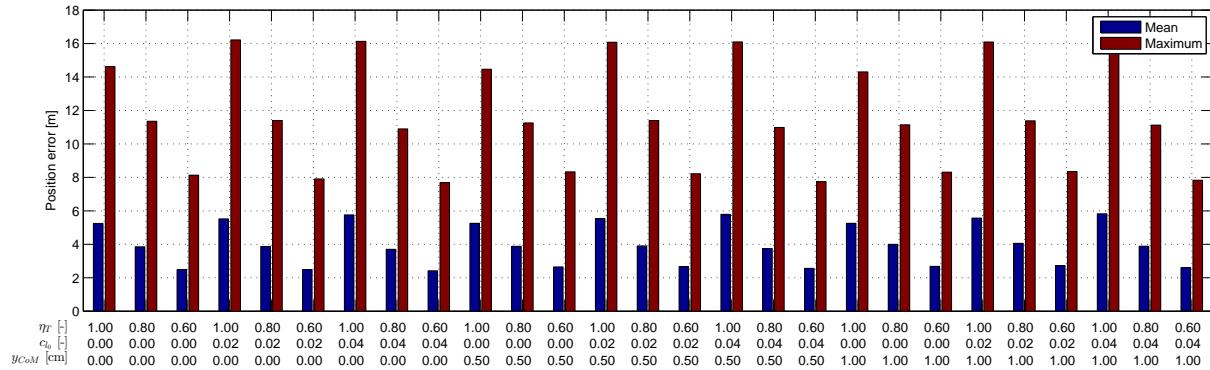


Figure 8.13: Position error for concept launcher flight

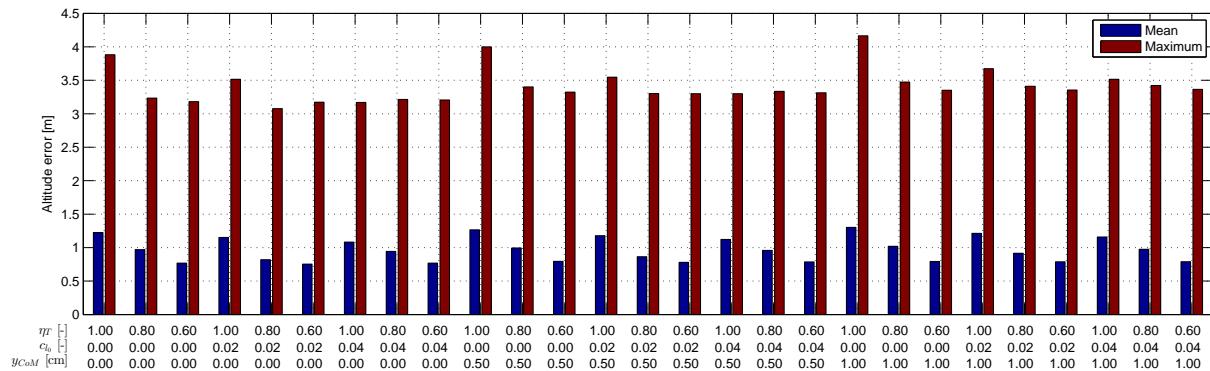


Figure 8.14: Altitude error for concept launcher flight

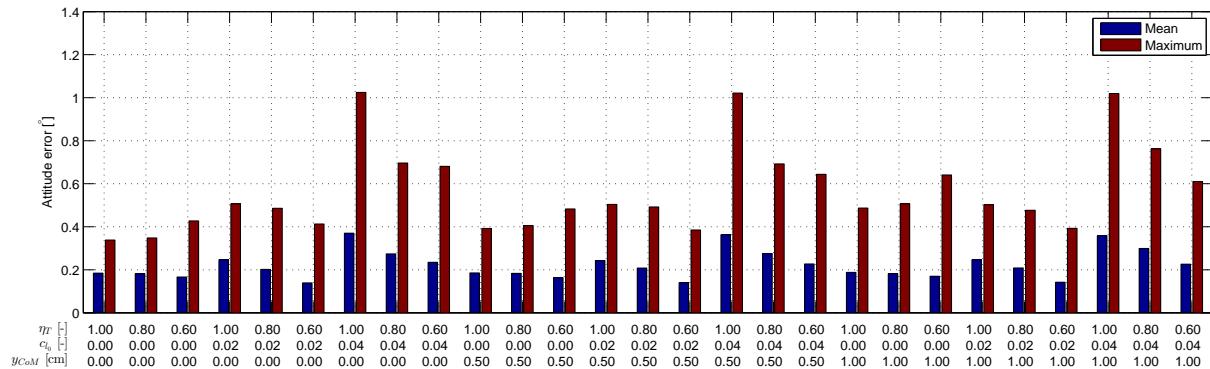


Figure 8.15: Attitude error for concept launcher flight

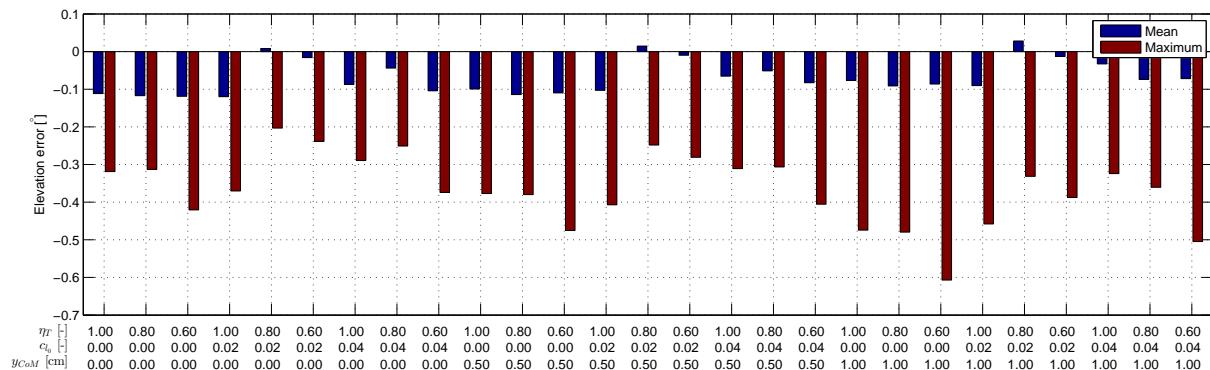
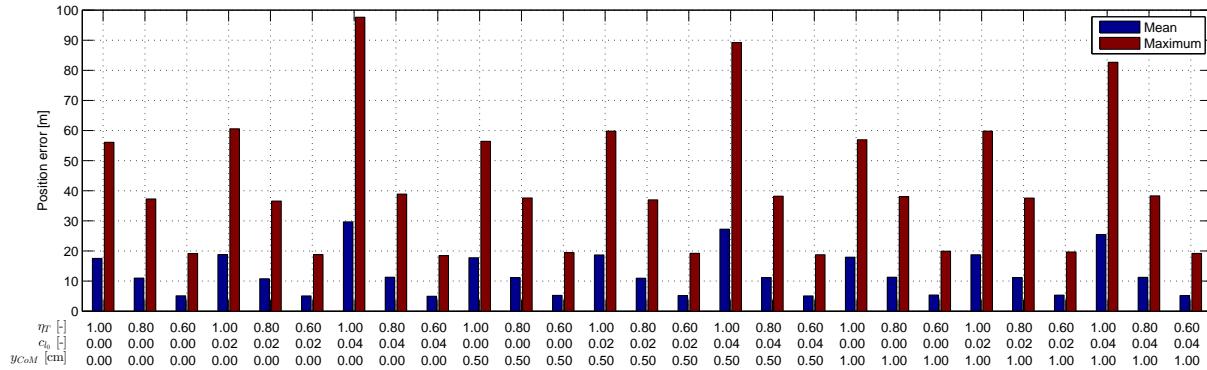


Figure 8.16: Elevation error for concept launcher flight



8.4 Sensor noise factors

Previously it was found that very good estimation results can be obtained with nominal sensors, regardless of the trajectory. In this section, noise factors on the sensors are studied, by creating a response surface from noise factors varied according to an orthogonal array. This is done to determine the noise factors that affect the estimation results the most. Then, the estimator is modified in order to also estimate these noise factors, and again the effect of noise factors on the modified estimator is studied. If sufficient information, in the form of measurements, is available to the estimator, it can be expected that the estimation error improves.

Similar to the Stratos II case, for both the accelerometer and gyroscope there are 12 noise factors, being 3 scale factor errors, 6 non-orthogonalities and 3 biases. As the magnetometer is absent, this yields in total 24 sensor noise factors. These noise factors are varied according to an orthogonal array, and a response surface around the nominal is created.

In respectively figure 8.21a and figure 8.21b the altitude estimation errors for estimation with and without pressure measurements is plotted. The first experiment is always the nominal case. As stated before, the pressure sensor does increase estimation error for the nominal case. However, in the presence of non-nominal sensors, the altitude estimation error does not vary significantly when the pressure measurements are included, whereas for estimation without pressure sensor, the altitude estimation error widely varies. This maximum altitude estimation errors peak to 300 m for the estimations without pressure sensors, but keep constant to about 120 m for estimations with pressure sensor.

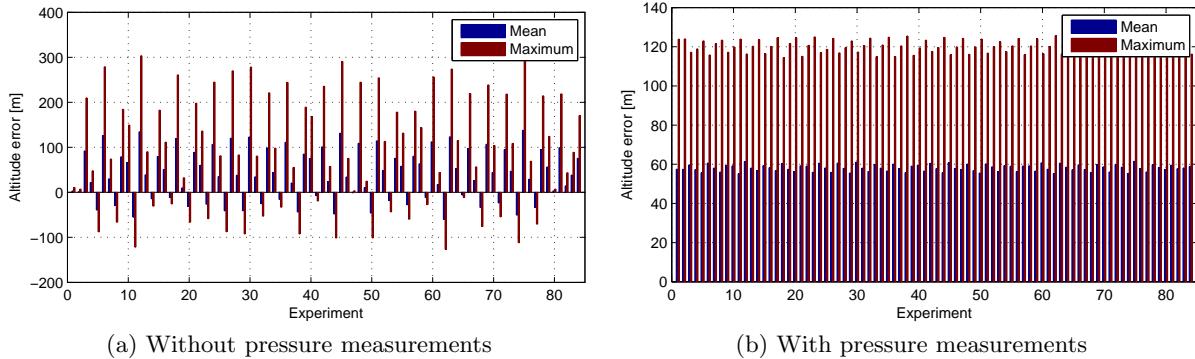


Figure 8.21: Altitude error without and with pressure measurements for non-nominal sensors

In figure 8.27, the estimation errors for position, altitude, attitude and elevation with varying sensor factors are presented for the pressure measurement case, and in figure 8.22 the corresponding normalized response surface coefficients are given. The axes of this figure have been scaled to accommodate the coefficients for position and altitude error. The attitude error coefficients are allowed to run off the plots, since in figure 8.27c, it can be found that the attitude errors are for all cases very reasonable. While there are many factors that affect the attitude estimation error, for the position error clear primary factors can be found. The largest effect on the position error are the non-orthogonalities of the accelerometer which cause projection of the x-acceleration on the two other axes. Therefore, in order to reduce the position estimation error, the two mentioned non-orthogonalities are added to the state vector in order to be estimated.

While tuning the estimator for these non-orthogonality estimations, it was found that the non-orthogonality estimations would converge much better if pressure measurements would only be considered while the rocket was on the ground. If pressure measurements were also considered while the rocket is flying, for some, unknown reason, the scale factor error would start to diverge again.

To show that the estimator is properly able to estimate the augmented non-orthogonalities,

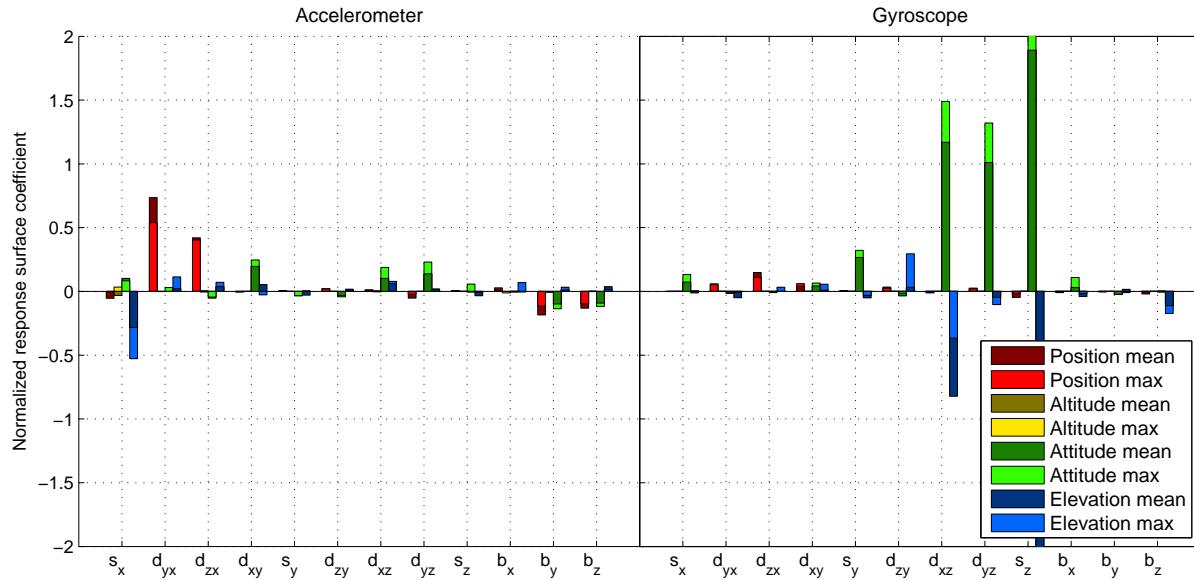


Figure 8.22: Normalized response surface coefficients with pressure measurements

in figure 8.23 the estimated non-orthogonalities, and their convergence to the real values, for a flight where both non-orthogonalities are present, is shown. The estimation errors for all experiments are shown in figure 8.28, and the response surface coefficients in figure 8.24. It is clear that in all cases the position estimate is improved, and in many cases also the altitude estimate. Attitude and elevation estimation were almost unaffected by the two added non-orthogonalities. However, both for position and altitude there appear to be still factors that affect the estimation result, as there is still variation in the estimation errors. From the response surface coefficients, this factor is found to be the scale factor of the accelerometer in x-direction, and this is the next sensor noise factor that will be estimated.

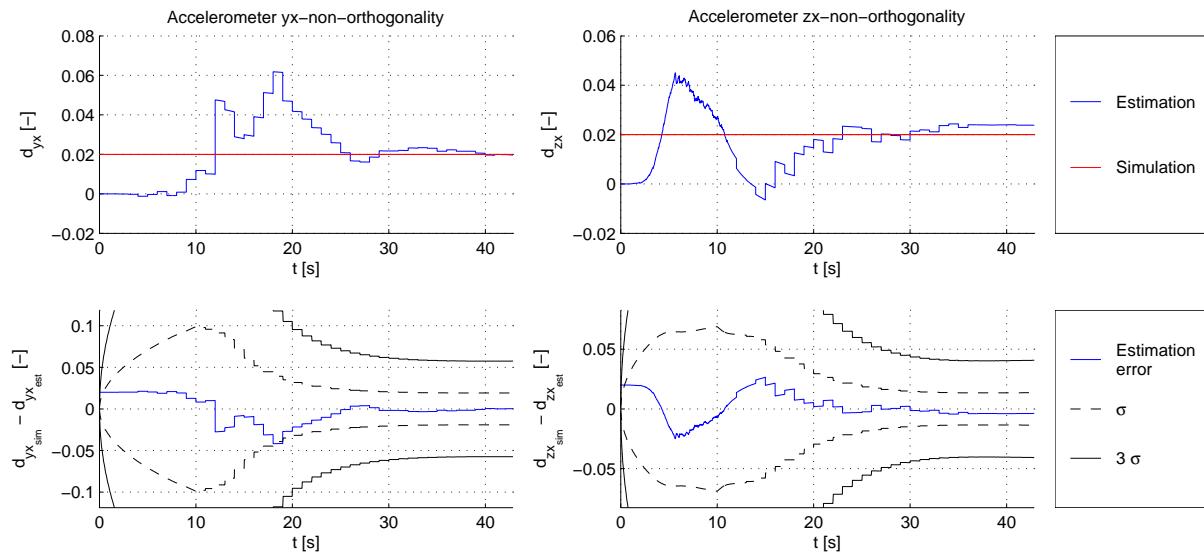


Figure 8.23: Estimation of the accelerometer non-orthogonalities

The estimation with accelerometer x-scale factor has been performed with the results in figure 8.29. It is clear that neither the position nor the altitude error is affected significantly any longer by varying any sensor noise factor, as both the mean and maximum position and altitude error are constant, irrespective of sensor noise characteristics. The attitude error continues to

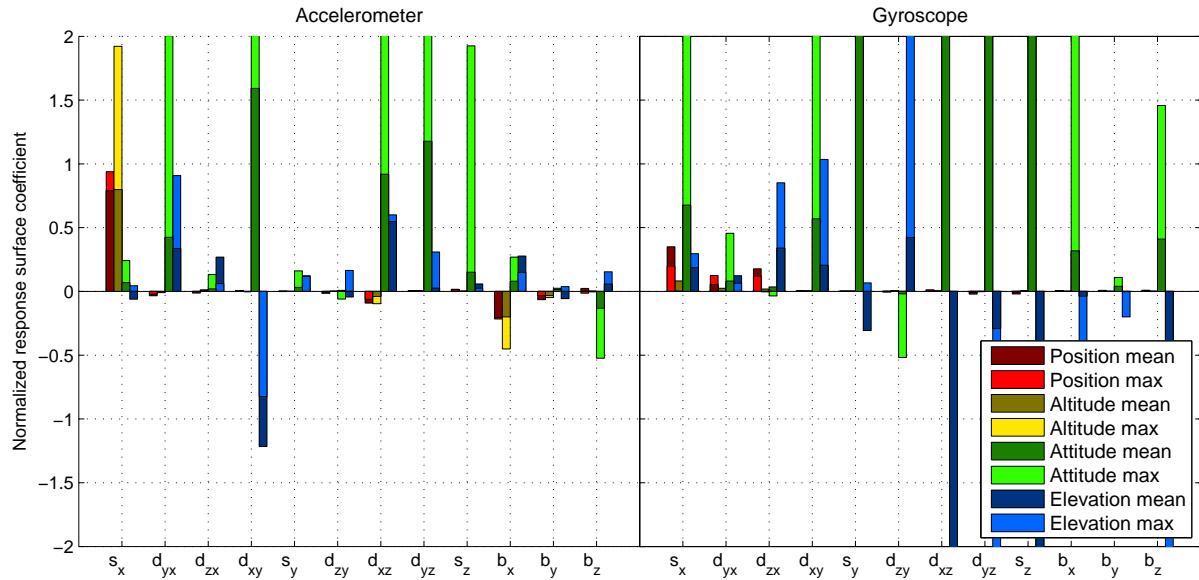


Figure 8.24: Normalized response surface coefficients with pressure measurements and accelerometer non-orthogonality estimation

be affected by varying sensor noise, but as shown in figure 8.29c, never exceeded 10° , which is reasonably small, and therefore no consideration was given to improving the attitude error. However, should the attitude error be improved, then the next factors which should be estimated are the gyroscope scale factor errors, and in particular the z-scale factor error. In figure 8.29c, aside from the nominal experiment, two experiments can be found where the attitude error is much lower than all other experiments. These are experiments 46 and 78, and these correspond to the experiments without any scale factor error for the gyroscope. In figure 8.29d this effect is even more clear: groups of larger elevation estimation errors, separated by three or six experiments with lower estimation errors. The groups with lower estimation errors are those where the gyroscope z-scale factor error is zero, the larger errors are those where these errors are present. In any case, without measurements that contain attitude information, such as magnetometer measurements, it is unlikely that the estimator is able to estimate these scale factor errors.

For all estimations, the maximum position estimation error was reached halfway in descent, between 30 s and 35 s. In all cases, the position and altitude estimation errors at apogee were almost equal, meaning that the position error at apogee is almost fully due to errors in the altitude. Incidentally, position error at apogee is almost equal to the mean position error, being approximately 40 m.

The estimated scale factor for a configuration where the estimated non-orthogonalities and scale factors are present, is shown in figure 8.25. It can be found that both non-orthogonalities converge, although slowly. The scale factor error does not converge correctly to its actual value. It does, however, stay near the estimated error covariance. The black covariance lines stay constant near the end of flight, indicating that the estimator is not able to reduce the estimation error further. Despite this, it was found that estimation of the scale factor error did improve the estimation result, and is therefore still performed.

Concluding these results, it can be stated that estimation errors in position or altitude can be attributed to the accelerometer, and only to the scale factor error in x-direction, and the non-orthogonalities that project the x-component of the acceleration on the other axes. All other sensor noise factors do not significantly affect the position and altitude estimation result. The fact that in all cases the x-component of the acceleration is involved, is a consequence from the fact that the x-component is by far the greatest in magnitude, and therefore the effects of any errors with it are large, especially for the non-orthogonalities, since even a small non-

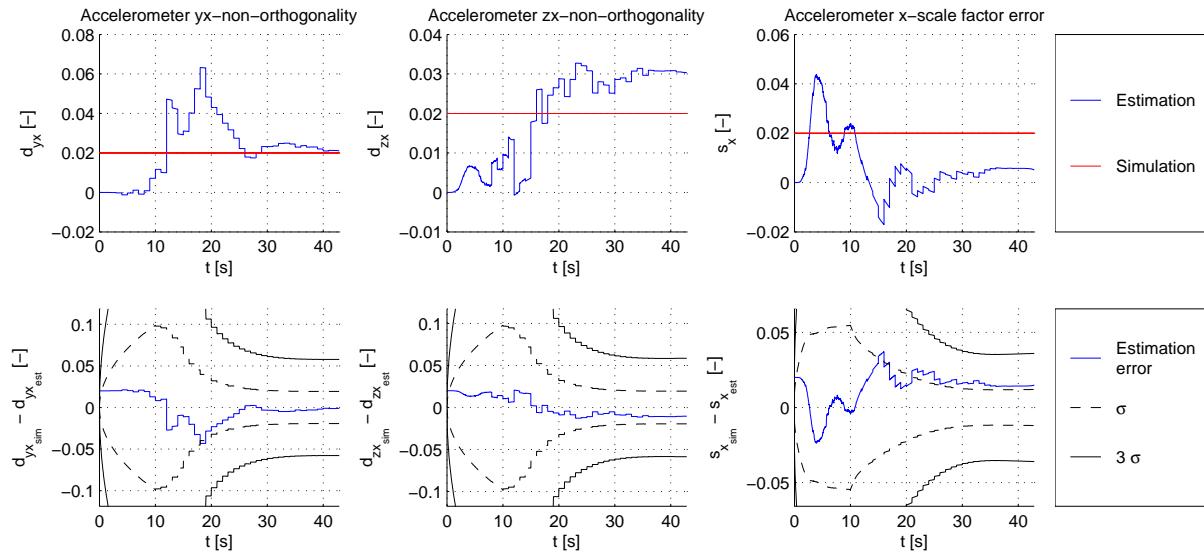


Figure 8.25: Estimation of the accelerometer non-orthogonalities and scale factor

orthogonality will yield an error on the affected axis of the same order or greater order than the nominal magnitude of the acceleration component on that axis. It would be expected that also gyroscope errors affect the position or altitude estimation errors, as the DCM is present in the translational equation of motion. Apparently, the errors in attitude introduced by gyroscope errors are sufficiently small that they are not significant to the position estimation result.

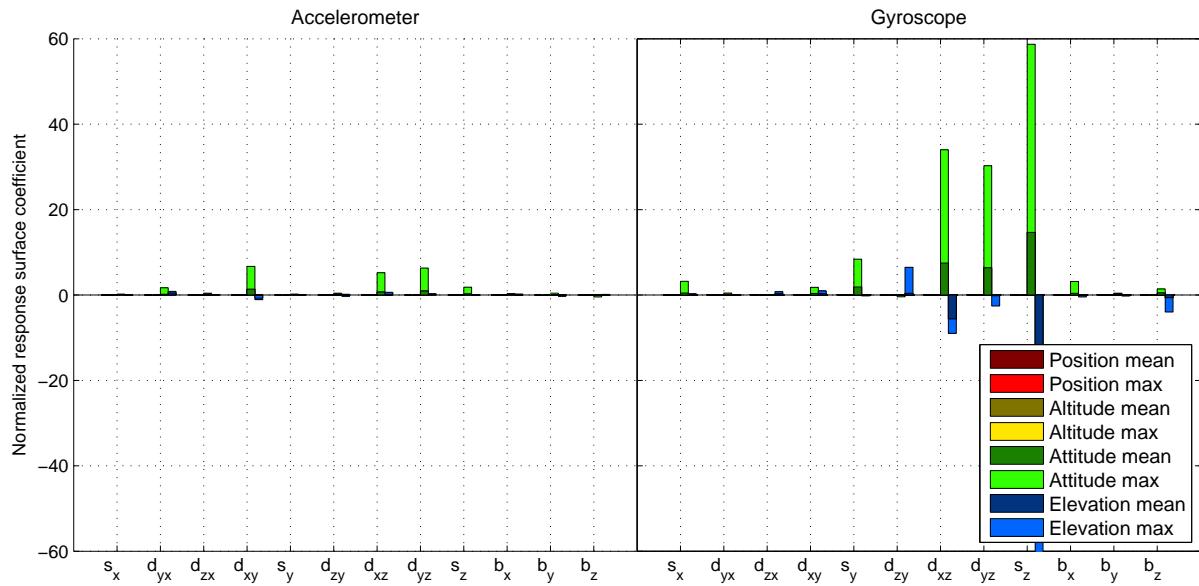


Figure 8.26: Normalized response surface coefficients with pressure measurements and accelerometer non-orthogonality and scale factor estimation

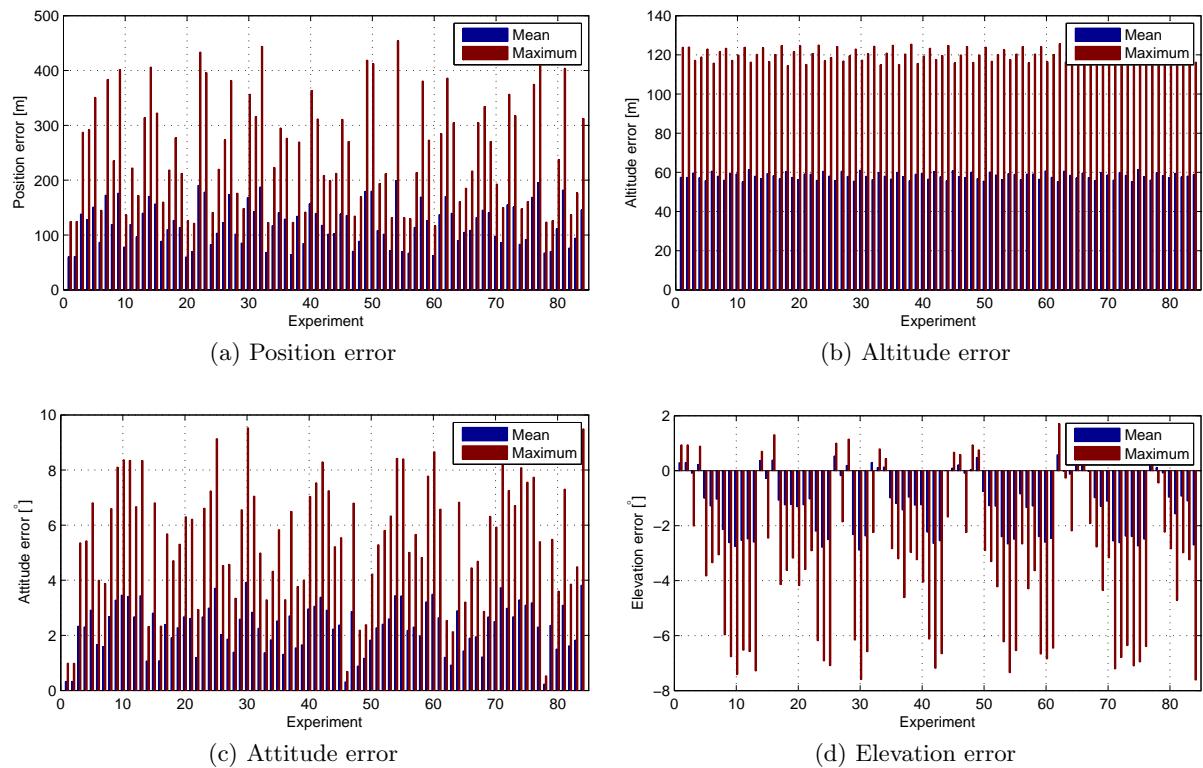


Figure 8.27: Estimation errors with sensor noise factors varied by orthogonal array and with pressure measurements

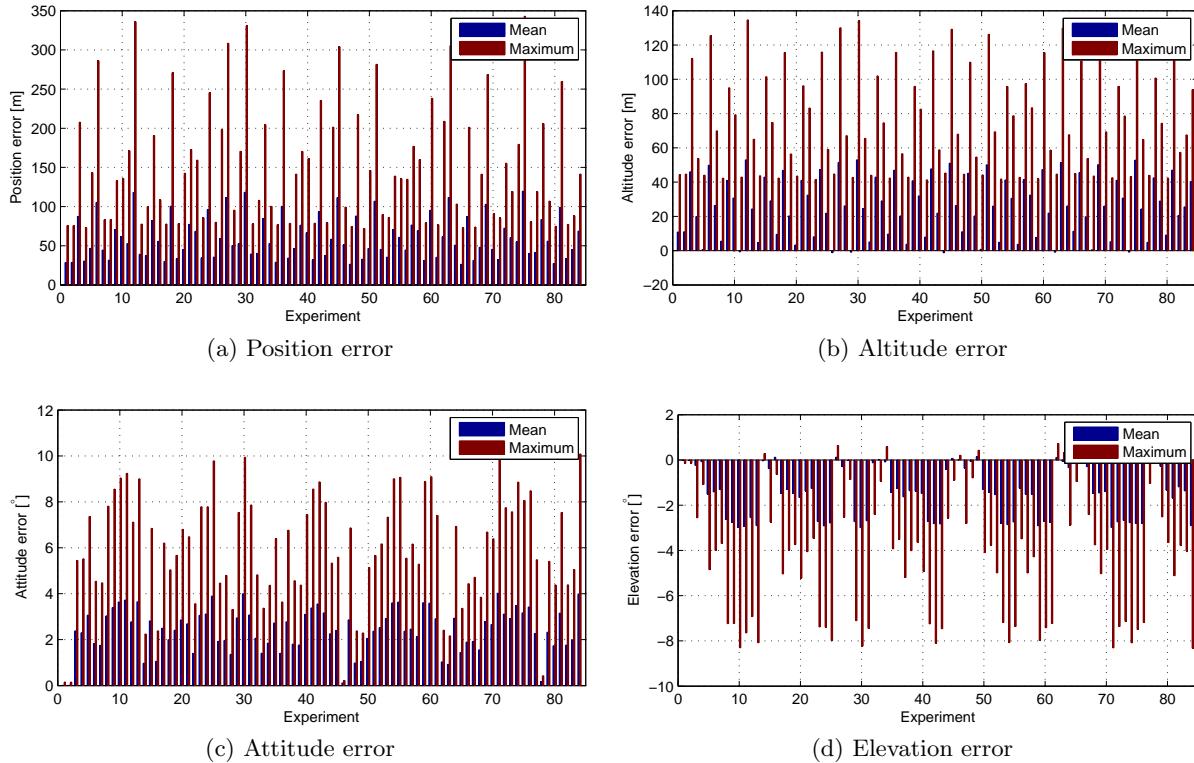


Figure 8.28: Estimation errors with sensor noise factors varied by orthogonal array and with pressure measurements and accelerometer non-orthogonality estimation

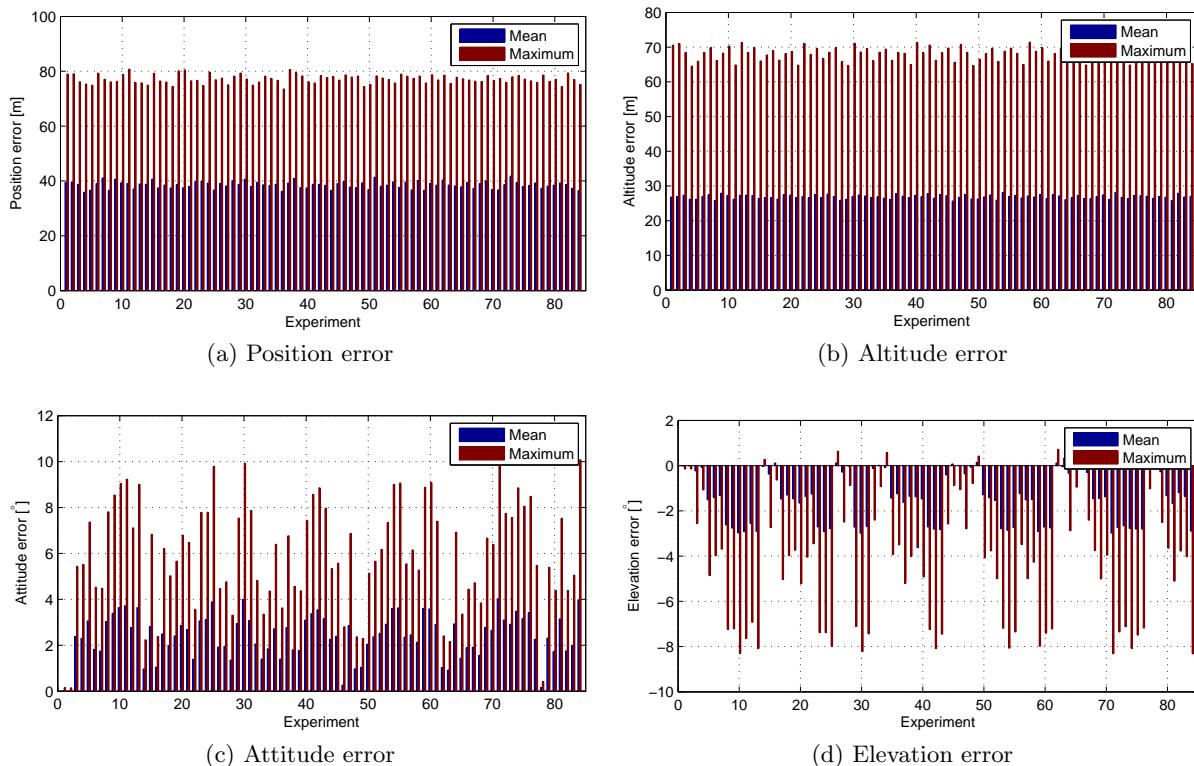


Figure 8.29: Estimation errors with sensor noise factors varied by orthogonal array and with pressure measurements and accelerometer non-orthogonality and x-scale factor error estimation

8.5 Estimation results of flight

Finally, the earlier discussed data from the flight can be used for estimation. The results for the position are in figure 8.30, for the velocity in figure 8.31 and for the attitude in figure 8.32. The effects of the motor malfunctioning are clear: an apogee altitude of 541 m is reached and in the simulations it was found that reduced motor performance significantly reduced the apogee altitude. Apogee is reached 16.2 seconds after ignition. The time of lift-off is 3.4 seconds, so the time to apogee is 12.8 seconds. The maximum velocity is 101 m/s. The estimated direction of flight is northwards, and slightly to the east. As for the elevation, it can be found that the rocket makes the expected gravity turn, having turned about 90° at the estimated apogee. From the azimuth it can be concluded in the ascent the rocket pointed in a straight direction.

In the descent, after 24 seconds, a sudden change of direction occurs. This coincides with the maximum descent velocity. Also, the azimuth changes rather abruptly by more than 100° . It is physically unlikely that this movement really occurred in flight. For it to have occurred, the vehicle should have generated aerodynamic lift, since other than aerodynamic forces and gravity no other forces were acting upon it. As the nose cone was fully separated, this lift generation could not have been present. The only remaining causes for this could be measurement or estimation errors. It was shown previously that angular dynamics have severe consequences for the estimation result and as the attitude plots show that the anomaly occurs in a highly dynamic regime of the flight, it is most likely that the cause is an estimation error due to strong angular dynamics.

8.5.1 Comparison with simulation

As can be expected, the estimated trajectory of the real flight does not correspond to the nominal simulated trajectory. However, the fact that many non-nominal trajectories were also simulated, allows finding a trajectory that is closer to the flight, so that statements about the performance of the rocket can be made.

In figure 8.33, the estimated and nominal ground track are shown. It is clear that the rocket traveled much further into the field, and slightly more eastward than in simulation. This can be attributed to slight differences in tower settings, or in an (intermediate) fluctuation in wind profile. This shows that for an unguided rocket, a nominal simulation is only of very limited use in determining the ground track in advance.

To analyze the aerodynamic and motor performance, in figure 8.34, the acceleration in the

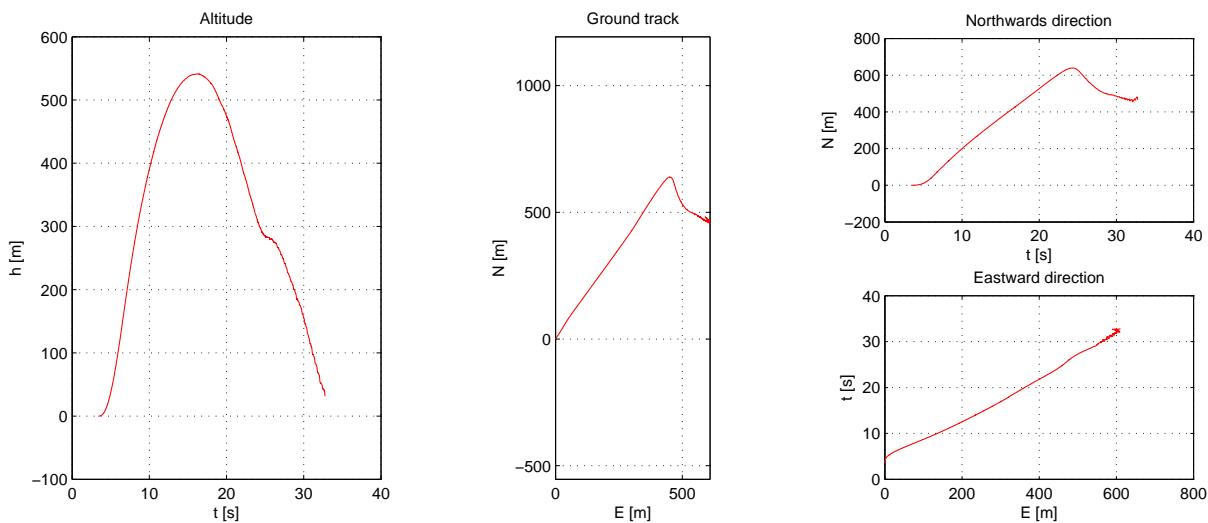


Figure 8.30: Estimated position from flight measurements

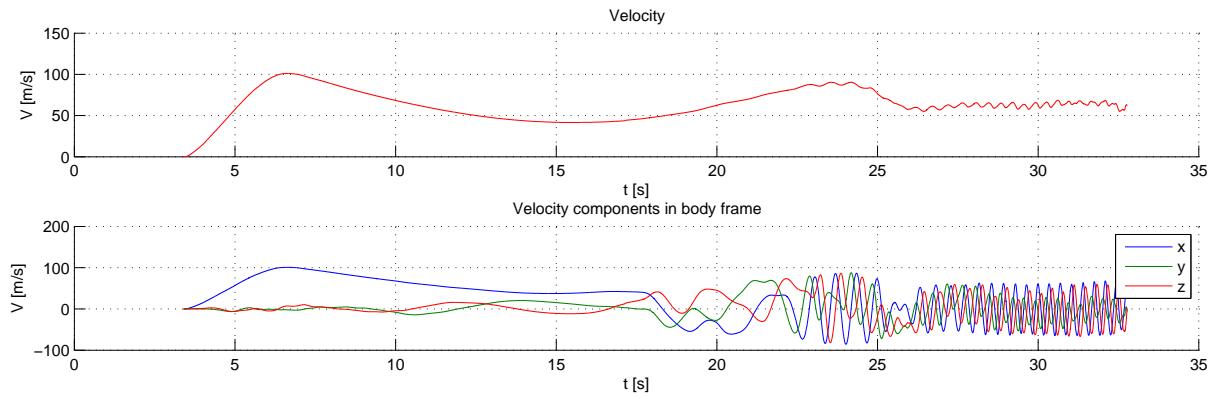


Figure 8.31: Estimated velocity from flight measurements

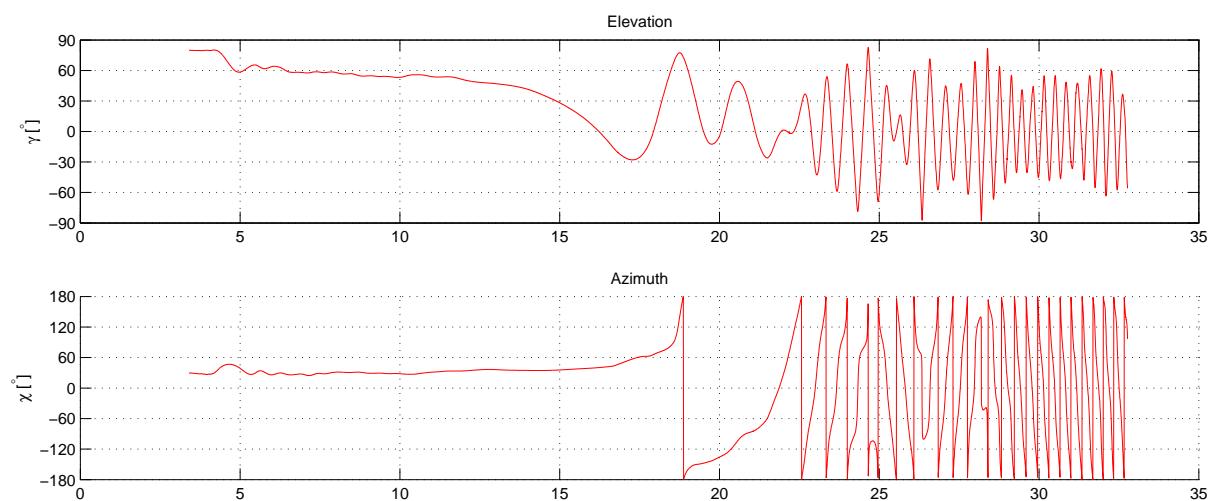


Figure 8.32: Estimated attitude from flight measurements

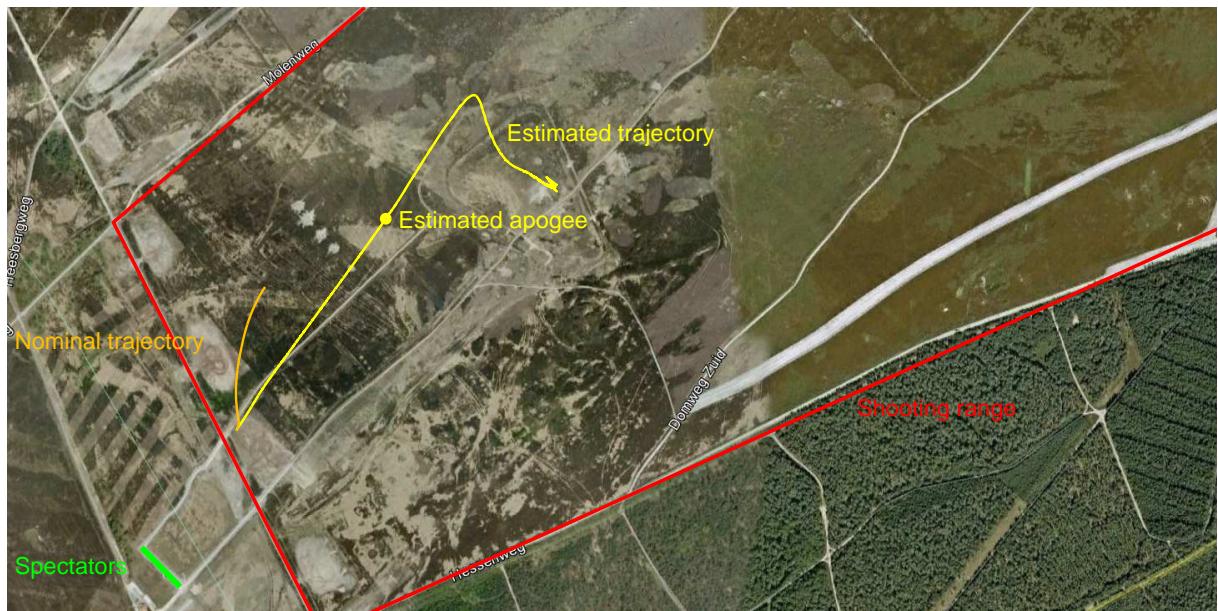


Figure 8.33: Estimated and nominal ground track of the flight at 't Harde, the Netherlands. Background by Google Earth

body frame has been plotted. Then, the estimated velocity and altitude have been used as input to ROSIE to obtain the simulated axial force coefficient. This axial force coefficient is only valid for the unseparated rocket. The aerodynamic acceleration in x-direction has then been plotted, followed by the residual acceleration, the acceleration minus the simulated aerodynamic acceleration and gravitational acceleration. In the non-powered phase, the residual acceleration should be zero; any non-zero value could be attributed to mismodeling or estimation errors. The errors are indeed very small until nose separation, which is a good indication that no large mismodeling and estimation errors are present. During motor burn, thrust is the dominant force, and it can be assumed that the residual acceleration is almost fully equal to the thrust acceleration. This is shown in the figure on the bottom left, along with the expected thrust. It can be seen that it is only possible to estimate the thrust, once the thrust has reached a level at least equal to the weight of the rocket. As expected, the actual thrust is much less than the simulated thrust. Inspection reveals that the thrust is between 0.6 and 0.7 times the nominal value. The estimated apogee altitude of 545 m is a reasonable value, considering that in simulation a thrust coefficient of 0.6 corresponds to an apogee altitude of 482 m, and a coefficient 0.8 to 816 m. Linear interpolation yields a thrust coefficient of 0.64 to reach that altitude. This shows that it is important that the simulated thrust corresponds to the actual thrust in order to obtain a good simulation result of the apogee altitude.

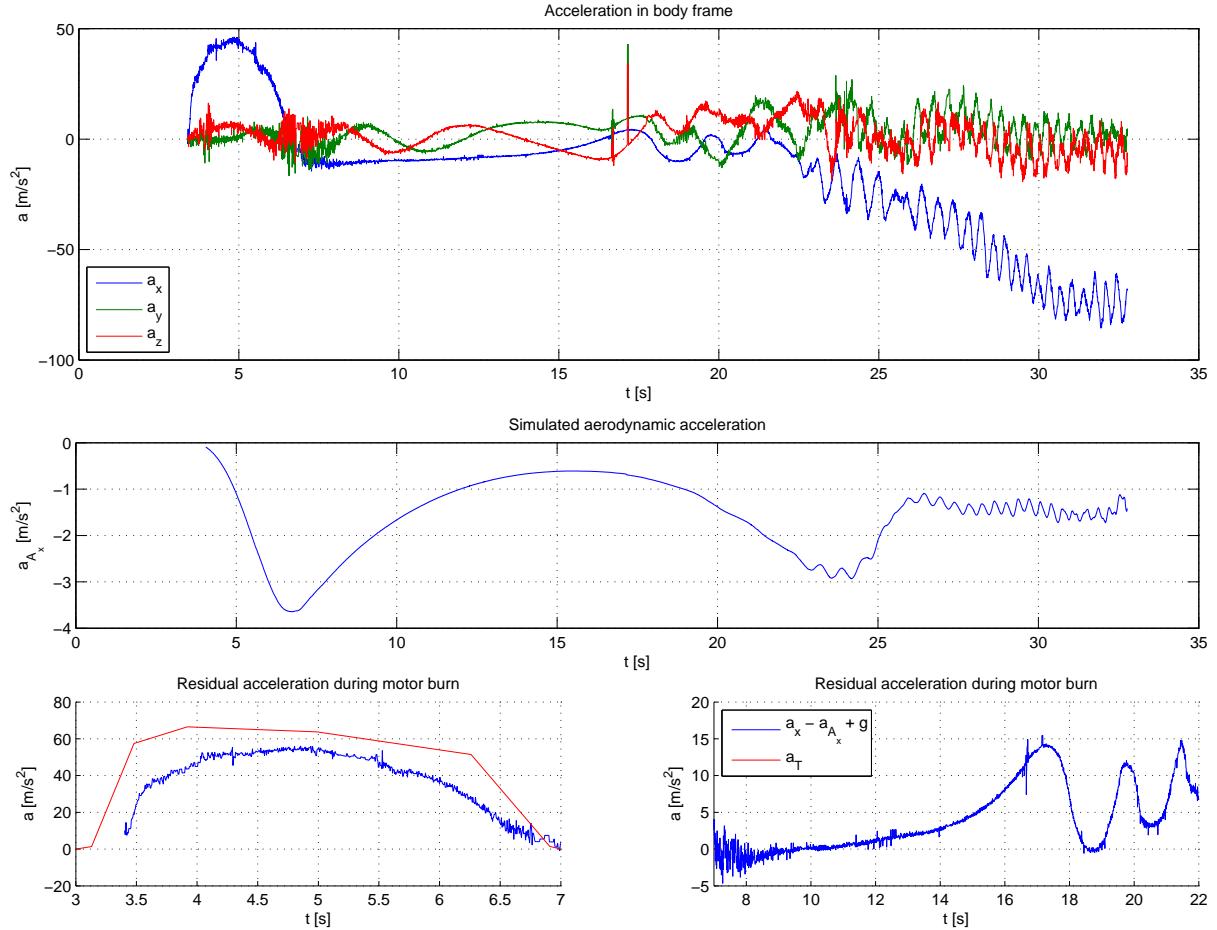


Figure 8.34: Estimated accelerations

8.5.2 Validity and accuracy of the estimation result

It is hard to make any definite statements about the validity of the estimation result. There was no other validated estimator device on board or tracking equipment of the rocket that

could provide benchmark data, nor is the impact point of the rocket known, as the persons responsible for retrieving the remains of the rocket, did not note the impact spot. There are several indications though that provide confidence that the estimation is a reasonable estimate of the ascent part of the real trajectory.

Even with the unrealistic descent behavior, the rocket stays within the shooting range, as expected. From the ground track it can be seen that from the viewpoint of the spectators, the rocket moved approximately in a straight line. This viewpoint is confirmed by video footage.

The performance of the estimator can be judged from the innovations. Figure 8.35 shows the pressure innovations. For a proper filter, the innovations are zero-mean, Gaussian distributed. This is not the case for the pressure measurements, which indicates a mismodeling. However, it was already recognized in the previous simulation sections that indeed the pressure model was imperfect. This mismodeling will in particular affect the estimated altitude. Still, as shown in the previous section, the estimated altitude reasonably corresponds to the expected altitude for the estimated thrust profile.

The angular motion according to the estimator corresponds to the expected angular motion: the rocket makes a gravity turn, reaching horizontal flight around apogee, and being aerodynamically stable, reaches a mostly constant azimuth in ascent, after some initial weathercocking.

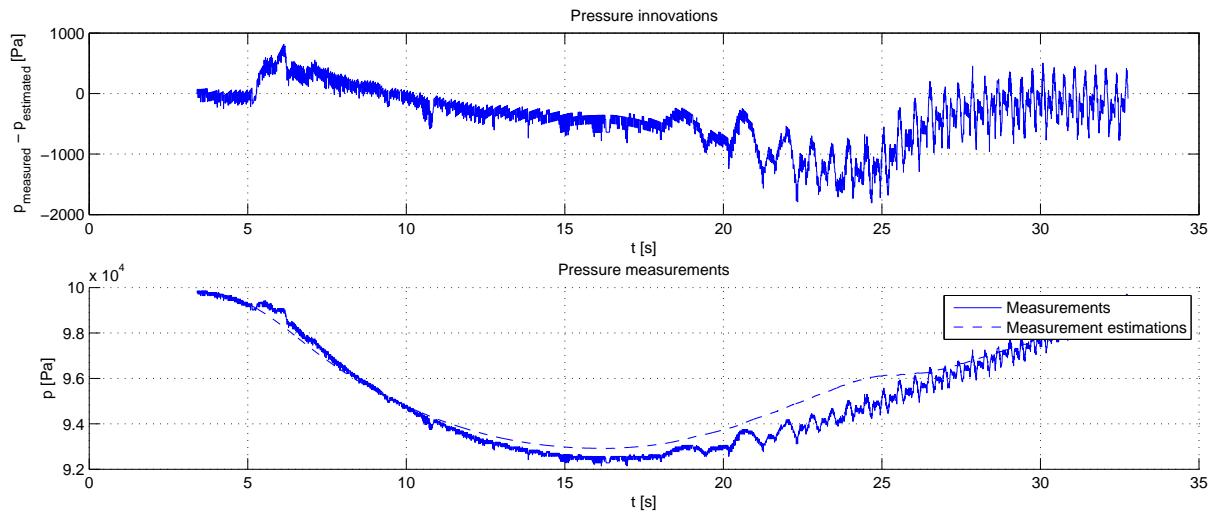


Figure 8.35: Pressure measurements estimations

Indications about the accuracy of the estimation can be obtained from the simulation results in chapter 8.3. The thrust magnitude is near the lowest simulated thrust, which has positive consequences for the estimation result: the lowest estimation errors in position and altitude corresponded to the flights with a low thrust magnitude.

Next to the flight profile, also the sensor noise factors were found to be of influence. Their estimated values have been plotted in figure 8.36. Clearly, the non-orthogonalities are found to be very small; about two orders of magnitude smaller than previously simulated. The estimator finds a scale factor error to be present. There strong divergence after apogee. This divergence is again most likely due to the strong dynamics after apogee. Further more, with the strong rotational motion, the offset of the accelerometer from the center of mass may start to play a role – an effect that earlier could be neglected.

With these results, it can be concluded that – in ascent – the estimation errors are likely of the same magnitude as the nominal estimation errors, or even smaller due to the lower thrust magnitude. This means position and apogee estimation errors around 70 m, and attitude errors around 1°. For the descent phase no numbers can be given for the estimation accuracy. This phase was not simulated, and it was already explained above that the estimator yields questionable results in this flight regime. Therefore, it must be concluded that the estimation

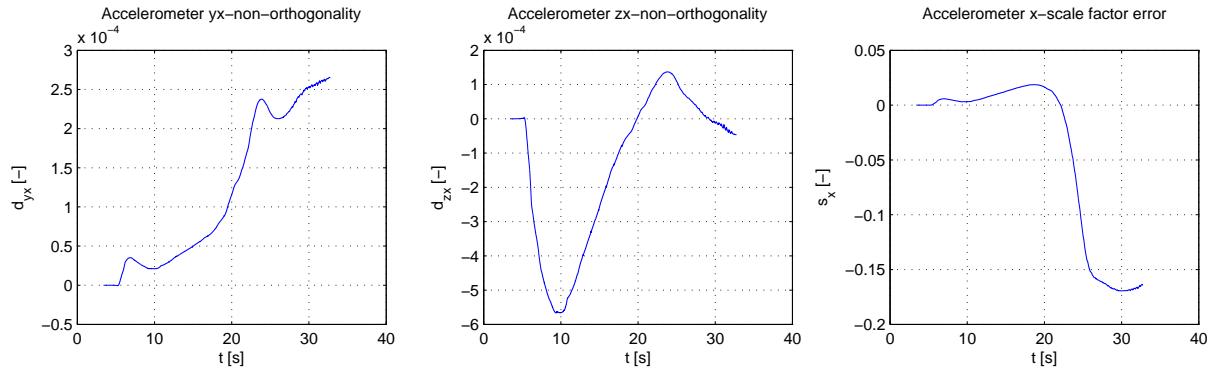


Figure 8.36: Flight sensor noise estimation

result is only accurate in ascent, and that the accuracy in descent cannot be stated until better simulations are performed.

To determine the effect of the filter itself, the estimation has also been performed with the RTS smoother and the UKF. However, it was found that changing the filter does not significantly affect the result. The only visible changes in the results are small deviations for the ground track post-apogee. This is shown in figure 8.37. The fact that only after apogee the estimation result differs, is again likely due the dynamics of the flight; only in parts of the flight where large dynamics occur, will higher order estimators start to make a difference.

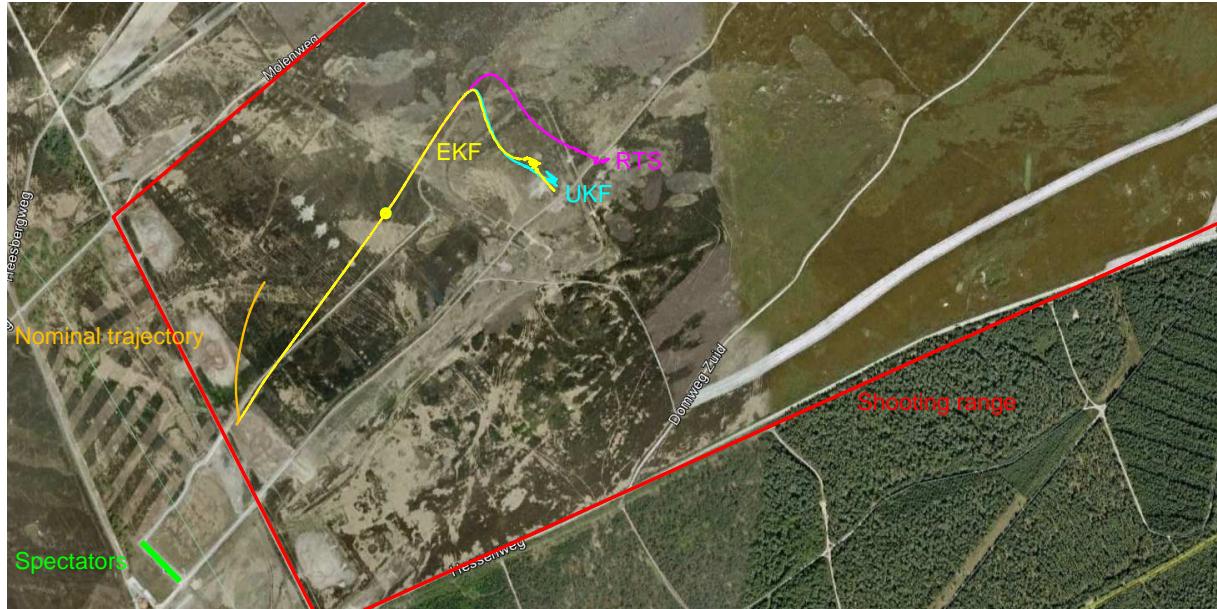


Figure 8.37: Estimated and nominal ground track of the flight at 't Harde, the Netherlands for different estimators. Background by Google Earth

Chapter 9

Conclusions and recommendations

With the move of sounding rocketry equipment from the traditional cold war surpluses to the development of new types of sounding rockets, new opportunities arise in the development of subsystems of these rockets with the latest technology. The position and attitude determination system of the Stratos II is one of the subsystems of such a newly developed rocket. The development of this system has been the main topic of this thesis, whose main goal has been to answer the following research question:

How can the position and attitude of the Stratos II sounding rocket be determined on-board and in real-time?

This thesis has sought to answer this research question. The conclusions are presented in section 9.1, while recommendations on future work that may improve the results are found in section 9.2.

9.1 Conclusions

To answer the research question, this thesis has sought to divide the topic into multiple subtopics, each which is a prerequisite to answering this main question, those being estimation, measurement and simulation. To verify and evaluate the performance of the estimator, and to design the measurement system, simulations of the flight are required. Development of a simulator was outside the scope of this thesis; rather, the simulator developed in Engelen, 2012, ROSIE, was used to generate flight trajectories and measurements. Next to this, also a simplified simulator was developed, with simplifications with respect to ROSIE and real-life, in order to be able to analytically express the state transition. To feed the estimators with information about the flight, a measurement system has been developed, which measured the following variables: acceleration, angular velocity, magnetic field, GPS, and to a certain extend atmospheric pressure. This system has also been tested in practice, by flying it on the *Stratos II concept launcher*. Estimation of position and attitude is the eventual goal of the research. This estimation is based on measurements obtained during flight, and the performance of the estimator has been evaluated using simulated trajectories. In this thesis, several forms of the Kalman filter have been used to estimate the position and attitude, and their performance under various sensor, vehicle and environmental characteristics have been evaluated.

Design and calibration of the measurement system showed that the designed measurement system allows measuring the specific force in the range of $-/+ 16 \text{ g}$, with an RMS noise of 0.15 m/s^2 , the angular velocity in the range of $-/+ 2000 \text{ }^\circ/\text{s}$ with an RMS noise of $0.3 \text{ }^\circ/\text{s}$ and atmospheric pressure in the range of 0 Pa to sea-level pressure, with an RMS noise of 0.7 hPa . Using this measurement system during the flight of the Stratos II concept launcher, showed that these sensors are able to measure the flight conditions of this rocket. However, the magnetometer and GPS that were also present, did not function as expected, with the

magnetometer being totally unusable, and the GPS only being usable on a limited part of the flight. From the simulations of the Stratos II, it was found that the designed measurement system requires some modification to be useful: the Stratos II will accelerate with a maximum value of almost 200 m/s^2 , or 20 g, which is outside the range of the accelerometer. In addition, the pressure sensor and GPS will only be usable on part of the flight.

From the estimator models studied in this thesis, it was found that the kinematic model, which treats the accelerometer and gyroscope measurements as dynamics rather than measurements, is superior to a model where the flight model of the rocket is incorporated, and the accelerometer and gyroscope are considered measurements in the sense of a Kalman filter. The use of this model in conjunction with an extended Kalman filter (EKF) is preferred over an unscented Kalman filter (UKF), as the UKF severely slows down the computation, while yielding only limited improvement in the estimation result. In the case of an ideal rocket flight, with sensor specifications as those of the designed measurement system, the maximum position estimation error is 473 m, while the maximum attitude error is 6° .

Sensor bias severely impacts the estimation result, but this impact can be reduced by estimating the bias in-flight. Estimating the accelerometer bias is only possible in the presence of GPS measurements. The difference between loose coupling of the GPS measurements, where the position estimate of the GPS receiver is used as input to the EKF, and tight coupling, where the pseudoranges to the GPS satellites are input to the EKF, was studied, and it was found that tight coupling provides slightly better position estimation results than loose coupling, when five satellites are available at orthogonal directions in the sky. With less than five satellites available, the improvement of the estimation result strongly depends on the geometry of the satellites as seen from the receiver. In fact, when only a single satellite is available, there is a possibility that the estimation result will deteriorate, due to misestimation of the accelerometer bias. For the attitude estimation, no clear correlation between the availability of GPS and the estimation result was found: depending on the configuration of the GPS satellites, the attitude estimate may improve or deteriorate a little.

In general the estimation errors decrease with increasing frequency and decreasing noise. Conversely, decreasing the frequency and increasing the noise has as a consequence that the estimation error increases. This effect is stronger for the position estimate than the attitude estimate. An order of magnitude decrease of the estimation error may be obtained by increasing the measurement frequency a factor 4 and decreasing the RMS noise a factor 4. In addition, it was also found that the dominating sensor is the gyroscope, as varying the measurement frequency and noise of the gyroscope affects the estimation result much more than varying the measurement frequency and noise of the accelerometer or magnetometer.

Noise factors other than white noise also impact the estimation result. However, the effects can be mitigated by estimation of these noise factors. The best estimation results for noisy sensors are obtained when next to the state and the accelerometer and gyroscope biases, also the accelerometer x-scale factor error and yx- and zx-non-orthogonalities are estimated, and no further noise sources are estimated. Still, even with estimations of these noise factors, the estimation result in presence of these noise factors is worse than in absence of them, and it is preferable if they are removed by pre-flight calibration. In addition, the total absence of certain measurements, for example due to sensors failure was found to render the estimation result unusable. The accelerometer, gyroscope and magnetometer were all found to be required for a proper estimation result. Not all failures are equally catastrophic, but in general it cannot be expected that good estimation results are obtained without any of these sensors. In analyzing the absence of certain sensors, the coupling between the position and attitude estimation was found. It was clear that without a proper attitude estimate, no position estimate can be made. However, if a proper position estimate cannot be made, then trying to estimate this position will significantly affect the attitude estimation result. It might therefore be worthwhile to investigate decoupling the position and attitude estimation.

Finally, it was concluded that for the Stratos II that non-nominal flight can have severe consequences for the estimation result. In particular, more severe dynamic characteristics, such as angular velocity and velocity, have adverse effect on the estimation results. It was tentatively concluded that the increased estimation errors are not critical to the flight, but further research would be required to confirm this.

For the Stratos II concept launcher, it was found that important flight factors that affect the estimation error were the aerodynamic roll coefficient and the thrust magnitude. Also, to obtain a proper estimate of position, the yx - and zx -non-orthogonalities and x -scale factor of the accelerometer should be estimated as well. To improve the attitude estimation error the gyroscope scale factor errors should be estimated, but since these errors were already reasonable, and there was likely insufficient measurement data to actually estimate them, this was not performed, and the flight estimation was performed with only the estimation of the aforementioned accelerometer noise factors.

The estimated flight trajectory did not correspond well to the simulated nominal trajectory. It was found that the apogee altitude was much lower due to motor malfunctioning, and that the range of the rocket was much larger than simulated. The motor was estimated to deliver only 64 % of the nominal thrust. On the contrary, the aerodynamics in axial direction from the simulation did correspond well to the real rocket. No conclusions on either the validity of the simulations or the validity of estimation result in descent could be drawn, because due to parachute failure, the rocket descended in a tumbling motion, for which no simulation model was available.

Concluding this thesis, it can be stated that the developed estimator is able to provide good estimation of the flight with the characteristics of the sensors of the developed measurement system. With the Stratos II concept launcher flight, it was shown that estimation of flight is possible, with the note that the validity of the result is subject to some uncertainty, and it would be certainly be advisable to perform another test flight. Also, deviations of the nominal flight or sensor characteristics do severely impact the estimation result. On the positive side, it was found that the estimation result could be improved by improving the gyroscope performance. Further improvements to the result could be made by determining sensor noise factors pre-flight as accurately as possible. If this is done, it can be concluded that the developed estimator with the developed measurement system can be used for the Stratos II.

9.2 Recommendations

No matter how good the results are, there is still opportunity to improve the research, given more time. In this section, this the future work is discussed. Two types of future work can be differentiated: those regarding results, and those regarding to practical implementation. With more work on the results, more confidence on them can be obtained, or new insights may be obtained that can be used to improve the estimator for the Stratos II. Recommendations regarding results are:

- In this research, sensor control, sensor noise and flight noise factors have been considered separately. However, it may be possible that these factors interact with each other. In particular, it should be researched if better estimation results for non-nominal flight can be obtained with better sensors or higher measurement frequencies.
- This research was performed with only very coarse requirements regarding the navigation system. To be able to better judge the results, better requirements should be determined that the navigation system is subject to.
- Several assumptions were made with respect to the sensors, such that only zeroth- (bias) and first-order (scale factor effects and non-orthogonalities) effects were considered. It

could be possible that other effects, such as g-sensitivity and higher-order effects affect the result.

- Assumptions with respect to the position of the GPS satellites in the sky were made. More realistic results could be obtained by simulating the actual GPS constellation.

More work on the practical implementation is required to show that the theoretical results obtained during this thesis results can be realized in practice. Of course, this is already partially shown with the development and flying of the measurement system, but more work is required:

- Obviously, the most important future work consists of implementing the estimator in an actual on-board system. Performing real-time estimation on an actual rocket, along with a validated navigation solution, would greatly improve the validity of the obtained results.
- The designed measurement system contains sensors that were easily available and relatively cheap, due to the requirements imposed on the design. Putting more work into obtaining more expensive or less easily obtainable sensors could significantly improve the estimation result.
- A better computational requirements evaluation should be performed. In particular, the number of *floating point operations per second* (FLOPS) should be counted. This information must then be used to judge whether the currently selected microcontroller is sufficiently powerful, or a more powerful one should be selected.
- Since it was found that sensor failure is essentially catastrophic, thought should be given to equipping a single measurement system with multiple sensors of the same kind, and on how to handle sensor failure. The latter part, giving thought to how to handle sensor failure, is of particular importance, as a wrong way of thinking about this topic was the ultimate cause of failure of the Stratos I parachute deployment.

Bibliography

- Aggarwal, P., Syed, Z., Noureldin, A., and El-Sheimy, N. *MEMS-Based Integrated Navigation. GNSS Technology and Applications*. Artech House, 2010. ISBN 9781608070435.
- Ambrosius, B. and Wittenberg, H. *Lecture Notes Rocket Motion & Re-Entry (AE4870)*. Delft University of Technology, November 2009.
- Anderson, J. *Introduction to flight*. McGraw-Hill, fifth, international edition, 2005.
- Betts, K. and Rutherford, R. Navigation error analysis for the rocketplane XP. *Aerospace and Electronic Systems Magazine, IEEE*, 24(2):27 –33, feb. 2009. ISSN 0885-8985. doi: 10.1109/MAES.2009.4798989.
- Bosch SensorTech. BMA180 digital, triaxial acceleration sensor, December 2009. URL <http://www.sparkfun.com/datasheets/Sensors/Accelerometer/BST-BMA180-DS000-03.pdf>. Last accessed 12 September 2012.
- Chu, Q. Spacecraft attitude dynamics and control course notes (AE4305). Delft University of Technology, 2010.
- Crassidis, J. and Markley, F. Attitude estimation using Modified Rodrigues Parameters. In *NASA Conference Publication*, pages 71–86. NASA, 1996.
- Eiche, M., Hermsen, R., Blom, W., Olthof, H., Fraters, A., Tong Minh, B., and Wagenaar, S. Stratos II conceptual design document. Technical report, Delft Aerospace Rocket Engineering, 2012.
- Engelen, F. Quantitative risk analysis of nominal suborbital rocket trajectories. Master's thesis, Delft University of Technology, 2012.
- Eurocircuits N.V. Making a pcb - educational movies - step 18 - electrical test, 2012. URL <http://www.eurocircuits.com/index.php/making-a-pcb-eductional-movies/50-electrical-test>. Last accessed 18 July 2012.
- Farrell, J. *Aided navigation: GPS with high rate sensors*. McGraw-Hill professional engineering: Electronic engineering. McGraw-Hill, 2008. ISBN 9780071493291.
- Fornberg, B. Generation of finite difference formulas on arbitrarily spaced grids. *Mathematics of Computation*, 51(184):699–706, 1988.
- Grewal, M. and Andrews, A. *Kalman Filtering: Theory and Practice Using Matlab*. Wiley, 2008. ISBN 9780470173664. URL <http://books.google.nl/books?id=zDHncuewKsQC>.
- Haeussermann, W. Developments in the field of automatic guidance and control of rockets. *Journal of Guidance and Control, vol. 4, May-June 1981, p. 225-239.*, 4:225–239, June 1981.
- InvenSense. IMU-3000 product specification, May 2011. URL <http://www.invensense.com/mems/gyro/documents/PS-IMU-3000A.pdf>. Last accessed 12 September 2012.

- Jongkind, W. *Lecture notes Spacecraft Mechatronics*. Delft University of Technology, September 2003.
- Julier, S. J. and Uhlmann, J. K. A new extension of the Kalman filter to nonlinear systems. pages 182–193, 1997.
- Kalman, R. E. A new approach to linear filtering and prediction problem. *Journal of Basic Engineering Transactions*, 82(1):34–45, 1960.
- Langley, R. Dilution of precision. *GPS world*, 10(5):52–59, 1999.
- Maus, S., Macmillan, S., McLean, S., Hamilton, B., Nair, M., Thomson, A., and Rollins, C. The US/UK world magnetic model for 2010–2015. Technical report, NOAA National Geophysical Data Center and British Geological Survey Geomagnetism Team, 2010. URL <http://www.ngdc.noaa.gov/geomag/WMM/DoDWMM.shtml>. Last accessed 18 July 2012.
- McGee, L. A. and Schmidt, S. F. Discovery of the Kalman filter as a practical tool for aerospace and industry. Technical report, National Aeronautics and Space Administration, Ames Research Center, Moffett Field, California, 1985. URL http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19860003843_1986003843.pdf.
- Measurement Specialties. MS1451, June 2011. URL <http://www.meas-spec.com/downloads/MS1451.pdf>. Last accessed 12 September 2012.
- Montenbruck, O. GPS navigation systems for ballistic vehicles, December 2007. URL <http://www.weblab.dlr.de/rbtrt/GpsNav/Ballistic/Ballistic.html>. Last accessed 18 July 2012.
- Montenbruck, O. and Gill, E. *Satellite orbits: models, methods, and applications*. Springer, 2000. ISBN 9783540672807.
- Mooij, E. *The Motion of a Vehicle in Planetary Atmosphere*. Delft University Press, 1997.
- Mooij, E. *Aerospace-plane flight dynamics: analysis of guidance and control concepts*. PhD thesis, Delft University of Technology, 1998.
- Mulder, J., van Staveren, W., van der Vaart, J., and de Weerdt, E. *Flight Dynamics Lecture Notes (AE3302)*. Delft University of Technology, June 2007.
- NASA Goddard Space Flight Center. Sounding rocket program handbook (SRPH), July 2001.
- Olsder, G. and Woude, V. *Mathematical systems theory*. NUR 919. VSSD, 2005. ISBN 9789071301407.
- PNI Corporation. Micromag3 3-axis magnetic sensor module. URL <http://www.sparkfun.com/datasheets/Sensors/MicroMag3%20Data%20Sheet.pdf>.
- Schaub, H. and Junkins, J. Stereographic orientation parameters for attitude dynamics: A generalization of the Rodrigues parameters. *Journal of the Astronautical Sciences*, 44(1):1–19, 1996.
- Simon, D. *Optimal state estimation: Kalman, H [infinity] and nonlinear approaches*. Wiley-Interscience, 2006. ISBN 9780471708582.
- Siouris, G. *Missile guidance and control systems*. Springer, 2004. ISBN 9780387007267.
- S.P.K. Electronics Co., Ltd. GPS module Spec P/N:SPK-GPS-GS407, April 2010. URL http://www.spkecl.com/comm/upfile/p_110714_08505.pdf. Last accessed 18 July 2012.

- Teunissen, P., Simons, D., and Tiberius, C. *Probability and Observation Theory*. Delft University of Technology, October 2006.
- Török, J. *Analytical mechanics: with an introduction to dynamical systems*. A Wiley-Interscience publication. Wiley, 2000. ISBN 9780471332077.
- Uitendaal, M. The Stratos rocket – design, simulation and production of a record breaking rocket. Master's thesis, Delft University of Technology, 2009.
- Viertotak, M. Esrange safety manual, June 2010. URL [www.ssc.se/file/
esrange-safety-manual.pdf](http://www.ssc.se/file/esrange-safety-manual.pdf). Last accessed 18 July 2012.
- Wakker, K. *Astrodynamic-II Lecture Notes (AE874-II)*. Delft University of Technology, August 2007.
- Zarchan, P. and Musoff, H. *Fundamentals Of Kalman Filtering: A Practical Approach*. Number v. 208 in Progress in Astronautics and Aeronautics. American Institute of Aeronautics and Astronautics, 2005. ISBN 9781563476945.

Appendix A

Attitude derivations

A.1 Derivation of the MRP time derivative equation

The time derivative of the MRP can be derived from the time derivative from quaternions. Starting with equation (2.33), this gives:

$$\begin{aligned}\dot{\sigma} &= \frac{d}{dt} \left(\frac{\varrho}{1+q_4} \right) \\ &= \frac{\dot{\varrho}}{1+q_4} - \frac{\varrho q_4}{(1+q_4)^2}\end{aligned}\tag{A.1}$$

Substituting equation (2.31), this can be written as:

$$\dot{\sigma} = \frac{1}{2} \left(\frac{\varrho}{1+q_4} \times \omega + \frac{q_4}{1+q_4} \omega \right) + \frac{1}{2} \frac{\varrho}{1+q_4} \frac{\varrho^T}{1+q_4} \omega\tag{A.2}$$

With equation (2.34), it can be found that $\frac{q_4}{1+q_4} = \frac{1}{2} (1 - \sigma^2)$, resulting in:

$$\dot{\sigma} = \frac{1}{2} \sigma \times \omega + \frac{1}{4} (1 - \sigma^2) \omega + \frac{1}{2} \sigma \sigma^T \omega\tag{A.3}$$

Appendix B

Partial derivatives

In this thesis, extensive use is made of partial derivatives. In this appendix, the derivation of the partial derivatives is given, if they were not provided in the body of the thesis.

Before starting the derivation, first two notations are introduced that allow for a more shorthand notation of some partial derivatives. These are the notation of the unit vector, and the column-wise cross product.

$\hat{\mathbf{e}}_i$ is the i -th unit vector. This is a vector of length 3, and zeros on all entries except for the i -th entry, which is one. From this it follows that the unit vectors $\hat{\mathbf{i}}$, $\hat{\mathbf{j}}$ and $\hat{\mathbf{k}}$ are respectively equal to $\hat{\mathbf{e}}_1$, $\hat{\mathbf{e}}_2$ and $\hat{\mathbf{e}}_3$.

$\mathbf{a} \times \mathbf{B}$ is the column-wise cross product of \mathbf{a} with the columns of \mathbf{B} . If \mathbf{B} is given as $\mathbf{B} = [\mathbf{b}_1 \ \mathbf{b}_2 \ \mathbf{b}_3]$, then this cross product is given as:

$$\mathbf{a} \times \mathbf{B} = [\mathbf{a} \times \mathbf{b}_1 \ \mathbf{a} \times \mathbf{b}_2 \ \mathbf{a} \times \mathbf{b}_3] \quad (\text{B.1})$$

B.1 Rules of differentiation

Some differentiations of elementary linear algebra operations, such as vector norms and cross products, occur often in the to be differentiated equations. Therefore, in this section generic rules for them will be established.

B.1.1 Differentiation of vector norms

Given a vector \mathbf{a} of length n , with components a_1 to a_n , the vector norm and the vector squared norm are given respectively as:

$$\|\mathbf{a}\| = \sqrt{a_1^2 + \dots + a_n^2} \quad (\text{B.2})$$

$$\|\mathbf{a}\|^2 = a_1^2 + \dots + a_n^2 \quad (\text{B.3})$$

The partial derivative of the vector norm and squared norm with respect to the vector then become respectively:

$$\begin{aligned}\frac{\partial}{\partial \mathbf{a}} \|\mathbf{a}\| &= \left[\frac{\partial \|\mathbf{a}\|}{\partial a_1} \quad \dots \quad \frac{\partial \|\mathbf{a}\|}{\partial a_n} \right] \\ &= \frac{1}{2\sqrt{a_1^2 + \dots + a_n^2}} \begin{bmatrix} 2a_1 & \dots & 2a_n \end{bmatrix} \\ &= \frac{\mathbf{a}^T}{\|\mathbf{a}\|}\end{aligned}\tag{B.4}$$

$$\begin{aligned}\frac{\partial}{\partial \mathbf{a}} \|\mathbf{a}\|^2 &= \left[\frac{\partial \|\mathbf{a}\|^2}{\partial a_1} \quad \dots \quad \frac{\partial \|\mathbf{a}\|^2}{\partial a_n} \right] \\ &= \begin{bmatrix} 2a_1 & \dots & 2a_n \end{bmatrix} \\ &= 2\mathbf{a}^T\end{aligned}\tag{B.5}$$

B.1.2 Differentiation of frame transformations

The frame transformation of vector \mathbf{a} to \mathbf{b} , both of length 3, is given by multiplication with matrix \mathbf{C} :

$$\mathbf{b} = \mathbf{Ca}\tag{B.6}$$

Provided that \mathbf{C} is neither a function of \mathbf{a} or x , by writing out equation (B.6), two partial derivatives can be trivially found:

$$\frac{\partial \mathbf{b}}{\partial \mathbf{a}} = \mathbf{C}\tag{B.7}$$

$$\frac{\partial \mathbf{b}}{\partial x} = \mathbf{C} \frac{\partial \mathbf{a}}{\partial x}\tag{B.8}$$

From the latter equation it can also be derived that for a vector \mathbf{x} the following relation holds:

$$\frac{\partial \mathbf{b}}{\partial \mathbf{x}} = \mathbf{C} \frac{\partial \mathbf{a}}{\partial \mathbf{x}}\tag{B.9}$$

The equations become more complicated if \mathbf{C} is also a function of x and \mathbf{x} respectively. The the following product rule can be found for the scalar case:

$$\frac{\partial \mathbf{b}}{\partial x} = \mathbf{C} \frac{\partial \mathbf{a}}{\partial x} + \frac{\partial \mathbf{a}}{\partial x} \mathbf{C}\tag{B.10}$$

Then, if \mathbf{x} has length 3 with components x_1 , x_2 and x_3 , the partial derivative with respect to \mathbf{x} becomes:

$$\frac{\partial \mathbf{b}}{\partial \mathbf{x}} = \mathbf{C} \frac{\partial \mathbf{a}}{\partial \mathbf{x}} + \left[\frac{\partial \mathbf{C}}{\partial x_1} \mathbf{a} \quad \frac{\partial \mathbf{C}}{\partial x_2} \mathbf{a} \quad \frac{\partial \mathbf{C}}{\partial x_3} \mathbf{a} \right]\tag{B.11}$$

B.1.3 Differentiation of the cross product

The cross product $\mathbf{a} \times \mathbf{b}$ can also be written in vector-matrix-multiplication form:

$$\mathbf{a} \times \mathbf{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & a_1 \\ -a_2 & -a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}\tag{B.12}$$

The partial derivative with respect to \mathbf{b} follows from equation (B.7):

$$\frac{\partial}{\partial \mathbf{b}} (\mathbf{a} \times \mathbf{b}) = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & a_1 \\ -a_2 & -a_1 & 0 \end{bmatrix}\tag{B.13}$$

Alternatively, using the column-wise cross product from equation (B.1), the partial derivative can be written as:

$$\frac{\partial}{\partial \mathbf{b}} (\mathbf{a} \times \mathbf{b}) = \mathbf{a} \times \mathbf{I}_3 \quad (\text{B.14})$$

Finally, using the identity $\mathbf{a} \times \mathbf{b} = -\mathbf{b} \times \mathbf{a}$, the derivative with respect to \mathbf{a} can be found:

$$\frac{\partial}{\partial \mathbf{a}} (\mathbf{a} \times \mathbf{b}) = -\mathbf{b} \times \mathbf{I}_3 \quad (\text{B.15})$$

Differentiation of vector-vector-transpose

The vector-vector-transpose of vector \mathbf{a} is defined as $\mathbf{a}\mathbf{a}^T$. Writing out this equation and differentiating with respect to a_i gives when applying the product rule:

$$\frac{\partial}{\partial a_i} (\mathbf{a}\mathbf{a}^T) = \mathbf{a}\hat{\mathbf{e}}_i^T + \hat{\mathbf{e}}_i\mathbf{a}^T \quad (\text{B.16})$$

B.2 Auxiliary partial derivatives

In the derivation of the partial derivatives of the state derivative, several dependent variables occur. The partial derivatives of these dependent variables are worked out here.

B.2.1 Aerodynamic partial derivatives

The auxiliary aerodynamic variables are the density ρ , the Mach number M , and the aerodynamic angles α and β . They were given in section 2.7.4 respectively as:

$$\rho = \rho_0 e^{-\frac{h}{H}} \quad (\text{B.17})$$

$$\alpha = \tan 2 \frac{V_{A_z}}{V_{A_x}} \quad (\text{B.18})$$

$$\beta = \tan 2 \frac{V_{A_y}}{\sqrt{V_{A_z}^2 + V_{A_x}^2}} \quad (\text{B.19})$$

$$M = \frac{V}{a} \quad (\text{B.20})$$

Density

With the altitude given as $h = r - r_0$, with r the position and r_0 the local radius of the Earth, the partial derivative of density with respect to position can be found by applying equation (B.4):

$$\begin{aligned} \frac{\partial \rho}{\partial \mathbf{r}} &= -\frac{\rho_0}{H} e^{-h} H \frac{\partial h}{\partial \mathbf{r}} \\ &= -\frac{\rho_0}{H} e^{-h} H \frac{\mathbf{r}^T}{r} \\ &= -\rho \frac{\mathbf{r}^T}{rH} \end{aligned} \quad (\text{B.21})$$

Mach number

Assuming that the speed of sound a is relatively constant, the Mach number only depends on velocity. The partial derivative with respect to velocity follows straightforwardly from equation (B.4):

$$\frac{\partial M}{\partial \mathbf{V}} = \frac{\mathbf{V}^T}{Va} \quad (\text{B.22})$$

Aerodynamic angles

Applying the chain rule gives the partial derivatives of the aerodynamic angles with respect to state:

$$\begin{aligned}\frac{\partial \alpha}{\partial \mathbf{V}} &= \frac{1}{1 + \frac{V_{A_z}^2}{V_{A_x}^2}} \frac{\partial}{\partial \mathbf{V}} \left(\frac{V_{A_z}}{V_{A_x}} \right) \\ &= \frac{V_{A_x}^2}{V_{A_x}^2 + V_{A_z}^2} \left(\frac{\partial V_{A_z}}{\partial \mathbf{V}} V_{A_x} - V_{A_z} \frac{\partial V_{A_x}}{\partial \mathbf{V}} \right)\end{aligned}\quad (\text{B.23})$$

$$\begin{aligned}\frac{\partial \beta}{\partial \mathbf{V}} &= \frac{1}{1 + \frac{V_{A_z}^2}{V_{A_x}^2 + V_{A_z}^2}} \frac{\partial}{\partial \mathbf{V}} \left(\frac{V_{A_y}}{\sqrt{V_{A_z}^2 + V_{A_x}^2}} \right) \\ &= \frac{1}{1 + \frac{V_{A_z}^2}{V_{A_x}^2 + V_{A_z}^2}} \frac{1}{V_{A_z}^2 + V_{A_x}^2} \left(\frac{\partial V_{A_y}}{\partial \mathbf{V}} \sqrt{V_{A_z}^2 + V_{A_x}^2} - \frac{V_{A_y} \left(2V_{A_x} \frac{\partial V_{A_x}}{\partial \mathbf{V}} + 2V_{A_z} \frac{\partial V_{A_z}}{\partial \mathbf{V}} \right)}{2\sqrt{V_{A_z}^2 + V_{A_x}^2}} \right) \\ &= \frac{1}{V^2} \left(\sqrt{V_{A_z}^2 + V_{A_x}^2} \frac{\partial V_{A_y}}{\partial \mathbf{V}} - V_{A_y} \frac{V_{A_x} \frac{\partial V_{A_x}}{\partial \mathbf{V}} + V_{A_z} \frac{\partial V_{A_z}}{\partial \mathbf{V}}}{\sqrt{V_{A_z}^2 + V_{A_x}^2}} \right)\end{aligned}\quad (\text{B.24})$$

In section 2.7.4, it was stated that the aerodynamic velocity is simply the velocity in the body frame. Therefore, all partial derivatives with respect to variables other than the velocity or the attitude are zero. These two non-zero partial derivatives of the aerodynamic velocity become:

$$\frac{\partial \mathbf{V}_A}{\partial \mathbf{V}} = \mathbf{C}_{IB}^T \quad (\text{B.25})$$

$$\frac{\partial \mathbf{V}_A}{\partial \boldsymbol{\sigma}} = \begin{bmatrix} \frac{\partial \mathbf{C}_{IB}^T}{\partial \sigma_1} \mathbf{V} & \frac{\partial \mathbf{C}_{IB}^T}{\partial \sigma_2} \mathbf{V} & \frac{\partial \mathbf{C}_{IB}^T}{\partial \sigma_3} \mathbf{V} \end{bmatrix} \quad (\text{B.26})$$

B.2.2 Attitude partial derivatives

For the derivation of the attitude partial derivatives, first the following notations are introduced:

Direction cosine matrix

The direction cosine matrix from equation (2.36) can be rewritten with the column-wise cross product of equation (B.1) as:

$$\mathbf{C} = Q^{-1} \mathbf{R} \quad (\text{B.27})$$

$$\sigma^2 = \|\boldsymbol{\sigma}\|^2 \quad (\text{B.28})$$

$$\mathbf{R} = 8\boldsymbol{\sigma}\boldsymbol{\sigma}^T + ((1 - \sigma^2)^2 - 4\sigma^2) \mathbf{I}_3 - 4(1 - \sigma^2)\boldsymbol{\sigma} \times \mathbf{I}_3 \quad (\text{B.29})$$

$$Q = (1 + \sigma^2)^2 \quad (\text{B.30})$$

The partial derivative with respect to σ_i is given by the quotient rule:

$$\begin{aligned}\frac{\partial \mathbf{C}}{\partial \sigma_i} &= \frac{1}{Q^2} \left(\frac{\partial \mathbf{R}}{\partial \sigma_i} Q - \mathbf{R} \frac{\partial Q}{\partial \sigma_i} \right) \\ &= \frac{1}{Q} \left(\frac{\partial \mathbf{R}}{\partial \sigma_i} - \mathbf{C} \frac{\partial Q}{\partial \sigma_i} \right)\end{aligned}\quad (\text{B.31})$$

Working this out results in:

$$\frac{\partial \mathbf{R}}{\partial \sigma_i} = 8(\boldsymbol{\sigma} \hat{\mathbf{e}}_i^T + \hat{\mathbf{e}}_i \boldsymbol{\sigma}^T) + (-4\sigma_i(1 - \sigma^2) - 8\sigma_i) \mathbf{I}_3 - (-2\sigma_i \boldsymbol{\sigma} + (1 - \sigma^2) \hat{\mathbf{e}}_i) \times \mathbf{I}_3 \quad (\text{B.32})$$

$$\frac{\partial Q}{\partial \sigma_i} = 4\sigma_i(1 + \sigma^2) \quad (\text{B.33})$$

Shadow transform

The shadow transform was given by equation (2.35) as:

$$\boldsymbol{\sigma}^S = -\frac{\boldsymbol{\sigma}}{\|\boldsymbol{\sigma}\|^2} \quad (\text{B.34})$$

The partial derivative can be derived using the quotient rule:

$$\begin{aligned} \frac{\partial \boldsymbol{\sigma}^S}{\partial \boldsymbol{\sigma}} &= \frac{\partial}{\partial \boldsymbol{\sigma}} \left(-\frac{\boldsymbol{\sigma}}{\|\boldsymbol{\sigma}\|^2} \right) \\ &= -\frac{1}{\|\boldsymbol{\sigma}\|^4} \left(\frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\sigma}} \|\boldsymbol{\sigma}\|^2 - \boldsymbol{\sigma} \frac{\partial}{\partial \boldsymbol{\sigma}} \|\boldsymbol{\sigma}\|^2 \right) \\ &= -\frac{1}{\|\boldsymbol{\sigma}\|^4} (\|\boldsymbol{\sigma}\|^2 \mathbf{I}_3 - 2\boldsymbol{\sigma}\boldsymbol{\sigma}^T) \end{aligned} \quad (\text{B.35})$$

B.3 State derivative partial derivatives

Finally, in this section the partial derivatives of the simplified state derivative from equation (4.3) are worked out.

B.3.1 Translational state derivative partial derivatives

The two translational state derivatives are given as follows:

$$\dot{\mathbf{r}} = \mathbf{V} \quad (\text{B.36})$$

$$\ddot{\mathbf{r}} = \mathbf{a}_G + \mathbf{a}_T + \mathbf{a}_A \quad (\text{B.37})$$

The partial derivative of velocity is trivial:

$$\frac{\partial \dot{\mathbf{r}}}{\partial \mathbf{x}} = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{I}_3 & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix} \quad (\text{B.38})$$

The partial derivative of acceleration is:

$$\frac{\partial \ddot{\mathbf{r}}}{\partial \mathbf{x}} = \frac{\partial \mathbf{a}_G}{\partial \mathbf{x}} + \frac{\partial \mathbf{a}_T}{\partial \mathbf{x}} + \frac{\partial \mathbf{a}_A}{\partial \mathbf{x}} \quad (\text{B.39})$$

Partial derivatives of gravitational acceleration

The gravitational acceleration is given by equation (2.90). It depends only on the position, so all other partial derivatives are zero. The partial derivative can be found from the equations for the partial derivative of the norm of the position vector as:

$$\frac{\partial \mathbf{a}_G}{\partial \mathbf{r}} = -\frac{\mu_e}{r^3} \left(\mathbf{I}_3 - \frac{3}{r^2} \mathbf{r} \mathbf{r}^T \right) \quad (\text{B.40})$$

Partial derivatives of thrust

The thrust is constant in the body frame, irrespective of any of the state variables. Therefore, the thrust acceleration in the inertial frame only depends on the attitude, and all other partial derivatives are zero. The partial derivative thus becomes:

$$\frac{\partial \mathbf{a}_T}{\partial \sigma_i} = \frac{\eta_T T}{m} \frac{\partial \mathbf{C}_{IB}}{\partial \sigma_i} \hat{\mathbf{i}}_B \quad (\text{B.41})$$

$$\frac{\partial \mathbf{a}_T}{\partial \eta_T} = \frac{T}{m} \mathbf{C}_{IB} \hat{\mathbf{i}}_B = \frac{1}{\eta_T} \mathbf{a}_T \quad (\text{B.42})$$

Partial derivatives of aerodynamic force

The aerodynamic acceleration and force were given in section 2.7.4 as, respectively:

$$\mathbf{a}_A = \frac{1}{m} \mathbf{C}_{IB} \mathbf{F}_A \quad (\text{B.43})$$

$$\mathbf{F}_A = \begin{bmatrix} \eta_{C_X} C_X \\ C_{Y_\beta} \beta \\ C_{Z_\alpha} \alpha \end{bmatrix} \frac{1}{2} \rho V^2 S_{ref} \quad (\text{B.44})$$

The aerodynamic acceleration partial derivatives are given by:

$$\begin{aligned} \frac{\partial \mathbf{a}_A}{\partial \mathbf{x}} = & \frac{1}{m} \mathbf{C}_{IB} \left[\begin{array}{cccccc} \frac{\partial \mathbf{F}_A}{\partial \mathbf{r}} & \frac{\partial \mathbf{F}_A}{\partial \mathbf{V}} & \frac{\partial \mathbf{F}_A}{\partial \boldsymbol{\sigma}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \frac{\partial \mathbf{F}_A}{\partial \rho} \end{array} \right] + \\ & \frac{1}{m} \left[\mathbf{0} \quad \mathbf{0} \quad \left[\begin{array}{ccc} \frac{\partial \mathbf{C}_{IB}}{\partial \sigma_1} \mathbf{F}_A & \frac{\partial \mathbf{C}_{IB}}{\partial \sigma_2} \mathbf{F}_A & \frac{\partial \mathbf{C}_{IB}}{\partial \sigma_3} \mathbf{F}_A \end{array} \right] \mathbf{0} \quad \mathbf{0} \quad \mathbf{0} \quad \mathbf{0} \right] \end{aligned} \quad (\text{B.45})$$

which follows from the fact that the aerodynamic force only depends on the position, velocity, attitude and density, and that the DCM only depends on the MRP. The partial derivative of aerodynamic force with respect to position only depends on the partial derivative of density with respect to position:

$$\frac{\partial \mathbf{F}_A}{\partial \mathbf{r}} = \begin{bmatrix} \eta_{C_X} C_X \\ C_{Y_\beta} \beta \\ C_{Z_\alpha} \alpha \end{bmatrix} \frac{1}{2} \frac{\partial \rho}{\partial \mathbf{r}} V^2 S_{ref} \quad (\text{B.46})$$

The partial derivative of aerodynamic force with respect to velocity is more complex, since both the aerodynamic angles as the V^2 -term depend on velocity. Applying the chain rule, and the differentiation rule of vector squared norms given earlier in this chapter yields:

$$\frac{\partial \mathbf{F}_A}{\partial \mathbf{V}} = \begin{bmatrix} \eta_{C_X} C_X \\ C_{Y_\beta} \beta \\ C_{Z_\alpha} \alpha \end{bmatrix} \rho \mathbf{V}^T S_{ref} + \begin{bmatrix} 0 \\ C_{Y_\beta} \frac{\partial \beta}{\partial \mathbf{V}} \\ C_{Z_\alpha} \frac{\partial \alpha}{\partial \mathbf{V}} \end{bmatrix} \frac{1}{2} \rho V^2 S_{ref} \quad (\text{B.47})$$

Finally, the aerodynamic force dependence on MRP is only found in the aerodynamic angles, giving:

$$\frac{\partial \mathbf{F}_A}{\partial \boldsymbol{\sigma}} = \begin{bmatrix} 0 \\ C_{Y_\beta} \frac{\partial \beta}{\partial \boldsymbol{\sigma}} \\ C_{Z_\alpha} \frac{\partial \alpha}{\partial \boldsymbol{\sigma}} \end{bmatrix} \frac{1}{2} \rho V^2 S_{ref} \quad (\text{B.48})$$

B.3.2 Rotational state derivative

Partial derivatives of MRP time derivative

With the column-wise cross product from equation (B.1), the MRP time derivative from equation (2.40) can be written as:

$$\dot{\boldsymbol{\sigma}} = \frac{1}{2} \boldsymbol{\sigma} \boldsymbol{\sigma}^T \boldsymbol{\omega} + \frac{1}{4} (1 - \sigma^2) \boldsymbol{\omega} - \frac{1}{2} \boldsymbol{\sigma} \times \boldsymbol{\omega} \quad (\text{B.49})$$

Using the equations derived in section B.1, this results in:

$$\frac{\partial \dot{\boldsymbol{\sigma}}}{\partial \boldsymbol{\sigma}} = \frac{1}{2} \boldsymbol{\sigma} \boldsymbol{\omega}^T + \frac{1}{2} \boldsymbol{\sigma}^T \boldsymbol{\omega} \mathbf{I}_3 - \frac{1}{2} \boldsymbol{\omega} \boldsymbol{\sigma}^T + \frac{1}{2} \boldsymbol{\omega} \times \mathbf{I}_3 \quad (\text{B.50})$$

$$\frac{\partial \dot{\boldsymbol{\sigma}}}{\partial \boldsymbol{\omega}} = \frac{1}{2} \boldsymbol{\sigma} \boldsymbol{\sigma}^T + \frac{1}{4} (1 - \sigma^2) \mathbf{I}_3 - \frac{1}{2} \boldsymbol{\sigma} \times \mathbf{I}_3 \quad (\text{B.51})$$

Partial derivative of angular acceleration

The angular acceleration was given by equation (2.89) as follows:

$$\begin{aligned}\dot{p} &= \frac{M_x + (I_{yy} - I_{zz})qr}{I_{xx}} \\ \dot{q} &= \frac{M_y + (I_{zz} - I_{xx})rp}{I_{yy}} \\ \dot{r} &= \frac{M_z + (I_{xx} - I_{yy})pq}{I_{zz}}\end{aligned}$$

where the sum of moments consists only of aerodynamic terms, given by equation (2.103) as follows:

$$\sum \mathbf{M} = r_A \hat{\mathbf{i}}_B \times \mathbf{F}_A + \begin{bmatrix} C_{l_0} + c_{l_p} \frac{pl_{ref}}{2V} \\ c_{m_\alpha} \alpha + c_{m_q} \frac{ql_{ref}}{2V} \\ c_{n_\beta} \beta + c_{n_r} \frac{rl_{ref}}{2V} \end{bmatrix} \frac{1}{2} \rho V^2 S_{ref} l_{ref}$$

In this equation, the moment part will be designated \mathbf{M}_A .

The partial derivatives of the force term follow from the partial derivatives of the aerodynamic force, since the aerodynamic arm is constant:

$$\frac{\partial}{\partial \mathbf{x}} (r_A \hat{\mathbf{i}}_B \times \mathbf{F}_A) = r_A \hat{\mathbf{i}}_B \times \frac{\partial \mathbf{F}_A}{\partial \mathbf{x}} \quad (\text{B.52})$$

Similar to the aerodynamic force, the only dependence of \mathbf{M}_A on position is given through the density:

$$\frac{\partial \mathbf{M}_A}{\partial \mathbf{r}} = \begin{bmatrix} C_{l_0} + c_{l_p} \frac{pl_{ref}}{2V} \\ c_{m_\alpha} \alpha + c_{m_q} \frac{ql_{ref}}{2V} \\ c_{n_\beta} \beta + c_{n_r} \frac{rl_{ref}}{2V} \end{bmatrix} \frac{1}{2} \frac{\partial \rho}{\partial \mathbf{r}} V^2 S_{ref} l_{ref} \quad (\text{B.53})$$

The partial derivative with respect to the velocity is slightly more complex, due to the many appearances of the velocity:

$$\frac{\partial \mathbf{M}_A}{\partial \mathbf{V}} = \begin{bmatrix} C_{l_0} + c_{l_p} \frac{pl_{ref}}{2V} \\ c_{m_\alpha} \alpha + c_{m_q} \frac{ql_{ref}}{2V} \\ c_{n_\beta} \beta + c_{n_r} \frac{rl_{ref}}{2V} \end{bmatrix} \rho \mathbf{V}^2 S_{ref} l_{ref} + \begin{bmatrix} -c_{l_p} \frac{pl_{ref}}{2V^2} \\ c_{m_\alpha} \frac{\partial \alpha}{\partial \mathbf{V}} - c_{m_q} \frac{ql_{ref}}{2V^2} \\ c_{n_\beta} \frac{\partial \beta}{\partial \mathbf{V}} - c_{n_r} \frac{rl_{ref}}{2V^2} \end{bmatrix} \frac{1}{2} \rho V^2 S_{ref} l_{ref} \quad (\text{B.54})$$

The only dependence on MRP is given by the dependence of the aerodynamic angles on them:

$$\frac{\partial \mathbf{M}_A}{\partial \boldsymbol{\sigma}} = \begin{bmatrix} 0 \\ c_{m_\alpha} \frac{\partial \alpha}{\partial \boldsymbol{\sigma}} \\ c_{n_\beta} \frac{\partial \beta}{\partial \boldsymbol{\sigma}} \end{bmatrix} \frac{1}{2} \rho V^2 S_{ref} l_{ref} \quad (\text{B.55})$$

Finally, the partial derivatives of the aerodynamic moment with respect to the angular velocity can be derived trivially as:

$$\frac{\partial \mathbf{M}_A}{\partial \boldsymbol{\omega}} = \begin{bmatrix} c_{l_p} \frac{l_{ref}}{2V} & 0 & 0 \\ 0 & c_{m_q} \frac{l_{ref}}{2V} & 0 \\ 0 & 0 & c_{n_r} \frac{l_{ref}}{2V} \end{bmatrix} \frac{1}{2} \rho V^2 S_{ref} l_{ref} \quad (\text{B.56})$$

As seen in equation (2.89), there are other components than the moment which cause a dependency on the angular acceleration. The partial derivatives of these coupling terms are given by:

$$\frac{\partial}{\partial \boldsymbol{\omega}} \begin{bmatrix} I_{xx}^{-1} (I_{yy} - I_{zz}) qr \\ I_{yy}^{-1} (I_{zz} - I_{xx}) rp \\ I_{zz}^{-1} (I_{xx} - I_{yy}) pq \end{bmatrix} = \begin{bmatrix} 0 & I_{xx}^{-1} (I_{yy} - I_{zz}) r & I_{xx}^{-1} (I_{yy} - I_{zz}) q \\ I_{yy}^{-1} (I_{zz} - I_{xx}) r & 0 & I_{yy}^{-1} (I_{zz} - I_{xx}) p \\ I_{zz}^{-1} (I_{xx} - I_{yy}) q & I_{zz}^{-1} (I_{xx} - I_{yy}) p & 0 \end{bmatrix} \quad (\text{B.57})$$

Appendix C

Estimator covariance properties

In this appendix, the initial covariance and the process noise matrices for the simplified estimator and the kinematic model estimator are presented.

C.1 Simplified estimator

In table C.1 the initial error covariance is presented. The initial error covariance is a diagonal matrix, each corresponding to a state variable. In the table, also the corresponding state variables are given. Also the process noise matrices are diagonal matrices, given in table C.2. As seen in the table, there are separate process noise matrices for each of the flight phases. For the flight phases where multiple process noise matrices are given, the value in the t_p row corresponds to the time in the phase that the corresponding process noise matrix will be used.

Table C.1: Initial covariance for the simplified estimator model

Corresponding variable	$\text{diag } \mathbf{P}_0$
x-coordinate	0.000001
y-coordinate	0.000001
z-coordinate	0.000001
x-velocity component	0.000001
y-velocity component	0.000001
z-velocity component	0.000001
MRP 1	0.000000001
MRP 2	0.000000001
MRP 3	0.000000001
x-angular velocity component	0.000001
y-angular velocity component	0.000001
z-angular velocity component	0.000001
thrust coefficient	0.1
rolling moment coefficient	0.001
axial force coefficient	0.000001
square root density	0.1
gyroscope x-bias	0.01
gyroscope y-bias	0.01
gyroscope z-bias	0.01
accelerometer x-bias	0.000001
accelerometer y-bias	0.000001
accelerometer z-bias	0.000001

Table C.2: Process noise matrix for simplified estimator model

C.2 Kinematic estimator model

Table C.3: Initial covariance for the kinematic estimator model

Corresponding variable	diag \mathbf{P}_0
x-coordinate	1.00E-09
y-coordinate	1.00E-09
z-coordinate	1.00E-09
x-velocity component	1.00E-12
y-velocity component	1.00E-12
z-velocity component	1.00E-12
MRP 1	1.00E-09
MRP 2	1.00E-09
MRP 3	1.00E-09
gyroscope x-bias	1.00E-04
gyroscope y-bias	1.00E-04
gyroscope z-bias	1.00E-04
accelerometer x-bias	1.00E-01
accelerometer y-bias	1.00E-01
accelerometer z-bias	1.00E-01
accelerometer x-scale factor error	1.00E-02
accelerometer zx-non-orthogonality	1.00E-02
accelerometer yx-non-orthogonality	1.00E-02

Process noise matrix for kinematic estimator model:

$$\text{diag } \mathbf{Q} = (t_k - t_{k-1}) \cdot \begin{bmatrix} 0 & 0 & 0 & 10^{-2} & 10^{-2} & 10^{-2} & 10^{-6}\|\boldsymbol{\omega}\|^2 & 10^{-6}\|\boldsymbol{\omega}\|^2 & 10^{-6}\|\boldsymbol{\omega}\|^2 \end{bmatrix}$$

Appendix D

Estimation with ROSIE

In principle, it is also possible to use the ROSIE simulator with an EKF, by using numerical derivatives to obtain the state transition matrix. Due to the simplicity of this, this has been the first estimator to be implemented. However, during the evaluation of it using the Stratos II concept launcher it was found that it was very computationally expensive already for the short concept launcher flight, and would require unacceptable computation times for a single Stratos II flight. Furthermore, it was found that using numerical derivative would often lead to instabilities in the filter. Therefore, it was soon decided to only use the simplified estimator and the model-free estimator. The results that were already obtained using ROSIE as estimator, are presented in this appendix.

Simulations were performed tailored to the flight of the concept launcher. It was observed from the raw measurement data that the two aspects in which the flight deviated most from the nominal simulations were the motor performance and the rolling motion. Therefore, simulations were performed with an underperforming motor, and with an increased rolling moment coefficient.

D.1 Motor underperformance

To determine the influence of motor performance, four simulations were performed, with the motor performing as respectively 100%, 90%, 80% and 70% of its nominal performance. Then, the generated measurements were inputted to the EKF. Figure D.1a shows the simulated and estimated altitude. The figure shows the dominating effect that the motor performance has on the obtained altitude. It also shows the excellent correspondence of estimation and simulation. This is due to the inclusion of the thrust in the state vector; it was found that without including this parameter in the state, no reasonable estimate of altitude could be obtained.

Figure D.1b shows the error in position and in altitude of the estimated result. The dashed lines are the 3-sigma lines according to the estimated error covariance matrix. It is clear that the estimated position does in fact diverge from the simulated position. Furthermore, after apogee it also significantly diverges from the 3-sigma estimate. From both observations it can be concluded that the implemented EKF is not an optimal filter, as for an optimal filter the estimation would not diverge from the simulation. Also, the error covariance would be an accurate representation of the error.

Despite the divergence, the implemented EKF still delivers an acceptable estimate. Especially from lift-off until apogee, the mission segment where the navigation system is the most critical, the estimation error stays within reasonable bounds: the position error is below 20 m, and the altitude estimate never diverges more than 1 m from the simulation. Figure D.2a shows the simulated and estimated ground track.

The attitude estimation error for the varying thrust levels is shown in figure D.2b, as the scalar part of the difference quaternion. It can be seen that the attitude error remains very

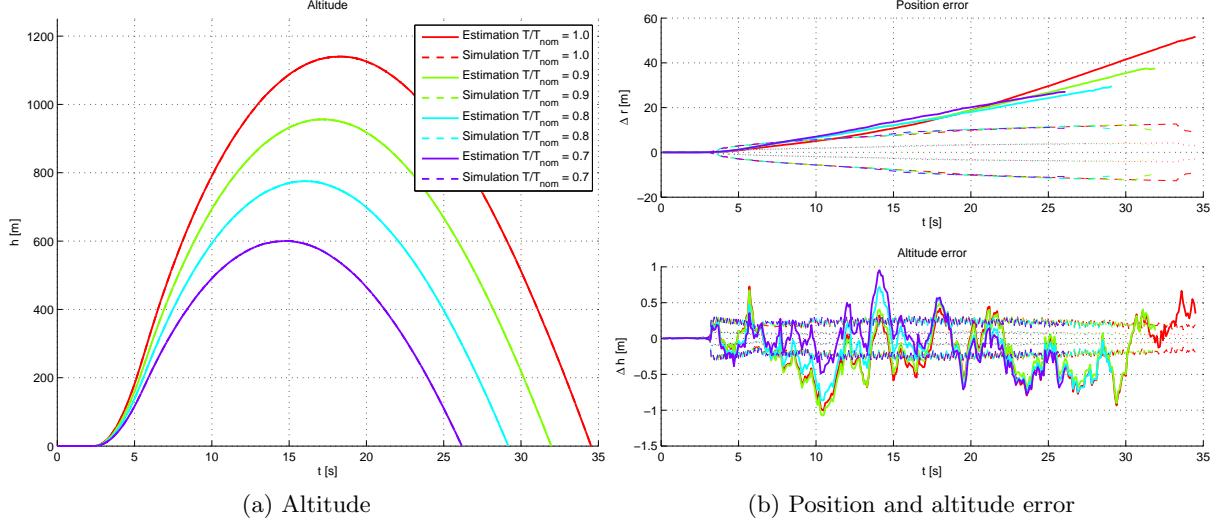


Figure D.1: Altitude and position error for varying thrust levels

small for the first three cases, and becomes slightly larger for the third case, but never diverges.

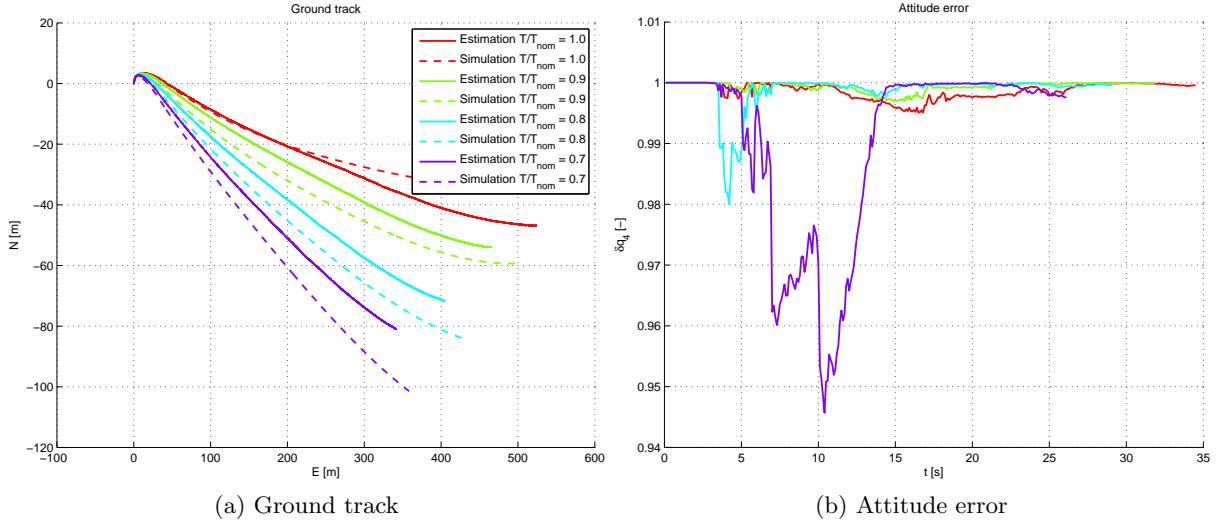


Figure D.2: Ground track and attitude error for varying thrust levels

D.2 Roll motion

Next to thrust variations, also roll motion was investigated. It was observed that the nominal simulation showed almost no roll motion, whereas the raw measurement data suggested its presence. This is most likely caused by the fact that roll motion is caused by imperfect mounting of the fins. To simulate this, extra static rolling moment coefficients of 0.01, 0.02 and 0.03 have been added to the aerodynamic model.

Figure D.3a shows the estimated ground track, and figure D.3b the position and altitude estimation error. The position error shows the same trend as for the thrust variations: the position error is slowly diverging, but nevertheless remains acceptable, and the altitude error is very small.

The inclusion of the roll coefficient in the state vector, allows the estimator to take this

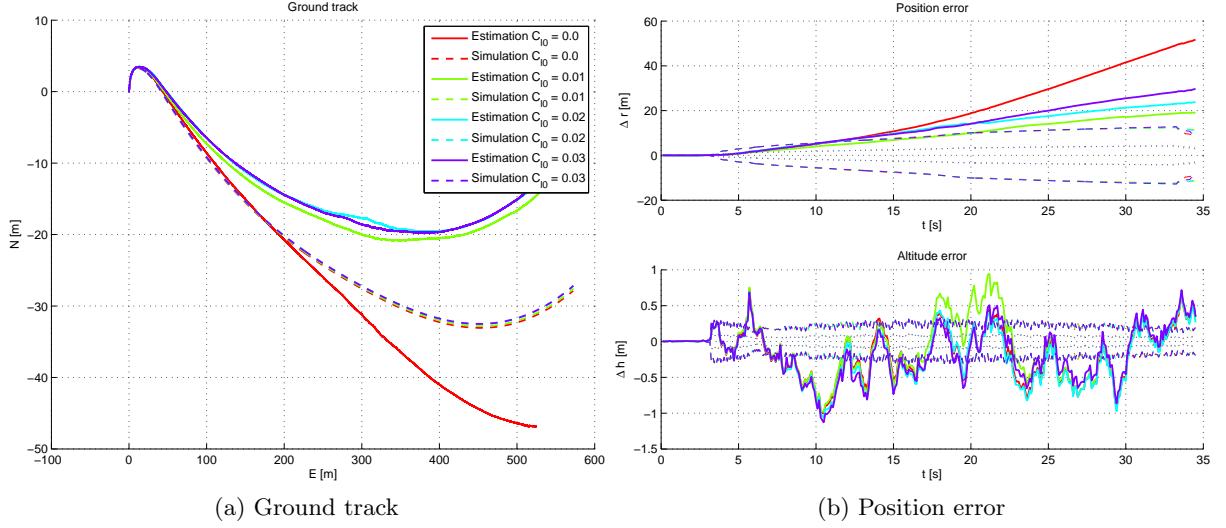


Figure D.3: Ground track and position error for various roll coefficients

deviation from the model into account. Figure D.4a shows the estimated coefficient, which always stays close to the real value. The attitude error is shown in figure D.4b. The results are not as good as those for the thrust variations case. Before apogee the attitude estimation error increases, but only strongly for a single case. Near apogee the attitude error is very small. After apogee the attitude error diverges strongly.

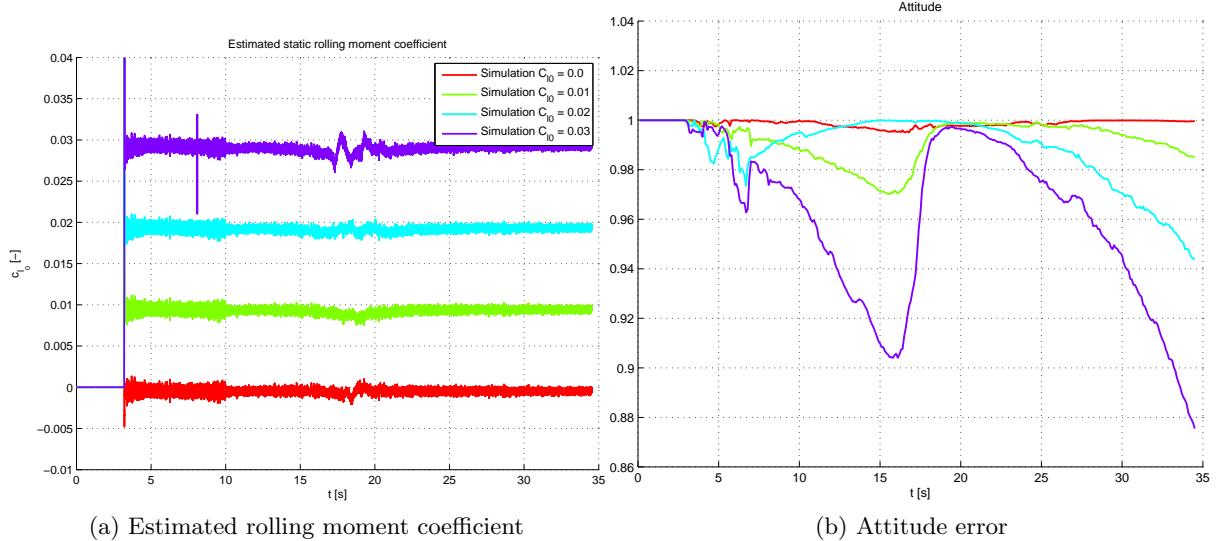


Figure D.4: Estimated rolling moment coefficient and attitude error for various roll coefficients

D.3 Performance of the estimator

Based on the performed simulations, the performance of the estimator can be evaluated. In all cases, the estimated altitude is excellent. The altitude estimation error never exceeds 1 m. The position estimate is reasonable. Up until apogee the estimate is good: the worst case error around apogee is 20 m. The quality of the attitude estimate is varying. Near apogee the estimate is excellent, however in the phase before the error can grow significantly for certain,

specific cases. Post-apogee the estimate may diverge greatly from its real value.

The contrast in estimation quality between the position estimate and the attitude estimate, can most likely be explained by the lack of an absolute measurement for the attitude. The position, and in particular the altitude can be indirectly measured by the pressure sensor. For the attitude however, only rate gyroscopes are available. Small biases and noise in those rate gyroscopes can result in the built-up of attitude estimation errors. It can be expected that with an absolute attitude sensor such as a magnetometer, the performance in attitude estimation can be greatly improved.

Another aspect in the performance of the estimator is the computational load associated with the estimator. As stated in the introduction, the current measurement system and estimator are a step in the development of a real-time position and attitude determination system for Stratos II. With the implemented EKF real-time estimation would not be possible: on a modern computer it takes at least five times the flight length to obtain an estimate. The main contribution to this is the numerical derivative calculation: approximately 70% of the estimation time is spent on the numerical derivative. An alternative to making the derivative calculations go faster would be to replace the EKF by another estimator, which does not require the numerical derivatives, such as the UKF.

It is clear that with the numerical derivative calculation real-time estimation is not possible. Still, even if all numerical derivatives were replaced by analytical derivatives, it would not be possible to perform real-time estimation on a modern computer, as the remaining run time is still at least 1.5 times the flight length. Onboard computers are much less powerful, so it would be absolutely impossible to perform onboard real-time estimation. The majority of the remaining time is spent on matrix operations. Due to the size of the state vector, there are many 20×20 matrix multiplications. Splitting the estimator in a separate attitude and position estimators, would make the matrix multiplications much less expensive, since the run time of a matrix multiplication of two $n \times n$ matrices is $O(n^3)$. Another option would be a lower measurement rate, since the matrix operations in the EKF have to be executed for every measurement. For the flight estimation, it means that the EKF equations are executed more than 1000 times per second.

Appendix E

Orthogonal arrays

E.1 3-level, 81-run orthogonal array

This 3-level, 81-run orthogonal array has been obtained from <http://www2.research.att.com/~njas/oadir/oa.81.40.3.2.txt>.

Run	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}	f_{21}	f_{22}	f_{23}	f_{24}	f_{25}	f_{26}	f_{27}	f_{28}	f_{29}	f_{30}	f_{31}	f_{32}	f_{33}	f_{34}	f_{35}	f_{36}
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
2	2	1	2	1	1	0	2	2	2	1	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	0	1	1	2	1	0	1	0	
3	1	2	1	2	2	0	1	1	1	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	2	0	0	
4	0	2	1	2	1	1	0	2	2	2	1	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	0	1	1	2	1	1	0	
5	2	0	0	0	2	1	2	1	1	0	2	2	2	1	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0
6	1	1	2	1	0	1	1	0	0	1	0	0	1	2	1	2	2	0	1	1	1	2	2	2	0	2	0	2	1	1	2	0	1	1	0	
7	0	1	2	1	2	2	0	1	1	1	2	2	2	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	2	0	0	
8	2	2	1	2	0	2	2	0	0	2	0	0	2	1	2	1	1	0	2	2	2	1	1	1	1	0	1	0	1	2	2	1	0	2	0	
9	1	0	0	0	1	2	1	2	2	0	1	1	1	2	2	2	2	0	2	0	2	1	1	2	0	1	0	2	2	1	2	0	0	2	0	
10	0	0	2	1	2	1	1	0	2	2	2	1	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1
11	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	2	0	0	2	0	0	2	1	2	1	1	0	2	2	2	1	1	1	
12	1	2	0	0	1	1	2	1	0	1	1	0	0	0	1	2	1	2	2	0	1	1	1	2	2	2	0	2	0	2	1	1	1			
13	0	2	0	0	0	2	1	2	1	1	0	2	2	2	1	1	1	0	1	0	1	2	2	1	0	2	0	0	1	1	2	1	1			
14	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	0	0	2	0	0	2	1	2	1	1	0	2	2	2	1		
15	1	1	1	2	2	2	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	2	0	0	0	2	2	1	2	1		
16	0	1	1	2	1	0	1	1	0	0	1	0	0	0	1	2	1	2	2	0	1	1	1	2	2	2	0	2	1	1	2	0	1			
17	2	2	0	0	2	0	0	0	2	1	2	1	1	0	2	2	2	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	
18	1	0	2	1	0	0	2	2	1	2	0	2	2	0	0	2	0	0	2	1	2	1	0	2	2	2	1	1	1	0	1	0	1	0	1	
19	0	0	1	2	1	2	2	0	1	1	2	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	2	0	2	2	1	2	0	2		
20	2	1	0	0	2	2	1	2	0	2	0	0	2	0	0	0	2	1	2	1	1	0	2	2	2	1	1	1	0	1	0	1	2	2		
21	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	1	0	0	1	2	1	2	2	0	1	1	1	2	2		
22	0	2	2	1	2	0	2	2	0	0	2	0	0	0	2	1	2	1	1	0	2	2	2	1	1	1	0	1	0	1	2	2	1	0	2	
23	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	0	1	2	1	2	2	0	1	1	1	2	2	2	0	2	0	2	0	2		
24	1	1	0	0	1	0	0	0	1	2	1	2	2	0	1	1	2	2	2	2	0	2	1	1	2	0	1	0	2	1	0	0	2			
25	0	1	0	0	0	1	2	1	2	0	1	1	1	2	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	1	2		
26	2	2	2	1	1	1	1	0	1	0	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	0	0	1	0	0	1	2			
27	1	0	1	2	2	1	0	2	0	1	1	2	1	0	1	1	0	0	1	0	0	1	2	1	2	2	0	1	1	1	2	1	2			
28	0	0	0	2	1	2	1	1	0	2	2	1	1	1	1	0	1	0	1	2	2	1	0	2	0	1	1	1	2	1	0	1	0	1		
29	2	1	2	0	2	2	0	0	2	0	0	2	1	2	1	0	2	2	2	1	1	1	0	1	0	1	2	2	1	0	2	0	1	0	1	
30	1	2	1	1	0	2	2	2	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	1	
31	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	0	2	0	0	2	1	2	1	1	0	2	2	2	1	1	1			
32	2	0	0	2	0	0	0	2	1	2	1	0	2	2	2	1	1	1	0	1	0	1	2	2	1	0	2	0	0	1	1	1	1	1		
33	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	0	0	2	0	0	2	1	2	1	0	2	2	2	1	1	1	1	1		

34	0	1	2	0	0	1	1	2	1	0	1	1	0	0	1	0	0	1	2	0	0	1	1	2	2	0	2	0	2	1				
35	2	2	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	1	2	1			
36	1	0	0	2	2	1	2	0	2	0	0	2	0	0	2	1	2	1	1	0	2	2	1	1	1	0	1	0	1	2	2	1		
37	0	0	2	0	0	0	2	1	2	1	1	0	2	2	1	1	1	0	1	0	1	2	2	1	0	2	0	1	1	2				
38	2	1	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	0	1	0	0	1	2	1	2		
39	1	2	0	2	2	0	0	2	0	0	2	1	2	1	1	0	2	2	2	1	1	1	0	1	0	1	2	2	0	1	2			
40	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	0	2	2	0	0	2	1	2	1	1	0	2	2	2	
41	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	0	2	2	0	0	0	2	1	2	1	1	0	2	2
42	1	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	0	1	2	1	2	2	
43	0	1	1	1	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	0	0	0	2			
44	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	0	2	2	0	0	0	2	1	2	1	1	0	2
45	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	0	1	2	1	2	2	0	1	1	1	2	2	2	0	2		
46	0	0	1	1	2	1	0	1	1	0	0	1	0	0	1	2	1	2	2	0	1	1	2	2	2	0	2	0	2	1	1	2	0	
47	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	1	0	0	1	2	1	2	2	0	1	1	1	2	2	0	
48	1	2	2	0	1	1	1	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	2	0		
49	0	2	2	0	0	2	0	0	2	1	2	1	1	0	2	2	2	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	
50	2	0	1	1	1	2	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	0	0	2	0	0	0	0		
51	1	1	0	2	2	2	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	0	1	0	0		
52	0	1	0	2	1	0	0	2	2	1	2	0	2	2	0	0	2	0	0	2	1	2	1	1	0	2	2	1	1	1	0	1	0	
53	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	0	2	0	0	0	2	1	2	1	1	0	
54	1	0	1	1	0	0	1	0	0	1	2	1	2	2	0	1	1	1	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	
55	0	0	0	1	2	1	2	2	0	1	1	2	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	
56	2	1	2	2	0	1	1	1	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	0	2		
57	1	2	1	0	1	1	0	0	1	0	0	1	2	1	2	2	0	1	1	1	2	2	2	0	2	1	1	2	0	1	0	2		
58	0	2	1	0	0	2	2	1	2	0	2	0	0	2	0	0	2	1	2	1	1	0	2	2	2	1	1	1	1	0	1	0	2	
59	2	0	0	1	1	2	1	0	1	0	0	1	0	0	1	2	1	2	2	0	1	1	2	2	2	2	0	2	0	2	1	1	2	
60	1	1	2	2	2	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	2	0	0	2	0	0	2	1	2	
61	0	1	2	2	1	0	2	0	1	2	1	0	1	2	1	0	0	1	0	0	0	1	2	1	2	2	0	1	1	1	2	2		
62	2	2	1	0	2	0	1	2	0	1	1	2	1	0	1	1	0	0	1	0	0	2	1	2	2	0	1	1	1	2	2	2		
63	1	0	0	1	0	0	1	2	1	2	2	0	1	1	2	2	2	2	0	2	1	1	2	0	1	0	2	1	0	2	2	2		
64	0	0	2	2	1	2	0	2	2	0	0	0	2	1	2	1	1	0	2	2	2	1	1	1	0	1	0	1	2	2	1	0		
65	2	1	1	0	2	2	2	1	1	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	1	0	
66	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	2	0	0	2	0	0	1	2	1	1	0	2	2	2	1	1	1	0	
67	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	1	0	0	1	2	1	2	2	0	1	2	2	2	0	2	0		

68	2	0	2	2	0	0	2	0	0	0	2	1	2	1	1	0	2	2	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0				
69	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	1	0	0	0	1	2	1	2	0			
70	0	1	1	0	0	1	0	0	0	1	2	1	2	2	0	1	1	1	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0			
71	2	2	0	1	1	1	2	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	2	0	0	2	0	0		
72	1	0	2	2	2	1	1	1	1	0	1	0	1	2	2	1	0	0	2	0	0	1	1	2	1	0	1	1	0	0	1	0	0	0	0	0		
73	0	0	1	0	0	0	1	2	1	2	2	0	1	1	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1				
74	2	1	0	1	1	0	0	1	0	0	1	2	1	2	2	0	1	1	1	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1				
75	1	2	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	2	0	0	2	0	0	2	1	2	1			
76	0	2	2	2	1	1	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	1	0	0	0	1		
77	2	0	1	0	2	1	0	0	2	2	1	2	0	2	2	0	0	2	0	0	2	1	2	1	1	0	2	2	2	1	1	1	1	0	1			
78	1	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	1	0	0	1	2	1	2	2	0	1			
79	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	1	0	0	1	2	1	2	2	0	1	1	1			
80	2	2	2	2	0	2	0	2	1	1	2	0	1	0	2	1	0	0	2	2	1	2	0	2	2	0	0	2	0	0	2	1	2	1	1			
81	1	0	1	0	1	2	2	1	0	2	0	1	2	0	0	1	1	2	1	0	1	1	0	0	1	0	0	0	1	2	1	2	2	0	1	1		

Appendix F

Results

In this chapter results are presented that belong to chapter 7, but were too long for inclusion there.

F.1 GPS tight coupling

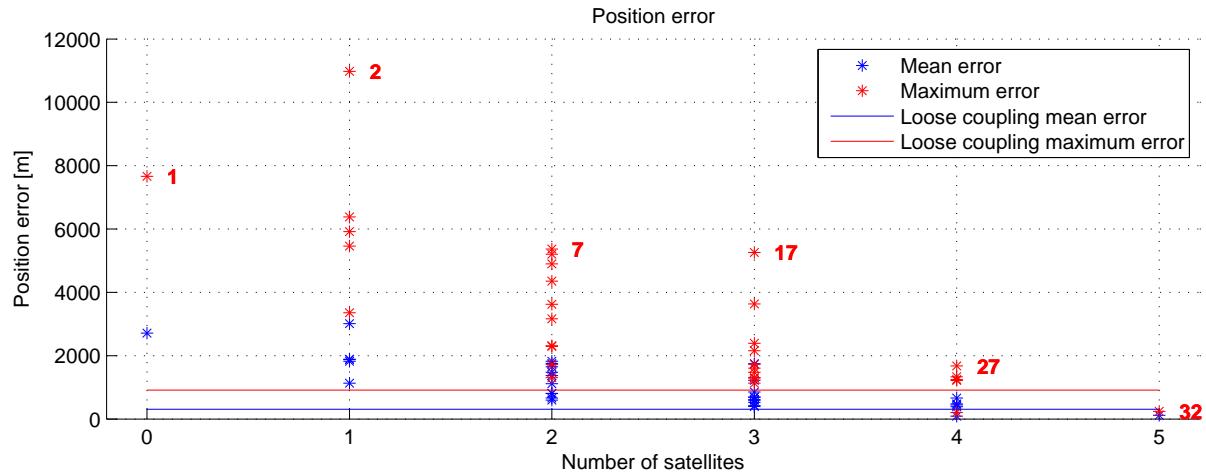


Figure F.1: Annotated position error for various levels of GPS coupling

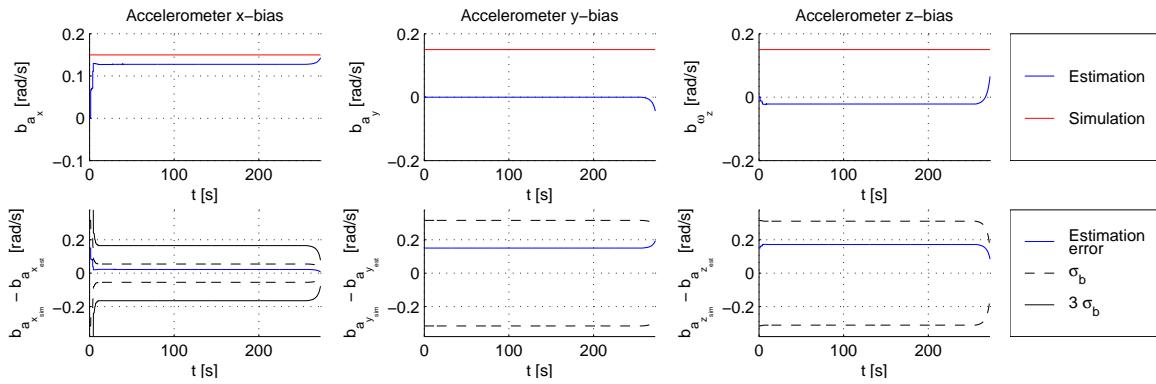


Figure F.2: Estimated bias with tight GPS coupling case 1 (0 satellites)

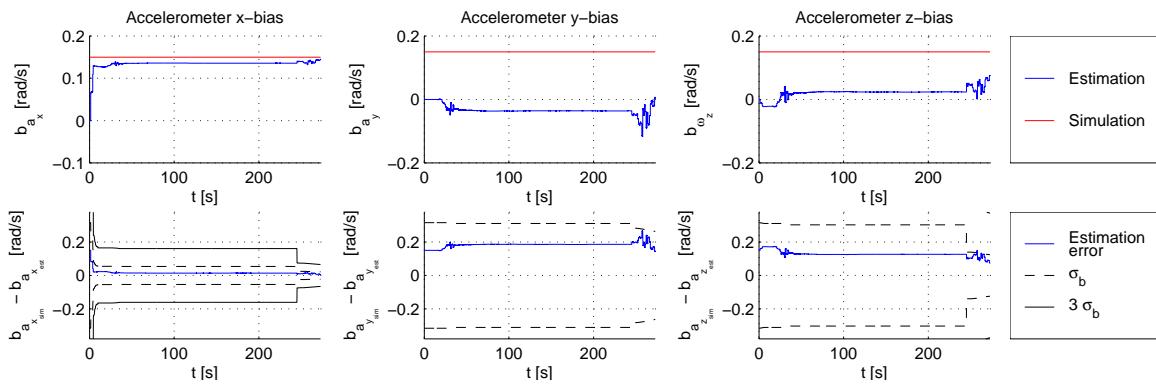


Figure F.3: Estimated bias with tight GPS coupling case 2 (1 satellite)

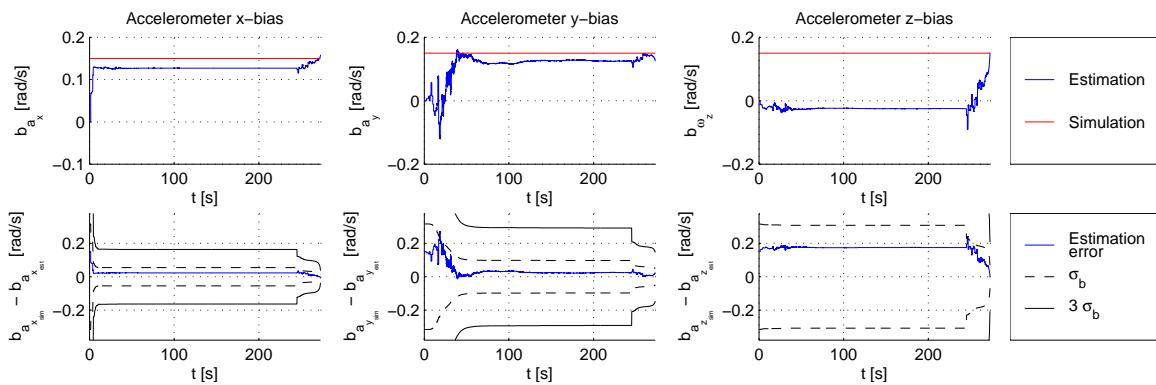


Figure F.4: Estimated bias with tight GPS coupling case 3 (1 satellite)

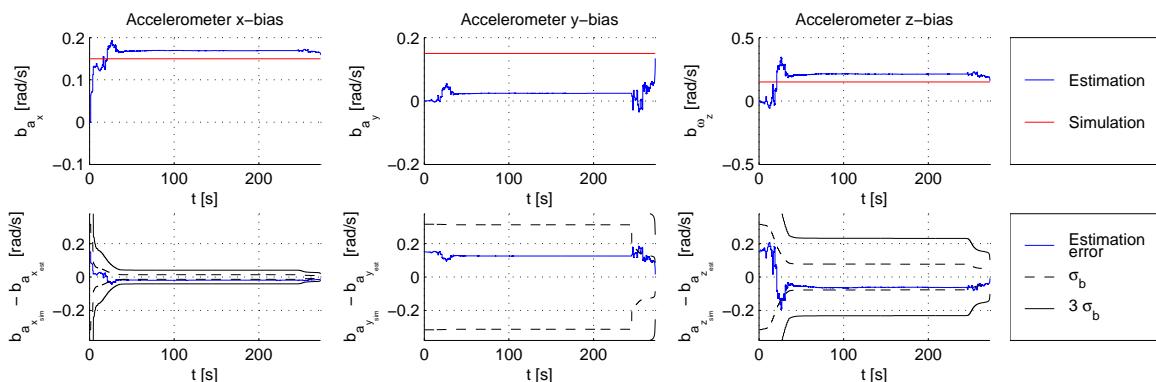


Figure F.5: Estimated bias with tight GPS coupling case 4 (1 satellite)

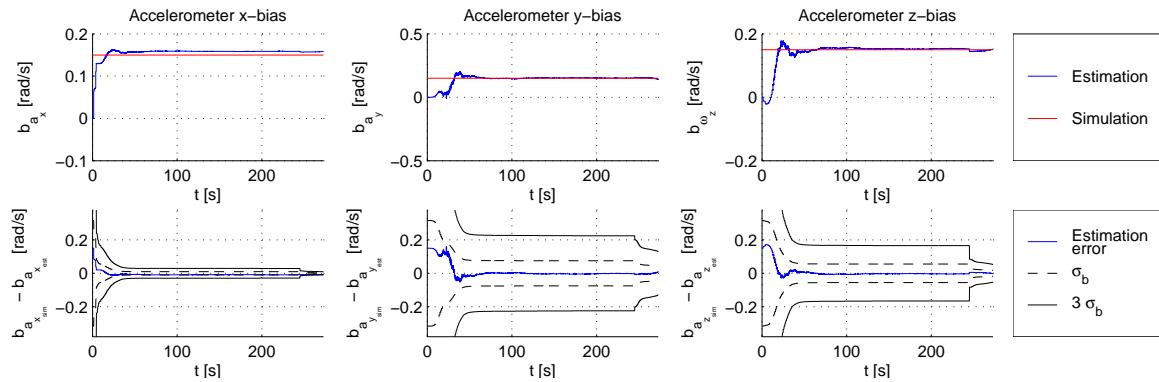


Figure F.6: Estimated bias with tight GPS coupling case 31 (4 satellite)

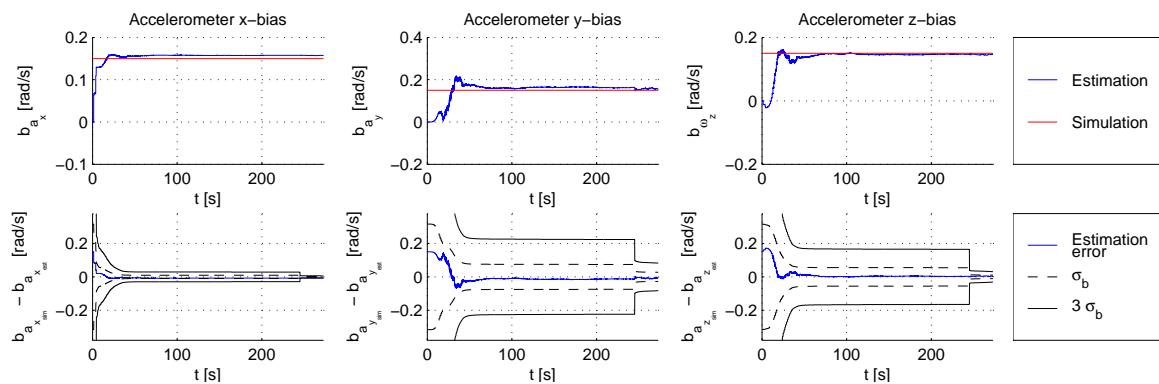


Figure F.7: Estimated bias with tight GPS coupling case 32 (5 satellite)

F.2 Sensor control factors

The following are the estimation errors for varying noise and frequency levels, followed by the response surface coefficients for these factors.

Table F.1: Position mean error for varying noise and frequency [m]

Noise factor	Frequency factor						
	0.25	0.50	0.75	1.00	1.50	2.00	4.00
0.25	639.77	320.66	162.11	169.65	134.64	64.33	38.51
0.50	2128.99	645.25	366.14	343.43	353.26	77.05	96.13
1.00	6332.50	1179.43	858.86	737.95	998.72	465.75	342.67
1.50	9806.03	1500.97	1403.12	1165.93	1872.09	1115.55	715.60
2.00	12657.11	1600.17	1875.92	1615.36	2872.46	1921.41	1170.52
3.00	16681.52	1501.46	2787.28	3214.50	5051.84	3652.14	2164.49
4.00	19435.15	1765.69	3692.15	5318.69	7499.74	5362.34	3125.93

Table F.2: Position max error for varying noise and frequency [m]

Noise factor	Frequency factor						
	0.25	0.50	0.75	1.00	1.50	2.00	4.00
0.25	1578.54	859.22	545.60	344.24	385.47	152.30	159.77
0.50	7755.17	1791.24	1406.85	757.11	1029.12	259.89	279.93
1.00	23199.81	3781.48	3976.18	1664.40	2806.76	1209.50	924.74
1.50	32261.47	5414.38	6431.02	2513.37	5158.16	2718.49	1856.67
2.00	39406.24	6283.32	7427.67	3571.01	7781.75	4636.66	2986.53
3.00	48931.04	6432.24	6725.26	6826.12	13294.12	8668.29	5399.14
4.00	55170.03	7162.09	9122.37	10919.98	19434.05	12582.47	7721.28

Table F.3: Altitude mean error for varying noise and frequency [m]

Noise factor	Frequency factor						
	0.25	0.50	0.75	1.00	1.50	2.00	4.00
0.25	127.36	29.21	7.89	32.22	27.05	26.01	5.54
0.50	372.02	63.50	28.27	9.48	58.17	29.28	0.14
1.00	1025.24	113.23	83.76	40.23	149.32	73.32	28.32
1.50	1567.32	122.92	129.46	97.25	271.74	155.16	76.35
2.00	2007.11	79.61	114.01	162.55	410.89	261.90	137.86
3.00	2597.07	84.95	52.22	402.08	710.71	498.43	277.79
4.00	2961.97	160.76	407.25	717.62	1044.15	737.81	416.72

Table F.4: Altitude max error for varying noise and frequency [m]

Noise factor	Frequency factor						
	0.25	0.50	0.75	1.00	1.50	2.00	4.00
0.25	391.87	75.04	33.16	62.69	75.95	59.50	25.66
0.50	1427.68	180.51	109.78	42.64	159.72	66.44	28.54
1.00	3854.99	403.86	449.75	143.00	397.73	177.15	89.76
1.50	5332.40	572.24	817.19	271.26	717.19	375.66	206.51
2.00	6473.18	620.45	984.94	392.19	1074.14	632.89	349.26
3.00	7896.20	693.54	929.82	917.54	1823.57	1183.91	693.81
4.00	8729.56	1060.30	1404.54	1569.42	2662.61	1724.54	1040.91

Table F.5: Attitude mean error for varying noise and frequency [degrees]

Noise factor	Frequency factor						
	0.25	0.50	0.75	1.00	1.50	2.00	4.00
0.25	0.64	0.41	0.41	0.57	0.40	0.33	0.32
0.50	1.61	0.73	0.64	0.88	0.58	0.33	0.32
1.00	4.50	1.27	1.23	1.54	1.16	0.47	0.49
1.50	7.06	1.65	1.88	2.22	1.92	0.96	0.79
2.00	9.24	1.87	2.46	2.93	2.78	1.61	1.15
3.00	12.39	2.60	3.61	4.90	4.65	3.04	1.91
4.00	14.48	3.46	4.89	7.33	6.71	4.44	2.63

Table F.6: Attitude max error for varying noise and frequency [degrees]

Noise factor	Frequency factor						
	0.25	0.50	0.75	1.00	1.50	2.00	4.00
0.25	4.86	4.95	5.01	5.48	4.64	4.61	5.05
0.50	5.06	5.12	5.17	6.07	4.58	4.33	5.21
1.00	11.51	5.72	5.71	7.25	4.68	3.79	5.54
1.50	15.89	6.50	6.45	8.43	5.11	3.28	5.88
2.00	19.12	7.33	7.29	9.59	5.55	3.62	6.22
3.00	22.78	9.03	10.33	12.12	7.01	5.43	6.90
4.00	24.53	10.67	13.96	15.25	10.31	6.79	7.55

Table F.7: Elevation mean error for varying noise and frequency [degrees]

Noise factor	Frequency factor						
	0.25	0.50	0.75	1.00	1.50	2.00	4.00
0.25	0.10	0.01	0.02	0.06	0.04	0.01	0.01
0.50	0.24	0.01	0.01	0.10	0.06	0.01	0.01
1.00	0.60	0.01	0.00	0.19	0.12	0.02	0.01
1.50	0.90	0.04	0.01	0.29	0.21	0.06	0.02
2.00	1.12	0.10	0.07	0.39	0.30	0.12	0.04
3.00	1.39	0.26	0.30	0.66	0.52	0.24	0.10
4.00	1.55	0.34	0.64	0.99	0.75	0.36	0.17

Table F.8: Elevation max error for varying noise and frequency [degrees]

Noise factor	Frequency factor						
	0.25	0.50	0.75	1.00	1.50	2.00	4.00
0.25	0.60	0.67	0.66	0.73	0.58	0.63	0.68
0.50	0.62	0.68	0.67	0.81	0.58	0.60	0.70
1.00	1.77	0.75	0.71	0.95	0.60	0.55	0.74
1.50	2.41	0.85	0.78	1.10	0.64	0.50	0.79
2.00	2.85	0.94	0.86	1.24	0.73	0.60	0.84
3.00	3.45	1.32	1.22	1.51	1.08	0.85	0.94
4.00	3.93	1.80	1.76	1.90	1.44	1.11	1.03

F.3 Sensor noise factors

Here, the response surface coefficients of sensor noise factors are presented.

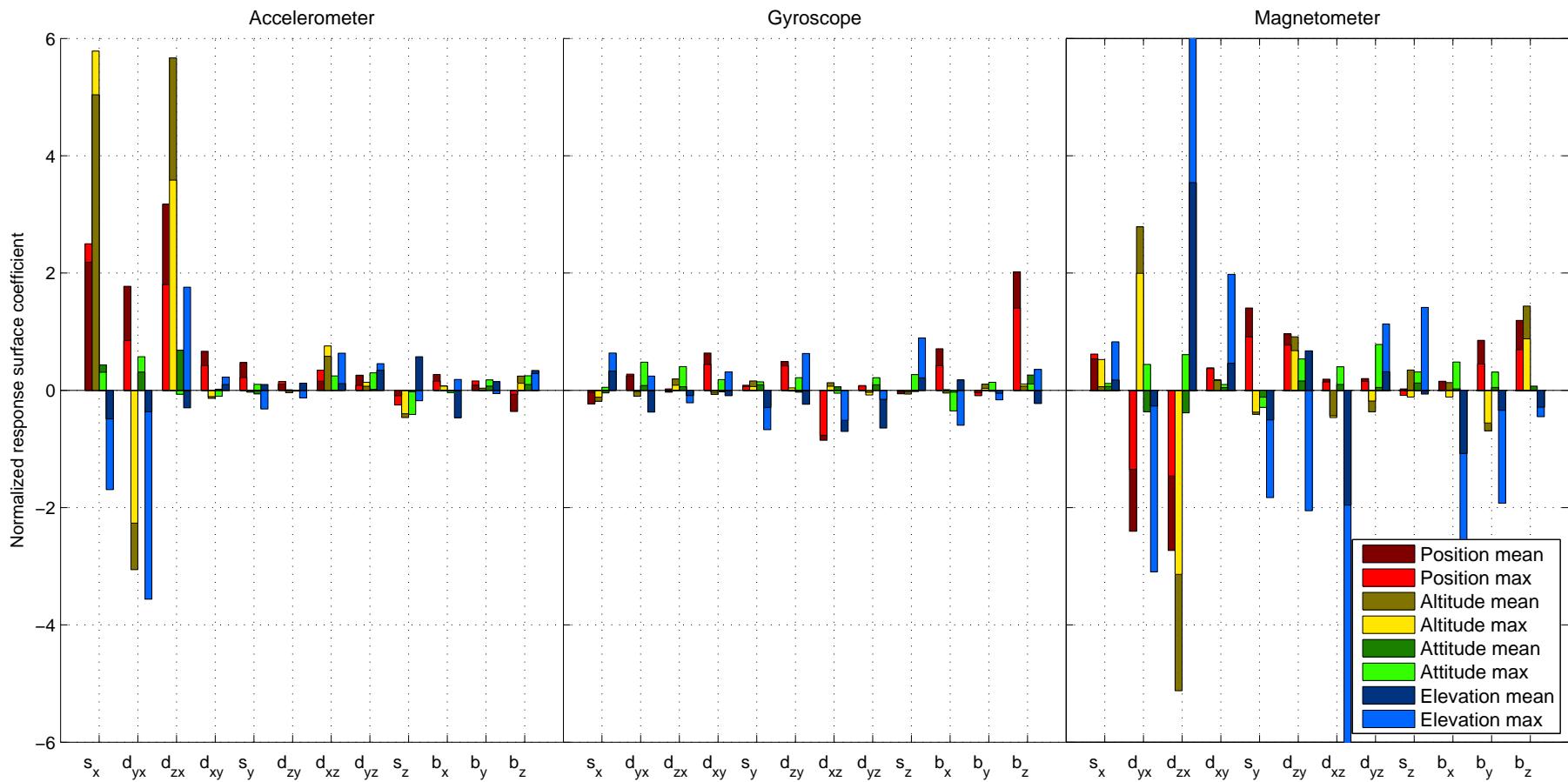


Figure F.8: Normalized response surface coefficients with only bias estimation

Table F.9: Response surface coefficients with bias estimation

Accelerometer	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	265.88	2940.99	664.72	225.15	479.23	112.07	58.18	39.55	91.23	22.10	-66.79	41.92	42.73	-18.55
Position mean	864.46	11820.24	1887.98	1533.00	2743.15	575.83	409.78	91.90	133.53	223.29	-79.07	234.16	76.25	-312.42
Attitude max	-216.88	1333.66	1254.87	-491.44	777.12	-23.11	0.89	1.71	164.21	29.54	-86.68	15.36	-0.73	25.76
Position max	-779.44	4641.20	3928.23	-2382.53	4418.94	-106.30	-25.14	-31.33	452.91	54.80	-362.91	56.93	27.72	190.16
Elevation mean	0.47	3.78	0.15	0.27	-0.03	-0.05	0.05	-0.01	0.12	0.14	-0.19	-0.02	0.08	0.12
Altitude mean	4.19	11.99	1.82	1.31	2.86	0.08	-0.26	-0.01	0.02	0.06	-0.11	-0.06	0.30	0.41
Elevation max	-0.05	-0.45	-0.08	-0.18	0.09	0.01	-0.02	-0.01	0.03	0.02	-0.01	0.01	-0.00	0.01
Altitude max	-0.59	-1.20	-0.29	-0.22	-0.18	0.06	0.06	0.07	0.07	0.20	0.34	-0.27	0.09	0.20
Gyroscope	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	265.88	2940.99	-8.26	73.58	6.15	117.44	17.25	110.78	-203.90	19.62	-4.81	111.70	-24.16	372.53
Position mean	864.46	11820.24	-198.67	213.31	-27.07	549.50	74.30	424.14	-735.83	71.40	-51.59	612.18	-38.99	1744.69
Attitude max	-216.88	1333.66	-25.26	-4.07	19.04	-6.80	14.05	8.82	15.13	-17.40	-5.90	2.21	7.30	23.05
Position max	-779.44	4641.20	-145.43	-79.00	149.28	-54.02	125.36	-11.52	100.32	-25.85	-50.63	-36.12	81.45	55.56
Elevation mean	0.47	3.78	0.02	0.23	0.19	0.09	0.07	0.10	-0.02	0.10	0.13	-0.17	0.07	0.05
Altitude mean	4.19	11.99	-0.18	0.35	0.26	-0.11	0.40	-0.13	0.24	0.39	-0.09	-0.13	-0.06	1.10
Elevation max	-0.05	-0.45	0.03	0.01	-0.01	0.02	-0.03	0.03	-0.03	-0.01	0.04	-0.03	-0.01	0.02
Altitude max	-0.59	-1.20	0.19	-0.22	-0.05	-0.05	-0.17	-0.14	-0.41	-0.38	0.12	0.10	-0.03	-0.13
Magnetometer	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	265.88	2940.99	164.12	-358.28	-388.48	98.92	242.15	204.91	37.87	40.65	-22.98	41.41	119.21	184.27
Position mean	864.46	11820.24	465.76	-2076.54	-2360.90	331.52	1212.52	832.00	164.30	172.76	20.81	128.53	737.69	1030.84
Attitude max	-216.88	1333.66	114.09	432.97	-680.52	39.47	-80.47	146.37	-100.67	-39.48	-25.01	-24.80	-120.97	190.52
Position max	-779.44	4641.20	48.28	2174.04	-3993.23	133.02	-318.50	710.51	-339.48	-284.81	270.71	103.38	-539.60	1120.29
Elevation mean	0.47	3.78	0.06	0.21	0.29	0.05	-0.14	0.25	0.19	0.37	0.15	0.23	0.15	-0.00
Altitude mean	4.19	11.99	0.28	-1.54	-1.61	0.19	-0.49	0.67	0.42	0.21	0.52	0.10	0.21	0.31
Elevation max	-0.05	-0.45	0.04	-0.15	0.55	0.10	-0.09	-0.10	-0.42	0.06	0.07	-0.25	-0.09	-0.02
Altitude max	-0.59	-1.20	0.10	-0.16	2.08	0.27	-0.30	0.40	-1.15	0.19	-0.04	-0.63	-0.20	-0.17

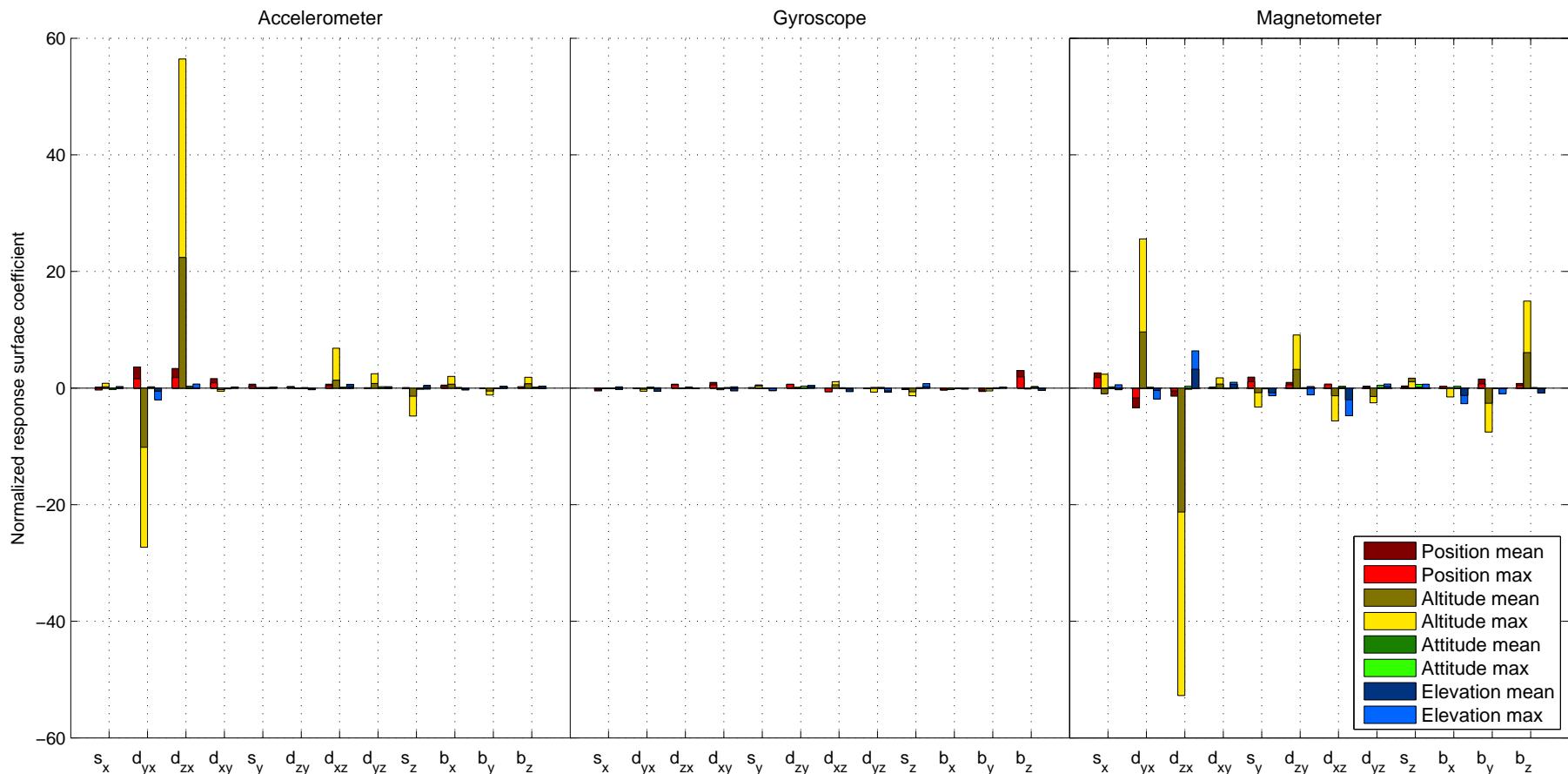


Figure F.9: Normalized response surface coefficients with accelerometer x-scale factor error estimation

Table F.10: Response surface coefficients with accelerometer x-scale factor error estimation

Accelerometer	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	193.12	2457.25	-65.34	308.68	346.04	170.54	61.06	49.81	76.93	8.50	-20.72	55.77	6.79	48.55
Position mean	538.77	10883.50	76.25	1945.89	1812.64	878.49	357.91	103.06	360.37	37.00	50.26	275.47	-1.80	13.03
Attitude max	17.26	287.13	14.24	-471.20	974.31	-9.47	-0.23	0.05	118.07	42.30	-82.71	34.58	-20.54	31.94
Position max	-227.57	1350.21	50.00	-2310.89	5098.30	-58.02	19.22	9.01	306.86	182.86	-318.86	152.73	-115.08	166.64
Elevation mean	0.82	3.70	-0.20	0.17	-0.02	-0.11	0.07	0.07	0.11	0.19	-0.18	0.09	-0.04	0.14
Altitude mean	4.97	10.37	-0.05	0.84	1.59	-0.02	-0.11	-0.07	0.09	0.05	-0.23	0.01	0.06	0.09
Elevation max	0.09	-0.35	0.00	-0.19	0.06	0.02	0.00	-0.03	0.01	-0.01	-0.02	-0.02	0.02	-0.01
Altitude max	0.59	-0.89	0.16	-0.33	-0.01	0.05	0.10	-0.07	0.38	0.14	0.26	-0.21	0.17	0.21
Gyroscope	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	193.12	2457.25	-42.07	4.93	108.89	106.18	15.02	121.06	-119.15	4.31	-41.81	-65.49	-66.77	374.16
Position mean	538.77	10883.50	-260.56	-0.07	362.91	526.84	24.25	358.12	-339.74	-29.54	-128.12	-170.08	-314.08	1620.83
Attitude max	17.26	287.13	-2.02	-9.17	-1.67	-4.80	5.87	2.44	18.82	-11.76	-22.77	-4.66	-8.62	-3.02
Position max	-227.57	1350.21	-2.66	-48.98	-2.92	-21.59	124.71	-7.13	126.51	25.83	-147.68	-1.95	-37.99	-27.10
Elevation mean	0.82	3.70	-0.07	0.12	0.15	0.08	-0.05	0.28	-0.04	0.07	0.07	-0.03	0.05	0.21
Altitude mean	4.97	10.37	-0.15	0.04	0.11	0.22	0.18	0.11	0.30	0.47	0.06	-0.09	-0.45	1.05
Elevation max	0.09	-0.35	-0.02	0.00	0.00	0.02	-0.04	0.04	-0.00	-0.04	0.07	-0.02	0.01	0.00
Altitude max	0.59	-0.89	0.12	-0.33	-0.06	-0.29	-0.27	0.14	-0.36	-0.43	0.14	-0.01	0.05	-0.25
Magnetometer	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	193.12	2457.25	346.05	-326.42	-96.62	19.90	212.38	106.63	113.99	42.07	70.60	56.58	136.64	88.74
Position mean	538.77	10883.50	1403.34	-1812.62	-753.31	86.88	1008.89	528.85	377.66	188.98	112.78	174.98	832.59	433.95
Attitude max	17.26	287.13	41.12	441.19	-910.46	30.02	-56.17	157.02	-97.00	-43.87	18.59	-26.29	-130.08	257.48
Position max	-227.57	1350.21	-218.45	2191.32	-4839.42	160.05	-180.85	731.64	-298.44	-327.91	388.19	20.65	-593.31	1384.96
Elevation mean	0.82	3.70	0.12	0.11	0.25	-0.12	-0.11	0.03	0.24	0.39	0.49	0.24	0.03	0.07
Altitude mean	4.97	10.37	1.06	-0.46	-1.06	-0.43	-0.11	0.33	0.52	0.38	0.67	-0.00	0.02	0.23
Elevation max	0.09	-0.35	0.05	-0.17	0.59	0.09	-0.12	-0.11	-0.44	0.06	0.06	-0.25	-0.09	-0.01
Altitude max	0.59	-0.89	-0.17	-0.25	1.92	0.36	-0.46	0.15	-1.18	0.15	-0.04	-0.75	-0.02	-0.51

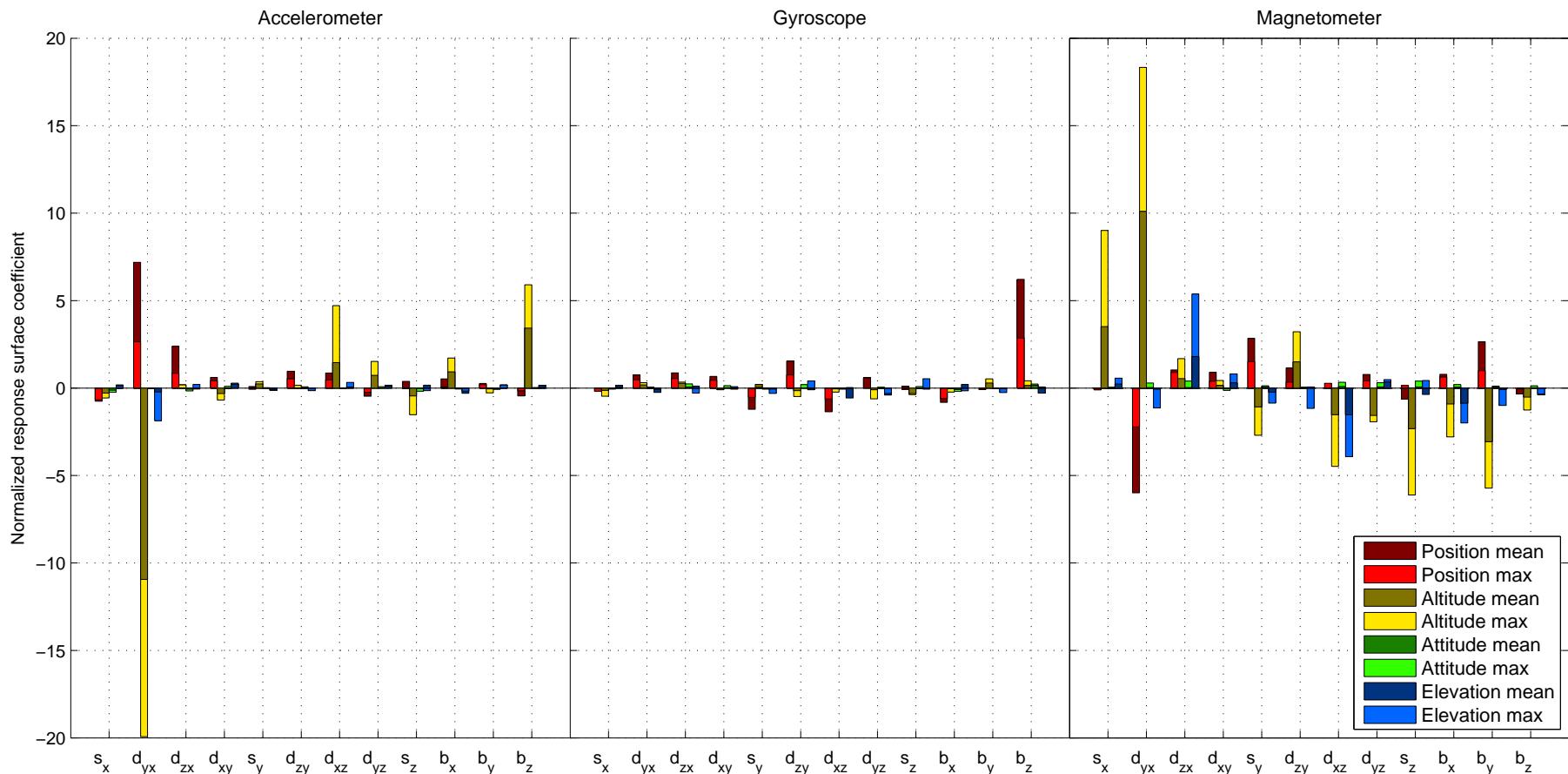


Figure F.10: Normalized response surface coefficients with accelerometer zx-non-orthogonality estimation

Table F.11: Response surface coefficients with accelerometer zx-non-orthogonality estimation

Accelerometer	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	154.70	1856.81	-103.99	409.00	130.08	64.89	-10.42	81.86	72.45	-37.62	-5.66	8.12	30.90	-24.60
Position mean	334.28	7795.49	-249.75	2403.45	799.46	203.23	28.57	322.01	286.46	-155.07	126.74	176.42	88.31	-149.40
Attitude max	25.41	-14.64	-14.41	-506.36	4.62	-17.39	9.27	3.94	119.73	38.65	-38.70	43.53	-6.86	150.10
Position max	-228.95	-177.73	-66.61	-2504.08	43.75	-72.55	54.20	37.21	332.75	166.92	-102.06	212.65	-11.26	784.45
Elevation mean	1.01	3.57	-0.24	-0.01	-0.17	0.10	0.01	0.07	0.02	0.08	-0.18	-0.06	-0.06	0.02
Altitude mean	6.98	8.29	-0.85	-0.02	-0.46	-0.09	0.16	-0.05	-0.00	0.08	-0.06	-0.22	-0.37	0.20
Elevation max	0.11	-0.36	0.01	-0.21	0.02	0.03	-0.01	-0.02	0.04	0.02	-0.02	-0.02	0.02	0.00
Altitude max	0.94	-0.71	0.17	-0.21	-0.05	0.21	-0.12	0.00	0.05	0.10	0.16	-0.28	0.18	0.15
Gyroscope	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	154.70	1856.81	-27.89	73.48	84.69	69.92	-82.17	116.10	-96.74	92.42	-11.97	-92.52	-13.27	441.93
Position mean	334.28	7795.49	-62.32	249.97	291.16	222.84	-405.73	514.52	-452.07	191.79	35.07	-274.28	-2.42	2078.02
Attitude max	25.41	-14.64	-11.77	7.85	8.53	-2.45	1.00	-12.19	-5.93	-15.62	-9.45	-5.90	13.28	10.64
Position max	-228.95	-177.73	-29.50	41.05	62.19	-14.77	47.12	-33.93	-10.49	-21.45	-71.60	-6.03	66.14	33.54
Elevation mean	1.01	3.57	-0.04	0.05	0.25	0.14	-0.06	0.20	-0.04	0.05	0.09	-0.19	0.01	0.24
Altitude mean	6.98	8.29	-0.39	-0.08	0.47	0.15	-0.00	0.08	-0.42	-0.03	-0.13	-0.53	-0.27	1.08
Elevation max	0.11	-0.36	0.01	-0.01	-0.03	0.01	-0.03	0.05	0.00	-0.03	0.06	-0.02	-0.03	0.01
Altitude max	0.94	-0.71	0.15	-0.24	0.11	-0.06	-0.05	-0.09	-0.53	-0.37	-0.06	0.19	-0.02	-0.27
Magnetometer	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	154.70	1856.81	-16.07	-343.43	138.48	59.82	234.43	52.96	42.74	62.80	22.90	97.45	154.31	-4.48
Position mean	334.28	7795.49	-0.52	-2002.04	345.51	300.40	949.20	385.27	3.28	258.60	-212.20	262.83	885.51	-112.43
Attitude max	25.41	-14.64	228.83	465.77	42.66	10.90	-68.48	81.69	-113.69	-48.87	-155.46	-70.97	-145.35	-31.91
Position max	-228.95	-177.73	804.73	2313.11	124.43	34.84	-247.60	343.44	-348.15	-359.04	-532.24	-208.25	-701.38	-118.87
Elevation mean	1.01	3.57	0.04	0.29	0.40	-0.13	0.13	0.05	0.34	0.31	0.41	0.20	0.11	0.13
Altitude mean	6.98	8.29	-0.03	0.18	0.22	-0.24	0.51	0.14	0.60	0.47	0.46	0.31	0.35	0.14
Elevation max	0.11	-0.36	0.06	-0.13	0.61	0.09	-0.10	-0.13	-0.45	0.05	0.05	-0.23	-0.11	-0.04
Altitude max	0.94	-0.71	0.22	-0.06	1.69	0.28	-0.21	0.06	-1.44	0.33	-0.34	-0.82	-0.08	-0.37

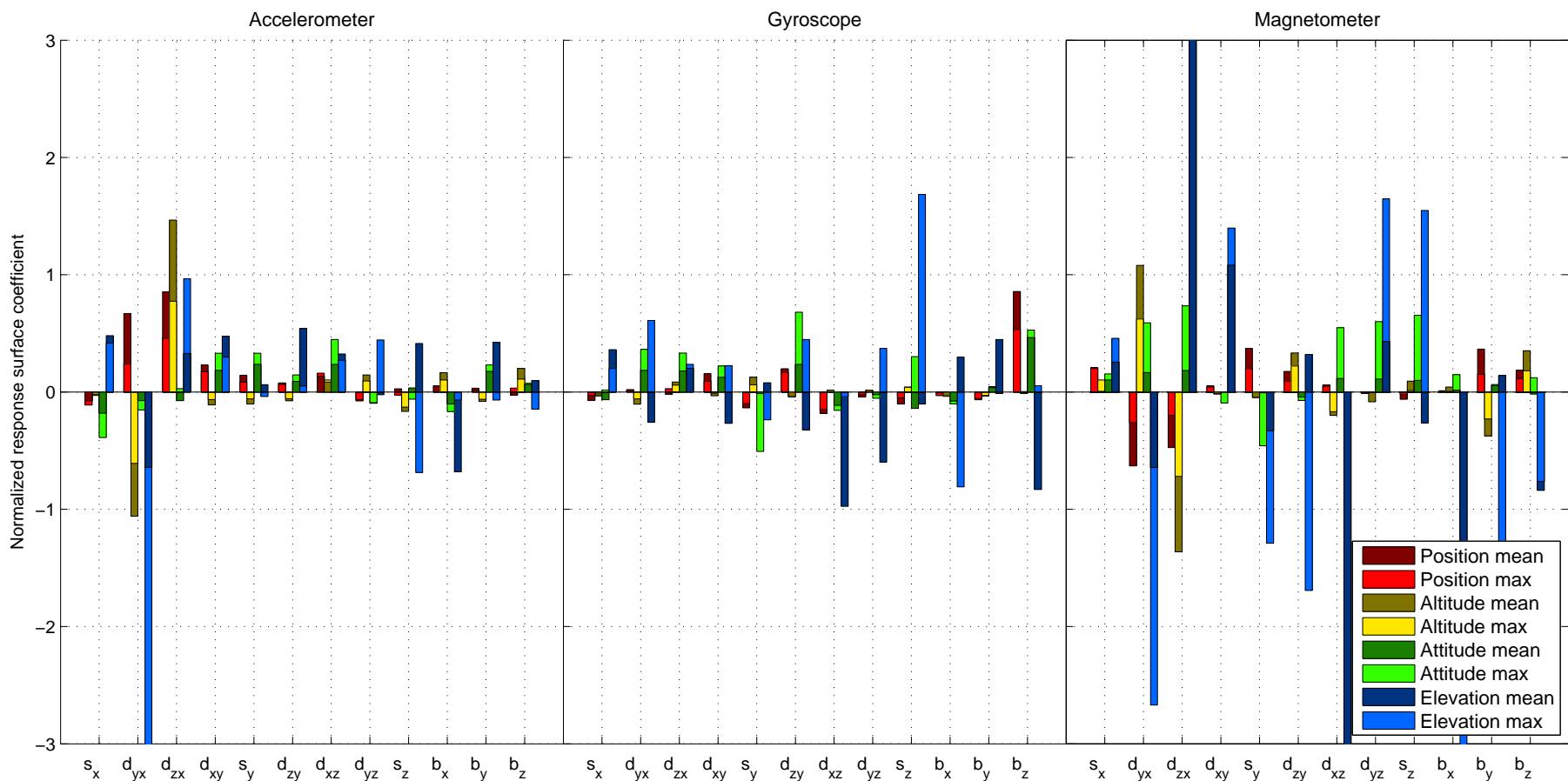


Figure F.11: Normalized response surface coefficients with magnetometer zx-non-orthogonality estimation

Table F.12: Response surface coefficients with magnetometer zx-non-orthogonality estimation

Accelerometer	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	839.36	2397.21	-93.22	199.83	384.68	145.24	69.79	54.99	134.35	-54.64	-22.65	1.16	27.44	27.83
Position mean	2472.89	9706.40	-186.22	1653.09	2113.44	569.53	354.52	191.81	335.52	-189.20	67.99	130.74	49.40	-66.96
Attitude max	780.90	902.46	-14.52	-475.19	604.51	-50.32	-45.58	-44.46	80.56	73.22	-100.70	80.39	-48.28	86.44
Position max	2174.63	2743.95	-64.39	-2304.43	3189.05	-238.54	-224.22	-163.58	185.92	315.58	-359.83	358.14	-174.57	437.36
Elevation mean	0.55	3.57	-0.22	-0.08	0.02	0.18	0.18	0.08	0.25	-0.05	-0.03	-0.09	0.13	0.04
Altitude mean	2.75	8.52	-0.50	-0.20	-0.20	0.51	0.65	0.25	0.64	-0.27	0.10	-0.28	0.48	0.17
Elevation max	-0.05	-0.31	0.02	-0.21	0.05	0.02	-0.00	0.00	0.01	0.02	-0.04	-0.00	-0.00	-0.01
Altitude max	-0.41	-0.64	0.20	-0.26	0.13	0.19	0.03	0.22	0.13	-0.01	0.17	-0.28	0.17	0.04
Gyroscope	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	839.36	2397.21	-23.63	17.12	23.72	77.85	-82.56	140.29	-126.23	-18.20	-41.96	-25.64	-47.05	447.30
Position mean	2472.89	9706.40	-174.79	23.74	-47.40	385.81	-334.84	489.95	-453.22	-105.42	-254.41	-18.50	-163.65	2119.80
Attitude max	780.90	902.46	-11.96	-45.46	45.91	-9.50	47.43	-30.77	10.81	12.16	30.97	-29.03	-22.86	2.08
Position max	2174.63	2743.95	-72.35	-225.38	184.67	-70.44	274.32	-85.28	34.44	32.03	90.94	-73.75	-71.55	-26.17
Elevation mean	0.55	3.57	0.01	0.20	0.18	0.12	-0.28	0.38	-0.09	-0.03	0.17	-0.06	0.03	0.29
Altitude mean	2.75	8.52	-0.18	0.51	0.49	0.34	-0.04	0.65	-0.31	-0.07	-0.38	-0.21	0.10	1.27
Elevation max	-0.05	-0.31	0.01	0.03	0.01	0.01	-0.01	0.02	-0.00	0.02	0.09	-0.04	-0.00	0.00
Altitude max	-0.41	-0.64	0.15	-0.11	0.08	-0.11	0.03	-0.13	-0.40	-0.25	-0.04	0.12	0.18	-0.34
Magnetometer	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	839.36	2397.21	168.92	-219.19	-167.10	36.62	166.60	79.15	41.87	-9.97	-18.50	8.88	127.12	95.62
Position mean	2472.89	9706.40	511.20	-1558.95	-1172.28	133.50	919.93	434.55	147.42	-31.25	-152.69	14.24	904.36	458.22
Attitude max	780.90	902.46	80.22	486.49	-561.69	-13.23	-35.93	173.29	-131.34	-0.01	8.92	5.32	-179.84	141.36
Position max	2174.63	2743.95	12.44	2346.99	-2963.85	-30.58	-99.20	722.89	-432.78	-183.28	200.38	92.76	-814.73	764.51
Elevation mean	0.55	3.57	0.09	0.33	0.41	-0.05	-0.25	-0.04	0.30	0.33	0.36	0.08	0.04	0.07
Altitude mean	2.75	8.52	0.28	0.45	0.50	-0.00	-0.02	-0.12	0.32	0.30	0.27	0.04	0.15	-0.05
Elevation max	-0.05	-0.31	0.02	-0.14	0.57	0.08	-0.07	-0.09	-0.46	0.09	0.08	-0.21	-0.10	-0.04
Altitude max	-0.41	-0.64	0.10	-0.26	1.80	0.44	-0.14	0.13	-1.37	0.18	-0.11	-0.69	0.06	-0.34

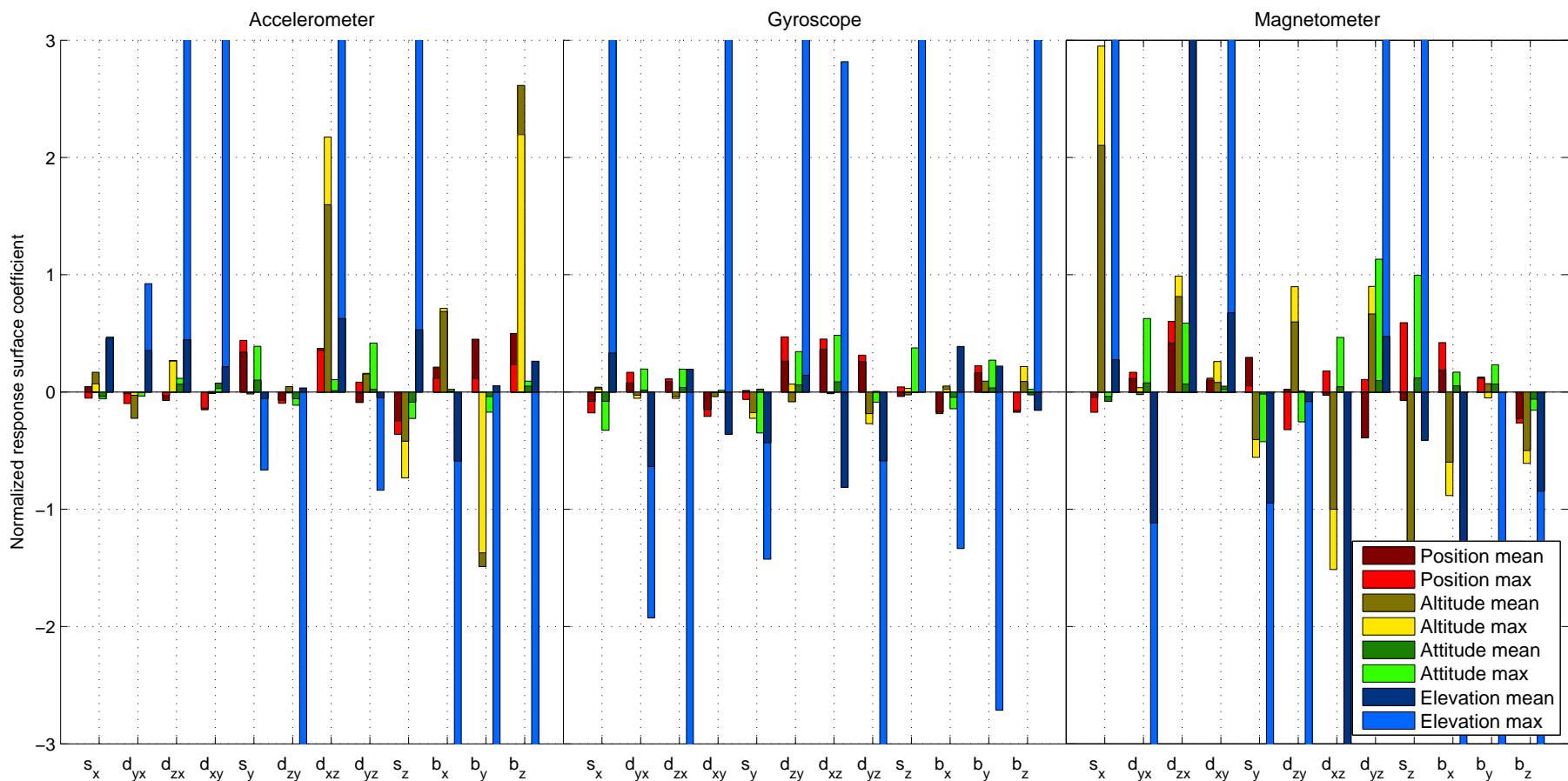


Figure F.12: Normalized response surface coefficients with accelerometer zx- and yx-non-orthogonality estimation

Table F.13: Response surface coefficients with accelerometer zx- and yx-non-orthogonality estimation

Accelerometer	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	208.27	873.96	-10.79	-20.56	-6.53	-28.80	91.64	-20.00	74.09	17.56	-75.11	24.47	24.24	49.02
Position mean	792.09	2701.42	35.73	-11.55	-56.89	-120.50	268.80	-54.95	294.39	-71.19	-194.86	167.56	356.98	395.65
Attitude max	-60.22	142.46	4.24	-1.72	15.91	0.27	0.02	-0.52	130.92	9.59	-44.05	42.94	-82.61	132.15
Position max	-266.65	581.61	44.91	-59.76	71.54	-3.39	-4.49	12.27	425.96	41.29	-112.07	183.19	-397.03	696.98
Elevation mean	0.55	3.27	-0.03	-0.02	0.06	0.02	0.21	-0.06	0.06	0.23	-0.12	0.01	-0.09	0.05
Altitude mean	4.22	7.01	-0.17	-0.01	0.28	0.32	0.43	-0.24	0.04	0.10	-0.37	0.03	-0.16	0.21
Elevation max	-0.00	-0.28	0.00	0.00	0.02	0.02	-0.00	-0.01	0.02	-0.00	0.01	-0.01	-0.04	-0.03
Altitude max	0.46	-0.54	0.21	0.16	0.21	0.10	-0.03	0.02	0.29	-0.02	0.25	-0.27	0.02	0.12
Gyroscope	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	208.27	873.96	-37.09	34.89	23.15	-43.37	-13.23	97.90	94.01	65.31	8.93	-38.47	46.68	-32.95
Position mean	792.09	2701.42	-63.57	61.51	69.98	-117.74	10.63	207.81	287.46	205.31	-30.36	-135.17	130.24	-137.58
Attitude max	-60.22	142.46	1.56	-3.23	-3.27	-0.14	-13.52	4.08	-0.43	-16.28	1.87	1.35	-0.02	13.10
Position max	-266.65	581.61	11.00	-7.26	-10.61	-10.64	-46.86	-22.28	-3.74	-49.32	-6.49	13.48	24.76	24.08
Elevation mean	0.55	3.27	-0.18	0.11	0.11	0.01	-0.19	0.19	0.27	-0.05	0.21	-0.08	0.15	0.01
Altitude mean	4.22	7.01	-0.34	0.07	0.16	0.00	0.10	0.26	0.36	0.02	-0.01	-0.19	0.15	-0.10
Elevation max	-0.00	-0.28	0.01	-0.01	-0.03	0.01	-0.00	0.05	0.01	-0.05	0.05	-0.00	-0.01	0.02
Altitude max	0.46	-0.54	0.15	-0.29	0.09	-0.17	-0.20	0.07	-0.38	-0.27	0.00	0.18	0.10	-0.07
Magnetometer	Nominal	Mean	s_x	d_{yx}	d_{zx}	d_{xy}	s_y	d_{zy}	d_{xz}	d_{yz}	s_z	b_x	b_y	b_z
Attitude mean	208.27	873.96	-36.04	35.12	125.25	24.61	10.88	-67.06	37.37	22.09	122.98	87.65	24.06	-55.03
Position mean	792.09	2701.42	-36.07	91.68	331.54	79.26	234.16	17.57	-22.93	-309.43	-57.53	149.17	101.49	-180.30
Attitude max	-60.22	142.46	177.65	2.26	59.48	15.67	-33.54	54.09	-91.13	54.28	-137.39	-53.20	-3.01	-36.63
Position max	-266.65	581.61	560.87	-5.80	216.93	21.63	-108.32	159.31	-266.62	177.57	-452.71	-159.61	19.11	-133.26
Elevation mean	0.55	3.27	-0.02	0.34	0.32	0.01	-0.23	-0.14	0.26	0.62	0.55	0.09	0.13	-0.09
Altitude mean	4.22	7.01	-0.34	0.33	0.29	0.21	-0.08	0.03	0.19	0.41	0.51	0.22	0.28	-0.26
Elevation max	-0.00	-0.28	0.03	-0.31	0.65	0.11	-0.10	-0.10	-0.47	0.11	0.07	-0.20	-0.03	-0.06
Altitude max	0.46	-0.54	0.13	-0.52	1.78	0.31	-0.44	-0.04	-1.45	0.22	-0.19	-0.70	-0.00	-0.39