Design Synthesis Exercise: HORUS

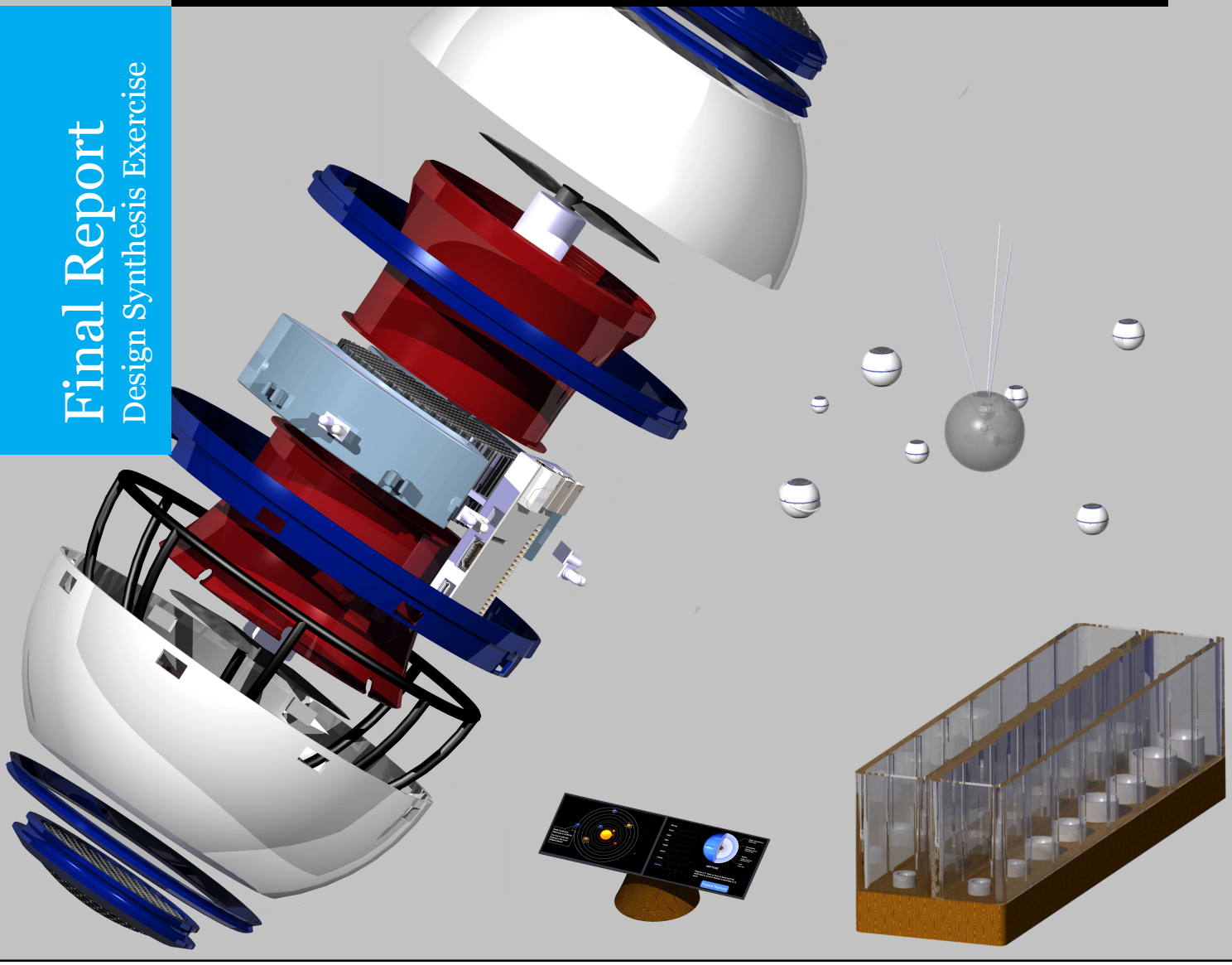
Heliocentric ORrery UAV System

Members Group S12 - Orrery

N.R. Atmopawiro	4291840
I.J. Donders	4164423
D.A.J. Douwes	4135601
R.A. Durrant	4204530
J. Engelsma	4285646
L. Ionescu	4306295
A.M. Krol	4292391
S. Stikkelorum	4287193
X.G. de Vrij	4093526

Tutor: Dr. R.M. Groves
Coaches: R. Sarma,
M. de Athayde Costa e Silva
Faculty of Aerospace Engineering
TU Delft

Final Report
Design Synthesis Exercise



TU Delft Delft University of Technology

Aerospace Engineering

June 28, 2016

Preface

The Final Report of Design Synthesis Exercise (DSE) group S12 is performed as part of the final project of the bachelor curriculum of the Aerospace Engineering Faculty at the TU Delft. The main goal of this DSE project is to design a solar system representation (an orrery) using spherical UAVs as planets. The purpose of this Final Report is to specify design choices and present the final design.

This report in divided into eight parts. Subjects related to the functionality of the system can be found in Part I. Part II is dedicated to the orbital model. Part III elaborates on the UAV Design. Readers interested in the communication system are referred to Part IV. The Sun and ground station design are described in Part V. Different analyses are performed, such as RAMS Analysis and Sustainable Development, which can be found in Part VI. Part VII describes the commercial aspects of HORUS. Subjects related to future implications of HORUS are stated in part VII. This report is wrapped up with the conclusion and recommendation.

The group members of groups S12 would like to thank Dr. R.M. Groves for his guidance and advice during the synthesis of this report, and throughout the project. We would also like to thank coaches R. Sarma and M. de Athayde Costa e Silva for their assistance during the project.

Group S12
Delft, June 2016

Nomenclature

AC	Alternating Current
$ACeP$	Advanced Colour e-paper
BLE	Bluetooth Low Energy
BPM	Brooks, Pope and Marcolini
C	Custom Parameter
CBS	Cost Breakdown Structure
CFD	Computational Fluid Dynamics
$CoFD$	Communication Flow Diagram
$COTS$	Component off-the-shelf
DC	Direct Current
DOD	Depth of Discharge
$e-paper$	Electronic Paper
F	Vale Parameter
FBS	Functional Breakdown Structure
FFD	Functional Flow Diagram
$FPGA$	Field-Programmable Gate Array
GS	Ground Station
HBD	Hardware Block Diagram
HIC	Head Injury Criterion
$HORUS$	Heliocentric ORrery UAV System
IC	Integrated Circuit
ID	Identifier Parameter
IMU	Inertial Measurement Unit
IR	Infrared
LED	Light Emitting Diode
$Li-Po$	Lithium-ion Polymer
M	Orbital Mode Parameter
O	Override Parameter
$OLED$	Organic Light Emitting Diodes
P	Parameter

$PD\&D$	Project Design and Development
PTM	Path Tracking Mode
PWM	Pulse Width Modulation
$RAMS$	Reliability, Availability, Maintainability, Safety
RPM	Revolutions per Minute
$RSSI$	Received Signal Strength Indication
SoC	State of Charge
SPL	Sound Power Level
$SWOT$	Strength, Weakness, Opportunities and Threats
TP	Thunder Power
UAV	Unmanned Aerial Vehicle
UI	User Interface

List of Symbols

Symbol	Unit	Description
α	°	Angle of Attack
β	-	Variable
β_0	-	Variable
γ	-	Variable
γ_0	-	Variable
Δ	m	Distance from Point B' to line AB
δ	m	Boundary Layer Thickness
η_{bat}	-	Battery-to-Load Transmission Efficiency
η_{coax}	-	Co-axial Factor
η_{motor}	-	Efficiency Motor
θ	rad	Angle
θ_B	rad	Angle
θ_C	rad	Angle
μ	$Pa \cdot s$	Kinematic viscosity
ρ	kg/m^3	Density
σ	Pa	Stress
σ_d	-	Expansion Ratio
ϕ	rad	Angle
ψ_{AB}	rad	Angle of Deformation
ψ_{CD}	rad	Angle of Deformation
ω	rad/s	Rotational Speed
A	-	Spectral Shape Function 1
A_e	m^2	Diffuser Exit Area
A_R	-	Variable of A
A_r	m^2	Disk Area Rotor
a	m/s^2	Acceleration
a_0	-	Absolute Value of the Logarithm of the Ratio of Strouhal Number
a_r	m/s^2	Radial Acceleration
$a_{r_{set}}$	m/s^2	Desired Centripetal Acceleration
a_t	m/s^2	Tangential Acceleration
B	-	Spectral Shape Function 2
B_R	-	Variable of B
b	-	Network Environment Variable
b_0	-	Absolute Value of the Logarithm of the Ratio of Strouhal Number
b	m	Flange Length
C	Wh	Capacity
C_d	-	Drag Coefficient
C_l	-	Lift Coefficient
C_T	-	Thrust Coefficient
c	-	Network Environment Variable
D	N	Drag
D_H	-	Directivity functions
DOD	-	Depth of Discharge
d	m	Diameter
d_c	m	Control Distance
d_{min}	m	Approach Distance
d_{prop}	m	Propeller Diameter
dh	m	Height Difference
dt	s	Time Step
dv	m/s	Speed Difference
E	Pa	Young's Modulus
E_{kin}	J	Kinetic Energy

Symbol	Unit	Description
E_{pot}	J	Potential Energy
e	m	Maximum Deviation Distance
e_2	m	Hovering Distance
F_{side}	N	Side Force
F_{design}	N	Force Design Load
F_{sit1}	N	Force Situation 1
F_{sit2}	N	Force Situation 2
$F_{x,y,z}$	N	Applied Net Force in x-, y-, or z-direction
f	Hz	Frequency
g	m/s^2	Gravitational Constant (9.81 m/s^2)
H	m	Distance Between Rotors
H/r	-	Rotor Spacing
H_1	N	Horizontal Force at Location 1
H_2	N	Horizontal Force at Location 2
HIC	-	Head Injury Criterion
h	m	Web Length
h	m	Height
h_{actual}	m	Actual Height
h_{ref}	m	Reference Height
I	m^4	Moment of Inertia
i	A	Current
i_p	-	Iteration Parameter
J	-	Advance Ratio
K	-	Gain Factor
ΔK	-	Amplitude Function 3
K_1	-	Amplitude Function 1
K_2	-	Amplitude Function 2
L_{AB}	m	Length of Line AB
L_{BC}	m	Length of Line BC
M	-	Mach Number
Ma	-	Mach Number
M_{AB}	Nm	Moment at AB
M_{BA}	Nm	Moment at BA
M_{BC}	Nm	Moment at BC
M_{CB}	Nm	Moment at CB
M_{CD}	Nm	Moment at CD
M_{DC}	Nm	Moment at DC
M_c	Nm	Control Moment
M_{max}	Nm	Maximum Moment
m	kg	Mass
N	-	Number of Windings
N_{bat}	-	Number of Batteries
N_r	-	Number of Refreshes per Cycle
n	rev/s	Rotational Speed
n_{beam}	-	Number of beams
P	N	Load
P_c	W	Power Required During a Cycle
P_{in}	W	Power In
P_{out}	W	Power Out
Q	Nm	Torque
R_1	Ω	Resistance 1
R_2	Ω	Resistance 2
R_3	Ω	Resistance 3
R_4	Ω	Resistance 4
Re	-	Reynolds Number
Re_{δ_p}	-	Reynolds Number on Pressure Side
ROI	-	Return on Investment
$RSSI$	dB	Received Signal Strength Indication
r	m	Radius
r_c	m	Hollow Circle Middle Radius

Symbol	Unit	Description
r_{mesh}	m	Radius of the Mesh
r_s	m	Semi Circle Middle Radius
r_{UAV}	m	Radius of the UAV
re	-	Source-Receiver Distance
rpm	rev/min	Rotational Speed
S	m^2	Projected Area
SF	-	Safety Factor
SPL	dB	Sound Power Level
SPL_{tot}	dB	Total Sound Power Level
SPL_v	dB	Sound Power Level Velocity Side
St	-	Strouhal Number
s	m	Distance
T	N	Thrust
T_c	hr	Time of One Cycle
T_{OR}	N	Thrust by an Open Rotor
T_{prop}	N	Thrust by a Single Propeller
T_{SR}	N	Thrust by a Shrouded Rotor
T_{total}	N	Total Thrust
t	s	Time
t_1	s	Time 1
t_2	s	Time 2
t_c	m	Hollow Circle Thickness
$t1$	m	Flange Thickness
$t2$	m	Web Thickness
t_s	m	Semi Circle Thickness
V	V	Voltage
V_1	N	Shear Force at Location 1
V_2	N	Shear Force at Location 2
v	m/s	Velocity
v_∞	m/s	Free Stream Velocity
$v_{x,y,z}$	m/s	Velocity in x-, y-, or z-direction
W	N	Weight
X_{act}	m	Actual Position in x-direction
x	m	Position in x-direction
Y_{act}	m	Actual Position in y-direction
y	m	Position in y-direction
Z_0	m	Mean Orbital Altitude
Z_{act}	m	Actual Position in z-direction
z	m	Position in z-direction

Subscript	Description
α	Angle Side
I	I-beam
max	Maximum
min	Minimum
n	n^{th} Coordinate
p	Pressure Side
R	Hollow Rectangle
s	Suction Side
set	Set
start	Initial position
x	In x-direction
y	In y-direction
z	In z-direction

Summary

In the previous report, the design options for the Heliocentric ORrery UAV Sytem (HORUS) were narrowed down to one preliminary design. In this report the preliminary design is worked out into further detail. The aim of this report is to elaborate on the process of this design and provide the system and performance characteristics.

HORUS, an interactive and educational system, consists of two sets of eight spherical Unmanned Aerial Vehicles (UAVs). Each set consists of UAVs representing the first eight planets in the solar system. Four sizes have been assigned to the UAVs, namely small, medium, large and extra large. The relative sizes of the UAVs correspond to the real sizes of the planets they represent, although on a non-linear scale. The Sun is suspended in the centre of the flight envelope of the system, and will be realistically represented by emitting light and a characteristic surface pattern. A ground station will enable communication with the system.

First the functions of the system has been defined logically and hierarchically using a Functional Flow Diagram and a Functional Breakdown Structure. The system has three top level functions: Perform Systems Check, which tests if all systems work properly, Run Default Mode, which is the primary operation mode, and Run Shutdown Sequence, which shuts down the system by flying the UAVs to their charge stations, after which it checks whether a system encountered any problems or displayed erratic behaviour, and logs all data retrieved from the system.

The system has different operation modes, which are divided into four categories. The user modes consist of the default mode, in which the UAVs will fly according to the orbital model with a mean altitude of 2.5 m, and identification mode, in which a selected UAV will lower to 2 m to identify itself. The supervisor modes will consist of the interaction, realistic, presentation and external emergency mode. In the interaction mode, the altitude will be lowered to 1.6 m to allow interaction between the UAVs and the visitors. In the realistic mode, all terrestrial planets will be represented by a single UAV and all gaseous planets will be orbiting with a realistic spacing. In the presentation mode a selected UAV will hover next to the presenter. The last mode of the supervisor modes is the external emergency mode, in which the observers and UAVs will be brought to safety. The autonomous modes are the charging mode, in which the UAVs will travel to a charge station when they reach a low state of charge, and internal emergency mode, in which internal faults of a UAV will be solved autonomously. Finally, the administrator mode allows the operator to change the parameters the system uses.

These modes define the desired semi-major axis, eccentricity, inclination and velocity of the UAVs. They are used as reference to determine the centripetal acceleration required for the orbit. The semi-major axis, velocity and centripetal acceleration are put into a negative feedback loop together with the real and estimated position of the UAV to maintain the orbit. The position of the UAVs will be determined using five Bluetooth beacons for long range. The distance between a UAV and beacon is determined using the received signal strength indication. For short range positioning four small photo-resistors on the UAVs and Light Emitting Diodes (LEDs) on the charge station will be used for an accurate landing. The path tracking model allows the UAVs to move from any position to the orbit and vice versa. This is done with waypoints along a specified path. For collision avoidance the distance between the UAVs and other object will be determined using infrared sensors and receivers. When a potential collision has been detected in orbit, the outer UAV will increase its orbit radius to avoid the inner UAV. During charge mode the UAV will increase its altitude to avoid all the other UAVs. Photo-resistors are used to detect the objects below a UAV.

The ground station will consist of a charge station, an interface, a server and a communication system. Charging will be done via conduction by having a copper surface making contact with the charge station. The charge station is placed within a 7.2 by 1.8 by 1.8 m clear glass cabinet with an open roof. There are two different interfaces: the user interface, which is accessible by the visitors via a screen on a stand, and the supervisor interface, which is only accessible by the supervisor via a tablet. The server sends and receives signals to and from the UAVs via a Bluetooth signal. The data will be sent using a Pulse Width

Modulation signal with a data rate of 2.7 kB/s. The data is handled by a Raspberry Pi onboard of the UAV and by a TU Delft Laptop as the server.

For the visualisation of the Sun four dimmable LEDs light bulbs will be used to create a bright and diffuse light source. The LEDs will be enclosed by a transparent Perspex sphere with a sticker applied on the surface of this sphere to represent the surface of the Sun more accurately. The Sun will have three suspension points at the ceiling with a fixed pulley at every point in order to vary the height. The Sun will have a total mass of 16.8 kg.

The UAVs will be spherical with a solid shell covered in coating and and a honeycomb mesh on the top and bottom. The solid shell will be made out of polyurethane foam with a supporting layer of plastic. The load carrying structure is placed internally and is able to withstand a fall from 5 m high. Only the bottom half will have this internal structure so that the UAV is bottom-heavy and will land only on the bottom half. The internal structure is made out of carbon fibre circular beams attached to two rings.

The UAV will be driven with a co-axial propeller configuration inside a shaft that is closed off with a aramid fibre honeycomb mesh at both ends. The mesh and shaft are both beneficial to thrust generation as these reduce thrust losses by removing turbulence from the air and the tip vortices from the propellers.

All the subsystems require a total power of 339.2 W for the smallest UAV and 519.2 W for the largest UAV. This power is delivered by two Lithium-ion Polymer batteries with a total capacity of 102.6 Wh and a mass of 0.490 kg. All the UAVs use two batteries except the largest UAV, which uses an additional battery to increase capacity with 5.00 Wh and mass with 44 g. The flight time with a depth of discharge of 90% is 15.3 minutes for the smaller UAV sizes and 18.5 minutes for the largest UAV. For convenience, all UAVs will fly for 15 minutes before heading to the charge station. The UAVs are all charged within 6.7 minutes, so HORUS can operate continuously.

The noise is predicted using the BPM aero-acoustic model with the assumption that both propellers will generate approximately the same amount of noise. It is estimated that the noise produced by both propellers will be approximately 62 dB at 2.5 m from the source, which is the minimum distance the visitors will be from the UAVs during normal flight. This is far below the maximum allowed 70 dB.

The system has a low risk and most failures occurring in the system will result in unscheduled maintenance. All components are easily accessible for quick maintenance with readily available parts. The system will be available for 8 hours a day, 7 days a week, due to the low risk for failure, the short time required for maintenance and repairs, and the second set of UAVs that flies during the charging of the first set. The main concern for safety is the possibility of sustaining a head injury after a UAV stops to function and crashes on someone's head. During operation the highest risk is that the UAV might not be charging properly. This risk can be mitigated by implementing an alarm on the supervisor interface.

The total cost for the prototype system will be €28,665. The largest contribution to the cost is the user interface with €13,500. HORUS is intended to be fully operational by the end of 2017. It is expected that HORUS will be profitable after 1 years if the total cost for production is €150,000. The main targets for selling the system are educational institutions and for rental are high schools and events. The UAVs could also be sold separately on the private market as a child-friendly UAV. The highest risk is that museums might be unwilling to buy HORUS if they have already invested in a more traditional orrery.

Sustainability is an important aspect during the design, operations and the end-of-life of the system. During the design the material required for production and the energy consumption during operations will be minimised. During production HORUS will create jobs by hiring employees for manufacturing, assembly, tests and transportation.

All orders should be placed for the commercially available parts and the raw materials and process these materials. Then the prototype subsystems will be manufactured, tested and, when necessary, altered. If all subsystems work properly then the system will be assembled, tested, validated and, again, altered where necessary.

During the operations of the system the aim of HORUS is to inspire and inform children about the solar system. The system will be easily maintainable, increasing its lifetime. At the end-of-life most of the system can be disassembled and be either resold, recycled or disposed independently.

Contents

1 Introduction 1

I General 2

2 System Description 3
2.1 Top-Level Decisions . . . 3
2.2 Final Configuration . . . 4
2.3 Functionality Diagrams . . . 4
2.3.1 Functional Flow Diagram . . . 4
2.3.2 Functional Breakdown Structure . . . 7

II Orbital Mode 8

3 Operating Modes 9
3.1 Mode Categories . . . 9
3.2 Mode Details . . . 10
3.2.1 User Modes . . . 10
3.2.2 Supervisor Modes . . . 10
3.2.3 Autonomous Modes . . . 11
3.2.4 Admin Mode . . . 12
3.3 Mode Code . . . 12
3.3.1 Verification and Validation . . . 12
3.4 Error Prevention . . . 14
3.5 Requirement Compliance . . . 14

III UAV Design 16

4 Visualisation 17
4.1 Flexible Screens . . . 17
4.2 Coating . . . 18

5 Aerodynamics 20
5.1 Aerodynamic Model . . . 20
5.1.1 Effect of Co-Axial Rotors on Thrust Generation . . . 21
5.1.2 Effect of a Shaft on Thrust Generation . . . 21
5.1.3 Effect of a Mesh on Thrust Generation . . . 21
5.2 Propeller Selection . . . 22
5.3 Motor Selection . . . 23

6 Electrical Power System 25
6.1 Power Budget . . . 25
6.2 Battery Sizing . . . 26
6.3 Electrical Architecture . . . 29
6.4 Battery Configuration and Shape . . . 30
6.5 Charge System . . . 30
6.5.1 Inductive Charging . . . 30
6.5.2 Conductive Charging . . . 31

7 Structure 33
7.1 Load Cases . . . 33
7.2 Materials . . . 34

7.3 Internal Structure Design . . . 34
7.3.1 Method . . . 35
7.3.2 Cross-Section Design Choice . . . 37
7.3.3 Structure Optimisation . . . 38
7.3.4 Verification and Validation . . . 39
7.4 Manufacturing . . . 41
7.5 Shell Structure . . . 41
7.6 Packaging . . . 42

8 Mass Estimation **47**

IV Communication & Control **48**

9 Communications **49**
9.1 Data Handling . . . 49
9.1.1 Bit Assignment . . . 49
9.1.2 Data Stream . . . 50
9.2 Hardware . . . 51
9.3 Communication Flow Diagram . . . 52

10 Positioning **53**
10.1 Long Range . . . 53
10.2 Short Range . . . 54

11 Control Forces **55**
11.1 Working Principle . . . 55
11.1.1 Conventional Variable Pitch Methods . . . 55
11.1.2 Modern Variable Pitch Technology . . . 56
11.2 Implementation . . . 57

12 Orbit **60**
12.1 Orbit Model Description . . . 60
12.2 Orbit Model Results . . . 60
12.3 Verification and Validation . . . 61

13 Path Tracking **62**
13.1 Requirements and Functionalities . . . 62
13.1.1 Path Tracking Requirements . . . 62
13.1.2 Path Tracking Functionality . . . 62
13.2 Model Predecessors . . . 63
13.2.1 1D Ascent . . . 63
13.2.2 2D Movement . . . 66
13.2.3 3D Axis-Based Movement . . . 67
13.3 Complete Path Tracking Model . . . 67
13.3.1 Parameter Creation . . . 68
13.3.2 Waypoint Iteration . . . 69
13.3.3 Situation Determination . . . 69
13.3.4 Velocity Anticipation . . . 70
13.3.5 Force Application . . . 72

14 Collision Avoidance **75**
14.1 Lay-Out . . . 75
14.2 Collision Avoidance System . . . 76
14.2.1 Orbit Collision Avoidance . . . 76
14.2.2 Path Collisions . . . 76
14.3 Verification and Validation . . . 77

V Contributing Elements of HORUS **78**

15 Sun Design **79**
15.1 Visualisation of the Sun . . . 79
15.2 Suspension of the Sun . . . 80
15.3 Mass Estimation . . . 81

16 Ground Station **82**
16.1 Charging . . . 82
16.2 Interface . . . 83
16.2.1 Terminal . . . 83
16.2.2 User Interface. . . . 83
16.2.3 Supervisor Interface . . . 85
16.3 Server . . . 85
16.4 Requirements Compliance . . . 86

17 Hardware **87**
17.1 Hardware Design Choices . . . 87
17.2 Hardware Block Diagram . . . 87

VI Performance of HORUS **89**

18 Compliance Matrix **90**
18.1 Feasibility . . . 91

19 Performance Analysis **92**
19.1 Flight Profile Diagram . . . 92
19.2 Noise Characteristics . . . 93
19.2.1 Noise Propagation . . . 93
19.2.2 Noise Estimation of the UAV. . . . 93

20 Operations & Logistic Concept **97**
20.1 System Logistics . . . 97
20.2 Normal Operations . . . 97

21 RAMS Analysis **99**
21.1 Fault Tree Analysis . . . 99
21.2 Hazard Analysis . . . 103
21.3 RAMS Analysis . . . 104
21.3.1 Reliability . . . 104
21.3.2 Availability . . . 104
21.3.3 Maintainability . . . 105
21.3.4 Safety . . . 105

22 Sensitivity Analysis **106**
22.1 System Parameters . . . 106
22.2 UAV Parameters . . . 107

23 Sustainable Development Strategy **108**
23.1 Environmental . . . 108
23.1.1 During Design Phase . . . 108
23.1.2 During Operations . . . 108
23.1.3 End-of-Life . . . 108
23.2 Societal . . . 109
23.3 Economical . . . 109

VII Commercial Aspect of HORUS **110**

24 Market Analysis **111**
24.1 Goal . . . 111
24.2 Target Market. . . . 111

24.3 Competitive Analysis . . . 111
24.4 SWOT Analysis . . . 111
24.4.1 Internal Analysis . . . 112
24.4.2 External Analysis . . . 112
24.5 Challenges . . . 112
24.6 Return on Investment . . . 113

25 Cost Estimation **114**

VIII The Future of HORUS **116**

26 Risk Assessment and Mitigation **117**
26.1 Risk Assessment . . . 117
26.2 Risk Mitigation . . . 118

27 Product Design & Development Logic **120**
27.1 Product Design & Development Logic . . . 120
27.2 Cost Breakdown Structure . . . 120

Conclusion **123**

Recommendations **125**

Bibliography **126**

A Gantt Charts **128**

B List of Requirements **131**
B.1 Stakeholder Requirements . . . 131
B.2 Internal Requirements . . . 132

C Propeller Characteristics **134**

1 | Introduction

The design of an orrery has been at a standstill since 1704, when a clockwork mechanism was used to represent the positions, motions and phases of the planets in the solar system [6]. With the introduction of new aerospace technology came the Unmanned Aerial Vehicles (UAVs). In today's society the use of UAVs is becoming increasingly popular. They are created for a wide range of functions, from recording footage with cameras to providing aid in emergency situations (e.g. the Ambulance Drone).[a] Furthermore, UAVs provide a spectrum of educational possibilities, from control theory to programming and from hardware to astrodynamics. The definition of an orrery is:

> 'Orrery [awr-uh-ree, or-]; *an apparatus for representing the positions, motions, and phases of the planets, satellites, etc., in the solar system.'* [b]

In Heliocentric ORrery UAV Sytem (HORUS) the use of UAVs is combined with an orrery, to create an educational and interactive solar system. This system is to be installed in an educational institution such as a science centre. The UAVs will be representing the planets and orbit around a sun. Both the orbits and the UAVs will be scaled to fit inside the exhibition space. Human interaction will be facilitated using several operation modes, and through the use of a safe design.

The purpose of this Final Report is to specify design choices, provide the system and performance characteristics and present the final design of HORUS. The project objective is as follows:

> Represent the solar system planetary motion by designing spherical UAVs flying in formation which shall be operational by 2017 with a budget of €1M, of which the design part of the project is to be completed by 9 students within 11 weeks.

The report is divided into eight parts. The report starts with describing the general system description and the functionalities of HORUS in Part I. This includes the final configuration of the entire HORUS system, the Functional Flow Diagram and the Functional Breakdown Structure. Part II describes the orbital model. In this part the different modes and the structure of the orbital model itself are explained. In Part III the UAV Design will be evaluated. The UAV design consists of the elements visualisation, structure and aerodynamics. The electrical power system architecture can also be found in this part. This part will end with a mass estimation of the UAVs. Part IV is dedicated to the communications and control of the HORUS system. This part elaborates on the communication between UAVs and between UAV and server. Also the positioning, path tracking and collision avoidance system will be described. The remaining elements of HORUS can be found in Part V. These elements are the Sun and ground station, the latter consisting of the charge station, interface and sever.

Now the design part is extensively described, the entire HORUS system can be analysed. These different analyses are performed in part VI. This part starts with the compliance matrix, in which the requirements are evaluated. After, the performance, RAMS and sensitivity analyses can be found. Furthermore, one chapter is dedicated to the operations and logistics of the system. During the design phase, choices are made taking sustainability into account. Information regarding this can be found at the end of this part, together with the commercial aspects of HORUS.

To make HORUS operational in the future, the HORUS elements should be produced. Also, the risks it entails should be carefully evaluated. These two elements are described in the final part, Part VIII. This report will be wrapped up with the conclusion and recommendation.

[a] The Ambulance Drone, URL `http://www.tudelft.nl/actueel/laatste-nieuws/artikel/detail/ambulance-drone-tu-delft-vergroot-overlevingskans-bij-hartstilstand-drastisch/`, accessed on 18 April 2016

I

General

2 System Description

HORUS is defined as an educational and interactive system, operational in educational institutions. It will represent the solar system, whereby the planets are represented by UAVs with a spherical exterior. The exterior will have a realistic representation of the planets it represents on the surface of its exterior. Also, the UAVs will manoeuvre according to the orbits of the planets within the solar system. HORUS is set to function in a flight envelope of 10 by 10 by 5 meters. The Sun in placed at the centre of this flight envelope.

The UAVs are to be scalable, implying that a central design is made which is applicable for each size of UAV. The UAV also has to be designed so that the structure can withstand collisions and the propulsion system has to allow for speeds of 4 km/h. Collisions, however, are preferably to be avoided as much as possible by the implementation of a collision avoidance system. To ensure the system and its UAV are usable in the vicinity of children, the UAVs are not allowed to cause injury during operation or malfunctions. Therefore, the surface of the UAVs has to comply with the regulations set for playground surfaces. The requirements imposed on the system can be found in Appendix B.

2.1. Top-Level Decisions

During the conceptual design phase, a number of important design choices were made which have moulded and shaped the system.

The first decision was to exclude the dwarf planet Pluto from the system, as the orbit of Pluto deviates significantly from that of the other planets in our solar system. These deviations would complicate the orbital model significantly as it needs to fit within the flight envelope. The addition of Pluto would furthermore complicate the relative sizing of the UAVs. Therefore, the decision has been made to exclude Pluto, simplifying the orbital model and UAV sizing.

Based on the first decision, the number of UAVs was determined. The number of UAVs was set to eight, one for each of the eight planets in the solar system, from Mercury to Neptune. This allows for **Hor-17** to be considered met. The Sun was initially set as a UAV as well. However, due to the size and weight this UAV would have, this decision was altered to having the Sun suspended by cables connected to a pulley system.

The next decision was to exclude the moons of each planet and the asteroid belt between Mars and Jupiter. The challenge of getting eight individual UAVs to autonomously fly in formation was deemed large enough. Implementing moons could be considered during a redesign of the system.

The stakeholder requirement **Hor-9** states that the system shall function 8 hours a day for 7 days per week over a minimal timespan of 1 year. The feasibility study performed in the Mid-Term report on UAV flight times yielded that UAVs fly for approximately 30 minutes [1]. Based on this, the decision was made to have two sets of identical UAVs present in the system. This allows for a UAV to charge while a second UAV is operating within the flight envelope. When the flying UAV needs to charge, the charging UAV shall take the flying UAV's place within the flight envelope. Incorporating two sets of UAVs fulfils requirement **Hor-13**. Having the UAVs change position allows for **Hor-14** to be met, assuming there are no malfunctioning UAVs. To ensure continuous operation, having the UAVs orbiting for eight hours a day, **Hor-Gnd-7** was set so that there is always one of the identical UAVs flying.

HORUS has been set to function indoors, meeting requirement **Hor-12**. The environmental conditions can be controlled and conditions such as rain and gusts of air generally do not occur indoors and thus are not accounted for in the design of HORUS.

The final decision was that the charge station for the UAVs is to be located outside of the flight envelope, allowing for maximum use of the set flight envelope. In the charge station, there will be room for two sets of UAVs to charge and each UAV has an allocated charging position. The UAVs will be able to enter the charge station from above, where a charge flight envelope is defined. Also, the charge station will not be accessible to the average users of the system.

2.2. Final Configuration

The final configuration of the entire HORUS system is shown in Figure 2.1. As can be seen, the system consists of a user interface, charge station and a Sun suspended to the ceiling. In total two sets of eight UAVs are present. In total four seizes of UAVs exist. Throughout this report, the different elements will be explained in detail.

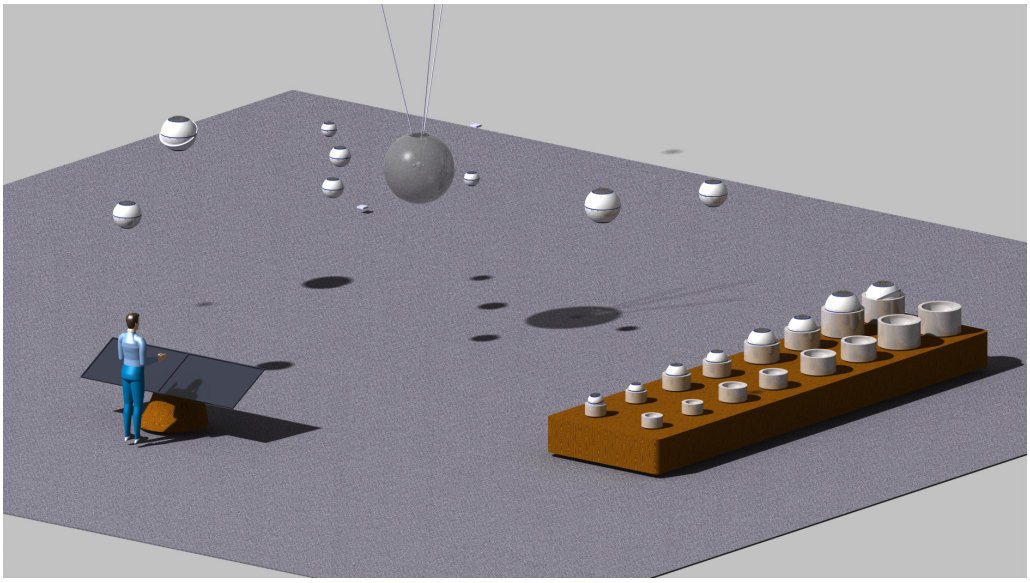


Figure 2.1: Final Configuration of the Entire HORUS System

2.3. Functionality Diagrams

To determine the design of the system, first the functions that need to be performed by the system should be identified. Section 2.3.1 shows the Functional Flow Diagram (FFD) of the system, which details the logical order of functions the system must perform. These functions are also shown hierarchically in the Functional Breakdown Structure (FBS) presented in Section 2.3.2.

2.3.1. Functional Flow Diagram

The functionality diagrams are visual representations of the operation sequence of the system. The FFD is shown in Figure 2.2. It uses a block flow representation to show the order of functions. In this flow process, there are three types of blocks:

1. End point blocks. These are rounded blocks which represent the start and finish points of (sub)-processes in the FFD.
2. Process blocks. These are the square blocks which represent the actions the system takes at a certain point in the process.
3. Decision blocks. These are diamonds which represent the points at which the system has to decide what path to take, based on the question posed in the diamond.

As illustrated by the three separated grey areas, the FFD consists of three levels. Each subsequent level describes in more detail the processes shown in the previous level. This set-up enables a clear overview of the most important processes and working method of the system, as well as a detailed description of the more complicated processes in the higher levels. Each process has a unique identifier, which also serves as an indicator for the next-level process description that is related to that process. An elaborated description on the exact nature of the processes can be found at the end of this section.

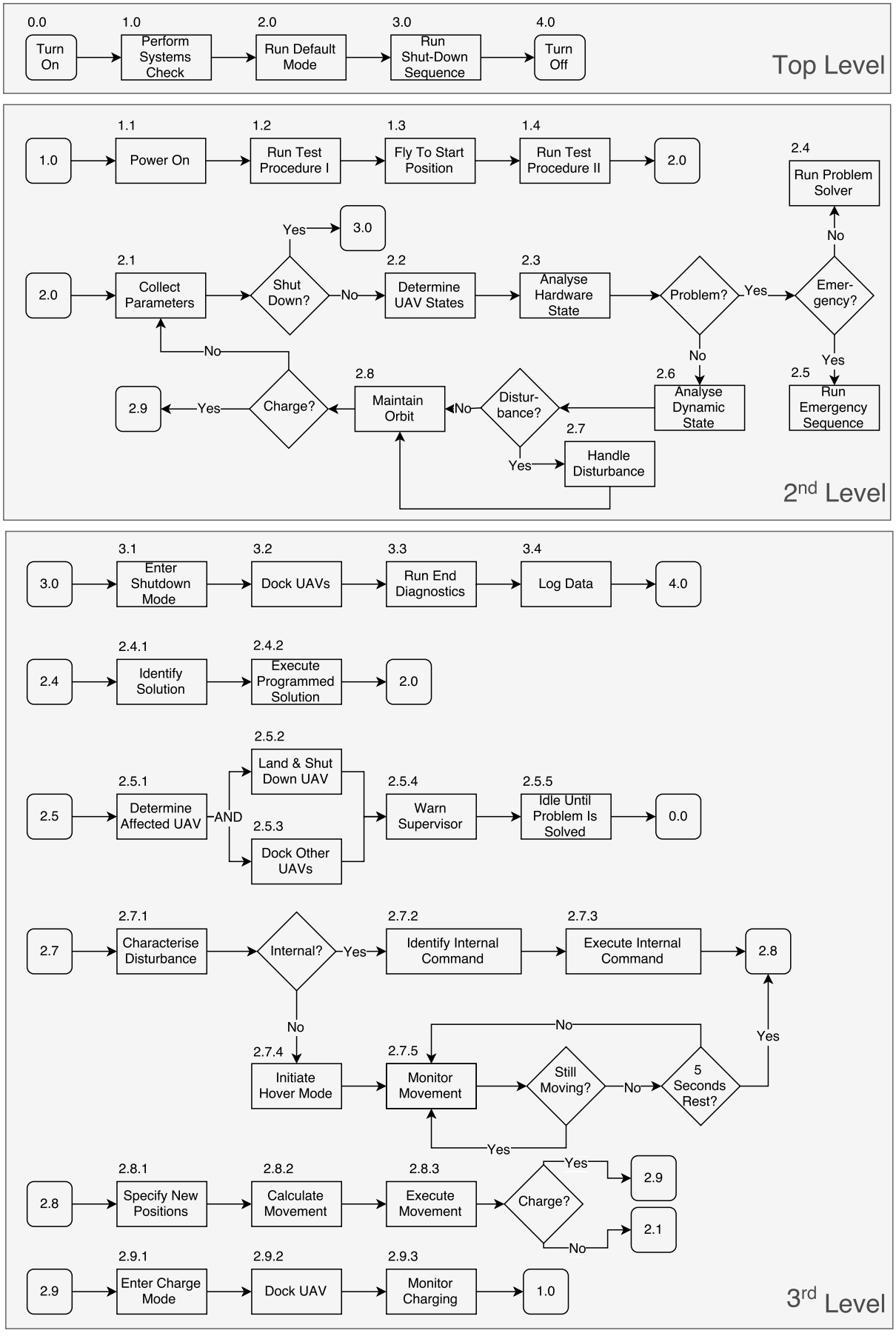


Figure 2.2: The Functional Flow Diagram of HORUS

The processes stated in Figure 2.2 are explained in numerical order below:

0.0	Turn On	
1.0	Perform Systems Check	Check if all systems are working properly
1.1	Power On	Turn on all systems
1.2	Run Test Procedure I	Perform test on the functionality of communications, sensors and mechanics
1.3	Fly to Start Position	Fly from docking position to starting position of the orbit
1.4	Run Test Procedure II	Perform test on functionality during flight
2.0	Run Default Mode	Continuously run the primary mode of operation of the system
2.1	Collect Parameters	Collect all parameters used within the system
2.2	Determine UAV State	Handle UAV-specific data
2.3	Analyse Hardware State	Analyse the status of the onboard hardware of the UAVs from the previously determined UAV state
2.4	Run Problem Solver	Execute pre-programmed problem handling code designed to self-correct predicted problems
2.4.1	Identify Solution	Identify the pre-defined solution that corresponds to the problem
2.4.2	Execute Programmed Solution	Execute the programmed solution that corresponds to the problem
2.5	Run Emergency Sequence	Execute standard protocol in case of an emergency
2.5.1	Determine Affected UAV	Determine which UAV is dealing with an issue
2.5.2	Land And Shut Down UAV	Ensure UAV cannot cause more damage to self or surroundings and can be handled on the ground
2.5.3	Dock Other UAVs	Protect properly functioning UAVs from damaged UAV, and make room for emergency handling
2.5.4	Warn Supervisor	Notify person in charge that an emergency is taking place, so corrective action can be taken
2.5.5	Idle Until Problem Is Solved	Wait until the problem has been taken care of and start-up can commence again
2.6	Analyse Dynamic State	Analyse and characterise the movement of the drone
2.7	Handle Disturbance	Handle the disturbance out of the default mode in order to return to the default orbit
2.7.1	Characterize Disturbance	Determine the nature of the disturbance
2.7.2	Identify Internal Command	Identify the movement that corresponds to the command
2.7.3	Execute Internal Command	Execute the movement that corresponds to the command
2.7.4	Initiate Hover Mode	Discontinue the orbital motion of the UAV and maintain a static position in the air
2.7.5	Monitor Movement	Analyse the movement of the UAVs
2.8	Maintain Orbit	Ensure that the UAVs follow the pre-specified orbital path
2.8.1	Specify New Positions	Calculate the new positions the UAVs should be in at the next time instance
2.8.2	Calculate Movement	Calculate the movements necessary to travel from the initial position to the new positions
2.8.3	Execute Movement	Perform the necessary movements to travel from the initial positions to the new positions, as calculated beforehand
2.9	Perform Charge Mode	Manage charging of (seperate) UAVs
2.9.1	Enter Charge Mode	Discontinue default orbital mode and maintain a static position
2.9.2	Dock UAV	Dock UAV with depleted battery for charging
2.9.3	Monitor Charging	Monitor the charge process of the UAV battery and update the charge status
3.0	Run Shutdown Sequence	Shut the system down
3.1	Enter Shutdown Mode	Discontinue default orbital mode and maintain a static position
3.2	Dock All UAVs	Fly all UAVs to docking station and position in a charging station
3.3	Run End Diagnostics	Check if all systems have encountered problems or displayed erratic behaviour/values
3.4	Log Data	Retrieve and store all the data from the system
4.0	Turn off	

2.3.2. Functional Breakdown Structure

The FBS in Figure 2.3 is an 'AND' tree which shows the functionalities that the system must have for different modes and sequences. The different modes of HORUS include: system check mode, default mode, emergency mode, disturbance handling mode, charging mode and shutdown mode.

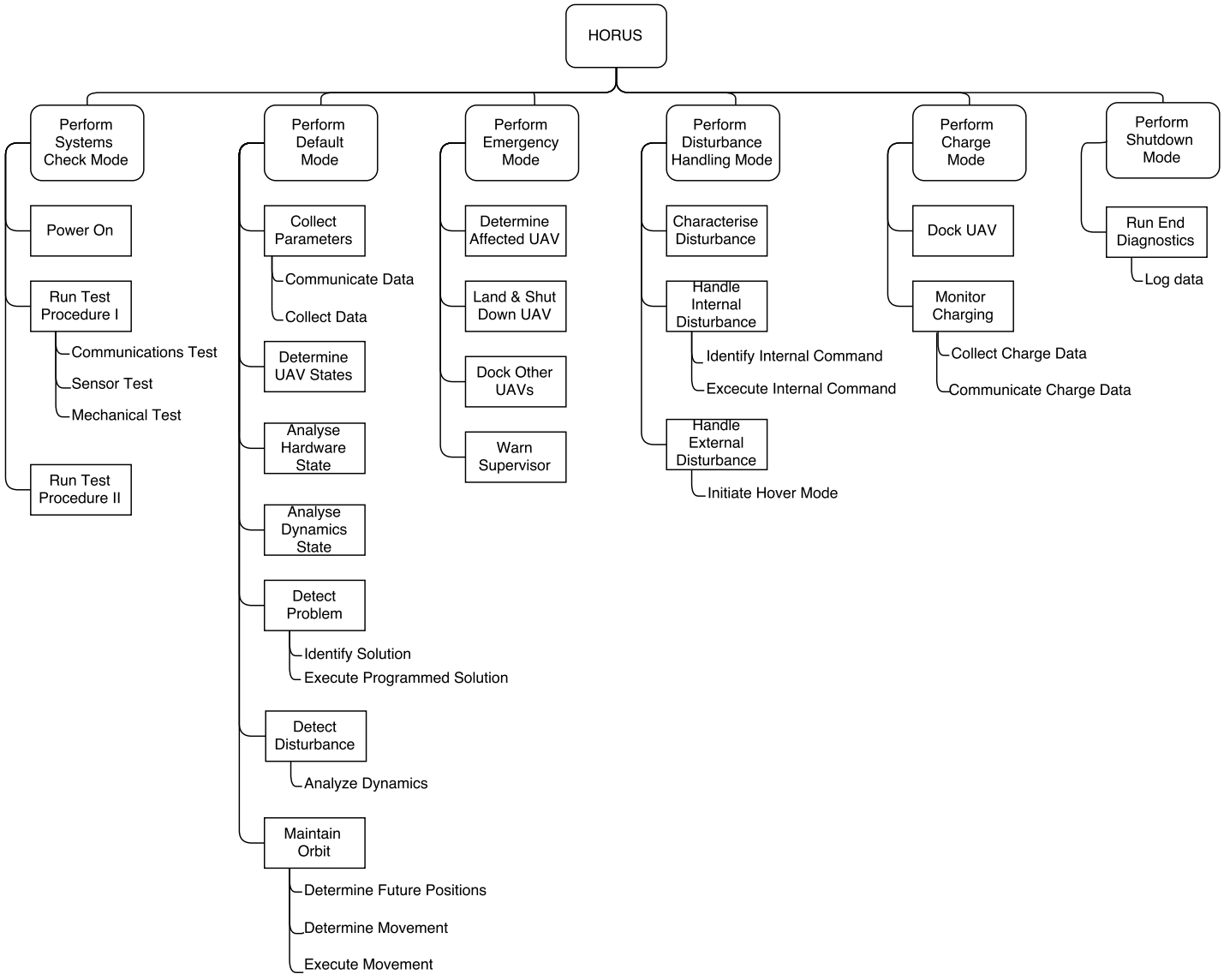


Figure 2.3: The Functional Breakdown Structure for HORUS

II

Orbital Mode

3 | Operating Modes

In this chapter, the different modes proposed for the system will be stated and elaborated on. First the different categories in which the modes are divided are discussed in Section 3.1, followed by a more elaborate explanation of the detailed functionality of the mode and their intended use in Section 3.2. With all the modes explained, the method used to switch between modes is explained in Section 3.3. Lastly, in Section 3.4, some explanation is provided on the system put in place to prevent errors when switching modes. Note that in this last section, the prevented system errors only have to do with the switching of modes, and not with errors in the entire system.

3.1. Mode Categories

There are ten modes proposed for the system, spread over four categories. These categories are the user modes, the supervisor modes, the UAV or autonomous modes, and the administrator modes. The user modes can be activated by any observer of the system. Due to the absence of any clearance required to execute commands to the system in this mode, the amount of control the user has over the system will be very limited. The supervisor modes are modes which can only be activated by an authorised person. As the name suggests, a supervisor will be present when the system operates in modes belonging to this category. Due to the presence of a supervisor, the UAVs are able to get into touching range with the observers in this mode, allowing them to physically interact with the UAVs and study them up close. The presence of a supervisor is assumed to deter observers from using the UAVs in inappropriate ways, which could potentially lead to unscheduled maintenance. The autonomous modes are functions the UAV has which can get activated without any external input. These functions are present to ensure the safe and continuous functioning of the system. The administrator mode functions provide the complete control over the system and its parameters, and can be used to verify the correct functionality of the system. Besides verification, this mode can be used to program in specific desired orbits into the system, allowing customisation of the system for the particular exhibition.

In addition to dividing the modes into four categories based on who can operate the system in the particular mode, the modes can be divided based on the corresponding movement the UAV performs after being assigned a certain mode. The first type, and the most common type, is “Orbiting”. Modes that belonging to this type spur the UAV to follow an orbiting path around a reference. Usually the UAVs will orbit at an average altitude of 2.5 m, and follow an orbit determined by the parameters listed in Table 3.1. It has been chosen to have to have 4 UAV sizes where two planets with similar sizes are grouped in one UAV size: Mercury and Mars as small, Venus and Terra as Medium, Uranus and Neptune as large, and Jupiter and Saturn as extra large. In the table Case A is defined as the distance between the centre of the orbit and the perihelion, minus the body radius. Case B is defined as the distance between the centre of the orbit and the aphelion, plus the body radius. The minimal spacing is the distance between Case A of the planet in question and the Case B of the previous planet. From this table it can also be seen that the requirements stating the minimum and maximum sizes of the UAVs, respectively **Hor-UAV-9** and **Hor-UAV-17**, are complient.

However, for some of the modes associated to Orbiting these parameters change slightly. Whenever this is the case, these changes will be listed in the detailed description of the mode. These other type modes can be associated with “go-to”. In this case the UAV will try to reach a specific destination, using the onboard pathfinding software. The specific destination the UAV put into such a mode will be listed in the detailed descriptions.

Table 3.1: Default Orbital Parameters

Celestrial Body	Body Radius [m]	Semi-Major Axis [m]	Eccen. [-]	Incl. [°]	Speed [m/s]	Case A [m]	Case B [m]	Minimal Space [m]
Sun	0.5	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.
Mercury	0.11	0.85	0.2	7	0.5	0.57	1.13	0.07
Venus	0.15	1.35	0	3.39	0.3697	1.2	1.5	0.07
Terra	0.15	1.7	0	0	0.3144	1.55	1.85	0.05
Mars	0.11	2.25	0.1	1.85	0.2542	1.925	2.575	0.075
Jupiter	0.25	3.25	0	1.3049	0.138	3	3.5	0.425
Saturn	0.25	3.8	0	2.485	0.1023	3.55	4.05	0.05
Uranus	0.2	4.3	0	0.772	0.0772	4.1	4.5	0.05
Neptune	0.2	4.75	0	1.769	0.0573	4.55	4.95	0.05

3.2. Mode Details

In this section the modes will be explained it more detail. Distinction is made between user modes, supervisor modes, autonomous modes and admin modes.

3.2.1. User Modes

Modes that belong to the user modes are the default mode and the identification mode.

Default Mode

The default mode will be the most used and, as the name implies, the standard mode the system will operate in. Whenever the system is activated with no further commands given it will start in this mode. The orbital parameters of the system are the ones listed in Table 3.1. The mean orbiting altitude for the UAVs in this mode is 2.5 m.

Identification Mode

The identification mode differs little from the default mode. The only change is that the UAV which was selected to operate in this mode will obtain a unique feature which will identify itself to the observers. From an orbit viewpoint this unique feature is a slightly lower mean altitude. The mean orbital altitude for a UAV in this mode is 2 m. Further unique features could be added by, for instance coloured lights. More of these options will be discussed in the section concerning visualisation in Chapter 4 The purpose of this mode is to allow users who are not acquainted with the arrangement of the planets to discover which planet is which. The selected UAV will remain in this mode for a set amount of time, or until an new planet is selected by the user.

3.2.2. Supervisor Modes

Modes belonging to the supervisor modes are the interaction mode, realistic mode, presentation mode and external emergency mode.

Interaction Mode

The interaction mode benefits the educational value by allowing the observers to interact with the UAVs directly. In this mode the inclination of all orbits gets increased by multiplying all inclination angles by five, and the mean orbital altitude is lowered to 1.6 m. This causes the system to fly low enough to be touched by observers on one side, and high enough to be walked underneath on the other side. This mode is the main reason to have a supervisor present. The UAVs will remain in this mode until a new mode is assigned to them. Due to the severe deviation from the default mode, this mode cannot be set for specific UAVs, but only for all UAVs at once. Since this mode facilitates interaction between the UAVs and the observers, it reinforces the compliance with requirement **Hor-5**.

Realistic Mode

The realistic mode is one of the modes which can add to the educational value of the system. In this mode, the orbits are more realistic than in default mode. The purpose of this mode is to demonstrate the vast scale of the solar system, especially the difference in distance between the terrestrial planets

(Mercury, Venus, Earth and Mars) and the gaseous planets (Jupiter, Saturn, Uranus and Neptune). In this mode the terrestrial planets could be represented by a single UAV of the smallest size in order to emphasise of the colossal size of the solar system, and the distances between the gaseous planets compared to the distances between the terrestrial planets. The remaining terrestrial planets will move to the charge station while this mode is active. Due to the desire to emphasise the vast distances, the orbital parameters differ vastly from the default mode. The parameters for this specific mode are displayed in Table 3.2. The mean orbital altitude is not different than the default altitude. The UAVs will remain in this mode until they are assigned a now mode. Due to the severe deviation from the default mode, this mode cannot be set for specific UAVs, but only for all UAVs at once.

Table 3.2: Orbital Parameters for the "Sense of Scale" Orbital Mode

Celestrial Body	Body [m] Radius	Semi-Major Axis [m]	Eccen. [-]	Incl. [°]	Speed [m/s]	Case A [m]	Case B [m]	Minimal Space [m]
Sun	0.5	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.
Terrestrial	0.11	0.65	0	0	0.22	0.54	0.76	0.04
Jupiter	0.25	1.41	0	1.3049	0.0102	1.16	1.66	0.4
Saturn	0.25	2.31	0	2.485	0.0041	2.06	2.56	0.4
Uranus	0.2	3.56	0	0.772	0.0014	3.36	3.76	0.8
Neptune	0.2	4.76	0	1.769	7.3e-4	4.56	4.96	0.8

Presentation Mode
The presentation mode is, like most other supervisor modes, used to increase the educational value of the system. A UAV in this mode will hover next to a presenter who is providing information regarding the specific planet to the observers. This allows the observers the see the UAVs better, and the possibility to display planetary weather conditions to increase education. The UAV will remain in this mode until either a new UAV is selected by the presenter, or the mode of the entire system is switched. Additionally, with the inclusion of this mode requirement **Hor-Ctrl-14** has been met.

External Emergency Mode
The external emergency mode is one of the two emergency modes which are implemented in the system. The purpose of this mode is to bring either the UAVs, or the observers to safety. This mode could be triggered in case of an emergency such as fire, flooding, of any other reason for evacuation within the building in which the system is exhibited. In order to lead observers to safety the UAVs could mark the path to the closest point of exit in order to provide evacuees with markers to the outside. If this mode is not supported by the particular exhibition, this mode would cause all the UAVs to bring themselves to safety, either by docking, or to follow other safety procedures. The UAVs will remain in this mode until a new mode is assigned to them.

3.2.3. Autonomous Modes
The charging mode and the internal emergency mode belong to the autonomous modes.

Charging Mode
The charging mode is one of the two autonomous modes in which the designed system will possess. This mode is activated by the detection of a low State of Charge (SoC) in the batteries. When this mode is activated, the UAV will make its way to the charge station where it will dock and recharge. A charging UAV will remain at the charge station until it has to relieve its twin. However, if the SoC of the charging battery is too low when its twin also charges it will remain in this mode until it has charged to the minimum specified level. This should not occur however, due to the similar batteries and the relation between the time it takes to charge and discharge a battery. In Chapter 6 this subject will be explained more in detail. Precaution will also be taken by not allowing the battery to drain completely, ensuring compliance with requirement **Hor-15**

Internal Emergency Mode
The internal emergency mode is the second autonomous mode which the UAV can get into without a command from the supervisor required and resembles the previously mentioned emergency mode with one key difference. This mode is triggered by the detection of a fault within the UAV which causes is to execute a specific set of commands. Two types of internal faults can be defined. The first type is a fault for which a contingency plan has been provided in the code which mitigates the damage from the specific fault. The second type are unforeseen errors for which no specific plan has been made. For this second type a generic contingency plan has been put in place. The UAV will remain in this mode until it is switched by a command specific for the particular UAV. In some of the emergencies it is desirable to have the faulty UAV to initiate landing. During this landing it will follow a safety landing path, which will initially force the UAV to ascent to a level of 2.5 m if needed, otherwise stay at its initial height. Following this it will go to the charge station and dock there. With this safe flight plan in place for emergencies the UAV will not cause any injuries to observers, and thus comply with requirements **Hor-Ctrl-6** and **Hor-Ctrl-7**.

3.2.4. Admin Mode
The admin mode has the most rights compared to the other modes. An operator who activates this mode is able to change the parameters the system uses. This means that the operator can tune the inclination, size of the orbits, and velocity, and the mean orbital height. However, due to the method used to determine these parameters in each of the modes, and to ensure the safety of the system, the ratios between the orbits will remain the same. Furthermore, the admin can at will activate and overwrite any other mode, including the autonomous mode. This function is implemented to provide a method by which the system can be checked for compliance in the room it is intended to operate in. A UAV will remain in this mode until the admin mode is switched off.

3.3. Mode Code
The method proposed for switching between orbital modes was decided upon as follows. Staring off, the ground station will sent a signal out, and each of the UAVs would receive this signal. In order to determine whether or not the received mode is intended for a specific UAV, the signal sent will start with an identifier. This identifier will either specify a single UAV, a predetermined group of UAVs (such as all UAVs representing terrestrial or gaseous planets) or call all UAVs. If the identity of the UAV does not meet any of these criteria, the UAV will continue on in its previously set mode. If the identifier does correspond to the UAV, it will switch to the new mode, setting the corresponding orbital parameters. As mentioned in the previous section, in the admin mode, the administrator has the ability to modify or scale parameters at will. If the admin wishes to utilise this feature, a switch would have to be toggled, telling the system to use custom parameters, instead of the default ones hardcoded into each of the UAVs. The decision to use custom parameters and the value of these parameters are also in the signal received by the UAV when applicable. The UAV then processes these parameters and outputs the desired semi-major axis, inclination, eccentricity, orbital velocity and mean orbital altitude, or the coordinates of waypoints to the pathfinding programs, whichever is applicable. To recap, the signal transmitted by the ground station and received by the UAVs contains an identifier, a key specifying whether or not to use custom parameters, each of the custom parameters, and the mode the system has to operate in.

3.3.1. Verification and Validation
The method used to verify the functionality of the Simulink model made to represent the mode switcher is as follows. As input to the model is a signal with the values of all parameters encoded into it. This signal would go into a decoder which would pull all information from the signal. Following this all the information goes into the mode selector, where the information is processed, and the semi-major axis, inclination, eccentricity, orbital velocity and mean orbital altitude come out of it, or the coordinates of waypoints, dependent on the mode. At each stage in the system where data was processed and transformed, a scope was placed to observe the inputs and outputs. This data was then compared to the expected outcome. Several telling results from this tests can be seen in tables 3.3 to 3.7. Note that all the data in these tables is Mars, except during the Realistic Mode test, here the data is taken from Saturn. In Table 3.5 the expected outcome of the program is a set of waypoints which will lead the UAV to its destination in a safe manner. However, since the exact location of the destination, in this case the charge

station, is not known, a temporary set of coordinates was used as expected result. From these results it can be concluded that the system as it is programmed in Simulink works according to expectation, and no further work is required on it.

Table 3.3: Verification & Validation Data from the UAV representing Mars, set in Default Mode. In this Specific Case all UAVs were set in this Mode Simultaneously, but only the Data from Mars is presented.

Inputs	Binary	Value	After rounding	Output	Expected	Simulated
UAV ID	11111	31	31	Semi-Major Axis	2.1	2.1
Admin Override	0	0	0	Inclination	1.85	1.85
State-of-Charge	n.a.	0.9	0.9	Eccentricity	0.1	0.1
Custom Parameters	0	0	0	Velocity	0.2542	0.2542
Orbit Mode	000	0	0	Mean Altitude	2.5	2.5
Max Orbit Size	0|00100|1100000000	4.75	4.75	Waypoints	n.a.	n.a.
Max Orbital Speed	0|00000|1000000000	0.5	0.5			
Inclination Scale	0|00001|0000000000	1	1			
Semi-Major Mode	01	1	1			
Eccentricity Mode	01	1	1			
Velocity Mode	01	1	1			
Mean Altitude	0|00010|1000000000	2.5	2.5			

Table 3.4: Verification & Validation Data from the UAV representing Saturn, set in Realistic Mode. In this Specific Case All UAVs were set in this Mode Simultaneously, but only the Data From Saturn is presented.

Inputs	Binary	Value	After rounding	Output	Expected	Simulated
UAV ID	11111	31	31	Semi-Major Axis	1.4	1.4
Admin Override	0	0	0	Inclination	1.3049	1.3049
State-of-Charge	n.a.	0.9	0.9	Eccentricity	0	0
Custom Parameters	0	0	0	Velocity	0.0041	0.0041
Orbit Mode	011	3	3	Mean Altitude	2.5	2.5
Max Orbit Size	0|00100|1100000000	4.75	4.75	Waypoints	n.a.	n.a.
Max Orbital Speed	0|00000|1000000000	0.5	0.5			
Inclination Scale	0|00001|0000000000	1	1			
Semi-Major Mode	01	1	1			
Eccentricity Mode	01	1	1			
Velocity Mode	01	1	1			
Mean Altitude	0|00010|1000000000	2.5	2.5			

Table 3.5: Verification & Validation Data from the UAV representing Mars, set in Charging Mode. In this Case only the State-of-Charge of Mars was set to a Level which would Trigger Charging Mode.

Inputs	Binary	Value	After rounding	Output	Expected	Simulated
UAV ID	00011	3	3	Semi-Major Axis	n.a.	n.a.
Admin Override	0	0	0	Inclination	n.a.	n.a.
State-of-Charge	n.a.	0.08	0.08	Eccentricity	n.a.	n.a.
Custom Parameters	0	0	0	Velocity	n.a.	n.a.
Orbit Mode	100	4	6	Mean Altitude	n.a.	n.a.
Max Orbit Size	0|00100|1100000000	4.75	4.75	Waypoints	[3.5,1.6,1.8]	[3.5,1.6,1.8]
Max Orbital Speed	0|00000|1000000000	0.5	0.5			
Inclination Scale	0|00001|0000000000	1	1			
Semi-Major Mode	01	1	1			
Eccentricity Mode	01	1	1			
Velocity Mode	01	1	1			
Mean Altitude	0|00010|1000000000	2.5	2.5			

Table 3.6: Verification & Validation Data from the UAV representing Mars, set in Interaction Mode after having its Charging overridden by an Admin.

Inputs	Binary	Value	After rounding	Output	Expected	Simulated
UAV ID	00011	3	3	Semi-Major Axis	2.1	2.1
Admin Override	1	1	1	Inclination	9.25	9.25
State-of-Charge	n.a.	0.06	0.06	Eccentricity	0.1	0.1
Custom Parameters	0	0	0	Velocity	0.2542	0.2542
Orbit Mode	010	2	2	Mean Altitude	1.6	1.6
Max Orbit Size	0|00100|1100000000	4.75	4.75	Waypoints	n.a.	n.a.
Max Orbital Speed	0|00000|1000000000	0.5	0.5			
Inclination Scale	0|00001|0000000000	1	1			
Semi-Major Mode	01	1	1			
Eccentricity Mode	01	1	1			
Velocity Mode	01	1	1			
Mean Altitude	0|00001|1001100110	1.599609375	1.6			

Table 3.7: Verification & Validation Data from the UAV representing Mars, set in an Orbit using Custom Parameters by an Admin.

Inputs	Binary	Value	After rounding	Output	Expected	Simulated
UAV ID	00011	3	3	Semi-Major Axis	2.343	2.343
Admin Override	0	0	0	Inclination	-3.885	-3.885
State-of-Charge	n.a.	0.8	0.8	Eccentricity	0	0
Custom Parameters	1	1	1	Velocity	0.0627	0.0626
Orbit Mode	000	0	0	Mean Altitude	1.85	1.85
Max Orbit Size	0|00101|0100110011	5.2998046875	5.3	Waypoints	n.a.	n.a.
Max Orbital Speed	0|00000|1000000000	0.4892578125	0.489			
Inclination Scale	0|00010|0001100110	-2.099609375	-2.1			
Semi-Major Mode	01	1	1			
Eccentricity Mode	00	0	0			
Velocity Mode	10	2	2			
Mean Altitude	0|00001|1001100110	1.849609375	1.85			

3.4. Error Prevention

In order to prevent errors from two modes from getting activated simultaneously, a ranking was put into the program. The ranking is the same as the order of the modes as listed above, with the exception of the external emergency mode, which supersedes the charging mode. Whenever two modes are activated simultaneously, only the mode with the highest rank gets activated. For clarity, the rank of the modes from lowest to highest are: *default*, *identification*, *interaction*, *realistic*, *presentation*, *charging*, *external emergency*, *internal emergency*, *admin*.

Default mode is ranked the lowest as this is the default mode the system operates in. Logically following from this is identification mode as it is the only remaining mode in the user category. Going to the supervisor category, the next mode in line is the interaction mode. There is no reason for order of the interaction and realistic mode, besides the fact that there has to be one. It was chosen to put interaction mode ahead of realistic mode. Following these two mode is the presentation mode. This mode was given a considerably high rank to allow a presenter to give an explanation regarding a specific planet, while the rest remain orbit behind the supervisor in the previously selected mode. The reason for switching the charging and external emergency mode stems from the fact that it is undesired to have the UAV go the the charging station when the emergency could potentially be at the charge station, such as a fire. The internal emergency mode is placed ahead of the external mode as it is undesired to have a malfunctioning UAV attempting to operate under such circumstances, or at all. Lastly, admin mode is given the highest priority for obvious reasons.

Another method implemented to prevent the system from malfunctioning are default modes to which the system will switch if the signal received does not lead to a valid mode switch. With these two systems in place the mode switcher software should reliably function.

3.5. Requirement Compliance

The way the orbital system is proposed, the UAVs will follow the approximate motions of the real planets in a non linear coordinate system. Because visitors are able to see the UAVs orbiting, read the informa-

tion presented regarding them, and listen to the presentation, the UAV can function as an educational tool. Furthermore, the interaction with the user interface, and with the UAVs in the interaction mode is possible. The maximum size of the orbit is by default set to fall within the flight envelop. Finally, the observers will be able to walk between the UAVs, except for the times where two near UAVs pass by each other. Because of all this the system complies with the requirements **Hor-2**, **Hor-3**, **Hor-4**, **Hor-5**, **Hor-6**, and **Hor-7**. Additionally, since the maximum orbital velocity if the system can be set by an admin, and is set by default at a value lower than 4 km/h, the system complies with requirement **Hor-UAV-5**. Furthermore, due to the fact that a mode is proposed which will get automatically activated when the UAV detects a low state of charge, it can comply with requirement **Hor-Ctrl-2**. However, whether or not the system actually complies with this requirement depends on the charge station. Because the orbital modes in which eight UAVs are desired facilitates eight UAVs in simultaneous orbit, requirement **Hor-14** can be complied with. However, whether or not this requirement is actually met will be determined by the control system. Due to the fact that the default mode is used as the standard whenever the system is enabled, it serves as a standard operating mode, complying with requirement **Hor-Ctrl-12**. Lastly, the fact that in this default mode the default mean orbital altitude is set at a height of 2.5 m, the system complies with the requirement **Hor-Ctrl-13**.

III

UAV Design

4 | Visualisation

The visualisation of the planets on the UAVs is an integral part of the HORUS design. When done properly it ensures each planet can be identified, as required by **Hor-1** and **Hor-UAV-4**, and enhances the visitors experience of the HORUS. In this chapter the method of displaying the planets using flexible screens will be described in Section 4.1 and the use of coating in Section 4.2.

In the Mid-Term Report it was chosen to use flexible screens as well as a coating on the surface to visualise each planet on an UAV [1]. After considering requirement **Hor-UAV-22**, this visualisation method is withdrawn from the UAV design. The additional weight of the flexible screens would make the UAVs too heavy. Requirement **Hor-UAV-4**, to have the UAVs mimic a specific planet in an identifiable manner, can still be attained by having a coating applied to the shell. Although flexible screens was withdrawn as a visualisation method, Section 4.1 can still be used as reference if the UAV design is improved upon in the future and the visualisation sub-system with flexible screens can be added. The alternative method for the representation of the planets, namely applying a coating on the shell of the UAV, will be used as final design choice for the visualisation of the planets.

4.1. Flexible Screens

The UAVs will have to be realistic representations of the planets in the solar system. The flexible screens will be mounted on the shell structure of the UAV. There are two types of flexible screens available: Organic Light Emitting Diodes (OLED) and Electronic Paper (e-paper). An OLED consists of a series of organic thin films between two conductors. In addition to being energy efficient, this technology also does not contain any bad metals and is made solely out of hydrogen and carbon, hence the organic interpretation. E-Paper on the other hand uses micro capsules filled with different colours that can rearrange themselves to graphics and text by using an electrical charge. This in turn also implies that the energy usage of e-paper could be zero during flight, if the visualisation on the UAV is loaded onto the e-paper while the UAV is docked and remains the same throughout the UAV flight time. Another advantage of the e-paper in comparison to OLED, is that e-paper is lightweight in relation to OLED. For these reasons, e-paper was initially chosen as a means to visualise the planets on the UAVs.

There are two methods to drape the e-paper on the UAV: polyconical and polycylindrical (see Figure 4.1). With the polyconical method the screen will be divided into tapes with varying radii and then attached to the shell. See Figure 4.1a for how the sphere is divided into tapes. This requires a very thin screen so the tapes can properly follow the curvature without buckling. With the polycylindrical method the the screen will be divided into single-curved cylindrical surfaces that will be draped from the top to bottom, see Figure 4.1b.

(a) **(b)**

Figure 4.1: Polyconical (a) and Polycylindrical (b) Method to approximate a Spherical Surface [17]

Because part of the top and bottom of the shell should remain open for the airflow, the methods should be adapted so they do not cover an entire sphere. For the polyconical method this means that the shortest tapes will not be manufactured. For the polycylindrical method the tips of the cylindrical surfaces should be cut off. The polyconical method is less complex for manufacturing and has less waste. However, the required maximum thickness to have a good placement of the e-paper on the spherical surface is

too small to be feasible. Therefore the polycylindrical method has been chosen for manufacturing the spherical shape of the shell.

To incorporate the distinctive colour of each of the planets in the visualisation a colour e-paper needs to be used. E Ink offers Triton, a matrix colour imaging film which is capable of showing 4096 colours.[a] The company also recently released information on their new prototype, Advanced Color ePaper (ACeP), with a range of 32,000 colours. It would be a possibility to switch the e-paper technology when ACeP goes into commercial production, which is expected to be in the next two years.[b] The custom shape of the e-paper display can be manufactured by Plasticlogic, experts in glass-free, flexible display manufacturing. E Ink has also indicated the openness to a joint development partnership, so both technologies can be combined to produce the visualisation system on the UAVs.[c]

A mass and energy estimation is made for each UAV size and can be seen in the Table 4.2. The surface area covered in e-paper is taken to be the complete surface area of the UAV minus the top and bottom cones where the mesh structure will be. The radius of the spherical zone is $\frac{4}{5} \cdot r$ of the bottom hemisphere and $\frac{1}{2} \cdot r$ on the top hemisphere, where r is the radius of the UAV. Plasticlogic's 15.4" Lectum display is taken as a reference for the mass per surface area kg/m^2 used in the mass estimation and the energy per surface area J/m^2 for the energy estimation.[d,e] Each UAV is expected to have two display changes during each battery cycle. The UAVs could also change their display while docked in the charge station.

Table 4.1: Estimated Mass and Energy of E-Paper Display

	Radius [m]	Mesh Radius [m]	Zone Height [m]	Area [m²]	Mass [kg]	Energy [J]
Small	0.11	0.11	0.10	0.10	0.13	0.42
Medium	0.15	0.15	0.13	0.24	0.24	1.04
Large	0.20	0.20	0.17	0.44		
Extra Large	0.25	0.25	0.22	0.68	0.66	2.90

There are two UAVs for each UAV size, for instance Mars and Venus have the same diameter. It would be preferable to have an interchangeability between these UAVs. Since the visualisation is done with e-paper, the image can change according to which planet the UAV represents.

4.2. Coating

As already mentioned before, the visualisation of the planets on the UAVs will not be done with the use of e-paper due to weight constraints. The alternative is to use a coating on the shell of the UAV. A simple representation of the actual planet with realistic (and not exaggerated) images is used. An indication of these representations can be found in Figure 4.2.

A base coating is applied on the parts of the structure of the UAVs that can not be completely covered in coating in detail, such as the the bottom structural ring and the mesh structure. The main colour of the coating is dependent on the main colour in the image that is used to visualise the planet. These main colors assigned to each planet can be found in Table 4.2. Note that for example Earth will use multiple colors to obtain a realistic representation, which can be seen in Figure 4.3b.

[a] E Ink Triton Data Sheet, URL `http://www.eink.com/sell_sheets/triton_spec_sheet.pdf`, accessed on 30 May 2016

[b] E Ink brings rich colour to ePaper, but not to e-readers, URL `http://techcrunch.com/2016/05/24/e-ink-brings-rich-color-to-epaper-but-not-to-e-readers/`, accessed on 30 May 2016

[c] E Ink Display Modules, URL `http://www.eink.com/modules.html`, accessed on 30 May 2016

[d] Plasticlogic Display Platform with Gate and Source Drivers, URL `http://www.plasticlogic.com/fileadmin/user_upload/Downloads/Product_brief/PlasticLogicGermany_PB60_15point4.pdf`, accessed on 30 May 2016

[e] Plasticlogic Display Module Datasheet 15.4", URL `http://www.plasticlogic.com/fileadmin/user_upload/Downloads/Display_datasheets/Display_Module_Datasheet_700238_700239_tiled_2-up_15_4inch.pdf`, accessed on 30 May 2016

[f] Planets - NASA Solar System Exploration, URL `http://solarsystem.nasa.gov/planets/`, accessed on 9 June 2016

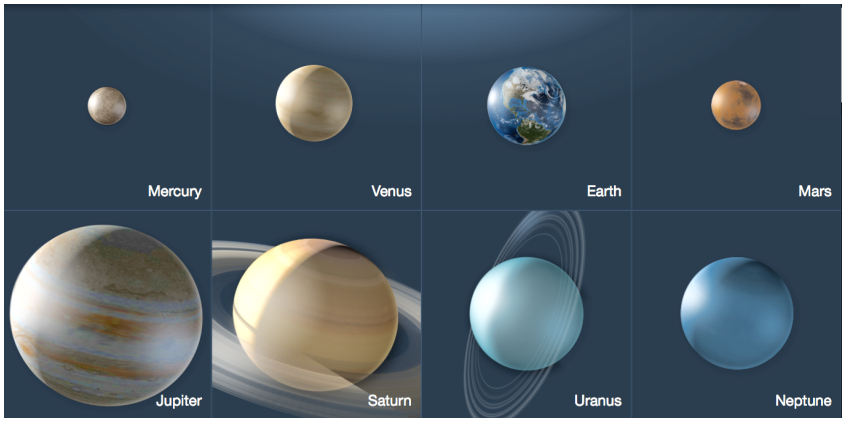


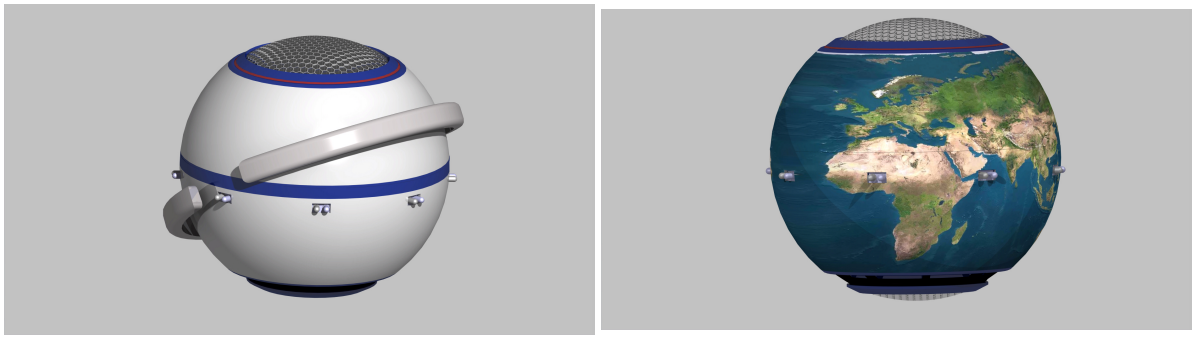
Figure 4.2: Graphics per Planet for Visualisation[f]

Table 4.2: Assigned Coating Colours for Each Planet

Planet	Color	Color Code	Color Desciption	Planet	Color	Color Code	Color Desciption
Mercury		#5E5144	Kabul Brown	Jupiter		#474B48	Nandor Grey
Venus		#695C46	Tabacco Brown	Saturn		#D2AC8F	Tan
Earth		#1F3046	Cloud Burst Blue	Uranus		#5C8694	Hoki Blue
Mars		#C78955	Antique Brass	Neptune		#1E3446	Cloud Burst Blue

In reality both Saturn and Uranus have a gas ring around them. However Saturn is the only planet that is commonly known for its rings. The rings of Uranus are also significantly smaller w.r.t. Saturn's rings. The outer radius of the rings of Uranus and Saturn is $97,700$ and $480,000$ km, respectively.[g,h] The relation between the radius of the ring and the radius of the planet is also significantly bigger for Saturn. For this reason, and the fact that adding the rings impedes the collision avoidance slightly, only Saturn is given a ring for the visualisation.

The rings are attached to the foam structure. For consistency in texture and exterior look, the rings will be cut out of the same foam. The foam shell of the extra large UAVs will be used for two different planets, for ease of production the rings will not be part of the foam shell but will be an extra part. This part will be attached to the foam shell with an adhesive. The rings will also be separated on the two hemispheres. This is illustrated in Figure 4.3a.



(a) The Ring attached on the Foam Shell of the UAV

(b) Coating Applied on the Earth UAV

Figure 4.3: Visualisation Aspects on both UAVs

[g]Uranus Rings Fact Sheet, URL `http://nssdc.gsfc.nasa.gov/planetary/factsheet/uranringfact.html`, accessed on 20 June 2016

[h]Saturn Ring Fact Sheet, URL `http://nssdc.gsfc.nasa.gov/planetary/factsheet/satringfact.html`, accessed on 20 June 2016

5 | Aerodynamics

In this chapter the aerodynamic performance of the UAV will be analysed. Propellers and motors that meet the specifications will be selected. In Section 5.1 the aerodynamic model used for the thrust determination will be described. Then, in Section 5.2, the requirements for the propellers will be defined and the propellers will be chosen. Finally, in Section 5.3, the motor selection process will be explained.

5.1. Aerodynamic Model

For the propulsion system, ANSYS CFX has been used to perform a computational fluid dynamics (CFD) analysis. For the simulation the propeller rotates at 9,000 revolutions per minute (RPM). The propeller is situated in air at a temperature of 25 $°C$ in a box that has an inlet and an outlet with a distance of 20 cm from the propeller and a pressure of $101,325\ Pa$ ($1\ atm$). The airflow is modelled as non-buoyant and there is no turbulence. The propeller is modelled to be GWS HD 4.5x4 using geometry from determined using PropellerScanner [9].[a] The sensitivity of the model has been tested by changing the boundary conditions and the time step. The model varies heavily for varying enclosure dimensions. Because the calculation scheme is done using the backwards Euler method, the time steps do not affect the convergence of the simulation. To validate the model the estimated thrust generated by the simulation was compared with the experimental data shown in Figure 5.1.

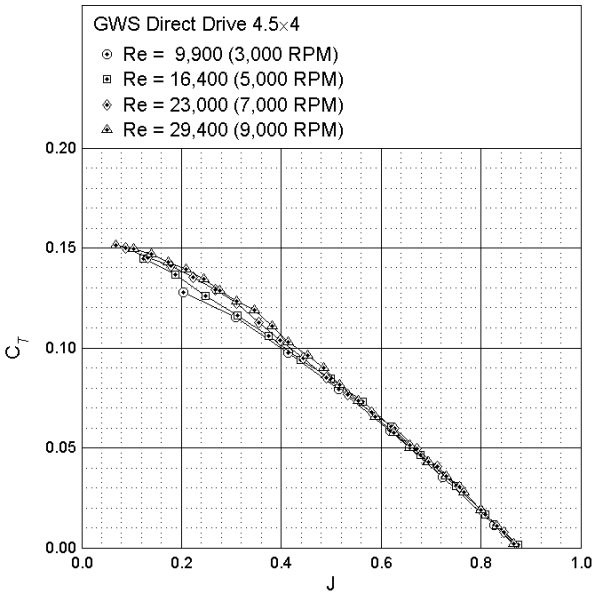


Figure 5.1: Thrust Coefficient Versus Advance Ratio of GWS HD 4.5x4 for Various RPM [9]

To determine the thrust coefficient the advance ratio was calculated with Equation (5.1) [12], with the free stream velocity $v_\infty = 1.0\ m/s$, the rotational speed of the propeller $n = \frac{9000\ rpm}{60\ s} = 150\ rev/s$ and propeller diamet $d_{prop} = 0.114\ m$. For an advance ratio of $J = 0.058$ the thrust coefficient will be $C_T \approx 0.15$ according to Figure 5.1. Then the thrust T can be determined with Equation (5.2) [12] using the same rotational speed n and propeller diameter d_{prop} and standard air density $\rho = 1.225\ kg/m^3$.

$$J = \frac{v_\infty}{n d_{prop}} = \frac{1.0}{150 \cdot 0.114} = 0.058 \tag{5.1}$$

$$T = C_T \rho n^2 d_{prop}^4 = 0.15 \cdot 1.225 \cdot (150)^2 \cdot (0.114)^4 = 0.70\ N \tag{5.2}$$

[a] Brandt, J. B., Deters, R. W., Ananda, G. K., and Selig, M., *UIUC Propeller Database*, University of Illinois at Urbana-Champaign, URL `http://m-selig.ae.illinois.edu/props/propDB.html`, accessed on 2 May 2016

The sum of the forces in flight direction in the CFD is 0.401 N with a rotational speed of 9,000 RPM, so the thrust generated in the CFD is off by 42.6%. The CFD model has therefore been deemed too inaccurate. Because increasing the accuracy would have taken too long, the effect of a co-axial system, a shaft and a mesh on thrust generation has been investigated using results from published research and experimental data.

5.1.1. Effect of Co-Axial Rotors on Thrust Generation

For co-axial rotors the upstream propeller generates about 55% of the total thrust generated by the coaxial configuration with a rotor spacing (the ratio between propeller distance H and radius r) of $H/r = 0.268$ [15]. The wake vortices generated by the top propeller can cause unsteadiness of the bottom propeller, which can cause fluctuation in the performance of both propellers. The upper propeller will become more steady with increasing rotor spacing, while the bottom propeller will initially become more steady, but the fluctuations will start to increase again when the rotor spacing becomes too big [15]. Therefore, the optimum spacing for the steadiness of the bottom propeller is $H/r = 0.446$ and the top propeller will then generate 57% of the total thrust [15]. For control it is important that the fluctuation in performance is as small as possible, and therefore a rotor spacing of 0.446 has been chosen.

For the smallest UAV, it is not possible to place the propellers that close to each other because of its small size. This close spacing would mean that the shaft takes up too much space, or the propeller diameter becomes too small. Therefore the propellers will be placed at the end of each shaft. To improve the performance of the bottom propeller a mesh will be placed between the two propellers to reduce the turbulence from the top propeller for the flow of the bottom propeller as explained later in Section 5.1.3. For simplicity, this has been done for all the UAVS. The resulting spacing can be found in Table 5.1.

Table 5.1: Rotor Spacing for every UAV Size. For the References for Propeller Radii see Table 5.3.

UAV Size	Distance [cm]	Radius Propeller [cm]	Rotor Spacing [H/r]
Small	12	6.5	1.85
Medium	19	8.89	2.14
Large	24	10.16	2.36
Extra Large	30	12.7	2.36

5.1.2. Effect of a Shaft on Thrust Generation

The shaft will be considered a shroud when the shaft properly encloses the rotors. The shroud will prevent the formation of vortices at the tips. This will improve the performance of the rotors by reducing the losses of thrust due to these tip vortices. The ratio between the thrust generated by a shrouded rotor and the thrust generated by an open rotor with the same power setting and same rotor disk area is given by Equation (5.3) [25]. With T_{SR} being the thrust generated by the shrouded rotor and T_{OR} the thrust generated by an open rotor.

$$\frac{T_{SR}}{T_{OR}} = (2\sigma_d)^{1/3} \tag{5.3}$$

The nozzle ratio $\sigma_d = \frac{A_e}{A_r}$, because the diffuser area A_e is equal to the disk area of the rotor A_r, $\sigma_d = 1$. The ratio between a shrouded rotor and open rotor will increase with a factor $\frac{T_{SR}}{T_{OR}} \approx 1.25$. The shaft will be made out of polypropylene homopolymer and 3D printed, just like most of the shell. The mesh will be attached to the shaft and the motors will be placed inside.

5.1.3. Effect of a Mesh on Thrust Generation

A mesh made out of a honeycomb structure can remove turbulence from the air when the length of the cell is six to eight times the cell diameter [19]. This is beneficial for the propeller performance as laminar flow will reduce the drag generated by the propellers, increasing the efficiency. The honeycomb should not allow fingers to come close to the propeller, therefore the diameter will be 0.75 cm. For laminar flow the honeycomb therefore needs at least a cell length of 4.5 cm. To have an as large as possible mass flow the cross section will be hexagonal.

The placement of the mesh between the two propellers will remove most of the interaction between the wake vortices. Therefore, it is assumed that both propellers will generate approximately the same

amount of thrust, so one propeller only has to generate about half the minimum required total thrust, so $\eta_{coax} = 0.5$.

The honeycomb structure is capable of following a simple curvature when the cells are over-expanded. The material used will be aramid fibre as this material has sufficient enough strength, has a low density, allows radio waves to pass through, dampens noise and is resistant to flames. [b,c]

5.2. Propeller Selection

The minimum thrust that has to be generated by one propeller depends on the mass of the current iteration plus a small margin to allow some growth in mass but also because the UAV should still be able to manoeuvre. For manoeuvring the difference between thrust and weight should be at least 0.1 N for the smallest UAV and at least 0.4 N for the largest UAV, as has been described in Chapter 11. Using the factors described in Section 5.1 an equation can be derived for the minimum thrust that has to be generated by one propeller T_{min}, see Equation (5.4). With η_{coax} and $\frac{T_{SR}}{T_{OR}}$ determined in Section 5.1.1 and Section 5.1.2 and m the current mass of the UAV.

$$T_{min} = SF\eta_{coax}\frac{T_{OR}}{T_{SR}}m \tag{5.4}$$

The required minimum thrust per propeller for every UAV size can be found in Table 5.2. To ensure the UAVs will a safety factor has been applied of $SF = 1.1$, so the thrust generated is at least 10% larger than the required thrust. The maximum propeller diameters (d_{prop}) that fit within the UAVs have been determined using Equation (5.5) and the radius of the UAV r_{UAV}.

$$d_{prop} = 2\sqrt{r_{UAV}^2 - \left(\frac{4}{5}r_{UAV}\right)^2} \tag{5.5}$$

Table 5.2: Minimum Thrust and Maximum Allowable Propeller Diameter for the Different UAVs with Their Estimated Masses

UAV Size	Mass [g]	Radius [m]	Minimum Thrust [g]	Maximum Propeller Diameter [m]
Small	1020	0.11	500	0.13
Medium	1320	0.15	715	0.18
Large	1640	0.20	935	0.24
Extra Large	2000	0.25	1100	0.30

To determine the thrust generated by one propeller at a certain rotational speed experimental data has been used as reference for designing the propellers in JavaProp.[d,e] The static thrust of the propellers have been measured using a static thrust stand and a digital scale.[f] The scale measures the downward force exerted by the propeller at different rotational speeds. After finding propellers that should be able to generate sufficient thrust, their dimensions, rotational speed and generated thrust were entered into JavaProp. All propellers use a Clark Y airfoil at $Re = 100,000$ and have a freestream velocity $v_\infty = 1.000\ m/s$. The results from JavaProp will only become inaccurate when the loading is too high, which is denoted with an exclamation mark.[e] None of the propellers had these warning. The chosen propellers can be found in Table 5.3. Their drag polar and distribution of the Reynolds and Mach number can be found in Appendix C.

[b] HexWeb A1 and A10 Datasheet, URL http://www.hexcel.com/Resources/DataSheets/Honeycomb-Data-Sheets/A1A10_eu.pdf, accessed on 13 June 2016

[c] PN2 Aerospace Grade Aramid Fiber Honeycomb, URL http://www.plascore.com/products/honeycomb-cores/aramid-fiber/pn2-aramid-fiber-honeycomb/, accessed on 13 June 2016

[d] Fly Brushless, URL http://www.flybrushless.com, accessed on 13 June 2016

[e] JavaProp - Design and Analysis of Propellers, URL http://www.mh-aerotools.de/airfoils/javaprop.htm, accessed on 13 June

[f] Test Stand Procedure, URL http://www.geocities.ws/dr.kiwitn/teststandprodcedure.html, accessed on 13 June 2016

Table 5.3: Propellers chosen for the Different UAVs with the Required Rotational Velocity and the generated Torque

UAV Size	Propeller	Diameter [m]	Thrust [g]	RPM [-]	Torque [Nm]
Small	Zagi Carbon 5.1x4.9 [g]	0.13	521	18900	0.02
Medium	APC E 7x6 [h]	0.1778	800	12600	0.03
Large	APC E 8x4 [i]	0.2032	936	11760	0.04
Extra Large	APC E 10x5 [j]	0.254	1330	10230	0.07

5.3. Motor Selection

The motors should be able to provide enough rotational speed for the propeller and be able to generate enough torque to counteract the torque generated by the propeller. Other selection criteria are the required power to generate the required torque and rotational speed, and the mass of the motor. The torque generated by the propeller has been determined using JavaProp and can be found in Table 5.3. The torque generated by the motor can be determined using Equation (5.6). The power out of the motor (P_{out}) can be determined using Equation (5.7), the power that goes into the motor (P_{in}), which is equal to the current i times the voltage V, and the efficiency of the motor η_{motor}. The rotational speed in RPM (rpm) can be converted to rotational speed in rad/s (ω) using Equation (5.8). Then the final equation for motor torque will be Equation (5.9).

$$Q = \frac{P_{out}}{\omega} \tag{5.6}$$

$$P_{out} = P_{in}\eta_{motor} = iV\eta_{motor} \tag{5.7}$$

$$\omega = \frac{rpm \cdot 2\pi}{60} \tag{5.8}$$

$$Q = \frac{iV\eta_{motor} \cdot 60}{rpm \cdot 2\pi} \tag{5.9}$$

The efficiency, current and voltage have been measured during the static thrust tests.[e] The motors that have been chosen for the different UAVs can be found in Table 5.4.

Table 5.4: Motors selected for the Different UAVs with their Rated Rotational Speed, Torque, Power and Mass

UAV Size	Motor	Rated RPM [-]	Maximum Torque [Nm]	Rated Power [W]	Mass [g]
Small	Suppo A2212-6 Gold [k]	18900	0.0635	167	51
Medium	KEDA TR2830/14 [l]	12600	0.116	219	59
Large	Justgofly 450T2 [m]	11760	0.121	213	63
Extra Large	Exceed Rocket 2215-3400 [n]	10230	0.192	257	63

The final configuration of the propulsion subsystem for each planet is given Table 5.5. The total thrust T_{total} was determined using Equation (5.10) and thrust of a single propeller T_{prop}.

$$T_{total} = \frac{T_{SR}}{T_{OR}} \cdot T_{prop} \frac{1}{\eta_{coax}} \frac{1}{SF} \tag{5.10}$$

[g] Zagi Carbon 5.1x4.9, URL `http://www.flybrushless.com/prop/view/89`, accessed on 9 June 2016
[h] APC E 7x6, URL `http://www.flybrushless.com/prop/view/208`, accessed on 9 June 2016
[i] APC E 8x4, URL `http://www.flybrushless.com/prop/view/123`, accessed on 9 June 2016
[j] APC E 10x5, URL `http://www.flybrushless.com/prop/view/40`, accessed on 9 June 2016
[k] Suppo A2212-6 Gold, URL `http://www.flybrushless.com/motor/view/50`, accessed on 9 June 2016
[l] KEDA TR2830/14, URL `http://www.flybrushless.com/motor/view/506`, accessed on 9 June 2016
[m] Justgofly 450T2, URL `http://www.flybrushless.com/motor/view/457`, accessed on 9 June 2016
[n] Exceed Rocket 2215-3400, URL `http://www.flybrushless.com/motor/view/672`, accessed on 9 June 2016

The smallest UAV has a difference between thrust and mass of 33.9 g, so the maximum control force during hover is 0.33 N and largest UAV has a difference of 647 g, so the largest UAV can generate a maximum control force of 6.34 N, therefore the subsystem meets requirement **Hor-Ctrl-21**. The UAVs are also capable of vertical take-off and landing, because the thrust generated is larger than the mass, thus meeting requirements **Hor-UAV-20** and **Hor-UAV-21**. All these components are commercially available.

Table 5.5: Final Configuration of the Propulsion Subsystem

UAV Size	Propeller	Motor	Total Power [W]	Total Mass [g]	Total Thrust [g]
Small	Zagi Carbon 5.1x4.9	Suppo A2212-6 Gold	334	108	1183
Medium	APC E 7x6	KEDA TR2830/14	438	124	1636
Large	APC E 8x4	Justgofly 450T2	426	132	1915
Extra Large	APC E 10x5	Exceed Rocket 2215-3400	514	132	2721

6 | Electrical Power System

The UAV power system has to be able to provide a continuous supply to each sub-system when in operation. In the Mid-Term Report it was decided that the UAVs of HORUS should charge autonomously. To incorporate this, it was chosen to use rechargeable Lithium-ion Polymer (Li-Po) batteries and an inductive charge system [1]. During the design it has been proven that an inductive charging system is infeasible due to the required current for charging, therefore the design has been changed to a conductive charge system.

A power budget is made for the different components in Section 6.1, for which the battery is sized in Section 6.2. The architecture of the electrical power system is described in Section 6.3. The final shape and position in the UAV is presented in Section 6.4. Lastly, Section 6.5 elaborates on the charging system.

6.1. Power Budget

The battery shall be sized based on the power required by all the sub-systems of the UAV. An initial estimate was made in the Mid-Term Report of 280 W [1]. In the detailed design phase the components of the sub-systems are chosen more specifically. Table 6.1 summarises the required power of the off-the-shelf products from each sub-system.

The initial power estimation differs greatly from the power budget, because the initial estimation was made on the basis of a statistical relationship between the power required by the UAV and the UAV mass, instead of taking the different components into consideration. Since the configuration of the designed UAVs differs greatly from other UAVs that are now on the market, this could be the main reason for the two estimates to differ this greatly.

Requirement **Hor-Ctrl-18** has not been met since the minimum power required is actually 339.2 W. This requirement was therefore a killer requirement to the design. It might be possible to re-evaluate the design in the future and to decrease the power required.

The power required by visualisation for one refresh of the e-paper is $15\ V \cdot 0.01\ A = 0.15\ W$. If more refreshes are required during flight this will increase the power budget. Due to requirement **Hor-UAV-22**, the visualisation sub-system with e-paper is discarded from the UAV design. For this reason it is also not included in the final power budget. The Inertial Measurement Unit (IMU) also uses a negligible amount of power and is not included in the final power budget. The main power consuming component of the UAV is the propulsion subsystem, because it requires high power motors. The total required power varies between the UAV sizes, since each UAV size uses different motors.

Table 6.1: The Power Budget for each UAV Size considering each Sub-System Power Component

	Voltage [V]	Current [A]	Amount of Components	Power [W] Small	Medium	Large	Extra Large
E-paper [a]	15.0	0.01	-	$0.15 \cdot N_r$ [b]	0.15	0.15	0.15
Raspberry Pi[c,d]	5.0	0.8	1	4.0	4.0	4.0	4.0
Internal Measurement Unit[e]	3.6	0.000145	1	0.000522	0.000522	0.000522	0.000522
Proximity sensor[f]	1.7	0.005					
Small			8	0.0680			
Medium			14		0.119		
Large			10			0.085	
Extra Large			16				0.136
Motor							
Small[g]	9.8	17.1	2	335.2			
Medium[h]	10.6	20.7	2		438.8		
Large[i]	10.6	20.1	2			426.1	
Extra Large[j]	10.6	24.3	2				515.2
Total Power				339.2	442.9	430.2	519.3

6.2. Battery Sizing

In the Mid-Term Report it was decided to use rechargeable Li-Po batteries to power the UAV, as they have a high energy density and charge rate [1]. This benefits the UAV design by having the possibility of lighter batteries and by reducing the time it takes to charge. In addition Li-Po offers more freedom in the shape of the battery. Different shapes are already on the market – cylindrical and prismatic –, but also custom shapes are offered by companies. To fit in each UAV the batteries will be shaped accordingly, which will be discussed in Section 7.6 Packaging. There is a range of Li-Po batteries on the market with different energy densities, capacities and C-rates. A comparison has been made in Table 6.2, where different battery options are summed up.

[a] E Ink TritonTM Imaging Film, URL `http://www.eink.com/sell_sheets/triton_spec_sheet.pdf`, accessed on 1 June 2016

[b] Dependent on the number of refreshes per cycle

[c] Raspberry Pi 2 has quad-core SoC, URL `http://hackerboards.com/raspberry-pi-gets-quad-core-soc-keeps-same-price/`, accessed on 9 June 2016

[d] Performance Testing the New Raspberry Pi 2, URL `https://www.linux.com/news/performance-testing-new-35-raspberry-pi-2`, accessed on 9 June 2016

[e] SN-IRS-01 infrared sensor, URL `http://www.cytron.com.my/p-sn-irs-01`, accessed on 10 June 2016

[f] SparkFun Triple Axis Accelerometer Breakout - ADXL345, URL `https://www.sparkfun.com/products/9836`, accessed on 10 June 2016

[g] Suppo A2212-6 Gold, URL `http://www.flybrushless.com/motor/view/50`, accessed on 9 June 2016

[h] KEDA TR2830/14, URL `http://www.flybrushless.com/motor/view/506`, accessed on 9 June 2016

[i] Justgofly 450T2, URL `http://www.flybrushless.com/motor/view/457`, accessed on 9 June 2016

[j] Exceed Rocket 2215-3400, URL `http://www.flybrushless.com/motor/view/672`, accessed on 9 June 2016

Table 6.2: Lithium-Ion Polymer Battery Type Characteristics

Battery Type	Discharge Voltage [V]	Discharge Current [A]	Charge Current [A]	Capacity [Wh]	Specific Energy [Wh/kg]	Energy Density [Wh/L]	Mass [kg]	Volume [L]
MultiStar 3S-5.2[k]	11.1	52	10.4	57.7	177.6	357.2	0.325	0.1616
Watson 3S-2.2[l]	11.1	44	2.2	24.4	138.8	138.8	0.176	0.0812
YUNEEC 3S-7.1[m]	11.1	71	35.5	78.8	186.8	273.4	0.422	0.2883
YUNEEC 3S-6.0[n]	11.1	60	30.0	66.6	160.1	223.3	0.416	0.2983
TP4500-6SHV[o]	22.8	27	9.0	102.6	213.3	388.0	0.481	0.2644
TP4500-3SHV[p]	11.4	27	40.5	51.3	209.4	382.1	0.245	0.1342
TP450-3SM70J[q]	11.1	31.5	5.4	5.0	113.5	241.7	0.044	0.0208
Tenergy 3S-5.0[r]	11.1	125	5.0	55.5	145.3	325.9	0.382	0.1703
Walkera 3S-5.2[s]	11.1	52	10.4	57.7	182.1	392.4	0.317	0.1471

Since the flight time of one UAV set will be shorter than one hour, a C-rate higher than one is desirable. In order to limit battery weight, the batteries need to provide a lot of current in a short time. The discharge current is derived from the capacity and the C-rate. HORUS will have two UAV sets, which means that the charge time of one set should be equal to or smaller than the flight time of one set. The charge current should thus be equal to or greater than the discharge current. It can be seen that the batteries of the manufacturer Thunder Power (TP) has the highest specific energy. Although the TP4500-6SHV battery has the highest specific energy, the charge current is significantly lower than the discharge current. TP4500-3SHV is a smaller version of the TP4500-6SHV with only 3 cells in series and has a much better charge current. The difference between the specific energy and energy density for both batteries is small enough to justify choosing the heavier battery. The dimensions of one cell is 33 by 36 by 113 mm. The discharge voltage and current of the battery are 11.1 V and 27 A, respectively.[p]
To size the battery, the required capacity of the battery is calculated with Equation (6.1) [16]. Here, C is the capacity of the battery in Wh, P_c the power required during a cycle and T_c the time duration of one cycle. DOD is the depth of discharge, dependent on the amount of cycles. Finally, η_{bat} is the battery-to-load transmission efficiency and N the amount of batteries.

$$C = \frac{P_c \cdot T_c}{DOD \cdot \eta_{bat} \cdot N_{bat}} \tag{6.1}$$

The battery-to-load transmission efficiency, η, is taken to be 0.9 [16]. With the initial decision to have HORUS running eight hours a day for thirty minutes, the amount of cycles per day is 8. For a lifespan of one year this amounts to 2912 cycles. From Figure 6.1 it can be seen that for this amount of cycles, the DOD can be met for every possible value. A DOD of 50% would mean that the capacity needed for the battery should be twice as big to account for the DOD. A DOD of 0.9 is chosen to ensure the UAV is never completely depleted of power and the batteries are not oversized. The batteries should then be replaced within 3.8 years, according to the number of cycles of approximately 8000 corresponding to a DOD of 0.9.

[k] MultiStar 3S-5.2, URL http://www.hobbyking.com/hobbyking/store/__56839__Multistar_High_Capacity_3S_5200mAh_Multi_Rotor_Lipo_Pack.html, accessed on 8 June 2016
[l] Watson 3S-2.2, URL http://www.bhphotovideo.com/c/product/1005779-REG/watson_rc_2220c3s_rc_lipo_battery_with.html, accessed on 8 June 2016
[m] YUNEEC 3S-7.1, URL http://www.venompower.com/products/yuneec-q500-series-3s-7100mah-11-1v-rc-drone-quadcopter-lipo-battery-by-venom, accessed on 8 June 2016
[n] YUNEEC 3S-6.0, URL http://www.venompower.com/products/yuneec-q500-series-3s-6000mah-11-1v-rc-drone-quadcopter-lipo-battery-by-venom, accessed on 8 June 2016
[o] TP4500-6SHV, URL http://www.thunderpowerrc.com/Products/4500-mAh/TP4500-6SHV, accessed on 8 June 2016
[p] TP4500-3SHV, URL http://www.thunderpowerrc.com/Products/4500-mAh/TP4500-3SHV, accessed on 8 June 2016
[q] TP450-3SM70J, URL http://www.thunderpowerrc.com/Products/450magna/TP450-3SM70J, accessed on 8 June 2016
[r] Tenergy 3S-5.0, URL http://www.all-battery.com/111volt-5000mah25cli-polylipobatterypack31289.aspx, accessed on 8 June 2016
[s] Walkera 3S-5.2, URL http://www.hobbyking.com/hobbyking/store/__53878__Walkera_5200mAh_3S_Lipoly_Battery_Pack_for_QR_X350_PRO.html, accessed on 8 June 2016

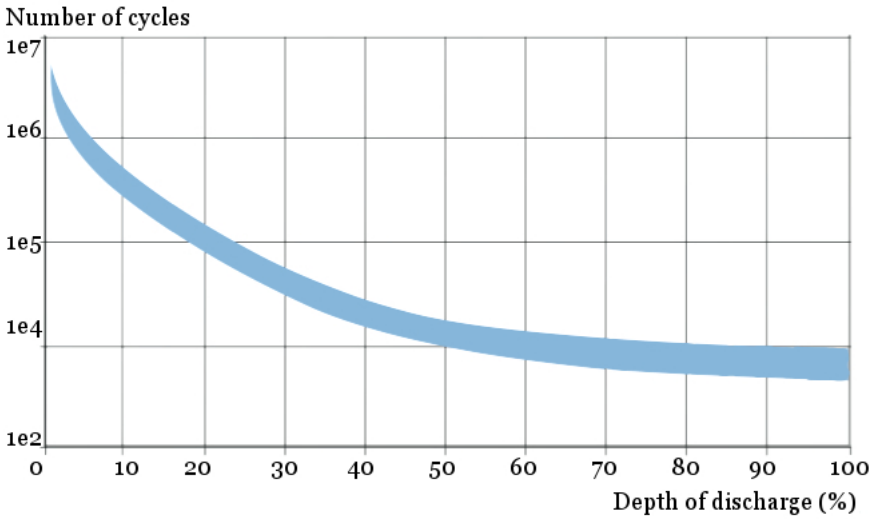


Figure 6.1: Cycle Life vs. Depth of Discharge at 25°C for Energy Applications at 70% Capacity at End-of-Life [t]

Initially, one UAV set of HORUS was sized to fly for 30 minutes, but this could not be accomplished. With an initial power estimate of 280 W and a cycle time of 30 minutes the battery mass would be 826 gram in total. With the new power budget for the small UAV of 339.2 W and a reduced cycle time of 10 minutes a complete battery mass of 410 g is theoretically feasible in relation to the specific energy. Unfortunately Li-Po batteries allow for a lot of flexibility with the cell shape but not with the capacity and voltage of one cell.[u,v] To have enough capacity it is necessary to have at least two batteries with a total capacity of 102.6 Wh for the smallest UAV, which is also preferable for redundancy. The excess in capacity is used to prolong the cycle time from 10 to 14.7 minutes. The mass of the batteries is 490 g in total. The battery size and mass of the other UAV sizes can be found in Table 6.3. An additional battery type is also chosen for the extra large UAV size because the battery capacity of the TP4500-3SHV is not high enough. The TP450-3SM70J provides the additional capacity, which results in a total battery weight of 535 g.

The mass of the cabling is estimated as a percentage of the total unit sub-system mass. The cabling that interconnects subsystems is 10-25% of the electrical-power system's mass [16].

Both Thunder Power batteries have a charge operational temperature range of 10.0 °C to 37.8 °C and a discharge operational temperature range of 10.0 °C to 54.4 °C. With this, requirement **Hor-Ctrl-16**, to have the battery not exceed 60 °C, is met.

The charge time is calculated for each separate battery, since they will be placed parallel to each other so that they can be charged simultaneously. The charge current implies the speed at which can be charged with the TP4500-32HV. This corresponds to 4.5 Ah/40.5 A = 0.11 hr = 6.7 min. As can be seen in Table 6.3 the combined charge time does not exceed 6.7 minutes for all the UAV sizes. Since this is smaller than the UAV cycle time the HORUS can be continuously operational with two sets of UAVs. With a cycle time of 15 minutes the extra down time can be used to let the batteries cool down before charging. The battery status will be monitored by the Raspberry Pi.[w] The status of the batteries can be communicated with the server and posssibly a supervisor. In combination with the second UAV set the HORUS can perform continuous operations for eight hours a day for a minimum of one year. Requirement **Hor-9** is hereby met. In addition requirement **Hor-Gnd-7** is also met because the charge time of all the UAVs

[t] Lithium-ion battery life, URL `www.saftbatteries.com/.../li_ion_battery_life__TechnicalSheet_en_0514`, accessed on 2 June 2016

[u] Custom Made Pack Formats, URL `http://www.all-battery.com/custommadepackforms.aspx`, accessed on 3 June 2016

[v] Design Considerations Using Lithium Polymer Batteries, URL `http://www.wirelessdesignmag.com/article/2010/08/design-considerations-using-lithium-polymer-batteries`, accessed on 10 June 2016

[w] Controlled shutdown duration test of Pi model A with 2 cell lipo, URL `http://raspi.tv/2013/controlled-shutdown-duration-test-of-pi-model-a-with-2-cell-lipo`, accessed on 14 June 2016

is lower than the discharge time. Charging at a high C-rate does have its disadvantages. In combination with chosing a DOD of 0.9, the battery life will be decreased. A good rule of thumb would be to replace the batteries after one year.

The Li-Po batteries are very sensitive to over- and undercharging. This means that the battery charging needs to be carefully monitored. The charging method is split into two phases: constant current and constant charging. The current is kept constant until the battery cell voltage reaches 4.1 V. After that the current is gradually diminished until it has a value of 0.1C and then charging is stopped. If the battery is still connected to the charger, the battery is periodically replenished to counteract the battery self-discharge. Each battery will feature a internal monitor specifically selected for single battery charging. The Thunder Power batteries have a specific charger, the TP820HC, which will be incorporated into the charge station.[x]

Table 6.3: Battery Size and Cycle Time for each UAV Size

UAV Size	Battery Type	Number of Batteries	Total Capacity [Wh]	Total Mass [g]	Dimensions per Battery [mm]	Cycle Time [min]	Charging Time [min]
Small	TP4500-3SHV	2	102.6	490	$33x36x113\ mm$	14.7	6.7
Medium	TP4500-3SHV	2	102.6	490	$33x36x113\ mm$	11.3	6.7
Large	TP4500-3SHV	2	102.6	490	$33x36x113\ mm$	11.6	6.7
Extra Large	TP4500-3SHV	2	102.6	490	$33x36x113\ mm$	10.6	6.7
	TP450-3SM70J	1	5.0	44.0	$13x30x53\ mm$	10.6	5.0

6.3. Electrical Architecture

In this section, the electrical architecture of the UAVs will be discussed. The single line diagram for the smallest UAV is depicted in Figure 6.2. As shown in the figure, the two batteries are placed in parallel so they can be charged simultaneously and the output current is increased. The three batteries of the extra large UAV are also placed in parallel. All components are placed before a resistor to lower the voltage and current to that required for the component as stated in Table 6.1. The motors and the number of proximity sensors changes for each UAV size and so the magnitude of the component resistor differs for each UAV size as can be seen in Table 6.4. Eight proximity sensors are placed in parallel with the micro-computer as can be seen in Figure 6.2. This parallel system has extra resistors to compensate for the difference in voltage and current. An extra connection is made from the proximity sensors to the micro-computer as an input. Different components are also connected directly to the micro-computer: the IMU and the four photo resistors used for landing. As mentioned in Section 6.1 the power consumption of the IMU is neglected. The photo resistors will not require any power.

The minimum current the batteries must supply is 31.5 A, which is lower than the nominal discharge current of 54 A (27 A per battery). The capacity of the batteries is then depleted at a lower rate because of the lower current. Therefore the run-time of the UAVs can be increased to 15.3 minutes for the small, medium and large UAVs and to 18.5 minutes for the large UAV. For an optimal functionality of HORUS, continuity is one of the main drivers. It would be preferable to have all the UAVs of one set switch to charging at the same time, which is possible because all the UAVs have a very similar discharge time. The run time of each UAV is set at 15 minutes, the discharge time of 0.3 minutes can be used as an overlap in run time of the both sets. The first set will start flying to the charge station after a run time of 15.15 minutes, while the second set starts flying to the position in orbit. In case the flight time of the UAV to get back to the charge station is higher than 0.15 minutes, the rest of the power can be drawn from the 10% that is left in the battery.

[x] TP820HVC, URL `http://www.thunderpowerrc.com/Products/Chargers_2/TP820HVC`, accessed on 14 June 2016

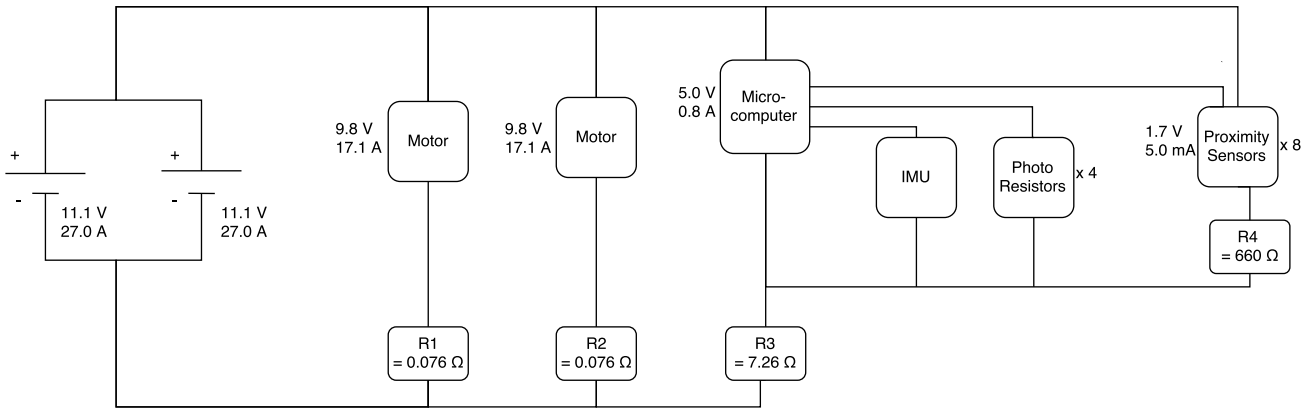


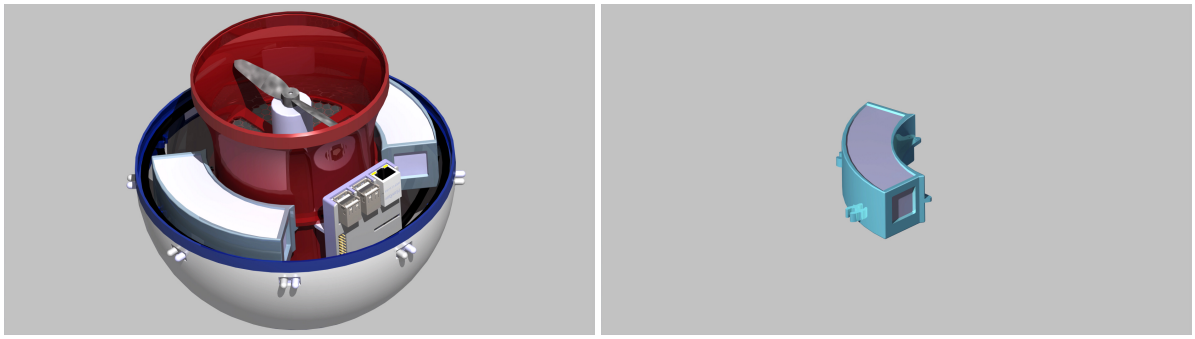
Figure 6.2: Electrical Architecture of the Smallest UAV

Table 6.4: Magnitude of Resistors for each Sub-System Component and for each UAV Size

Components	Resistor number	Small	Medium	Large	Extra Large
Motor	R1, R2	0.076 Ω	0.024 Ω	0.025 Ω	0.021 Ω
Micro-computer	R3	7.26 Ω	7.01 Ω	7.18 Ω	6.93 Ω
Proximity Sensor	R4	660 Ω	660 Ω	660 Ω	660 Ω

6.4. Battery Configuration and Shape

As mentioned before the Li-Po batteries can be subjected to different curvatures and can be manufactured in custom shapes. To fit the UAV design the batteries will be shaped according to the curvature of the shaft and shell, while maintaining the same volume. In Figure 6.3 the curvature and the location of the batteries can be noted. For the largest UAV, one extra battery will be present. In Section 7.6 the connection of the battery to the rest of the UAV is elaborated on.



(a) Battery Placement Inside the UAV **(b)** Battery Shape

Figure 6.3: Battery Shape and Configuration inside the UAV

6.5. Charge System

Due to the lower safety risk, the choice was made to use inductive charging to charge the batteries. Firstly, the weight of this system is determined. This weight turns out to be too high to meet **Hor-UAV-22**, so the design is changed to conductive charging. Finally, the lay-out and the mass of this final system is discussed.

6.5.1. Inductive Charging

Inductive charging works by transferring energy through a changing magnetic field. This principle is shown in Figure 6.4.

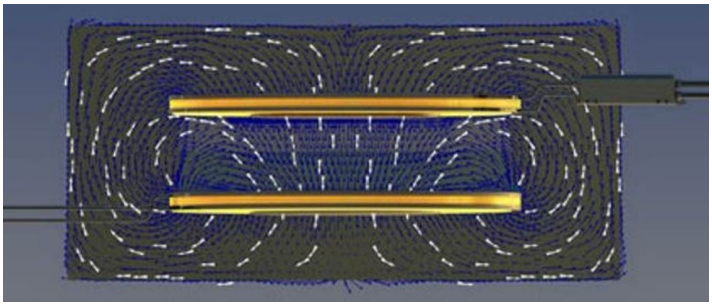


Figure 6.4: The Magnetic Field Distribution in the Air Core Transformer [2]

The batteries will need a voltage of 11.4 V to charge, and a current of 40.5 A. In order to limit the heat generated by the inductors, a flat spiral is used, which has the additional advantage of a lower separation between the two coils of the charging system. From [2], an indicative value of 200 μH has been taken for the inductor in the UAV. For the smallest UAV this will mean a flat coil with 25 turns of copper wire.[y]

Because the wire will need to carry 40 A of current, it will need to have a diameter of 2.5 mm [28]. This will lead to a mass of 1200 g, as calculated using Equation (6.2). In this equation, ρ is the density of copper (8.96 g/cm^3), r the radius of the wire, N the number of turns and d the diameter of the turns, which is on average 13 cm for the smallest UAV. Because the mass of the inductive charging system is too high, it has been decided that inductive charging will not be used.

$$m = \rho \cdot \pi r^2 \cdot N \cdot \pi d \tag{6.2}$$

6.5.2. Conductive Charging

The next best option is then conductive charging, which supplies the current directly to the batteries. As already mentioned in Section 6.2, the charging op the Li-Po batteries is a very sensitive process. To monitor the battery charging, an integrated circuit (IC) should be incorporated. The Thunder Power batteries have a specific charger, the TP820HC, which will be incorporated into the charge station.[z] This means the electrical block diagram of the system will look like Figure 6.5.

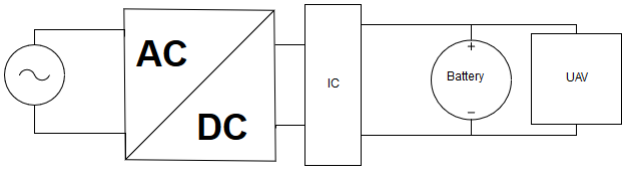
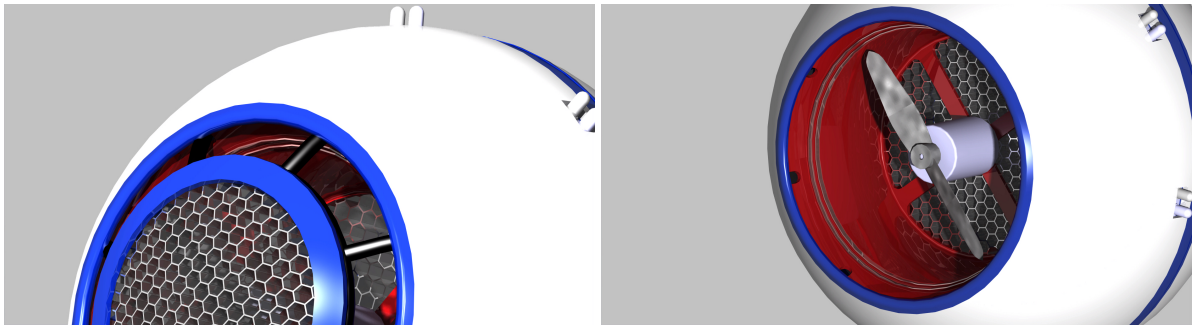


Figure 6.5: The Electrical Block Diagram of the Charging System.

The charge system will function through the use of two magnetic, conducting rings at the inside of the shaft. One of these rings will be attached to the positive poles of the batteries, and the other to the negative poles. This will mean the angle at which the UAV lands will not have to be very accurate. The rings are located close to the underside of the UAV, as shown in Figure 6.6. The charging station will have matching plates, and has an additional structure to provide stability to the UAVs. In Figure 6.6a the rings are shown with the mesh in place. The matching connectors from the Charge Station fit in the space between the mesh and the shaft. Figure 6.6b shows the rings more clearly. The two copper rings will add a weight of 17 g to the smallest UAVs, as calculated with Equation (6.2). The radius of the rings is approximately 65 mm, while the area is the same as copper wire with a diameter of 1.25 mm. This is half of what would be needed or inductive charging, as the two batteries can be charged in parallel and therefore the wire needs to carry only the current for one battery. The medium UAV will have a charging system that will weight 26 g, and the larger UAVs will have only a partial ring to charge. This will mean the UAV will need to land at a certain angle to the charge station, but that can be achieved using the

[y] Flat Spiral Coil Calculator, URL `http://www.pronine.ca/spiralcoil.htm`, accessed on 11 June 2016
[z] TP820HVC, URL `http://www.thunderpowerrc.com/Products/Chargers_2/TP820HVC`, accessed on 14 June 2016

positioning system. These partial rings will have a mass of 30 g and will be only the left-most and right-most parts of the ring and have a total surface area comparable to the medium UAV. The largest UAV will have the additional battery attached in parallel to one of the other UAVs.



(a) With the Mesh **(b)** Without the Mesh

Figure 6.6: The Charging Rings Inside the Shaft

7 | Structure

This chapter is dedicated to the structure of the UAV. In Section 7.1 different load cases are analysed and the design load is determined. Then, in Section 7.2 the comparison between materials for the optimised design are stated. In Section 7.3 the internal structure optimisation is performed. The manufacturing will be considered in Section 7.4. After, the shell structure will be described in Section 7.5. Lastly, in Section 7.6 the packaging will be discussed, which includes the method to connect the structure to other elements is discussed and the way all elements are put inside the UAV.

7.1. Load Cases

The largest load case for the structure is the impact force exerted on the UAV when it falls on the ground due to for instance propeller failure. In flight forces are estimated not to exceed 30 N. For this situation, the most critical fall is a fall from a height of 5 meters, the upper limit of the flight envelope. Two situations are analysed to determine the most critical loads. In the first situation the UAV has no horizontal velocity and falls straight down. The second situation incorporates an initial horizontal velocity. Both situations are depicted below, with the accompanying forces.

For this analysis the UAV is considered a point mass. The initial velocity in this situation is set at the maximum velocity that is acceptable for the UAVs. The surface area, S, is taken as the frontal area of the UAV and the C_d is set at 0.55 [20].

During impact with the floor, the kinetic energy is transferred into potential energy, see Equation (7.1) and Equation (7.2). The assumption is made that the UAV will not bounce up again once it has hit the ground. The velocity of the UAV at impact is v. The floor is modelled as a rigid surface and is assumed not to indent upon impact. However, an absorption factor of 15% is applied. The assumption is made that the structure of the UAV will indent, $dh = 1\ cm$. The deceleration a will determine the impact force applied onto the structure of the UAV.

$$E_{kin} = 0.5 \cdot m \cdot v^2 \tag{7.1}$$

$$E_{pot} = m \cdot a \cdot dh \tag{7.2}$$

In which,

$$a = \frac{T - D}{m} \tag{7.3}$$

$$D = \frac{1}{2} \cdot \rho \cdot v^2 \cdot S \cdot C_d \tag{7.4}$$

$$S = \pi \cdot r^2 \tag{7.5}$$

$$v = \int a dt \tag{7.6}$$

For the situation where an initial horizontal velocity is applied, the assumption is made that when the UAV hits the ground, the UAV will displace 1 mm and will not roll. Making use of Equation (7.1) and Equation (7.2), the horizontal impact force can be determined. Table 7.1 shows the impact forces for both situations for the four different UAVs.

Table 7.1: Load Cases for the Various UAV Sizes. Situation 1 is a Free Fall from a Height of 5 Meters. Situation 2 is a Free Fall from 5 Meters whereby the UAV has an Initial Horizontal Velocity of 1 m/s.

Radius [m]	Mass [kg]	F_{sit1} [kN]	F_{sit2} [kN]	F_{design} [kN]
0.10	1.3	5.32	5.35	5.89
0.15	1.5	5.92	5.95	6.55
0.20	1.7	6.41	6.46	7.26
0.25	2.0	7.26	7.31	8.04

The largest load occurs when the UAV has an initial horizontal velocity and falls straight down. The design forces are set by multiplying the highest occurring force for each size of UAV and applying a safety factor of 1.1. This safety factor is chosen to ensure structural integrity. The design loads can be found in Table 7.1, and range from 5.89 kN to 8.04 kN.

7.2. Materials

For the load carrying structure, different materials are analysed and compared. These materials and the relevant properties can be found in Table 7.2. A material study was performed in an earlier phase of the project, keeping the scope of the materials big. The materials in Table 7.2 were selected on the basis of the preferred material properties. The density is an important factor in the UAV design, to keep the mass of the UAV below 2 kg. Other properties such as yield strength, Young's modulus and Poisson ratio have also been considered.

Table 7.2: Properties of Materials considered for the Load Carrying Structure

Material	Density [g/cm³]	Yield Stength [MPa]	Young's Modulus [GPa]	Poisson Ratio [-]	Cost [$/kg]
Al 7075 [5]	2.8	503	71	0.33	9.7
HM Carbon Fibre-epoxy matrix [5]	1.6	570*	70	0.25	275
E-glass Fibre-epoxy matrix [5]	1.9	1020*	25	0.19	22
Nylon 6/6 [5]	1.4	55.1	3.8	0.39	6 (raw)
Polypropylene [5]	0.905	31.0	1.55	0.42	1.65
Kevlar Fabric [a,b]	1.4	190*	30	0.36	88

* For composites, instead of Yield Strength, the Ultimate Tensile Strength is taken in the direction of the fibres.

7.3. Internal Structure Design

The internal structure of the UAV should withstand all the load cases and still allow for accessibility and positioning of the internal components. As determined in the previous phase of the project, the load carrying structure will consist of a horizontal ring at the middle, a horizontal ring around the mesh and semi-circles connecting these [1]. The lay-out of the internal structure can be found in Figure 7.1. This is the design of the smallest planet. However, the design of the other planets is comparable to this.

[a] Mechanical Properties of Carbon Fibre Composite Materials, URL http://www.performance-composites.com/carbonfibre/mechanicalproperties_2.asp, accessed on 10 June 2016

[b] ArmorCO, URL http://www.armorco.com/shop/item.aspx?itemid=215, accessed on 10 June 2016

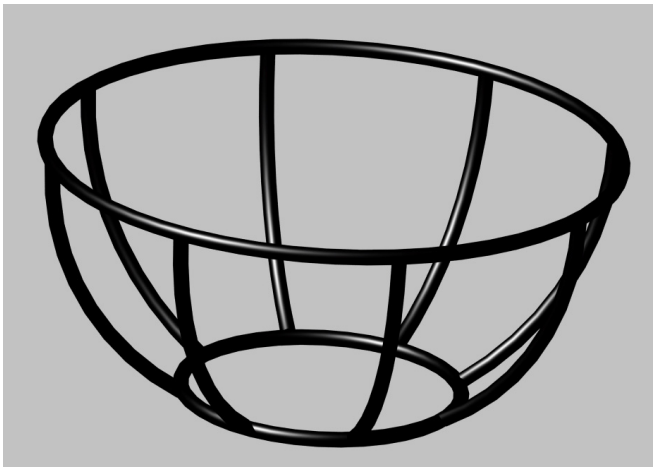


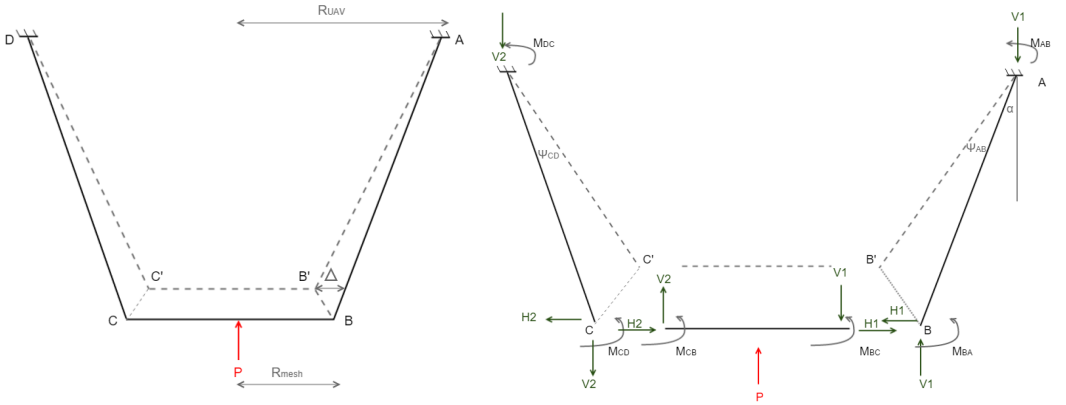
Figure 7.1: Lay-Out of the Internal Structure for the Smallest UAV

As stated in Section 7.1, the design loads range from 5.89 kN to 8.04 kN. The internal structure is designed such that when the UAV falls on the bottom, the internal structure can handle the loads and will not fail. The structure will be optimised in the following sections with four parameters: cross-section shape, cross-section dimensions, number of beams and material. The materials that are compared can be found in Table 7.2.

7.3.1. Method

To analyse the different combinations to find a optimal configuration, MATLAB is used. A simplified model is used to perform the optimisation. This simplified structure can be seen in Figure 7.2. In this figure the horizontal beam represents the mesh ring. The two diagonal beams represent the beams connecting the small circular ring at the bottom with the circular ring at the middle. The simplified model uses straight beams. The diagonal beams have clamped boundary conditions (zero deflection and zero rotation at clamped positions), to represent the attachment to the middle ring.

This simplified structure is statically indeterminate. To analyse a statically indeterminate structure, the displacement method is used [13]. For the analysis, a rigid frame is used, indicated by continuous lines in the figure. The deformed structure is indicated by dotted lines.



(a) Simplified Structure

(b) Simplified Structure Internal Moments and Angles

Figure 7.2: Simplified Structure used for Internal Structure Optimisation

In the theory applied to this problem, the following assumptions are made:

- BB' is perpendicular to AB and CC' is perpendicular to DC.

- Due to deformation AB rotates by an angle ψ_{AB}. Similarly, CD rotates by an angle ψ_{CD}. Deformation is counter-clockwise direction is assumed positive.
- During deformation bar CB stays horizontal.
- The ends of the diagonal beams have clamped boundary conditions and cannot rotate, $\theta_A = \theta_D = 0$

The angles over which beam AB and CD rotate during deformation, are given by,

$$\psi_{AB} = \frac{BB'}{L_{AB}} = -\frac{\Delta}{L_{AB}cos(\alpha)} \tag{7.7}$$

$$\psi_{CD} = \frac{\Delta}{L_{BC}cos(\alpha)} \tag{7.8}$$

The equations for the internal moments are as follows

$$M_{AB} = \frac{2EI}{L_{AB}}[\theta_B - 3\psi_{AB}] \tag{7.9}$$

$$M_{BA} = \frac{2EI}{L_{AB}}[2\theta_B - 3\psi_{AB}] \tag{7.10}$$

$$M_{BC} = M_{BC}^F + \frac{2EI}{L_{BC}}[2\theta_B + \theta_C] \tag{7.11}$$

$$M_{CB} = M_{CB}^F + \frac{2EI}{L_{BC}}[2\theta_C + \theta_B] \tag{7.12}$$

$$M_{CD} = \frac{2EI}{L_{CD}}[2\theta_C - 3\psi_{CD}] \tag{7.13}$$

$$M_{DC} = \frac{2EI}{L_{CD}}[\theta_C - 3\psi_{CD}] \tag{7.14}$$

and by approximation,

$$r_{mesh} = \frac{r_{UAV}}{2} \tag{7.15}$$

In which the L_{AB} is the total length of beam AB etc, and,

$$M_{BC}^F = -M_{CB}^F = \frac{P \cdot r_{mesh}^3}{2 \cdot r_{mesh}^2} \tag{7.16}$$

Due to equilibrium of joints B and C,

$$M_{BA} + M_{AB} = 0 \tag{7.17}$$

$$M_{CB} + M_{BC} = 0 \tag{7.18}$$

Considering beam AB and taking moments around the cut,

$$h \cdot H_1 - M_{AB} - M_{BA} + V_1 \cdot r_{mesh} \tag{7.19}$$

$$h \cdot H_2 - M_{CD} - M_{DC} + V_2 \cdot r_{mesh} \tag{7.20}$$

In which h is the vertical distance between A and B or B and C.

$$\sum F_x = 0 = H_1 + H_2 \tag{7.21}$$

$$\sum F_y = 0 = V_1 - V_2 - P \tag{7.22}$$

Now summing Equation (7.19) and Equation (7.20), using Equation (7.21) and Equation (7.22) and rewriting gives,

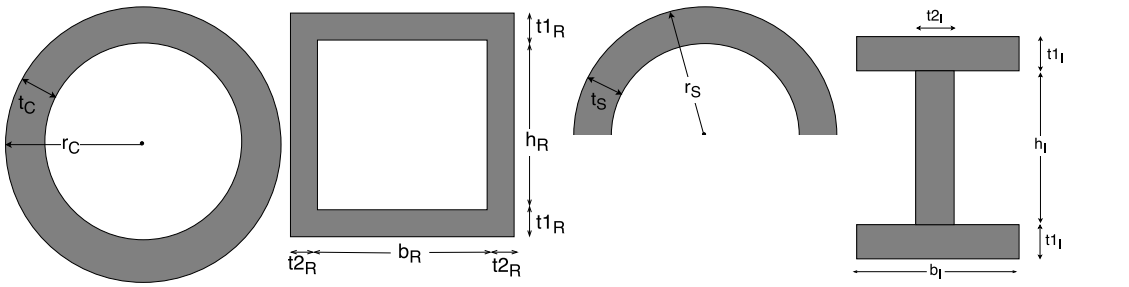
$$-M_{AB} - M_{BA} + M_{BC} + M_{CB} - M_{CD} - M_{DC} = 0 \tag{7.23}$$

When looking at Equation (7.17), Equation (7.18) and Equation (7.23), there are three equations and three unknowns θ_B, θ_C and Δ. This can be solved and these variables can be substituted back in the moment equations. Now considering the maximum moment and substituting this in Equation (7.24) gives the maximum stress in the material.

$$\sigma = \frac{M_{max} y}{I} \tag{7.24}$$

7.3.2. Cross-Section Design Choice

First of all, the shape of the cross-section is chosen. Four different shapes are analysed, namely a (hollow) circle, (hollow) rectangle, semi-circle and I-beam, see Figure 7.3.



(a) Hollow Circular Cross-Section **(b)** Hollow Rectangular Cross-Section **(c)** Semi-Circular Cross-Section **(d)** I-Beam Cross-Section

Figure 7.3: Cross-Sections analysed for Internal Structure Optimisation

In Table 7.3 the analysed range for the different variables can be found. The corresponding variables can also be seen in Figure 7.3. These ranges have been set considering practical issues. Too small dimensions would give handling problems, and would have greater impact on the surface the UAVs would be falling on. On the other hand, too large dimensions would require too much space in the UAV, not allowing for optimal packing. Furthermore, the number of beams is varied between 2 and 20, with a step size of 2. The program is written in such a way that solid circles and rectangles are also taken into account, which can be noted when looking at the boundaries of r_C, t_C, h_R and b_R.

Table 7.3: Range of Variables used for the Optimisation of the Internal Structure

Variable	Description	Range [m] (Step Size [m])
	Cross-Section: Hollow Circle	
r_C	Middle Radius	0.0015 - 0.006 (0.001)
t_C	Thickness	0.0015 - 0.0024 (0.001)
	Cross-Section: I-Beam	
h_I	Web Length	0.01 - 0.015 (0.005)
b_I	Flange Length	0.01 - 0.02 (0.005)
$t1_I$	Flange Thickness	0.0015 - 0.0035 (0.001)
$t2_I$	Web Thickness	0.0015 - 0.0025 (0.001)
	Cross-Section: Hollow Rectangle	
h_R	Web Length	0.0 - 0.015 (0.005)
b_R	Flange Length	0.0 - 0.02 (0.005)
$t1_R$	Flange Thickness	0.0015 - 0.0035 (0.001)
$t2_R$	Web Thickness	0.0015 - 0.0025 (0.001)
	Cross-Section: Semi Circle	
r_S	Middle Radius	0.0015 - 0.013 (0.001)
t_S	Thickness	0.0015 - 0.0035 (0.001)

Table 7.4 shows the results for an aluminium structure. These are representative for all the materials. From this, it follows that the circular cross-section gives the lightest structure for all sizes, except for the extra large UAV. However, if a different cross-section is chosen for this UAV, different manufacturing techniques might be required than for the rest of the UAVs, or even a different material. This would bring extra costs. Thus, for convenience of manufacturing, all UAVs will have the same cross-sectional shape. The circular shape is also more beneficial for the attachment of the inner components. Therefore, the structure will have a circular cross-section, and will be further optimised for this shape.

Table 7.4: The Mass of an Aluminium Internal Structure for Different Planet Sizes and Cross-Section Geometries

	Cross-Section Geometry			
UAV Size	I-beam	Rectangular	**Circular**	Semi-Circular
Small	198 g	264 g	103 g	118 g
Medium	297 g	396 g	197 g	239 g
Large	396 g	528 g	324 g	-
Extra Large	495 g	660 g	561 g	-

7.3.3. Structure Optimisation

With the shape of the cross-section chosen, the structure will be optimised for the cross-section dimensions, number of beams and materials. The cross-section is represented by two parameters, the radius and the thickness, as defined in Figure 7.3a. The MATLAB program takes the load and planet radius as inputs. It will then compare all the configurations that can be made for the defined ranges and materials. MATLAB calculates the maximum stress by solving the system of equations as explained above. This stress is compared with the yield stress of the respective material, and if it is lower, it is stored. Then, from all the combinations of beams and dimensions that can hold the load, the most lightweight option is chosen. This is done for each material. The most lightweight configuration per planet size and material can be found in Table 7.5.

Table 7.5: Iteration Results for the Hollow Circular Beams

Material	n_{Beam} [-]	Radius [mm]	Thickness [mm]	Mass [g]
Size: Small				
Carbon Fibre	**8**	**1.6**	**2.4**	**69**
Aluminium	8	2.2	1.5	103
Glass Fibre	6	2.7	1.5	76
Nylon	8	5.2	1.5	122
Kevlar	8	3.5	1.5	82.2
Polypropylene	16	6	1.5	133
Size: Medium				
Carbon Fibre	**14**	**1.5**	**2.4**	**131**
Aluminium	10	2.5	1.5	197
Glass Fibre	10	2.7	1.5	144
Nylon	10	6	1.5	236
Kevlar	10	4	1.5	158
Polypropylene	-	-	-	-
Size: Large				
Carbon Fibre	**10**	**2.3**	**2.4**	**221**
Carbon Fibre	14	1.9	2.4	221
Aluminium	12	2.8	1.5	324
Glass Fibre	12	3.0	1.5	236
Nylon	16	5.7	1.5	393
Kevlar	10	4.8	1.5	252
Polypropylene	-	-	-	-
Size: Extra Large				
Carbon Fibre	**8**	**3.0**	**2.4**	**322**
Aluminium	12	2.9	2	561
Glass Fibre	14	2.9	2	417
Nylon	20	6	1.5	601
Kevlar	10	5.6	1.5	367
Polypropylene	-	-	-	-

It can be seen that for Polypropylene there are no feasible design options for the medium, large and extra large sized UAVs. Therefore, Polypropylene is not considered a feasible design option. Furthermore, one can clearly see that the carbon fibre provides the best results for all UAVs and is thus chosen as the material used for the load carrying structure. For the large UAV there are two possible lightweight configurations using carbon fibre. The option with 10 beams and radius of 2.3 *mm* is chosen because this is more convenient for packaging. It has been decided that for every beam the same dimensions are used, even though the stress is not evenly distributed. For example, the horizontal ring at the centre of the UAV carries less loads compared with the smaller circular circle. However, this optimised configuration already has the minimum producible thickness. Also, when the radius would be decreased, the beams would become too fragile.

There are some limitations to the use of carbon fibre. First, the minimum thickness used for structural components is 2.4 *mm*, this is thus used as the minimum thickness in the MATLAB model used to iterate the dimension of the structure.[c] Furthermore, the corners in a carbon fibre body are the weak-points of the structure and should thus be manufactured with care. This is further discussed in Section 7.4.

7.3.4. Verification and Validation

As stated before, a MATLAB program has been used for the optimisation of the internal structure. To ensure that it gives reliable results, the program has been verified and validated. The procedure for this is discussed in this section.

[c] Standard Thicknesses, URL `http://www.protechcomposites.com/standard-thicknesses/`, accessed on 10 June 2016

Verification of the program has been done in two steps: firstly, individual units have been tested, after which a whole system test has been performed. The program units are: moment of inertia I, maximum moment M_{max}, maximum stress σ_{max} and weight W. The moment of inertia, maximum stress and weight are verified by comparing the result to manual calculations. Also, unit inputs have been used to quickly spot faults in the outcome. MATLAB solves a system of equations to calculate the maximum moment. The solution is inserted back into the equations, to verify that the solution complies with the equation. Furthermore, the system has been tried with zero input, to ensure that this gives zero stress as an output. For the system test, the units have been added one by one. It has been ensured that no variables have been given the same symbol, and that the units were placed in the right order. The loops have been tested by making lists of one unit length, and printing all the variables to see if the values are correct. It has also been checked that when a zero force is applied, if the resultant stress is zero.

To validate the program, the geometry for the smallest UAV has been created in CATIA. As a constraint, the upper ring is fixed, and a distributed load is applied on the lower ring (around the mesh). CATIA could then calculate the von Mises stresses on the structure. This can be seen in Figure 7.4. The highest stress given by the MATLAB program is 534 MPa, while the highest stress resulting from the stress analysis in CATIA is 409 MPa. This is a difference of 23%. This is within acceptable and expected margins, because a simplified 2D model is compared to a detailed 3D model. The two methods thus give reasonable results, with the MATLAB program being slightly higher. This yields a conservative result, and thus means that the internal structure is over designed. The MATLAB program is therefore suitable for the design of the internal structure.

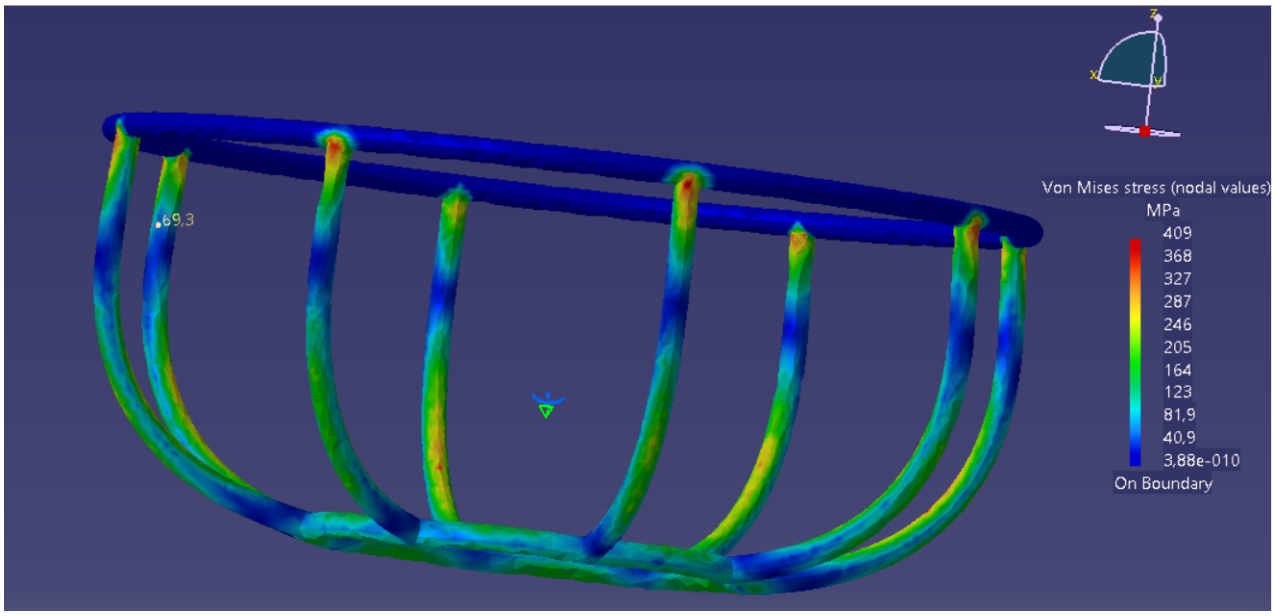


Figure 7.4: Static Analysis in CATIA of the Internal Structure

Having verified and validated the internal structure, it can now be concluded that the UAVs are scalable and crash resistant. Therefore, requirements **Hor-UAV-1** and **Hor-UAV-2** are met.

Due to the fact that structure is over-designed, it can be optimised even further. The parts of the internal structure carrying less critical loads can have optimised geometries. For example, taper can be introduced in the beams. However, the production techniques of carbon fibre will have to be further developed to allow for smaller thicknesses. Furthermore, the geometries could be adjusted as such, that the UAV is allowed to fall on one of the semi-circular beams or upper circular ring, which makes the structure even more reliable.

7.4. Manufacturing

As mentioned earlier in Section 7.3.3, during manufacturing of the internal structure, attention should be paid at the places where the beams intersect. Because of the fragile structure and the small radius, it cannot be easily made out of one piece. When made out of one piece, either the cross-section should be solid or an internal mould should be used that cannot be removed. Both options add weight to the internal structure and are hard to manufacture.

For these reasons, it is chosen to manufacture the beams separately and later attach them to each other. The carbon fibres are made wet and wound around a mould. As a mould a small stick is often used, such as the carbon rods used for the DelFly Micro.[d] When the fibres are put in place, they are impregnated with resin. Multiple layers can be build up in this way which are then put in vacuum to hold the layers together. The vacuum removes the air inside and impregnates the fibres with the epoxy in which it is covered. When the resin is dry, the mould can be removed and a (bended) tube with hollow cross-section is realised. This can be executed relatively easily by a machine.

To attach the beams to each other, carbon fibre tow is wound around the beams, in a way as shown in Figure 7.5a. After that, the resin is dripped on this intersection to reinforce it [8]. When this is dry, the connection is strong enough to handle the design loads. The carbon fibre tow can also be formed into a more flat shape, resulting in a more sheet-like tow, which would resemble a tape. An example of such a junction is shown in Figure 7.5b. Prepregging these sheets and then carefully laying it around the junction, the junction should become rigid and strong when it is dry.

(a) A Junction of Carbon Fibre Tubes by Means of Carbon Fibre Tow [e]

(b) A Junction of Carbon Fibre Tubes by Means of Carbon Fibre Sheets [f]

Figure 7.5: Attachment of the Carbon Fibre Beams

If it is found that the above approach does not give satisfying results during the test phase of the UAV, the carbon fibre beams can also be joined together by making use of carbon fibre connectors.[g] If none connectors can be found in the size that is required, custom connectors can be manufactured. The beams are then glued to this connector, and special care has to be taken when choosing this glue, so that the appropriate loads can be handled.

7.5. Shell Structure

Considering the UAV lay-out, the shell is merely a part of the UAV that provides a surface for the visualisation and closes off the UAV, making it safe for the user and giving it the spherical shape as required by the stakeholders in **Hor-UAV-8**. The shell will be designed such that it is not a load carrying structure and is made up of two components, both hollow hemispheres with a cut out to allow for airflow. The material choice made for the shell is based on Table 7.6.

[d] DelFly Micro, URL `http://www.delfly.nl/micro.html`, accessed on 15 June 2016
[e] Calfee Factory Tour, URL `http://www.bikerumor.com/2013/06/07/calfee-factory-tour-part-one-how-they-make-bikes/`, accessed on 17 June 2016
[f] Factory Tour Carbon Fibre Bikes, URL `http://www.bikerumor.com/2012/04/12/factory-tour-parlee-cycles-part-1-inside-look-at-how-they-make-custom-carbon-fiber-bikes/`, accessed on 17 June 2016
[g] Carbon Fibre Connectors, URL `https://goodwinds.com/connectors.html`, accessed on 17 June 2016

Table 7.6: A Selection of Materials considered for the Shell

Material	Density [kg/m^3]	Tensile Strength [MPa]
Aluminium [h]	26800	325
Nylon 12 [i]	1020	50
Polyurethane Foam [j, k]	48	1.05
Carbon Fibre [5]	1600	570
Polypropylene Homopolymer [l]	913	33

The shell is equal for each UAV size, varying only in size and number of collision avoidance sensors. All the shells have a thickness of 3 *mm*. To ensure the mass of the shell is as light as possible, polyurethane foam is chosen as the main material. As the foam is quite brittle, each shell hemisphere will consist of a plastic ring at each end of the foam. Plastic is included to provide and maintain the shape of the shell. The hemispheres are connected to each other by screwing them together. The lower hemisphere is connected to the load carrying structure by means of clips. The upper plastic of the lower hemisphere also houses the collision avoidance sensors for the drone.

The foam is light, so in the event that the shell hits the floor/person under it first, it will cause minimal to no damage to its environment. Recently, 3D printing of foams was introduced to the world. As it is a new concept, testing is being performed to analyse the properties of the foams. The assumption is made that when production of the system commences, the 3D printing foam technology will have advanced to the stage that it is commercially available.

The shell should be as soft as a playground surface or softer. The most common playground surface material is rubber. Since polyurethane foam has a lower hardness than rubber [22] [26], requirement **Hor-UAV-7** is met.

7.6. Packaging

In this section the packaging of the shell and the connections of the UAV systems to the internal structure will be presented. The packaging makes sure that all the systems fit within the UAV and are connected to form a coherent UAV. The packaging of all the required flight components within the UAVs spherical shape is essential, as the spherical shape required in **Hor-UAV-8** must be met. The overall packaging is presented in Figure 7.6, excluding the top hemisphere and top mesh, to show where all the systems are placed.

[h] Aluminium, URL `http://asm.matweb.com/search/SpecificMaterial.asp?bassnum=MA5052H19`, accessed on 16 June 2016

[i] Nylon 12, URL `https://www.smithmetal.com/pdf/plastics/nylon-12.pdf`, accessed on 16 June 2016

[j] Low Density Polyurethane Foam Block, URL `https://system.na1.netsuite.com/core/media/media.nl?id=5263&c=3937524&h=bd892aaaca96e75c3c50&_xt=.pdf`, accessed on 16 June 2016

[k] High Density Polyurethane Foam Block, URL `https://system.na1.netsuite.com/core/media/media.nl?id=5308&c=3937524&h=ba9b48253791ba39a75e&_xt=.pdf`, accessed on 16 June 2016

[l] Polypropylene Homopolymer, URL `http://sterlingplasticsinc.com/materials/polypropylene-homopolymer-pph/`, accessed on 16 June 2016

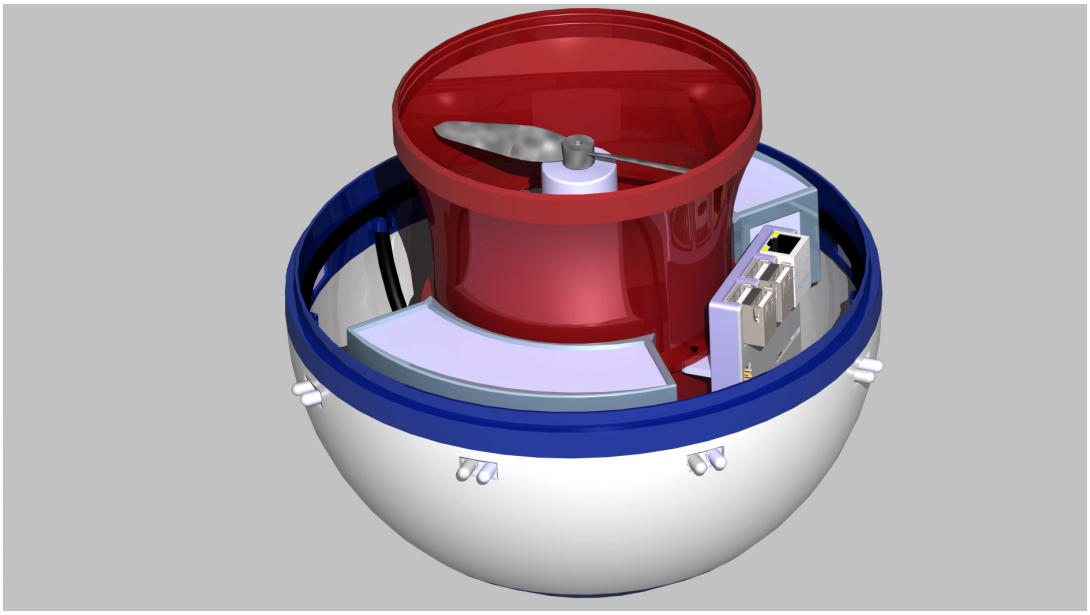
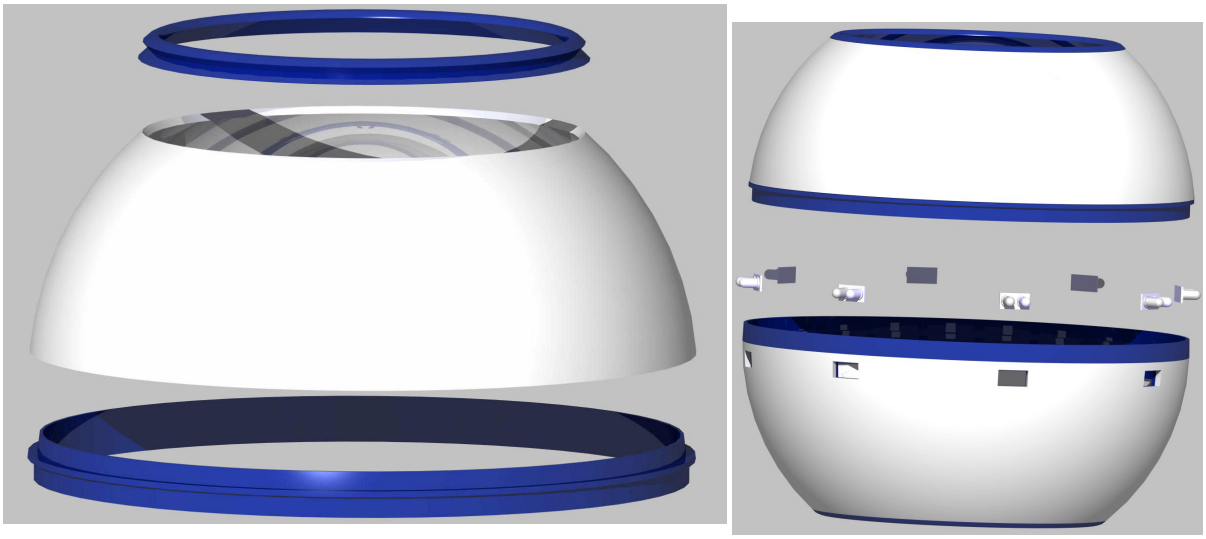


Figure 7.6: Inside View showing Packaging of UAV

First, the shell connections will be discussed, followed by the battery and Raspberry Pi connections. After these connections, the shaft connections and the housing of the propulsion system will be presented. Lastly, the mesh connections will be presented.

The shell consists of two hemispheres of which each hemisphere consists of three separate parts. The hemispheres are two plastic components and a polyurethane low density foam component as presented in Figure 7.7a, which are glued together. The two hemispheres are connected with a male thread on the upper hemisphere and a female thread on the lower hemisphere, see Figure 7.7b. The lower hemisphere houses the collision avoidance sensors in the plastic, with holes inserted into the foam for them to protrude through the shell as can be seen in Figure 7.6. The plastic also contains eight clips with which the shell can be connected to the eight beams of the internal structure, depicted in Figure 7.8.



(a) Hemisphere consisting of two Plastic Components and a Polyurethane Component

(b) Hemisphere and Sensor Connections

Figure 7.7: Hemisphere Connection

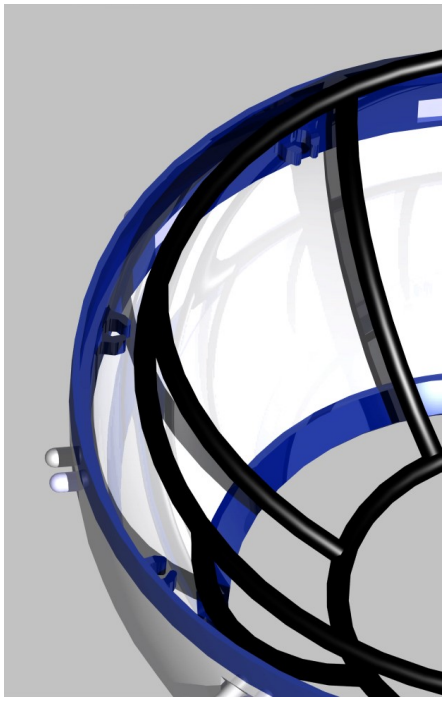
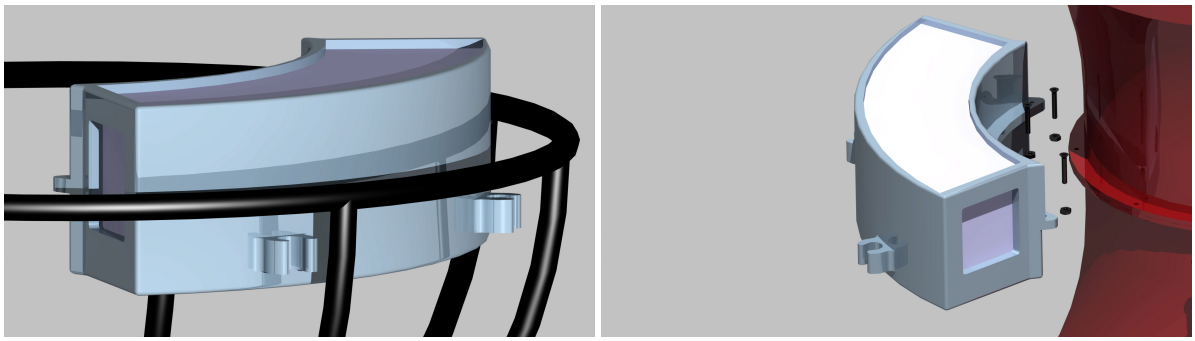


Figure 7.8: Shell Connection to Load Carrying Structure

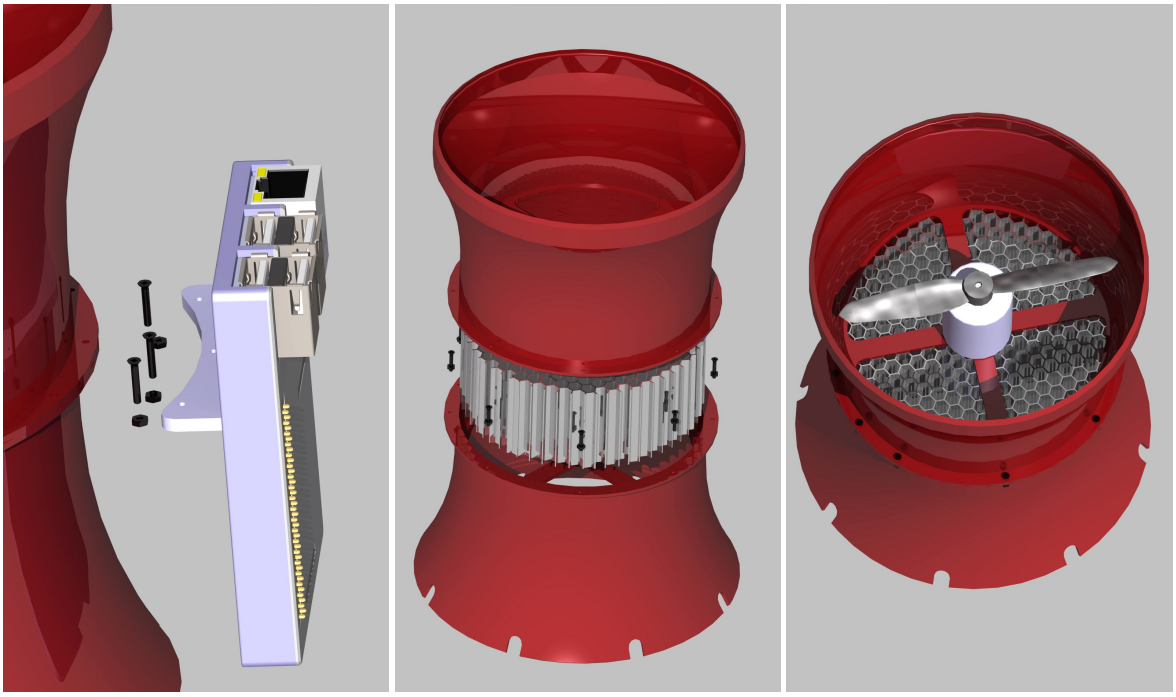
The battery is housed in a plastic casing, which is connected to the internal structure by two clips, similar to the shell connections with the internal structure. Furthermore, the batteries connected by three M3 bolts to the shaft. The two batteries will be connected to the shaft in a mirror image of each other, so that the centre of gravity is approximately on the centre line of the UAV. The Raspberry Pi micro-computer is placed in a plastic casing, which can be connected to the shaft by means of three M3 bolts as can be seen in Figure 7.10a. Furthermore, rubber spacers will be placed between the shaft and connecting systems to damp the vibrations generated by the propulsion system.



(a) Battery to Load Carrying Structure Connection **(b)** Battery to Shaft Connection

Figure 7.9: Housing of the Batteries

As depicted in Figure 7.10b, the shaft consists of two parts, which are connected to each other making use of ten M3 bolts, eight of which are placed every 45° and two extra to connect the micro-computer. The shaft houses the inner mesh between the two motors and propellers, held in place on either side by a cross configuration within the shaft. The cross configuration also provides mounting points for the propulsion system placed on the top and bottom of the mesh as can be seen in Figure 7.10c.



(a) Raspberry Pi Connection to Shaft

(b) Shaft Connections

(c) Motor to Shaft Connection

Figure 7.10: Housing of the Motors and Propellers

The shaft is connected to the internal structure in two ways. The first is by clipping the batteries into place, which are bolted to the shaft, a combination of the connections presented in Figure 7.9. The second way, to keep the shaft straight, is clipping it to the internal structure just above the bottom ring, as shown in Figure 7.11.

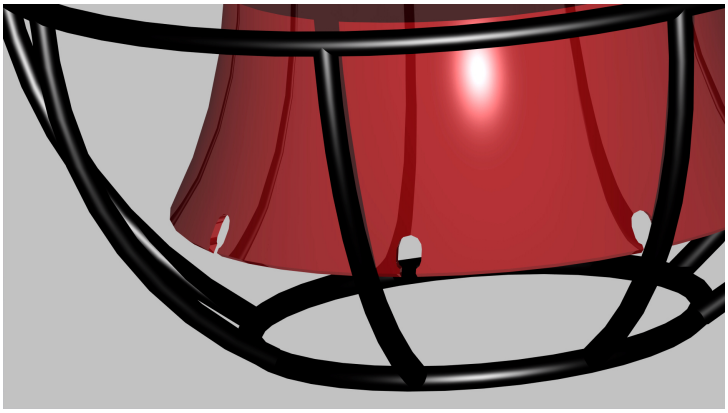
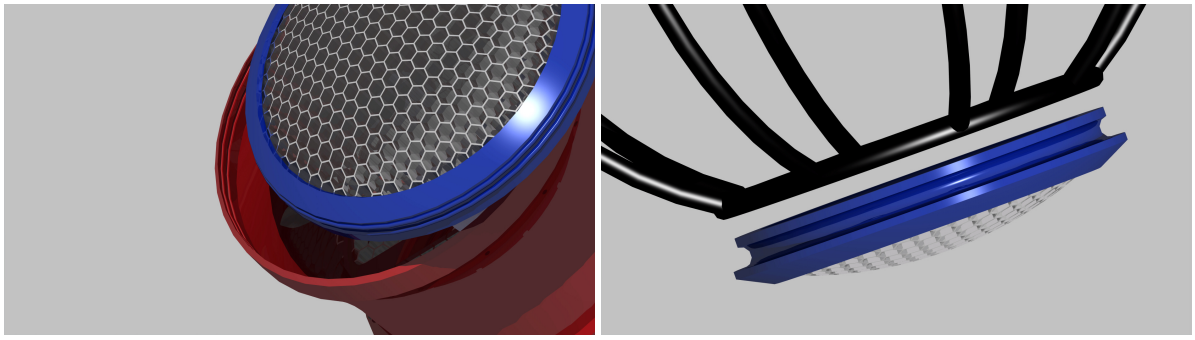


Figure 7.11: Shaft to Load Carrying Structure Connection

The top mesh is connected to the shaft by the use of a custom made connection mechanism shown in Figure 7.12a. Two ridges are present on the top mesh, with two corresponding groves within the shaft. The mechanism is created to hold the mesh in place within the shaft. The distance between the mesh and the propulsion mechanism is 10 *mm*.

The bottom mesh is connected to the internal structure, with a similar connection mechanism as the top mesh. However, a grove is present in the bottom mesh and the ring of the internal structure functions as the ridge. The ridge extends the diameter of the ring, so that when a collision occurs due to a fall the load is transferred from the mesh to the internal structure.



(a) Top Mesh Connection **(b)** Bottom Mesh Connection

Figure 7.12: Mesh Connections

The packaging presented is for the smallest UAV, as it is the most difficult to package due to the small volume available for all the systems. All connection methods presented are scalable to the larger UAVs. Packaging wise the shell, shaft, mesh, propellers and internal structure scale up with the increasing size of the UAVs. The batteries, motors and Raspberry Pi do not vary in size over the UAVs, which makes packaging easier for the larger UAVs. The batteries and Raspberry Pi shall be placed as low as possible, while still connecting to the shaft and internal structure. Finally, in Figure 7.13 an exploded view of the design is depicted.

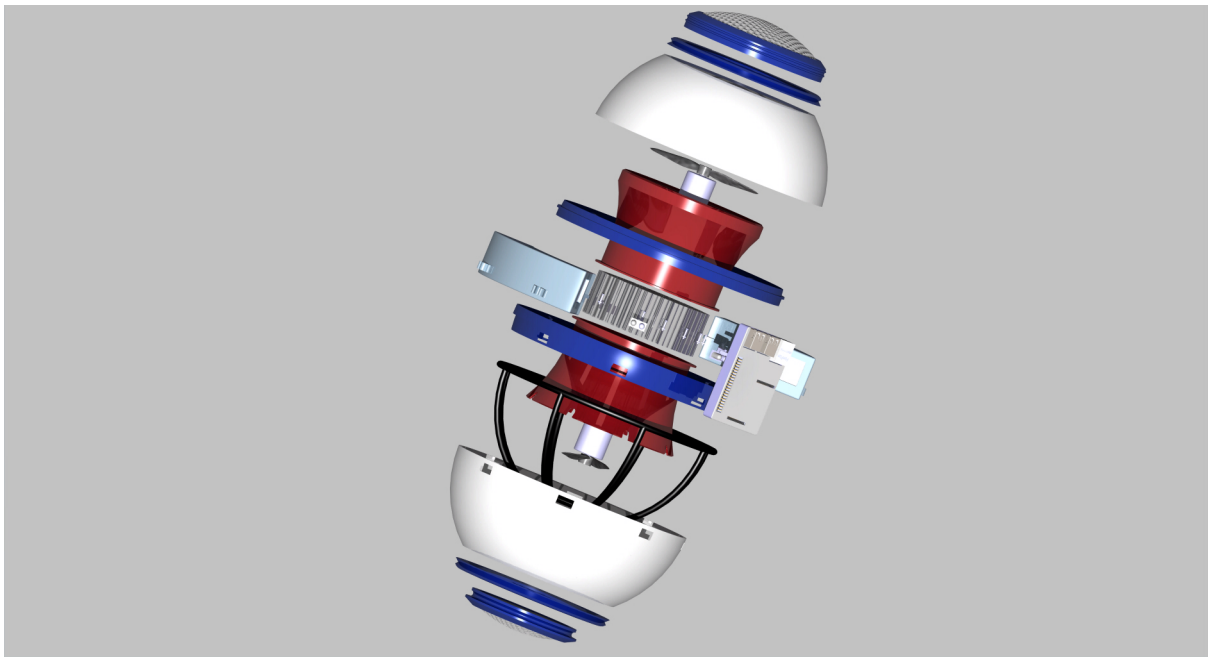


Figure 7.13: Exploded View of the Design

8 | Mass Estimation

The mass of the UAVs is an important aspect of the design. **Hor-UAV-22** states that any UAV can weigh a maximum of 2 kg. The mass of each system that make up the UAV have been collected and are presented in Table 8.1 and is listed for each size of UAV.

Table 8.1: Mass Estimation for each Size UAV

System	**Small** [g]	**Medium** [g]	**Large** [g]	**Extra Large** [g]
Prop	6	6	6	6
Motor	102	118	126	126
Battery	560	570	575	634
Visualisation	110	240	420	660
Structure	69	131	221	322
Rasp pi	45	45	45	45
Shaft + Mesh	159	201	374	250
Shell	94	130	206	428
Sensors	7	8	8	56
Wires	96.2	96.2	96.2	130.0
Charging system	17	26	30	30
Total	1265.2	1571.2	2107.2	2687.0

As can be seen in the table, the mass of the large and extra large UAV do not comply with the set mass requirement, exceeding the requirement by 4.9% and 34.1%, respectively. Also, the thrust that each UAV can produce is exceeded. Table 8.2 presents the available thrust for each UAV size in the first row, showing that the small and large UAV are not able to produce enough lift to take-off. This is due to the fact that for these two UAVs the thrust is lower then the mass showed in the table above. In order to reduce the mass of the UAVs and to make sure enough thrust is provided to fly, the decision was made to remove the visualisation from all the UAVs, as this is one of the less essential systems and can be replaced by adding a coating to the shell of the UAV. The final mass of the UAVs is also presented in Table 8.2.

Table 8.2: Final Mass Estimation for each Size UAV

	Small	**Medium**	**Large**	**Extra Large**
Thrust [g]	1183	1636	1915	2721
Initial Total [g]	1265.2	1571.2	2107.2	2687
Final Mass [g]	1155.2	1331.2	1687.2	2027.0

The final masses for all the UAVs, except the Extra Large, are below the 2 kg requirement. The Extra Large exceeds the requirement by 1.1%. Although the mass has exceeded the set limit, **Hor-UAV-22** is deemed as a fulfilled requirement as the percentage thrust available for the UAV is significantly higher than the exceeded mass.

IV

Communication & Control

9 | Communications

In this chapter the communication method used in the system is discussed. Initially, the data handling is discussed and the required amount of bandwidth is determined in Section 9.1. Following this, the chosen hardware which will facilitate this data handling and communication is mentioned and its benefits are listed in Section 9.2. In Section 9.3 the Communication Flow Diagram can be found.

9.1. Data Handling

There are two subjects which will be treated in this section. The first is the data flow throughout the system, followed by the determination of the required bandwidth for the communication model.

9.1.1. Bit Assignment

In this section first the number of bits that are needed to communicate to the UAVs are determined. In order to facilitate this, the transmitted information is divided into classes. All items within a class will have the same number of bits and each class is assigned an identifier to facilitate the calculation. The number assigned to each of the orbit modes, parameter determination mode and UAVs can easily be represented by binary numbers.

Modes

For each of the modes a binary number consisting of 3 bits is used to identify it, as there are eight unique modes the UAV can be set in. Later on in this section, when the data flow is discussed into more detail, parameters transmitted of this type and thus size, will be followed by an (M) for mode.

ID

Because there will be two sets of eight UAVs, so sixteen required unique numbers in total, a binary number of 4 bits can be used to transmit the ID. However, since it is desired to have groups of UAVs be callable with a single ID, one extra digit is added to add this functionality. This leads to a total of 5 bits. Parameters of this type will be identified by an (ID), for identity.

Custom Values

In order to specify the use of custom parameter values, only 1 bit is required. This data type identifier is (C), for custom.

Parameter

In addition to specifying custom parameters, a method has been implemented which allows an admin to switch the mode which is used to calculate the orbital parameters semi-major axis, eccentricity and inclination. Since no more than three different modes are provided for these parameters, only three bits are required to uniquely identify each mode. Data from this class is represented by a (P), for parameter.

Admin Overwrite

The last switch which can be toggled by an admin is the Admin Overwrite Mode. Because this is a switch, it will also only require 1 bit to identify the desired state. This data is identifiable by a (O), for overwrite.

Non-Integer Value

The method used to transmit the non-integer numbers over Bluetooth will be as follows. A binary number consisting of 16 bits will be used to represent each number. The first bit indicates the sign of the number. The following five represent the value of the number using binary code. The final ten are used to represent the decimals of the number, the first bit in this list has a value of a half, the second of quarter, the third of an eighth and so forth, to the tenth bit which will have a value of slightly less than 0.001, providing the system with the accuracy to transmit values accurate up to a millimeter. The maximum magnitude value the system can receive is equal to 31.999. These parameters will be identified by the

(*F*), for float, following their mention in the section to come. These parameter types, their identifier and bit value are summarised in Table 9.1.

Table 9.1: Type, Identifier and Bit Value of the Parameters Transferred over the Bluetooth Connection

Class	Identifier	Number of bits
Orbital Mode	M	3
Unit Identifier	ID	5
Use Custom Parameter	C	1
Parameter Mode	P	2
Use Admin Override	O	1
Non-integer Value	F	16

9.1.2. Data Stream

The data stream starts with a user (observer, supervisor or admin) setting the parameters on the User Interface (UI) to their desired value. From the UI the inputs are transferred internally to the ground station server and will be send out via Bluetooth signal to all the UAVs, complying with requirement **Hor-Ctrl-15**.

These parameters are *ID of the selected UAV* (ID), *admin overwrite marker* (O), *custom Parameters marker* (C), *selected orbital mode* (M), *maximum orbit size* (F), *maximum orbital speed* (F), *inclination scale* (F), *semi-major mode* (P), *eccentricity mode* (P), *velocity mode* (P) and *mean orbital altitude* (F). In addition to all these parameters, the Orbit Mode Selector also receives a signal from the UAVs dictating the State-of-Charge from the batteries. Using all these parameters, the Mode Selector will determine the semi-major axis, inclination, eccentricity, orbital velocity and mean orbital altitude. Additionally, if the active mode is not an orbit mode but a ”Go-To” mode such as the charge mode, it will provide a number of way point coordinates to the desired position the UAV can follow. The semi-major axis, inclination, eccentricity and orbital velocity will be send to the Orbital Pathfinding system, or the way point coordinates will be send to the directional Pathfinding system, dependent on the desired result.

In order to determine the required change in position, the current position of the UAV has to be determined. This is done by receiving a signal from five Bluetooth Beacons. This Bluetooth signal will only contain the beacon ID as data, as this is the only information required to determine the position besides the signal strength. With five beacons to identify, an identifier size of 3 bits is required for this signal.

With all this information the Orbital Pathfinder or the Directional Pathfinder determines the magnitude and direction of the force the UAV has to produce in order to follow the desired path.Additionally the pathfinders determine the velocity and acceleration of the UAV. The desired force is then sent to the Control Software which converts it into a Pulse Width Modulation (PWM) signal. This PWM signal finally is sent to the propeller controls.

The final Bluetooth transfer is the parameters the UAV sends to the ground station to be displayed on the UI. These parameters are *ID of the selected UAV* (ID), *current selected orbital mode* (M), *current maximum orbit size* (F), *current maximum orbital speed* (F), *State-of-Charge of the selected UAV* (F), *current semi-major axis* (F), *current eccentricity* (F), *current inclination* (F), *current X-position w.r.t. reference point* (F), *current Y-position w.r.t. reference point* (F), *current Z-position w.r.t. reference point* (F). The location of this reference point can vary, but will most likely be the Sun.

A graphical flow of the data between each section is illustrated in Figure 9.1. In this figure, the solid connections represent data streams that happen internally and the dashed connections represented data streams over Bluetooth. The text by some of the lines indicate the specific variables which flow into the next block. If this text is absent all the outputs from the previous block are required for the next block. The signals which are shown to go ”to display” will be sent to the Admin Display over the Bluetooth connection at the end of each calculation cycle.

A total of 80 bits needs to be transmitted to the UAV from the server for calculations and 136 bits transferred from the UAV to the server to be displayed. As stated before, this entire process needs to happen every 0.01 seconds, or at a frequency of 100 Hz. This means that a total bandwidth of 21,600 bits per

second (b/s) is needed, or data rate of 2.7 kB/s. By setting the refresh rate of the system at this frequency, the control system now complies with requirements **Hor-Ctrl-8**, **Hor-Ctrl-9**, **Hor-Ctrl-10**, **Hor-Ctrl-11**.

Since Bluetooth 4.0 has bandwidth possibilities ranging from 721.2kbps to 24Mbps [3], it can be concluded that currently very little of the possible bandwidth is used. This means that a more complex signal can be used, if during testing of the system it is discovered that the interference on the Bluetooth signal is too large to get clear data transmission. A more complex signal will provide each parameter with more unique characteristics, which allows for easier recognition in a noisy signal. This requires more power during transmission and is therefore left as a back-up.

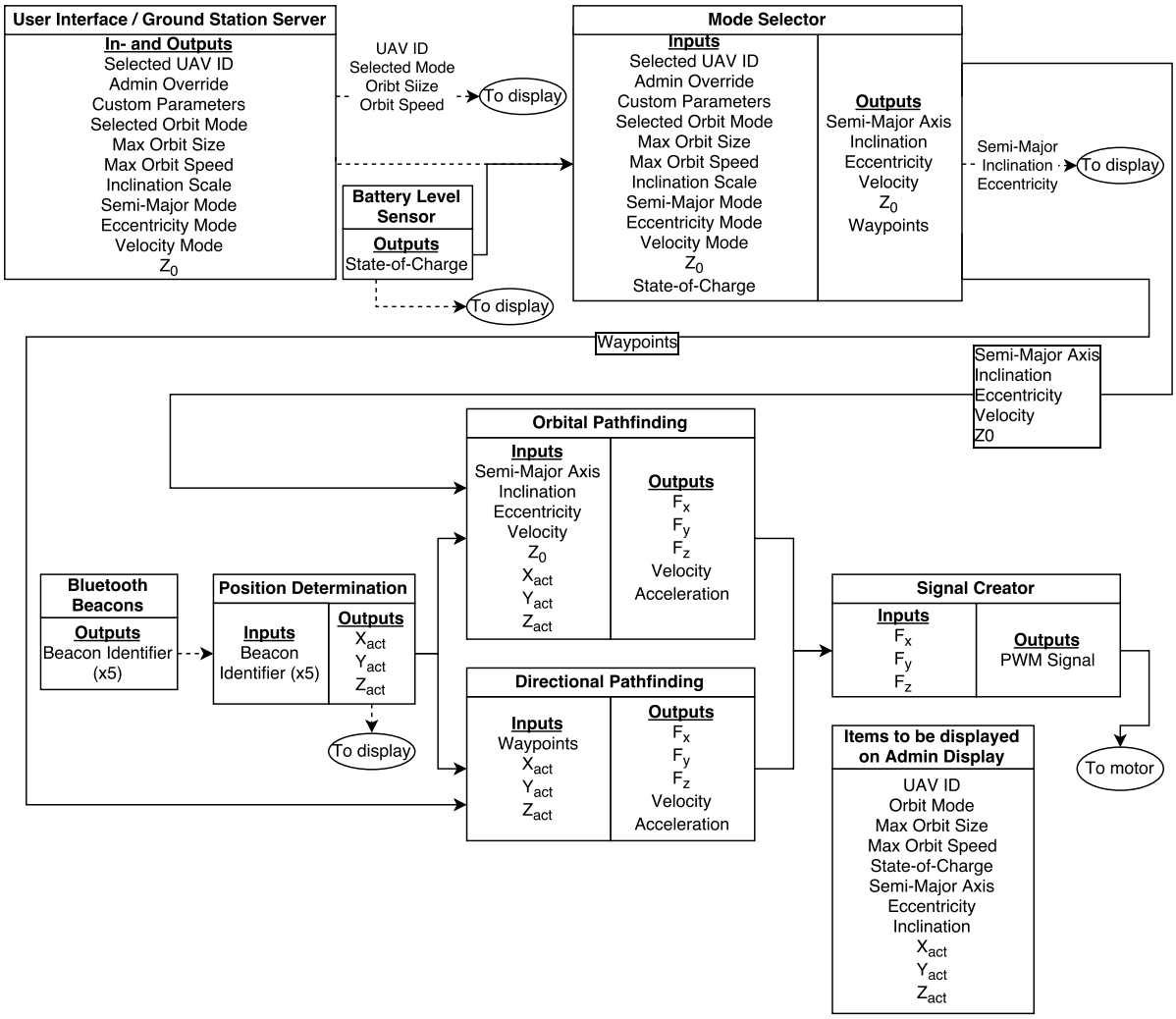


Figure 9.1: Illustration of the Data Flow of the System

9.2. Hardware

The hardware that will be used to execute the above mentioned data transmissions and handle the data on the UAV side will be a Raspberry Pi.[a] More specifically the Raspberry Pi Model B, Generation 3 micro-computer is chosen. This micro-computer has the benefit over other Raspberry Pi cards that it comes stock with Bluetooth 4.1 capabilities, facilitating the desired to use Bluetooth for both the communication, position determination and collision avoidance systems. Furthermore, this is the only version of the Raspberry Pi that comes equipped with a multi-core processor, it boasts the highest clock speed, it has the highest bit architecture (x64) and it comes with the most on-card memory, namely 1 GB. On

[a]Raspberry Pi 3 On Sale, URL `https://www.raspberrypi.org/blog/raspberry-pi-3-on-sale/`, accessed on 1 June 2016

the server side of things a HP EliteBook 8570w will be used. This laptop is chosen as it features native Bluetooth 4.0 capabilities, comes with a decent sized hard drive of 500GB which can be used for data logging and has a 64-bit architecture. Furthermore, it was decided to use a mobile workstation for the server as it facilitates any changes that have to be made to the software.

9.3. Communication Flow Diagram

The Communication Flow Diagram (CoFD) is shown in Figure 9.2. It shows the flow of data between the different elements of the system, and also between the system and its environment.

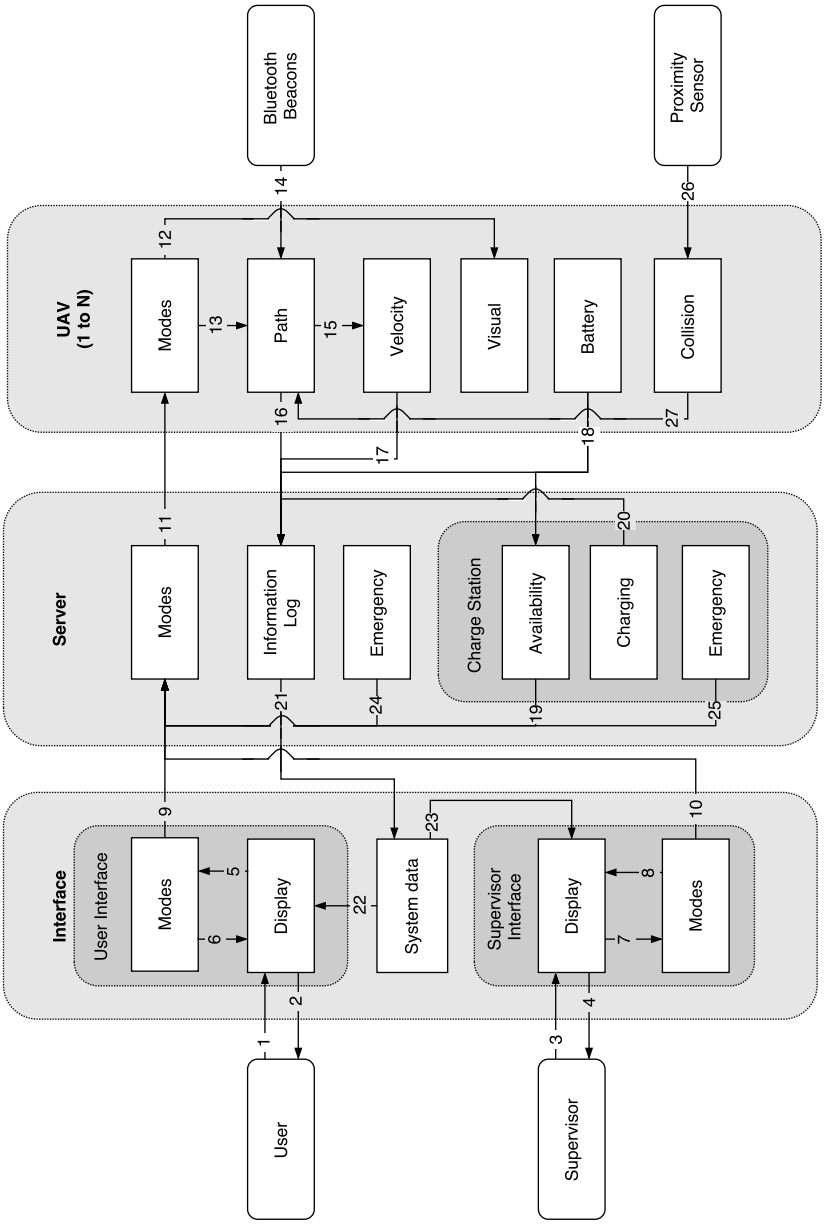


Figure 9.2: Communication Flow Diagram

10 | Positioning

In this section the position determination method of the UAVs is detailed. This positioning is split into long and short range determination, for the orbiting and landing position determination respectively.

10.1. Long Range

With the new Bluetooth Low Energy (BLE) protocol, it is no longer necessary to "pair" Bluetooth devices. This allows a device to easily obtain a list of the Received Signal Strength Indication (RSSI) of all nearby devices and beacons. This can then be used to obtain a distance estimate between the UAV and beacons in the exhibition space, using Equation (10.1) [29]. In this equation, the network environment variables b and c, the signal strength at 1 meter and the signal transmission constant respectively, are used to calculate the distance d from the RSSI.

$$RSSI = -(10b\,log_{10}d + c) \tag{10.1}$$

With Bluetooth positioning and a minimum of four beacons in the exhibition space, the position of the UAVs can be determined with an uncertainty of 3 cm [29]. It is decided to make use of five Bluetooth beacons, four of which are located on the corners of the flight ceiling and the fifth is located in the Sun. The floor under the Sun has been set as the origin of the coordinate system, because the orbits are flown with a set distance to the Sun. When the Sun is located at (0,0,z), the radius can easily be determined. The floor has been chosen as (z=0), because the ceiling and Sun might vary in height depending on system mode and the space HORUS is placed in.

Via distance provided from the RSSI, the position is determined by making use of three beacons. Because there are five beacons present, the remaining two beacons are implemented for redundancy. Equation (10.2) is used to calculate the current point in the flight envelope occupied by the UAV. n refers to the set coordinates of an arbitrary beacon. The true position will be determined from the different positions obtained using this equation using the Least Square Estimation. Figure 10.1 shows how the positions obtained from the RSSI estimation (blue dots), and the position estimation made with the least squares estimate (red plusses). The least squares gives very good approximations of the true position. This position will be broadcast to the server for data collection.

$$d = \sqrt{(x - x_n)^2 + (y - y_n)^2 + (z - z_n)^2} \tag{10.2}$$

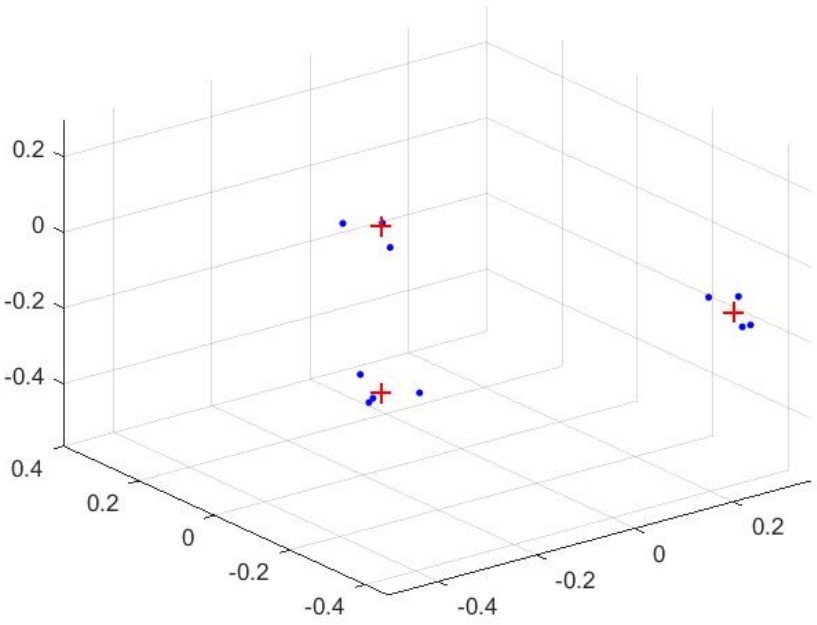


Figure 10.1: The Least Squares Algorithm

10.2. Short Range

For landing on the charge station a much higher accuracy is needed than for orbiting. Therefore an additional short range positioning system has been designed, which uses photo-resistors to determine UAV position when it is near the charge station. Therefore, four Light Emitting Diodes (LEDs) are placed on the charge station, which match the four photo-resistors that will be placed on the UAV. The UAV will land by maximising the light that reaches the resistors, therefore landing with high accuracy on these LEDs and by extension the charge station. This has been shown in Figure 10.2. This will allow the UAV to land and charge autonomously, and thus fulfils requirements **Hor-UAV-12** and **Hor-Ctrl-2**. These sensors are very lightweight and will add approximately 1 g per UAV.[a]

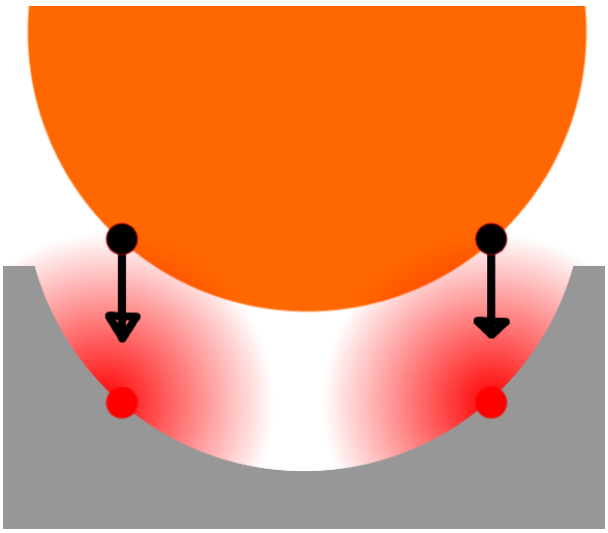


Figure 10.2: Short Range Positioning Near the Charge Station

[a] Photoresistors, URL `https://www.adafruit.com/product/161`, accessed on 20 June 2016

11 | Control Forces

This chapter covers the systems that provide the lateral control forces that are essential to the manoeuvrability of the UAV. These forces will enable the UAV to manoeuvre in a horizontal plane as opposed to sole ascent and descent. The technology of this system will be explained in Section 11.1 and the use of this technology in the UAV is detailed in Section 11.2.

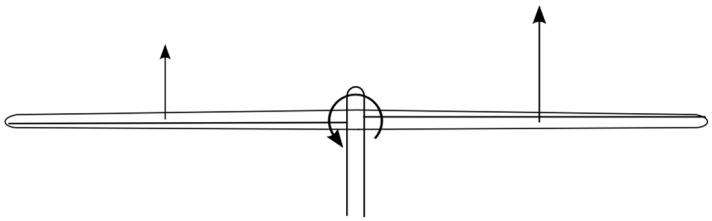
11.1. Working Principle

The chosen method of applying control forces is the principle of variable pitch. This means that the pitch of the propeller blades changes during parts of a revolution in order to create a moment around the centre of rotation. Section 11.1.1 will illustrate the principle of a variable pitch rotor in greater detail, as Section 11.1.2 shows how new developments aid in improving both the mass and performance of the variable-pitch system.

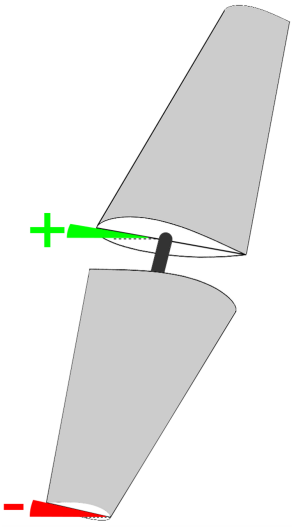
11.1.1. Conventional Variable Pitch Methods

In order to move sideways, a sideways force has to be applied to the UAV. This is possible in two ways: applying an added internal or external side force besides the thrust, or tilting the thrust vector in such a way that the UAV will start to move. The trade-off in the Mid-Term Report showed that tilting the thrust vector by means of variable pitch was by far the most advantageous option.

The principle of variable pitch rotors is illustrated in Figure 11.1 below. Since the pitch of the propeller blades is different – one blade has a lower pitch and one blade has a higher pitch than the neutral pitch angle – the perceived angle of attack is different due to the different pitch. The blades have a wing-like airfoil, which means that the blades will produce different magnitudes of lift. This force imbalance causes a net moment around the centre of rotation.



(a) A Schematic View of the Force Imbalance causing a Moment



(b) A 3D example to illustrate the Positive and Negative Pitch Angles

Figure 11.1: A Visualisation of the Variable Pitch Control Concept

This moment causes the entire UAV to tilt, which causes the thrust vector to be titled as well. This is the desired effect, as stated in the first paragraph of this section. The tilt of the vector means the thrust force now has a horizontal and a vertical component. The horizontal component is the part causing lateral movement: this is how lateral control is achieved.

Variable pitch propellers can be found in aircraft and ships, for example, but the application of the concept laid out above is found most in helicopters. The most frequently used system is that of the swashplate, which uses a ring that is on top of an inclined disk and connected to the blades by actuators. Figure 11.2 shows an actual example of a swashplate mechanism.

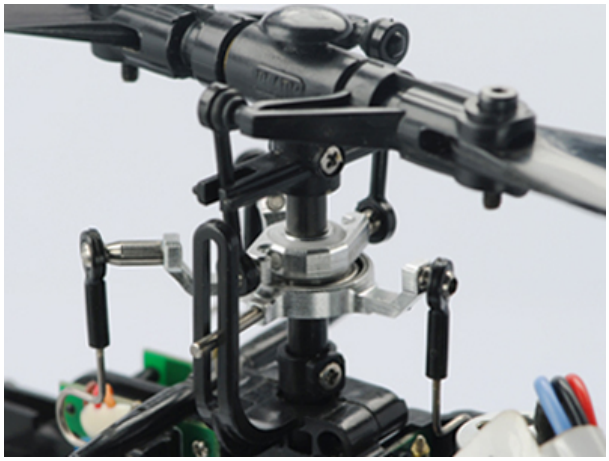


Figure 11.2: An Example of a Small-Scale Swashplate System[a]

There are a lot of advantages to the swashplate mechanism. It is the most used system because it has been around for over 100 years.[b] The concept is a proven one, and this also means the technology behind it has had ample time to develop. In essence, the swashplate mechanism is a proven fall-back option if all else fails.

However, there are also disadvantages. Clearly, the mechanism has a lot of moving parts and is therefore heavier than for example the variable pitch propellers described in the next section. This is not a problem for a helicopter or a larger UAV because it is very easy to scale up the power unit in order to accommodate for the extra mass. This is not the case for this UAV because the mass requirements are more strict.

11.1.2. Modern Variable Pitch Technology

The rather heavy but proven concept of the swashplate, illustrated in Figure 11.2, has been identified as a good reference point and a great fall-back. However, part of designing a state-of-the-art system is trying to push the envelope. This is why more advanced variable pitch methods and mechanisms are also considered.

One of the novel concept considered is the so-called swashplateless variable pitch control. This concept works by using a smart hinge system and sinusoidal impulses to the motor torque. Figure 11.3 shows a diagram of the concept.

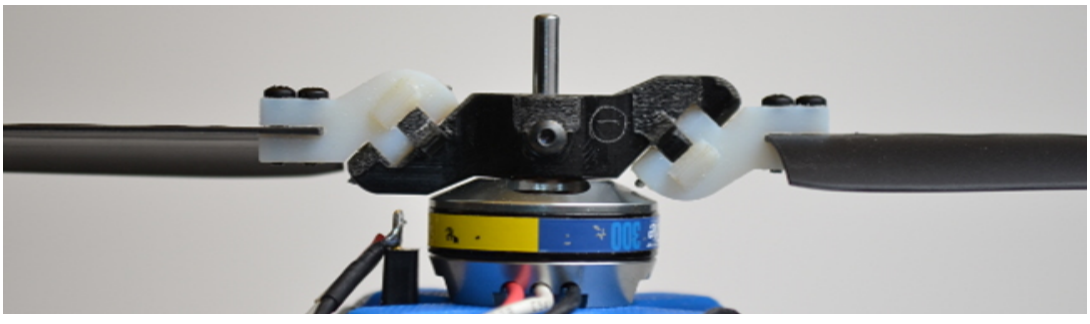


Figure 11.3: The Mechanism of the Swashplateless Technology [24]

[b] Principles of Helicopter Aerodynamics, p.13, URL `https://books.google.ru/books?id=nMV-TkaX-9cC&lpg=PA13&ots=CodTn9WMfz&pg=PA13&hl=nl#v=onepage&q&f=false`, accessed on 17 June 2016

When the mechanisms of this system are compared with that of the swashplate, it is clear that the swashplateless system is much smaller and made from lighter material. Therefore, the overall mass is much lower. This is critically important for this design. The effect of this observation is that this concept should be investigated further because of this mass advantage. In fact, further research has shown that this is a state-of-the-art system is the most advantageous system available. The concept itself had already been suggested in 2013 [23], but the concept has already been proven as recently as 2015 [24] my means of a UAV that has been demonstrated to work using this technology.

Therefore, the choice of technology was straightforward. The swashplateless system has several major advantages over the swashplate system. The biggest disadvantage of the new technology compared to the swashplate is the vast amount of concept that have been proven to work with a swashplate. The knowledge about swashplates is far greater due to the fact that this method of variable pitch control has been around for far longer.

However, the swashplateless technology has already been proven to work both as a standalone rotor as well as in a UAV [23] [24]. Therefore, this drawback does not outweigh the weight advantage of the swashplateless system. Also, the swashplateless system consists of a single rotor with only two hinges; way simpler than all the mechanical links in the swashplate system. This means that the modern technology is less sensitive to mechanical malfunctions, which greatly contributes to reliability and ease of maintenance.

11.2. Implementation

The system in Figure 11.3 is chosen as the variable pitch system to be implemented in our UAV design. The stated advantages outweigh the disadvantage of a less proven concept and the lack of experience with this concept. The known experiences with the technology are such that the system can be seen as a viable design option.

As shown, the only moveable parts in the mechanism are – apart from the drive shaft, of course – two hinges at an angle with the vertical. These hinges are essential to the functionality of the system. This is because the variable pitch is achieved by letting the blades rotate around these hinges, which causes a different angle of attack with respect to the neutral position. This principle is illustrated by Figure 11.4 below.

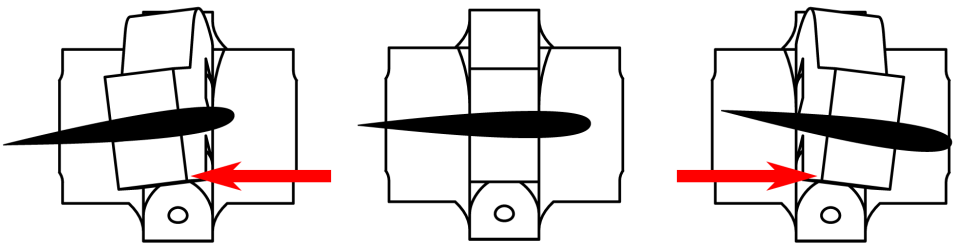


Figure 11.4: The Principle of Lead and Lag Causing Pitch Changes

This particular blade is the ”positive” blade, denoting that the top of the hinge points towards the drive shaft. It is the left blade in Figure 11.3. The red arrows indicate lag in the left situation and lead in the right situation. The lag and lead of the blades is achieved by increasing and decreasing the torque at the correct point during the rotation, causing the drive shaft to lead or lag the blades.

The application of these pulses is achieved by using PWM to add a sinusoidal signal to the requested torque. The result is a torque that is suddenly higher or lower than the previously applied constant torque. The inertia of the blades and the fact that there is a hinge between the blades and the drive shaft cause the blades to rotate along the angled hinge. This rotation translates in both a rotation in the rotor plane as well as along the blade axis. The other blade has a negative hinge, meaning that the top of the hinge points away from the shaft. This causes the exact opposite reaction: positive pitch when leading and negative pitch when lagging. This pitch difference causes the force imbalance and the ensuing moment illustrated in Figure 11.1a.

Now that the method of pitch control has been discussed, the performance of the system can be addressed. The function of the variable pitch system is to keep the UAV at a certain angle that causes

the thrust vector to tilt. This tilt causes sideways movement due to the fact that the trust vector can be decomposed into three perpendicular forces. Looking forward, the maximum acceleration as stated in Chapter 13 is 2 m/s^2. This means that the maximum net force on the UAV is equal to the mass times the maximum acceleration.

The net force spoken about in the previous paragraph is the sum of the net forces along the three axes. Figure 11.5 below shows the difference between the weight, the thrust force and the net force on the UAV. This is a very important distinction.

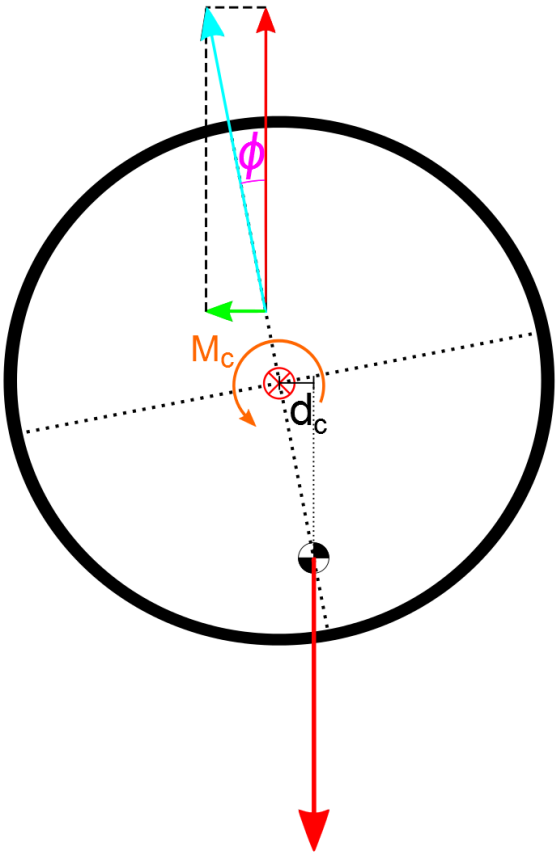


Figure 11.5: The Difference Between Weight (Red), Thrust (Blue) and Net Force (Green)

The entire UAV is tilted under the influence of the control moment M_c in orange created by the force imbalance discussed in Section 11.1.1 and schematically shown in Figure 11.1a. This moment has to be equal to the weight times the horizontal displacement d from the centre of the UAV. The largest possible value for this moment will occur if the center of gravity is at the very edge of the UAV, so that d is equal to $r * \tan(\phi)$. Here ϕ is the tilt angle of the UAV. The value for ϕ can be found by looking at the extreme case for the side force. The maximum possible side force is the mass of the UAV times the maximum acceleration. Since the vertical component of the thrust force is equal to the weight, ϕ can be found by using Equation (11.1):

$$\phi = \arctan\left(\frac{F_{side}}{W}\right) = \arctan\left(\frac{m \cdot a_{max}}{m \cdot g}\right) = \arctan\left(\frac{a_{max}}{g}\right) = \arctan\left(\frac{2}{9.81}\right) = 11.5^{\circ} \qquad (11.1)$$

The value for $\tan(\phi)$ can also be substituted directly to find the required moment:

$$M_c = W \cdot d_c = m \cdot g \cdot r \cdot \tan(\phi) = m \cdot g \cdot r \cdot \frac{a_{max}}{g} = m \cdot r \cdot a_{max} \qquad (11.2)$$

The largest possible value for M_c then becomes $2\ kg * 0.25\ m * 2\ m/s^2 = 1\ Nm$ for the two largest UAVs. The other UAVs will require smaller moments since both their masses and radii are smaller. From the

tests conducted with the prototype, the achieved thrust to moment ratio was approximately 1 to 20. So for every Newton of thrust generated, the rotor provided $1/20\ Nm$. If the assumption of a linearly scalable system is made, a UAV requiring $20\ N$ of thrust from two rotors would experience a moment of $1\ Nm$. This is exactly the required moment for the most extreme case.

At first glance, it seems like the moment provided by the rotors barely meets the required control moment. There are some mitigating factors, however. The distance from the centre of gravity to the centre of the UAV will not be equal to R. In fact, even a distance of $0.5 * r$ is conservative considering the fact that the UAV lay-out is nearly entirely symmetrical. This is not considering the fact that the centre of rotation of the UAV will probably not be in the centre of the UAV. However, for the purpose of this exercise – finding the highest required moment magnitude – this is assumed.

Also, the required moment is directly proportional to a_{max}. Since a_{max} is set by the design team, a lower value for this parameter can help overcome a possible moment shortage. In fact, the only effect that a lower value for a_{max} will have is the fact that the UAV will react slightly slower to target position changes. This will be elaborated upon in Chapter 13.

The last possible problem this method of applying lateral control is the fact that the required thrust force is higher than the weight itself if the thrust is tilted. However, the value of $\cos(\phi)$ with ϕ as calculated in Equation (11.1) is equal to 0.980. This means the thrust only has to increase 2% to accommodate for the lateral control forces. In fact, the largest possible requested force is if the desired acceleration is $2\ m/s^2$ upwards only. This would require a total force of $m * (g + a)$, which would lead to an thrust increase of $a/g = 2/9.81 = 20.4\%$ when compared to the situation where the thrust is equal to the UAV weight.

If this increase in thrust cannot be produced, the only drawback this would have is a lower acceleration during ascent, resulting in a longer ascension time. Since the goal of the system can still be realised if the UAVs ascend slightly slower, this problem is not considered to be of any hindrance to fulfilling the two most important control force requirements. These are Item **Hor-UAV-18**, which states that the UAV shall be able to translate along the three axes and Item **Hor-UAV-19**, which states that the UAV shall be able to rotate about the three axes. The former is automatically satisfied by being able to ascend, descend and move laterally. The latter has already been demonstrated for two axes; rotation around the vertical axis cannot be achieved by variable pitch, however. The solution is letting the two rotors rotate at a different speed, so a torque difference is created and the net torque rotates the UAV.

Concluding, the control method posed in this chapter seems to meet the requirements relating to the lateral control both during orbit mode as well as during path tracking mode. If the generated control moment still happens to be too small to reach the 11.5 ° tilt angle, the consequence will be a smaller side force, having the same effect as the ascent problem stated in the previous paragraph. This is not detrimental to the functionality of the system and can also be compensated in a number of ways. Therefore, the swashplateless variable pitch control technology will be implemented in the UAVs of the HORUS system.

12 | Orbit

To simulate the planets orbiting around the Sun, a control model needs to be developed to steer the UAV. From the mode determination, set in Chapter 3, the mode is set to ORBIT. The mode will set the desired semi-major axis, eccentricity and velocity. The model described in this chapter will use the inputs set by the mode and translate the inputs into desired forces the UAV needs to generate.

12.1. Orbit Model Description

The orbit control model is based on mimicking a centripetal force generated by the UAV to maintain a fixed distance from the Sun. The UAV is assumed to have a constant forward facing side.

The semi-major axis and eccentricity are used to set a reference radius the orbit needs to maintain. This radius and the desired velocity are used to calculate the desired centripetal acceleration, see Equation (12.1).

$$a_{r_{set}} = \frac{v^2}{r} \tag{12.1}$$

The desired centripetal acceleration, velocity and radius are fed into a negative feedback control loop. The loop uses polar coordinates to define the acceleration, namely the radial (a_r) and the tangential (a_t) acceleration. The polar coordinates are transferred to Cartesian coordinates (x- and y-direction). These accelerations are integrated twice to get the position in x- and y-coordinates. The coordinates are set coordinate estimation outputs. In the model, these estimated coordinates will be used for feedback. When the UAVs are in use, the feedback will be done using the real position and the estimated coordinates will only be used to calibrate the system. The coordinates will be converted to polar coordinates, so the radius r and the angle θ will be calculated. These variables are then used to calculate the tangential acceleration and the difference in radial acceleration with respect to the desired acceleration and the computed acceleration. The radial and tangential acceleration feedback loops are shown mathematically in Equation (12.2) and Equation (12.3).

These formulas incorporate the difference between the set radius (r_{set}) and the true radius (r), in the same direction as the difference. It also incorporates the radial velocity. This means that a_r becomes $a_{r_{set}}$ when the set radius has been reached. The gain factor, K, will be determined for each UAV size, as this is greatly depended on how the UAV reacts to disturbances and on the set orbit.

$$a_r = a_{r_{set}} - K\left[\frac{dr}{dt} + \left(\frac{r_{set}^2 - r^2}{v_{set}}\right)\right] \tag{12.2}$$

$$a_t = K\left[v - v_{set}\right] \tag{12.3}$$

The tangential acceleration is calculated using Equation (12.3). Because it uses the difference between the real tangential velocity and the set one, it will become zero when the correct value is reached. It is divided by the orbit radius, in order to scale the acceleration with the orbit that the UAV is flying. When these final accelerations are higher than the maximum acceleration of the UAV, it is set to the maximum acceleration of 2 m/s^2. This is to account for the maximum pitch of the propeller, which leads to a certain maximum control force that can be generated.

These two accelerations are transformed to a Cartesian coordinate system and integrated twice to get the modelled position of the UAV. This position is then used to calculate the radius, velocity and angle of the UAV. This system is capable of stabilising to the set radius and velocity, from any position or initial velocity. When the UAVs are flying, the real position and velocity will be used for feedback, which allows for real-time corrections of either.

12.2. Orbit Model Results

The orbit model is thus capable of reacting to arbitrary begin position and velocity. One of these responses has been shown in Figure 12.2b. The gains of the system will need to be tuned in order to minimise the chance of an overshoot taking the UAV outside of the flight envelope. If this does happen,

the collision avoidance system will avoid collisions with the walls or other objects just outside the flight envelope.

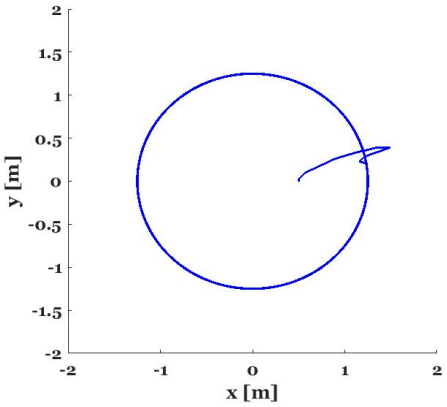
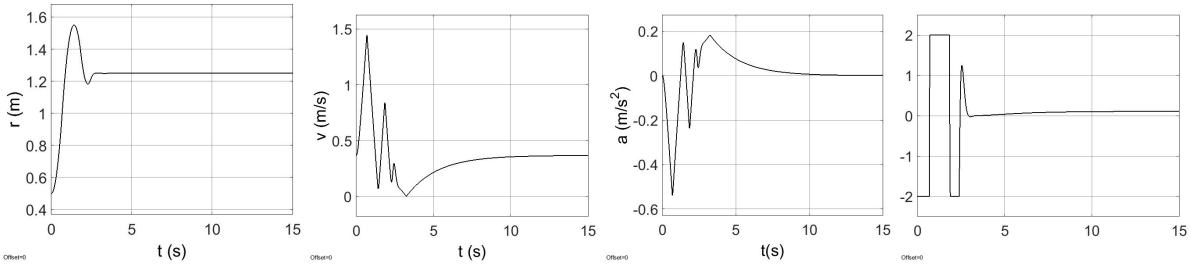


Figure 12.1: The Circular Orbit of the Venus UAV, with an Input that has the Final Speed and a Higher Radius than the Final One

As an example, Venus starts its orbit at the coordinates $(x, y) = (0.5; 0)\ m$, and the final velocity of $0.36\ m/s$. It can be seen that the radius of the orbit converges exponentially to the final value of $1.25\ m$ (Figure 12.2a), while the velocity first changes in order to get the UAV to the correct radius and then converges to the final value of $0.36\ m/s$ (Figure 12.2b). The radial acceleration at $t = 0$ is lower than the final value, but becomes bigger when the radial velocity increases. When the radial velocity has converged to zero, the radial acceleration converges to the final value of $0.1\ m/s^2$, as calculated with Equation (12.1) (Figure 12.2d). Meanwhile, the tangential acceleration works in parallel to bring the tangential velocity to the final value, and then levels out to zero (Figure 12.2c).



(a) The Radius of the Venus UAV Orbit

(b) The Velocity of the UAV

(c) The Tangential Acceleration of the UAV

(d) The Radial Acceleration of the UAV

Figure 12.2: The Responses of the Venus UAV For a Starting Position of (0.5,0)

12.3. Verification and Validation

The verification started with the individual code blocks of which the outputs could be checked by hand. Then, the feedback model was made in one dimension. This allowed the formula to be optimised on its own. Afterwards, a two dimensional system was developed. This only had radial feedback, and did not converge to the correct radius because it worked like a real planet in that it exchanged velocity for radius. This was corrected through the combination of the one dimensional and the two dimensional system, where the one dimensional system is used to correct the velocity of the UAV. When these two systems work in tandem, they lead to a system that quickly converges to the set radius and orbit velocity. The gains used in this system were optimised for each radius and orbit velocity, but can be further optimised using analytical tools.

Because the orbit model is based on a theoretical UAV, it will need to be experimentally tested when the true components are known. The necessary gain of the feedback is also heavily dependent on the outputs of the sensors. The individual components will thus be tested, and their values used to optimise the simulation. This can then be build into the Raspberry Pi, and outputs tested to see if they conform to the simulation. When that works, the whole control system can be assembled and tested.

13 | Path Tracking

This chapter explains the fundamentals of the non-orbital path tracking ability of the UAVs. In short, this enables a UAV to fly a prespecified path using 3D coordinates given by the server. First, the requirements of this mode, along with its functionalities and application, will be explained in Section 13.1. Next, Section 13.2 outlines the development and maturation of the software. It also describes the problems faced during this developmental process and the improvements made to overcome these problems. Finally, Section 13.3 will go over the details of the final model. The logic behind the software will be explained and suggestions for improvement will be given in the last part of this section.

13.1. Requirements and Functionalities

This section contains the requirements set by the control & communications team regarding the Path Tracking Mode (PTM) and the model simulating this mode, shown in Section 13.1.1. It also explains why it is such an important part of the software of the system, and when and how this mode is used to reach a desired position or system behaviour in Section 13.1.2.

13.1.1. Path Tracking Requirements

As stated before, one of the main goals of HORUS is to educate on the principles of the solar system and the movement of the planets. The orbital mode, detailed in Chapter 12, is the most important piece of software used to demonstrate these movements. This mode, however, does not include auxiliary movements, such as flying from the charging station to the orbit and back, or landing in a controlled manner.

One can compare it to a Formula 1 team: the most import goal of such a team is to win the race. However, behind the scenes there is much more going on than just driving around the track. Pit stops, transport to and from the track and so on are just as important to achieve the goal of setting the fastest possible time.

The entire system is composed of many more parts than only the central machine and there are many more factors in play besides striving to achieve the main goal using that machine. This is why the word ”auxiliary” was used in the previous paragraph: the PTM handles all flying actions *besides* the main goal of showing planets orbiting around the Sun.

Consequently, the requirements of the PTM can be easily identified: look at the entire set of requirements concerned with flying and compare this set to the abilities of the Orbit Mode. Everything but the ability of the main mode has to be accounted for by the PTM. This chapter will also treat several specific requirements found in Appendix B.

Since the main mode fulfils the primary task of the system – keeping the UAVS in orbit – the PTM does everything except for keeping the UAVs in orbit. This requirement essentially means that the PTM has to be able to bring a UAV from every possible position into a position from which it can go into orbit. Vice versa, it also has to be able to bring the UAV from an orbit to any possible position using a specified path.

13.1.2. Path Tracking Functionality

A logical place to start a flight analysis is the start of the flight. This flight path will logically start with the UAV in the charging dock. After the UAV start sequence has been completed and the UAV is ready to go into orbit, it still has to get there. This is where the PTM first steps in: it has to bring the UAV from the charging dock to a position from which it can start the orbiting mode. One of the requirements of the system is **Hor-UAV-3**, which states that the UAV shall be able to take-off after a crash landing. Provided that the UAV is mechanically still able to fly, the PTM works exactly the same in this situation. The UAV determines its position and its intended position, after which it simply is the same as a normal start.

This orbiting mode will carry on for a while, but it is possible that a supervisor selects a different mode. Take the presentation mode, for example. This mode requires one or more UAVs to hover at a

specific position, such as next to the supervisor. The PTM should be able to take care of this situation by keeping the UAV at a specified waypoint. This situation is also comparable to any other specific positioning request. The compliance of the PTM with these requirements can be summarized in one sentence: the PTM should be able to control a specific UAV and let it fly to any specific position.

After the UAV has flown for a certain amount of time, the state of charge of the battery will reach a lower threshold, causing the charging mode to be activated. The path to the charging station is then automatically calculated and relayed to the UAV so the PTM can take over and guide the UAV to the docking station. The UAV then will have to complete docking itself in the charging station as fast as possible or risk running out of power with all the consequences attached to that scenario. It is of utmost importance that the UAV makes its way to the dock as fast as possible, since any delays might result in the aforementioned scenario. This effectiveness will be elaborated upon in Section 13.3.

The ability to automatically follow this flight path is in line with **Hor-Ctrl-20**, stating that the UAV shall have autonomous path planning capabilities. Since flying a path means that the UAV has to fly to a specified point in the flight envelope and repeat this for successive points, the UAV automatically satisfies **Hor-Ctrl-22**, which states that the UAV shall be able to fly to a specified set of coordinates with an accuracy of 2.5 cm. It will become apparent that this requirement is the cornerstone of the PTM.

Since there are two identical UAVs, one UAV can charge while the other one is flying. After a certain period of time, however, there is a moment at which the UAVs have to switch roles. The PTM is capable of steering the twin UAVs in such a way that the risk of collision is minimal, even though the UAV charging docks are adjacent to each other. This is done by using two non-crossing paths combined with sufficient accuracy. The readily charged UAV will then resume the orbit that the nearly depleted UAV has left, so the entire system still functions as one solar system without any missing planets.

Summarizing, it can be said that the functionality of the PTM lies mostly with the regular and anticipated mode switches during standard operation. These switches are critical moments and should be able to be executed numerous times with different UAVs over a very long period of time. There is always the possibility that unique and unpredictable situations suddenly arise. It cannot always be said, however, that a prespecified reaction by the PTM exists. One must think of broken propellers, possible mesh damage and other emergencies in this sense. It sometimes is simply impossible to account for such situations, but a reaction to a comparable situation might be used in this case. Therefore, the most important functionality the PTM has is the guidance of the UAVs during planned non-orbital motions.

13.2. Model Predecessors

There are numerous models that were designed before the final model was constructed. This section will cover these models, as well as the mistakes made during their design and the lessons that have been learned from these mistakes. Section 13.2.1 will start with covering the very first model, the 1D model. The various 2D successors of that model will be detailed in Section 13.2.2 and, finally, Section 13.2.3 will illustrate the link from the 2D models to the 3D model and the initial imperfections with respect to the final model.

13.2.1. 1D Ascent

The first model made was the altitude model: a 1D model with the goal of simulating the take-off of a UAV to a specified height. The initial model used the difference between the target height and the actual height to specify a force upward for a positive height difference – the target height was higher than the actual height – and a force downward for a negative difference. These forces were the net forces on the UAV. The forces caused and acceleration, velocity and of course displacement of the UAV, which gave the first simulation results, shown in Figure 13.1.

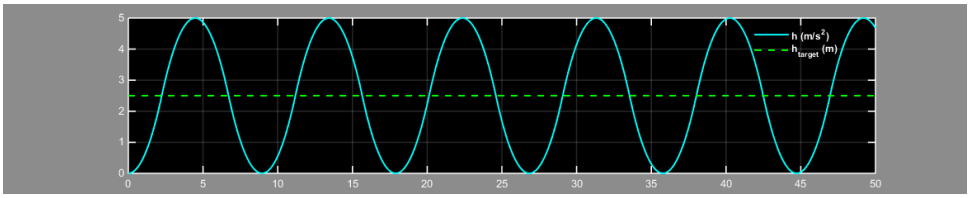


Figure 13.1: The Displacement of the UAV Versus the Time in the First 1D System

Clearly, the actual height in yellow oscillates around the target height of 2.5 m in purple, which of course is not the desired behaviour. The problem was the fact that the net force would only change sign after the UAV had passed its reference height. This was caused by the fact that the force was proportional to the height difference, with a maximum magnitude specified by $F_{max} = a_{max} \cdot m$. a_{max} was chosen to be $2\ m/s^2$. This value was used throughout all of the subsequent models to stay consistent.

The biggest problem was of course the oscillation. After all, what use is a UAV if it cannot stay at its intended height? There had to be a different approach to the problem of equal accelerating and braking: the UAV had to decelerate *before* it had reached the target height. The concept simple: reverse the force at a certain height difference in order to decelerate and reduce the speed to zero at the target height. The actual application of this idea proved to be a lot harder, though.

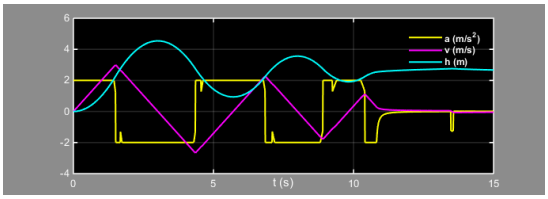
The underlying principle during braking is the following: apply enough force in order to decelerate the UAV to zero velocity at the reference height. The height difference was given as $dh = h_{ref} - h_{actual}$. This height will be traversed with a certain speed in a certain amount of time. During this amount of time, the deceleration must be such that the speed can completely be brought to zero. The relation between acceleration, speed difference and time is $dv = a \cdot dt$.

Since it is mandated that $dv = v$ due to the fact that the result speed should be zero, it can be observed that $v = a \cdot dt$ is obtained, which can be rewritten as $a = v/dt$, with v the speed at the start of the braking manoeuvre. Time is not of interest, so dt has to be expressed in other parameters. Since $v = dh/dt$, $dt = dh/v$ and using this relation in $a = v/dt$ yields $a = v^2/dh$. This is the governing equation for the braking manoeuvre, with the slight addition of mass into the equation such that instead of acceleration, the force can be expressed. The equation for the force applied during braking then becomes:

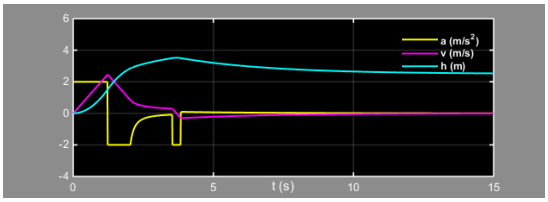
$$F = m \cdot \frac{v^2}{dh} \tag{13.1}$$

This force is limited in magnitude to $F_{max} = a_{max} \cdot m$ so the acceleration does not exceed the limit of $2\ m/s^2$. Using basic kinematic relations and generalizing where applicable, it is possible to come up with Equation (13.1) and use this equation in the second version of the 1D model. Using Equation (13.1), the next model was able to implement the desired behaviour, which is shown in Figure 13.2 below.

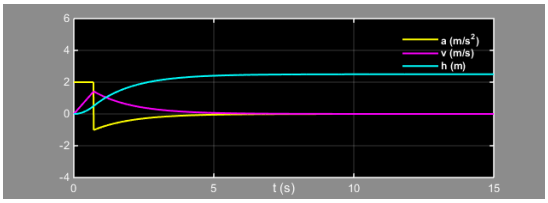
In every subfigure in Figure 13.2, the height is shown by the blue line, the speed is the purple line and the acceleration is the yellow line. These values all start at zero and vary over time, with positive values denoting an upwards speed or acceleration. Figure 13.2a shows the oscillation dying out over the course of about two periods. In this model, the height difference of 0.25 m at which the UAV starts braking is too small to fully decelerate the UAV, and it therefore overshoots the target height several times.



(a) Low Damping



(b) Medium Damping



(c) High Damping

Figure 13.2: Three Different Reactions of the Second 1D System

The graph in Figure 13.2b shows that the braking action starts much earlier, namely at a height difference of 1 m. This can be seen by the earlier reverse of the acceleration (the yellow line). There still is some overshoot, however, clearly visible by observing the peak of the blue line, which indicated that the actual height has to decrease again to reach the reference height. Although this is a better scenario, there is still overshoot and the distance at which braking commences has to be even larger.

When the braking starts at a height difference of 2 m, there is no overshoot and the system actually converges the fastest of the three options. The purple speed graph in Figure 13.2c shows that the UAV does not fully use its available top speed. In fact, the top speed goes unchecked and appears to be higher than 1 m/s in every situation. This means the UAV would not adhere to Item **Hor-UAV-5**, which is intolerable.

This can be counteracted by using the following condition: if the speed of the UAV is equal to (or higher than) its maximum speed, the applied net force will be zero. This way, the UAV maintains its maximum speed until the height difference is such that braking will commence. Since the speed with which the UAV will start braking is known, the minimum braking distance can be calculated. Given that $a_{max} = 2\ m/s^2$ and $v_{max} = 1\ m/s$, the time it will take to fully decelerate the UAV is $t = v/a = 1/2 = 0.5\ s$. The average speed during braking will be the average of start and end speed if constant deceleration is applied. This speed is $0.5\,m/s$. During a time period of 0.5 s and travelling at 0.5 m/s, the UAV will travel a distance of $dh = \overline{v} \cdot dt = 0.5 \cdot 0.5 = 0.25\ m$.

A distance of 0.25 m proved to be too small due to the discrete nature of the simulation program. It is also unwise to use the smallest margin possible from a real-life perspective; any small disturbance would throw off the exact nature of the deceleration. Therefore, a height difference of 0.3 m was chosen so a margin for error was created. The results of the third model are shown in Figure 13.3.

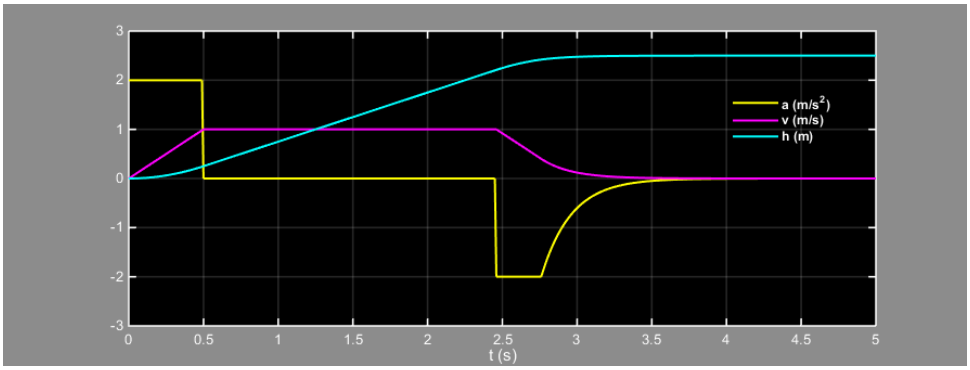


Figure 13.3: The Desired Response Given by the Third 1D System

It can clearly be seen that the purple line indicating velocity does not exceed 1 m/s and that the addition of 5 cm extra braking distance allows for a very gentle approach to the reference height. The acceleration is zero when the speed is at the maximum value, as decided upon in the previous paragraph. The elapsed time is also far lower than in the examples in Figure 13.2; ascent takes 3.5 seconds as opposed to 12, 10 and 6 seconds for the previous versions. The demonstration of this working principle is one of the most

important ones for the functionality of all future models and therefore has been elaborated upon at large.

13.2.2. 2D Movement

The last part of Section 13.2.1 showed that the 1D model has the ability to guide the UAV to a reference height and execute a controlled and efficient deceleration. The switch to the same principle in 2D is fairly easy: implement the used model for both x- and y-direction and set a 2D reference point. The maximum velocity of the UAV in either direction becomes $\frac{1}{2}\sqrt{2}\ m/s$, so the total maximum velocity does not exceed $1\ m/s$.

The problem with this method, however, is the fact that it looks at x and y independently. This is best understood by looking at Figure 13.4, from which it is clear that the UAV does not follow the most efficient path. This is of critical importance since the effectiveness of movements made during flight impact the flight time and therefore the measure with which the requirements are met. Figure 13.4 on the next page shows the actual path in blue, the starting point as a red circle and the end point as a red cross.

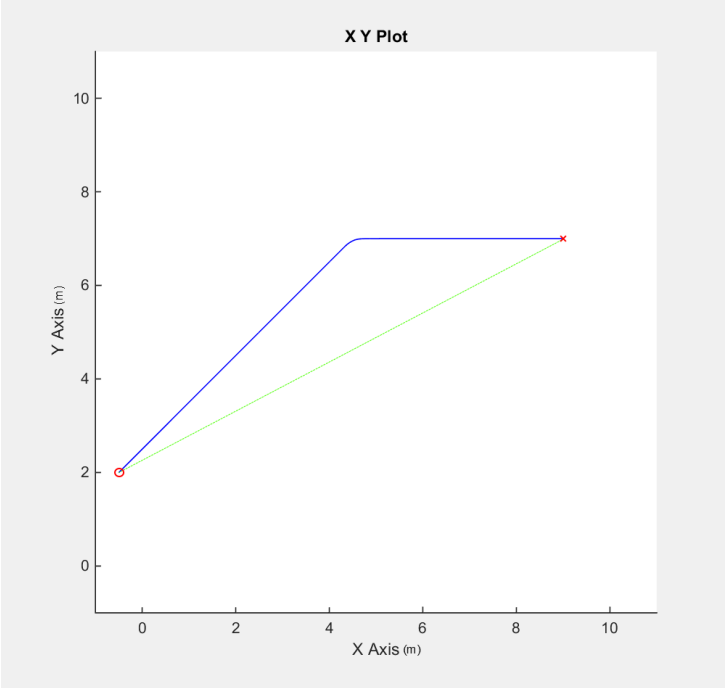


Figure 13.4: The Path Flown by the First 2D System

It is clear that the flight path from the first point to the second point is not a straight line like the ideal flight path in green: the flight path consists of two segments. The first segment is at a 45 degree angle to the axes, because the speed in x-direction and y-direction, vx and vy for short, are of equal magnitude. When the UAV approaches the specified y-coordinate, however, vy decreases until it is zero when the UAV is at the set y-coordinate, while vx remains unchanged. So not only is the length of the path longer than necessary, but the UAV speed does not equal its maximum speed during parts of the flight path. This is very undesirable and Section 13.3 will explain how this problem is solved.

The 2D model does lend itself for expansion with a waypoint module. This means that the 2D model offers the possibility of letting the UAV track an entire path as opposed to flying to one set point. This path is entered beforehand as a set of (x,y) coordinates. In order to achieve the goal of flying an entire path, the model checks the x and y distance to the current waypoint. If both of these distances are lower than a specified limit, this will trigger an if-statement in the code, causing the next waypoint to become the target x and y coordinates. This is done up to the last waypoint, which also has a stop clause: if the current waypoint is the last given waypoint and both x and y are within the limit, the model discontinues the simulation, because the UAV has reached its final waypoint. Figure 13.5 gives an example of such a waypoint sequence being followed by a UAV.

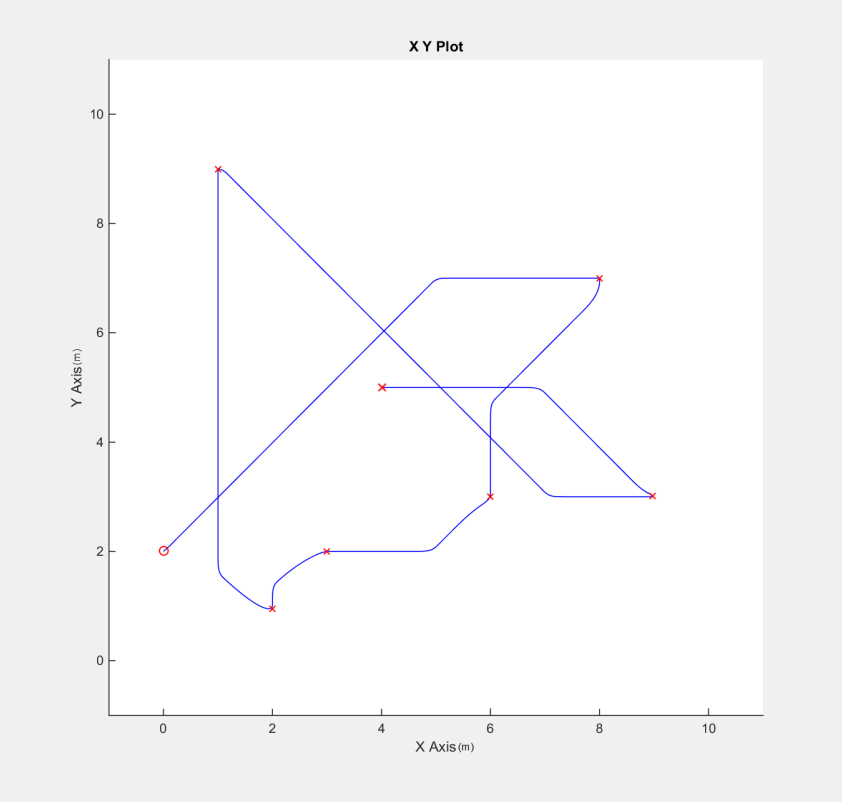


Figure 13.5: An Example Waypoint Track Flown by the Second 2D System

This results concludes the section on 2D models. The most important functionality the model has gained is the waypoint module: the UAV can fly a predetermined path consisting of several waypoints and can automatically stay at the last waypoint. The problem with the way the movements are calculated is the fact that the UAV does not fly as efficiently as possible.

13.2.3. 3D Axis-Based Movement

The biggest difference between the 2D and 3D models is that an extra dimension is added, obviously. But since the 3D model still works with the force, acceleration, speed and displacement along the x, y and z-axes independently of each other, the code itself is not that much different. In fact, the braking system explained in Section 13.2.1 is still used because it exhibits the exact desired behaviour. Essentially, the 3D model has the same functionality as the 2D model with the same order or performance, but the third dimension is added in order to simplify the transition from the proven 2D model with a non-optimized flight path to a 3D model with maximum efficiency. Section 13.3 will detail the final step to a more realistic force-based control system as opposed to the axis-based system used until now. Therefore, the 3D models can be seen as a non-innovative but essential step towards the final model.

13.3. Complete Path Tracking Model

This section will detail the entire final 3D model as used in the final simulations and as the base for the collision avoidance system detailed in Chapter 14. First, the prerequisites are given and explained in Section 13.3.1. Then, the waypoint iteration sequence – already touched upon previously – will be explained in detail in Section 13.3.2. The situation determination is next and will be discussed in Section 13.3.3. After that, the algorithm predicting the required velocity will be shown in Section 13.3.4 and the final subject of this section, the determination and application of the required force, will be elaborated upon in Section 13.3.5.

Before continuing in this section, it should be mentioned that the model that is to be discussed is a digital equivalent of the intended reality. There are some simplifications, which will be mentioned in the relevant subsections below. This model can therefore be looked upon as a means of visualizing the

intended system behaviour with the goal of implementing the underlying logic in the system.

13.3.1. Parameter Creation

Before the UAV will take off, there are certain parameters that are already set. These have to be set in the model as well. There are base parameters and tuning parameters. Base parameters are either inherited (cannot be changed) or otherwise specified. Changing these parameters would greatly influence model behaviour. Tuning parameters are minor parameters that can be changed to improve the overall performance of the model and do not have a large influence on the model. These parameters can be seen in Table 13.1.

Table 13.1: External Model Parameters

Type	Symbol	Value	Unit	Parameter	Remarks
Base	m	1.3-2.0	kg	UAV Mass	Inherited. Differs for every UAV
	v_{max}	1	m/s	Maximum Speed	Set by Item **Hor-UAV-5**
	a_{max}	2	m/s	Maximum Acceleration	Set by the Design Team
Tuning	e	0.025	m	Maximum Deviation Distance	Set by Item **Hor-Ctrl-22**
	e_2	0.01	m	Hovering Distance	Set by the Design Team
	d_{min}	0.3	m	Approach Distance	Follows from Section 13.2.1

Most parameters in Table 13.1 have already been touched upon. The parameters e and e_2 will now be explained. The value for e is given by **Hor-Ctrl-22**, and denotes the minimum accuracy a UAV must have. Therefore, e is the maximum distance from a waypoint for the UAV x-, y- and z-position before going to the next waypoint. Only if all three are within e of their target coordinates, the server will forward the next waypoint. Since e is already given, it might be considered to be a base parameter. However, a different value for e has very little influence on the model behaviour and output values, which is why it is considered a tuning parameter. The inverse is true for a_{max}, which is why it is a base parameter.

The parameter e_2 is the maximum deviation a UAV can have in x-, y-, or z-direction when the waypoint has been reached for that particular coordinate. In other words: x-, y-, and z-position are only corrected if they deviate more than e_2 from their target value. If the UAV is closer than e_2 in a direction, no force is exerted along that particular axis: the UAV hovers undisturbed near the target coordinate.

There are more parameters that have to be set before the model can function, but these are initial values and will change throughout the simulation as well as in real life. In reality, these parameters already have a value. For example, the speed will be zero at start-up, but this has to be specified in the model in order for it to run. The initial parameters are the following:

Table 13.2: Initial Value Parameters

Symbol	Initial Value	Unit	Parameter	Remarks
$v_{x,y,z}$	0	m/s	Velocity	All Zero due to Standstill Start
x	$[x_{start}]$	m	Initial x-position	Determined by the Positioning System
y	$[y_{start}]$	m	Initial y-position	Determined by the Positioning System
z	$[z_{start}]$	m	Initial z-position	Determined by the Positioning System
$F_{x,y,z}$	0	N	Applied Net Force	Checks Previous Iteration

The parameters v_x, v_y and v_z have to be specified before each calculation loop, so their initial values have to be set before the first iteration. Since the UAV is at a standstill at the start, these values are zero. The same goes for F_x, F_y and F_z. The difference, however, is the fact that the forces from the previous iteration are checked to ensure the program will function within the set limits and doesn't accidentally apply forces that are outside of the force limits.

The x-, y- and z-coordinates have to be specified beforehand since the entire code hinges on the use of the difference between a target coordinate and the actual coordinate. Therefore, a starting position

has to be given for the first iteration. After that, the code will feed back the UAV position and the starting values are no longer used.

13.3.2. Waypoint Iteration

After all external and initial parameters are set, the code can start simulating flight. In real life, the equivalent would be completing the start-up sequence and commencing flight. The first step is determining where the UAV should go. As mentioned before in Section 13.2.2, this is determined by comparing the UAV position to the current waypoint. The current waypoint is determined by using an iteration parameter called i. i starts at 1, indicating that the target coordinates are retrieved from the 1st set of (x,y,z) parameters.

The number of waypoints is determined by getting the column length of the waypoint matrix. The waypoint matrix is a nx3 matrix with every row being a point in the flight envelope and every column being one dimension. The iterator first checks i for every loop. If i is smaller than n, i.e. the current waypoint is not the last waypoint, it checks if the UAV is close enough to go to the next waypoint. If so, i increases by one and the new waypoint coordinates are used. If the current waypoint is the last waypoint and the UAV is close enough to that waypoint, i will become larger than n and the code will perform its final iteration.

```
while i<=size(wp,1) #check whether or not final waypoint has been reached
    i = round(i);
    x_wp = wp(i,1); #assign target values for x, y, and z
    y_wp = wp(i,2);
    z_wp = wp(i,3);
    if i<size(wp,1) && abs(x_wp - x)<e ...        #check if UAV is close
            && abs(y_wp - y)<e && abs(z_wp-z)<e; #enough to current waypoint
        i = i+1;         #use next waypoint
        x_wp = wp(i,1); #set new target values
        y_wp = wp(i,2);
        z_wp = wp(i,3);
    elseif i==size(wp,1) && abs(x_wp - x)<e3 ...  #check if UAV is close
            && abs(y_wp - y)<e3 && abs(z_wp-z)<e3 #enough to final waypoint
        i=i+1;           #will force last iteration
    end
```

13.3.3. Situation Determination

Once the target waypoint has been determined, the distance deficiencies can be determined. This is an easy operation and simply is the UAV distance from the waypoint along the three axes. The corresponding code is shown below. The total distance d is calculated because it is used in future calculations. From this point on, dx, dy and dz will be used to determine every situation and calculate every force.

```
dx = x_wp-x;
dy = y_wp-y;
dz = z_wp-z;
d = sqrt(dx^2+dy^2+dz^2);
```

The next part of the code is determining what calculation to use. This is done by determining three things for the UAV-waypoint distance along every axis. There are three if-statements that are either true or false.Therefore three binary parameters for every axis. The first if-statement is shown below.

```
if abs(dx)<=d_min
    ix = 1;
end
```

Before every iteration, ix is set to zero. If ix=1, it indicated that there has to be a braking action in x-direction because dx is smaller than the minimum braking distance. This check is repeated for y and z. This yields ix, iy and iz, all being either 0 or 1. The sum of these variables then ranges from 0 to 3. The control flow follows from this sum. This means there are 4 different scenarios, all with their own

variations. If the sum is 3, the situation is very straightforward. This will be shown in Section 13.3.4. If the sum is 1 or 2, the situation becomes a little more complicated. Both sum results have 3 possible outcomes: for the sum of 1, one of the three is braking, whereas for the sum of 2, one of the three is nót braking. Section 13.3.4 and Section 13.3.5 will go into further detail about what follows next.

The most complicated situation, however, is if there is no braking at all. First, the desired velocity is determined. This is done by calculating vx, vy and vz as the fraction of their respective distance from their target coordinate over the total distance multiplied by the maximum velocity. Simply put, the result is that vx:vy:vz as dx:dy:dz. This ensures the velocity vector points directly towards the waypoint so the most efficient path is flown. The code used to obtain these desired velocities is shown below.

```
vxd = dx/d*v_max
vyd = dy/d*v_max;
vzd = dz/d*v_max;
vd_v = [vxd vyd vzd];
```

This is an example of a calculation which uses d, mentioned in Section 13.3.1. The vector vd_v is used because this enables us to re-use code and changing vector element indices instead of typing code multiple times. The if-statements below checking magnitude and sign already use elements from the vector v_v containing the actual speeds; vx, vy and vz in that order.

```
if abs(vxd)>abs(v_v(1))
    cx = 1;
else
    cx = 0;
end
if sign(vxd) == sign(v_v(1))
    sx = 1;
else
    sx = 0;
end
```

Simply put, these two statements check if a) The desired speed is larger than the actual speed – so if acceleration is required – and b) if the desired speed is in the same direction as the actual speed. There are three scenarios for every speed element. The first scenario is if the sign is opposite, i.e. sx = 0. Then cx does not matter because the speed has to be reversed first. However, if sx = 1, cx is 0 or 1, depending on the speed deficiency. This means there are three options:

1. sx = 0 & cx = not relevant
2. sx = 1 & cx = 0
3. sx = 1 & cx = 1

Since there are three parameters with each three possible values, there are $3^3 = 27$ possible scenarios. When taking a closer look, there are seven different code executions because of the recombination of several outcomes. In total, it leaves us with 46 total situations, which can be grouped into 10 different code executions. The logic behind these code executions will be explained in Section 13.3.5

13.3.4. Velocity Anticipation

The biggest shortcoming the models described in Section 13.2 is the fact that the paths the UAV took were not the most efficient. This was first demonstrated in Section 13.2.1 and carried on in the 2D and 3D models. Figure 13.5 is an example of the 2D case.

This is where velocity anticipation comes into play. Instead of looking at each component separately, this method takes into account every possible scenario and determines the correct desired total speed for that particular situation. It ensures the UAV always flies directly to its target and that the total UAV speed remains 1 m/s during operation, making the most of the time spent in the air; a more efficient flight path and a higher average speed means a shorter path execution time. Every group of situations has its own desired velocity profile. The concept is simple:

1. Determine your actual velocity

2. Identify your desired velocity
3. Determine the force required to bridge the gap
4. Apply the force

The first two steps are quite straightforward. The third step is where it gets tricky. This is because the steps taken in the code are increasingly complex. Determining the position and UAV distance to the target are very basic steps. Determining the situation is simply checking a number of if-statements. Determining the desired velocity is a little harder, but the constant in this situation is that the velocity will always point to the target. The net force, however, is a totally different story. This is why, even though steps 2 and 3 are closely linked, step 3 will be discussed in Section 13.3.5

Taking one step back, an example of velocity profiling will be given. Assume the UAV is nearing a waypoint and that, as a result of the UAV position, the following applies:

- dx is lower than d_min: ix = 1
- dy is higher than d_min: ix = 0
- dz is higher than d_min: iz = 0

Due to the fact that ix = 1, Fx takes precedence over Fy and Fz. This is done because braking is more important than accelerating. This holds for every single situation. The situation is the following: vx has to be reduced to zero, whereas vy and vz have to be as high as possible. The first step is calculating Fx, since Fx has precedence. This is done using Equation (13.1), proving once more this braking principle is the cornerstone of the path tracking mode. The exact force determination will be given in Section 13.3.5.

Because known is that vx should become zero and that the braking force loop will take care of this. Next, the two remaining forces have to be calculated. This is done by velocity profiling the remaining speed. What is meant by this is that vx has a certain value which will decrease during braking. Since the maximum total speed is constrained, there is only a given magnitude of speed left for vy and vz to take up collectively. The code to determine the desired vy and vz is given below.

```
if (ix||iy||iz) ~= 0
    if ix+iy+iz == 1
        if iz == 1
            p = 1;
            q = 2;
            r = 3;
         elseif iy == 1
            p = 1;
            q = 3;
            r = 2;
        else
            p = 2;
            q = 3;
            r = 1;
        end

        v_over = sqrt(v_max^2-v_v(r)^2);
        d_over = sqrt(d_v(p)^2+d_v(q)^2);
        vdp = d_v(p)/d_over*v_over;
        vdq = d_v(q)/d_over*v_over;
```

This might look like an incomprehensible section of code, but some explanation will greatly simplify things. The first if checks whether or not there is any braking. Since ix = 1, this is the case. As iy and iz are both zero, the sum is 1 and the second if is entered. After that, the else statement is activated, since iz nor iy are equal to 1. This is where the use of vectors comes in. The letters p and q are the non-braking axes and r is the braking axis. Hence, p = 2 which is y, q = 3 which is z and r = 1 which is x. The magnitude of the 'leftover' velocity is calculated by subtracting the actual vx – since v_v(1) is vx – from v_max in three dimensions using Pythagoras. The same goes for the leftover positioning difference.

The goal is to get vy and vz up to speed as fast as possible with Fy and Fz being proportional to dy as d_v(p) with p = 2 and dz as d_v(q) with q = 3. This is achieved by using the ratios of dy/d_min and

dz/d_min and multiplying these with v_over. It can be proven that the square root of the sum of the squares of these speeds is equal to v_over:

$$\begin{aligned} vdp^2 + vdq^2 &= \frac{d\_v(p)}{d\_over} * v\_over^2 + \frac{d\_v(q)}{d\_over} * v\_over^2 \\ &= \frac{d\_v(p)^2}{d\_v(p)^2 + d\_v(q)^2} * v\_over^2 + \frac{d\_v(q)^2}{d\_v(p)^2 + d\_v(q)^2} * v\_over^2 \\ &= \frac{d\_v(p)^2 + d\_v(q)^2}{d\_v(p)^2 + d\_v(q)^2} * v\_over^2 \\ vdp^2 + vdq^2 &= v\_over^2 \\ \sqrt{vdp^2 + vpq^2} &= v\_over \end{aligned} \tag{13.2}$$

This can be proven for all velocity profiles. Every velocity profile type has a different calculation method, but every profile of the desired velocity will be equal to v_max if the three components are summed. The desired velocities are then used to further determine the situation the UAV is in in addition to ix, sx, cx and their y and z counterparts.

13.3.5. Force Application

The third step of the sequence given in Section 13.3.4 can now be taken: Determining the force response to the UAV situation. Using the fact that dx < d_min, the first force calculation loop at the very beginning of the code is entered.

```
if abs(dx) < e2
        Fx = 0;
        ix = 1;
elseif abs(dx)<=d_min
    if sign(dx) == sign(vx) && abs(vx) > 0.01
        Fx = -sign(vx)*vx^2*m/(abs(dx)+1E-10);
    else
        Fx = 10*dx;
    end
    ix = 1;
    if abs(Fx)>F_max
        Fx = sign(Fx)*F_max;
    elseif abs(Fx) < 0.1
        if sign(dx)~=sign(vx)
            Fx = sign(Fx)*0.1;
        elseif abs(Fx) < 0.01
            Fx = sign(Fx)*0.01;
        end
    end
end
```

There are a few simple if-statements here. If the UAV is within hovering distance, the UAV is so close to the target coordinate that the UAV remains undisturbed. This prevents unnecessary adjustments of very low magnitudes. If the UAV is not close enough to hover, the next statement checks if the UAV is close enough to still apply braking. If this is the case, the approach zone statement is entered. If the UAV is moving towards the target – dx and vx are of equal sign – and the UAV travels faster than 0.01 m/s, the braking equation from Section 13.2.1 applies. Then ix is set to 1, regardless of what happens before.

A few checks then follow this statement: if the requested force is larger than Fmax, Fx is reduced to Fmax, keeping in mind the sign of the requested force. There is also a minimum requirement on the force: if Fx is smaller than 0.1 N and the UAV is moving away from the waypoint, Fx is raised to 0.1 N. This ensures the UAV hovers back to the target. If the UAV is still moving towards the target but Fx is lower than 0.01 N, Fx is raised towards that value to ensure the applied forces are not unrealistically small. This loop is exactly the same for y and z.

Fx has now been determined, which means vx will be taken care of. In Section 13.3.4, the desired values for vy and vz have already been determined. This was done by identifying the UAV situation using ix, iy and iz. This leaves the application of the forces. The corresponding code is the following:

```
if abs(v_v(p))<abs(vdp) && abs(v_v(q))<abs(vdq)
    F_v(p) = d_v(p)/d_over*sqrt(F_max^2-F_v(r)^2);
    F_v(q) = d_v(q)/d_over*sqrt(F_max^2-F_v(r)^2);
elseif abs(v_v(p))<abs(vdp) && abs(v_v(q))>=abs(vdq)
    F_v(p) = sqrt(F_max^2-F_v(r)^2);
    F_v(q) = 0;
elseif abs(v_v(p))>=abs(vdp) && abs(v_v(q))<abs(vdq)
    F_v(p) = 0;
    F_v(q) = sqrt(F_max^2-F_v(r)^2);
else
    F_v(p) = 0;
    F_v(q) = 0;
end
```

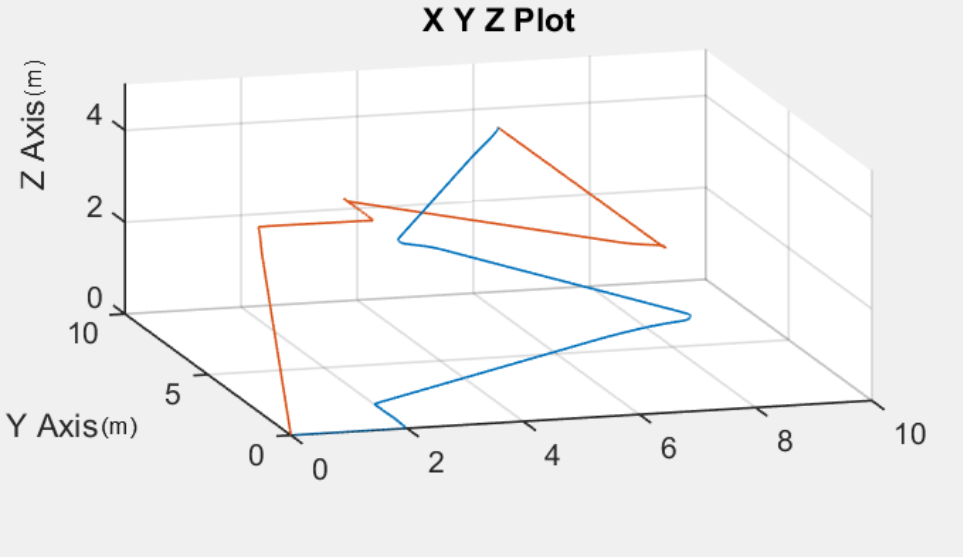
There are four more situations, depending on whether or not vdp and vdq are smaller than their actual counterparts or not. If both are smaller, there has to be an acceleration in both directions. The calculation of the forces is exactly analogous to the calculation of vdp and vdq, and Equation (13.2) can be applied in analogous fashion as well. If only one of the two is smaller, that force will be equal to the remaining force magnitude. Of course, if both v_v(p) and v_v(q) are equal to vdp and vdq, respectively, there is no force applied.

These forces are checked at the end of the code to ensure the total force F does not exceed F_max. If every calculation has been performed without errors, the new speed and position of the UAV is determined:

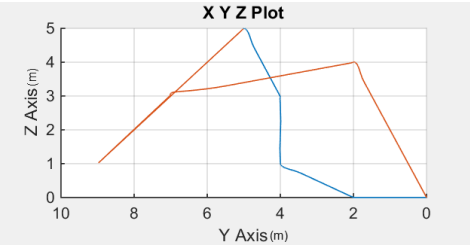
```
ax = Fx/m;
ay = Fy/m;
az = Fz/m;

vx = vx + ax*dt;
vy = vy + ay*dt;
vz = vz + az*dt;

x = x + vx*dt;
y = y + vy*dt;
z = z + vz*dt;
```

The steps to calculate vx, vy vz and then x, y and z are discrete integrations with a timestep of 0.01 s. These values are fed back into the start of the iteration as long as the UAV is not close enough to the final waypoint. The 3D flight path of 2 UAVs can be shown in a 3D simulation. The result of this simulation is shown in Figure 13.6 below.

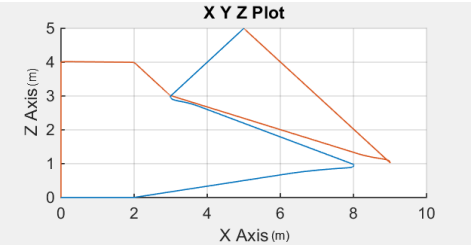
The plots in Figure 13.6 show that UAVs follow their specified path until both have reached (5,5,5), which causes the simulation to stop. A slight modification to the end conditions will enable the UAVs to comply with Item **Hor-UAV-10**, which specifies that the UAV shall have hover capability. If the server orders the UAV to maintain the same target coordinates over a period of time, the UAV will stay at that coordinate. This is the required hover capability. It also ensures the PTM enables the UAV to meet all applicable requirements treated in this chapter.



(a) 3D View of Both Paths



(b) YZ Side View



(c) XZ Side View

Figure 13.6: An Example of the 3D Path Tracking Ability, Showing Two UAV Paths

14 | Collision Avoidance

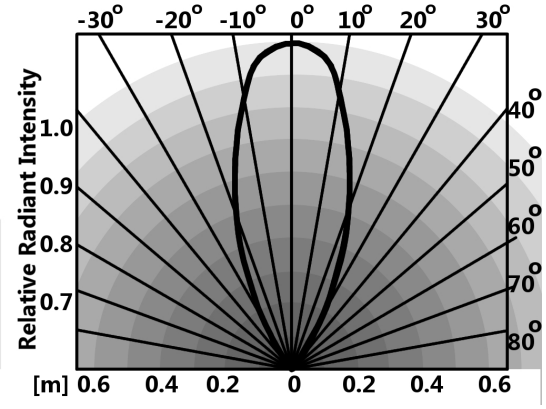
In this chapter the collision avoidance system shall be discussed, in order to comply with requirement **Hor-Ctrl-1**. First in Section 14.1, the lay-out of the sensors and type of sensors are determined. Then in Section 14.2 the algorithm used for the collision avoidance is described.

14.1. Lay-Out

The previously decided upon method of collision avoidance, ultrasonic sensors [1], has been determined to be infeasible because of the phenomenon called "cross-talk". The ultrasonic sensors all use sound-waves at approximately the same frequency, which means it is not possible to determine the time the sound wave takes to bounce against an obstacle when that obstacle "talks back". Because of the limited frequency range and high number of sensors, the use of different frequencies for the sensors is not an option.

Therefore, the detection system will use infrared (IR) sensors and receivers to provide information about close-by UAVs. These are depicted in Figure 14.1a.[a] The distance estimation will be done using signal strength, and any signal from neighbouring UAVs will be regarded the same as the reflected signals.

(a) The SN-IRS-01 IR Sender and Receiver



(b) Relative Radiant Intensity Versus Angular Displacement of the SN-IRS-01

Figure 14.1: The SN-IRS-01 IR Sensors

As can be seen in Figure 14.1b, the IR sensors have a field of view of approximately 20 degrees. In this field, any oncoming obstacles can be detected. This information is an analogue voltage value, which will need to be converted to a distance using a formula or a look-up table. This distance value will then be used to determine an appropriate collision avoidance manoeuvre.

In order to detect all UAVs, the sensors will need to provide information about any objects with a radius bigger than 0.1 m from at least 0.5 m away. This is stated in requirement **Hor-Ctrl-23**. For the smallest UAVs, this means a maximum angle between the UAVs of 18.9°, which is determined using Equation (14.1). r is the radius, s the minimum distance to the approaching UAV, and β is the resulting maximum angle between the IR sensors.

$$2 \cdot arctan\left(\frac{r}{r+s}\right) = \beta \tag{14.1}$$

Thus, 8 sensors will be used. For the biggest UAVs, with a radius of 0.25 m, this will mean a total of 16

[a] SN-IRS-01 infrared sensor, URL `http://www.cytron.com.my/p-sn-irs-01`, accessed on 10 June 2016

sensors. These sensors weigh 0.6 g, so the total weight added is $5 - 6 - 8 - 9$ g for the smallest to the biggest UAVs.[b]

14.2. Collision Avoidance System

In order to avoid in-flight collisions, a collision avoidance mechanism has been developed. Potential collision avoidance scenarios are split into two main categories: orbit collisions and path collisions. These are shown in Figure 14.2. The collision avoidance system is only qualitatively evaluated. Quantitative estimates will be made during the production phase.

14.2.1. Orbit Collision Avoidance

A UAV with a smaller orbit radius has a higher velocity than a UAV with a bigger radius. If the proximity sensors detect a nearby UAV, the slower UAV will temporarily fly an orbit with a bigger radius until the risk of collision has passed. Because there is more space between the UAV orbits to the outside of the system, the movement of the outer UAV to the outside of the system is less likely to cause another collision than if the UAV with the smaller radius changes its orbit. This goes for all the UAVs except Mercury, which will divert to an orbit with a smaller radius. This will mean both Mercury and Venus will divert from their orbit if they detect a collision.

(a) Two UAVs in Orbit

(b) Two UAVs That are flying a Path

Figure 14.2: The Collision Avoidance Protocols

As can be seen in Figure 14.2a, the outer UAV detects the inner one and moves to the outside. The UAV with a smaller orbit will not react to this collision risk, unless the UAVs fly so close that it will need to react as well. In that case, it will fly at a slower speed until the outer UAV has moved aside, and then continue in its orbit.

The UAVs will determine if they are the inner or the outer planet through the angle at which they detect the other UAV. If the UAV "sees" the other one at the left of itself, it will divert to the right, to an orbit with a bigger radius. If it sees a UAV to the right, it will not react until they are so close that it will need to slow down to avoid a collision.

14.2.2. Path Collisions

When either of the UAVs is not in an orbit but flying a path to a target position, a different protocol is in place. When a UAV detects a potential collision, the faster, overtaking UAV will divert its course. This is shown in Figure 14.2b, where the UAV at the left of the picture diverts its course to avoid the UAV in the upper right. In order to avoid these sort of collisions as much as possible, the UAVs that goes to the charge station will fly at a different altitude than the orbiting UAVs. UAVs that are leaving the charge station will have a different altitude as well.

A descending UAV will divert to avoid a UAV flying at a lower altitude. The photo-resistors will be used to detect these UAVs. This will allow the UAV to fly an unobstructed path to and from the charge station, and thus fulfils requirement **Hor-UAV-14**.

[b] Datasheet 5 mm LEDs, URL `http://www.dominant-semi.com/userfiles/file/5mmovallamphighbrightnessingan-catalogue-v2.pdf`, accessed on 20 June 2016

14.3. Verification and Validation

A more specific collision avoidance protocol can be determined when the capabilities of the sensors are more accurately known. This can be accomplished through contacting the supplier, or by ordering them and testing them. Verification of this system can be done through the use of a simulation, which will allow for a more complete overview of possible collision scenarios and the best tactics for their avoidance.

When production is started and the sensors are ordered, they will need to be tested to determine their capabilities. If a working prototype is built, the collision avoidance protocols will need to be tested. The protocols will need to be further developed, or more sensors added, if this testing proves that they do not sufficiently prevent collisions.

V

Contributing Elements of HORUS

15 | Sun Design

In this chapter the Sun design will be described. In Section 15.1 the visualisation of the Sun will be discussed. Also in this section the lay-out of the Sun is shown. Then in Section 15.2 the suspension system will be explained. Finally, a mass estimation will be performed in Section 15.3.

15.1. Visualisation of the Sun

For the light source four Feit LED BR40 Bulbs will be used. Four lights bulb are chosen to create a bright and diffuse light source. Important specifications of this light bulb are shown in Table 15.1.[a]

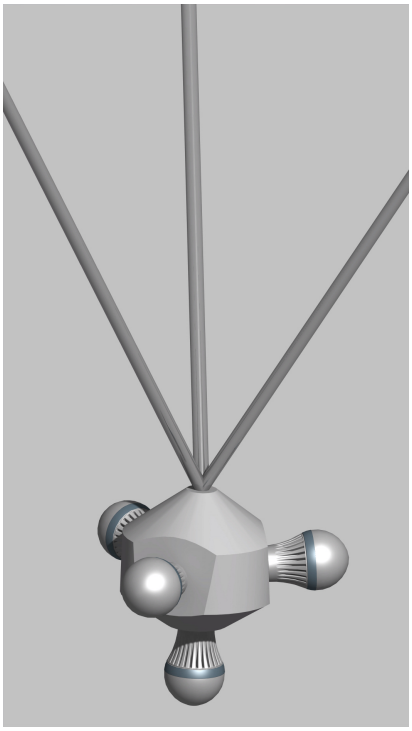
Table 15.1: Specifications Feit Light Bulb

Specifications Feit Light Bulb	
Volts	120 V
Life Year	22.8
Life Hours	25,000
Color Temperature	3000 K
Energy Used	36 Watts
Lumens	2500
Beam Angle	120°
Mass	0.05 kg
Features	Dimmable, Instant to Full Brightness

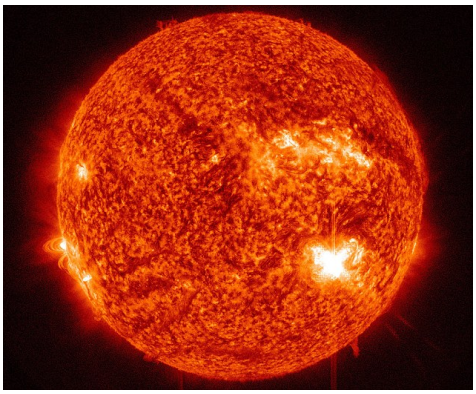
This light bulb is chosen because it is one that gives the brightest light available on the market. Another advantage is that it is dimmable. Using this dimmable light, the brightness of the Sun can be adjusted for different modes. Furthermore, the colour temperature is 3000 K, which is warm white and most suitable to represent the Sun. The light bulb is very sustainable; when operating for eight hours a day every day, the lifetime will be over eight years.

The configuration of the light bulbs can be seen in Figure 15.1a. In the horizontal plane three light bulbs will be placed. One bulb had a beam angle of 120°, using three bulbs will thus cover an entire circle. One extra bulb pointed to the floor is placed to ensure full light coverage. This configuration allows the cables to travel from the ceiling to the Sun without colliding with the bulbs. Furthermore, the horizontally placed bulbs are placed under a slight upward angle of 5° to increase coverage. An area of beam angle 50° at the top is not covered with light, in order to keep the cables in the shadow.

[a] 2500 Lumen 3000K Dimmable LED BR40, URL `http://www.feit.com/led-lamps/performance/performance_led/performance_led/br/r_and_br/br40-dm-2500-3k-led`, accessed on 27 May 2016



(a) Light Bulb Configuration



(b) Lay-Out Sun Sticker

Figure 15.1: Light Bulb Configuration and Sun Sticker

The material of the outer shell will be transparent Perspex. The thickness of the Perspex sphere will be 5 *mm*. The light transmittance of transparent Perspex is 92% for a thickness of maximum 6 *mm*.[b] A thickness of 5 *mm* is chosen, to allow for production uncertainty. A sticker representing the surface of the Sun will be applied to the Perspex sphere. This will allow for a more accurate representation of the Sun, due to the detailed surface representation and light colour. This sticker will also reflect visible light. Because the light bulbs used are very powerful, the decrease of light transmittance from the Perspex as well as the sticker will not be a problem. The sticker will have a lay-out such as the Sun surface shown in Figure 15.1b.

The Perspex sphere is produced in two halves, which will be glued together. At the top of the sphere, the cables enter via a opening of 30 *cm* diameter. This opening size allows for maintenance activities, as it is wide enough for a hand and arm to access the light bulbs.

15.2. Suspension of the Sun

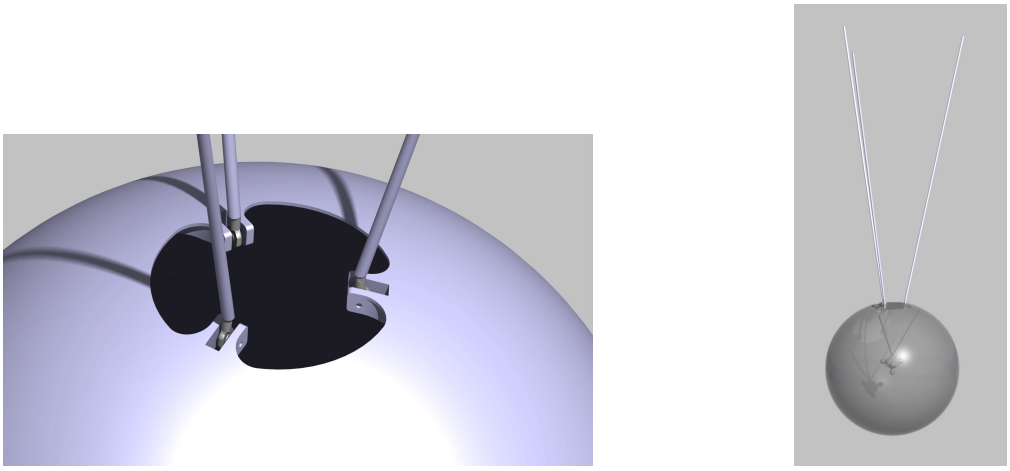
For the suspension, three suspension points at the ceiling will be used with at every suspension point a fixed pulley. No loose pulleys will be used, because this will interfere with the representation of the Sun as the suspension system is more easily noticeable. Furthermore, the loads can be handled by a single pulley/motor system so adding extra loose pulleys to reduce the loads is not necessary, this complies with **Hor-16**. Also, because three suspension points are used, the load is divided by three and therefore a less powerful motor is needed.

It shall be possible to change the vertical height of the Sun. This will be done by the use of a motor. This single motor regulates the length of the three ropes. The height can be set by the supervisor. This complies with **Hor-16**.

The cables are connected to the sphere using rod ends. A rod end is placed in the “fork” shaped part of the Perspex sphere and connected using a bolt. This connection can be seen in Figure 15.2a. The total Sun design can be found in Figure 15.2b.

[b] Perspex- Optical and Transmission Characteristics, URL `http://www.plexiglas.com/export/sites/plexiglas/.content/medias/downloads/sheet-docs/plexiglas-optical-and-transmission-characteristics.pdf`, accessed on 30 May 2016

The cables of the internal structure are connected to the sphere using the same method. It uses rod ends placed in the same "fork" shaped part of the Perspex sphere. When the bolts are removed, the internal structure can be taken out of the shell for maintenance activities. For the connection of the rod end of the cables connected to the ceiling and the rod ends of the cables of the internal structure, separate bolts are needed. This is done in order to make the movements independent of each other. The cables providing the light bulbs from electricity, will be wound around the cables rising to the ceiling.



(a) Rod End Connection **(b)** Entire Sun Design

Figure 15.2: Sun Lay-Out

15.3. Mass Estimation

The mass of Perspex is 5.32 kg/m^2 for a thickness of 5 mm.[c] Because the diameter of the Sun is 1.0 m, as determined in Chapter 3, the area of the sphere is 3.1 m^2. The gap at the top of the sphere is accounted for in this calculation. The mass of the Perspex shell will be 16.33 kg. Adding the four light bulbs chosen in Section 15.1 will add 200 g. The cables inside the sphere are made out of steel. The inner structure to hold the light bulbs is made out of plastic. The mass of this inner structure, excluding the light bulbs, is calculated using CATIA and has a value of 237 g. This totals at 16.767 kg which can also be seen in Table 15.2. It is assumed that the sticker does not add any mass to the Sun design.

Table 15.2: Mass Estimation of the Sun

Component	Amount [-]	Mass [kg]
Perspex	-	16.33
Light Bulbs	4	0.05
Inner Structure	-	0.237
Total	-	16.767

[c] Plexiglass - Weights per square foot, URL `http://www.eplastics.com/Plastic/plastics_library/plexiglass-weight-per-square-foot`, accessed on 27 May 2016

16 | Ground Station

In this chapter the final design of the ground station will be described. First, in Section 16.1, the charge station components and configuration will be treated. This involves the geometry of the cabinet as well as the way the UAVs will enter the charge station. Then, in Section 16.2, the user interface and the supervisor interface will be discussed. In Section 16.3, the server architecture will be evaluated.

16.1. Charging

The charging system will provide the contact surfaces for the conductive charging, and the LEDs for the precision landing. Because there is no weight requirement on the ground station, the main focus of the system will be a good fit to the UAV. This will mean that charging resistances will be as low as possible, as well as the structural loading. Because Li-Po batteries will need to be carefully monitored and controlled during charging, the charge station will include a battery charging integrated circuit. This is necessary to assure a high battery life of the Li-Po batteries as well as to prevent combustion. The charge station will also include an Alternating Current (AC) to Direct Current (DC) converter, because the batteries are charged using DC, while the mains connection provides AC.

In Figure 16.1 and Figure 16.2 the charge station is depicted. The height of the charge station is 1.8 m. As can be seen the UAVs will enter the charge station from the top and then land on the pedestals, which can be placed in any order. Each UAV will have an individual landing station and will charge via conductive charging. The dimension of the charge station are 7.2 by 1.8 by 1.8 meters. The dimensions are chosen such that the charge station can be part of the HORUS experience as well as having a safe place for the UAVs to charge. The UAVs cannot be reached by visitors, so that they cannot be stolen or damaged. The UAVs can always be reached by a supervisor, however, via small doors at the side of the cabinet.

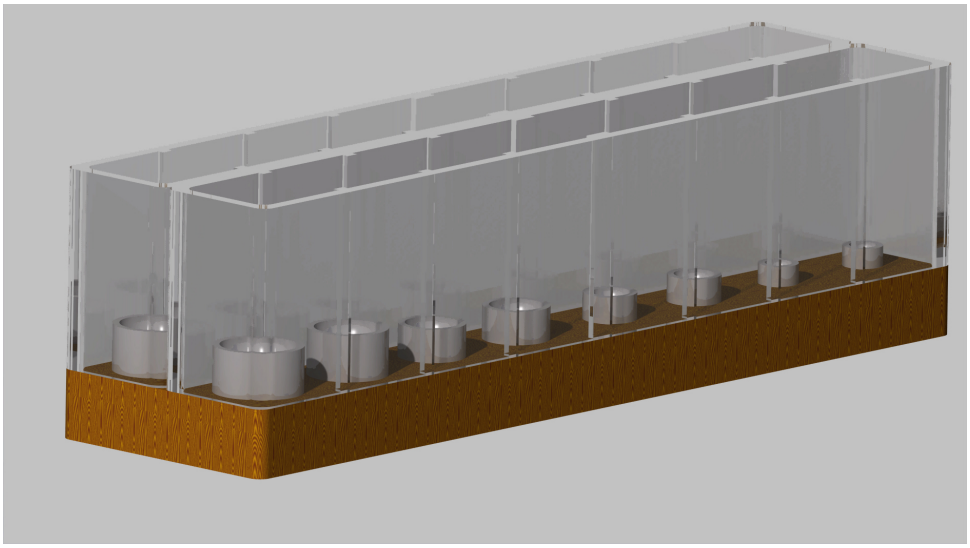


Figure 16.1: Charge Station Lay-Out

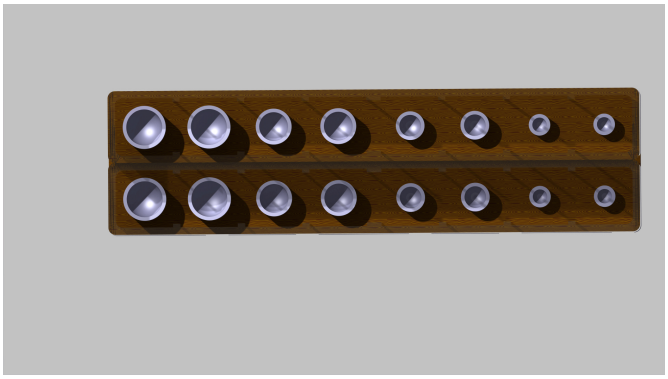


Figure 16.2: Top View of Charge Station

16.2. Interface

In this section, the interfaces of the terminal as well as the user interface will be presented. This will be done in Section 16.2.1 and Section 16.2.2 respectively.

16.2.1. Terminal

In this section the terminal will be defined geometrically as well as aesthetically. The design should be such that children as well as adults are able to easily access and use the terminal for its intended purposes. In Figure 16.3 the proposed layout of the terminal is depicted. The terminal consists of a touch screen display of dimensions $1x2\ m$. It is placed on a stand and the lower side of the screen has a height of $0.5\ m$. The screen itself has a width of $2.7\ m$ and a height of $0.83\ m$.

Figure 16.3: Terminal Lay-Out

The material used for the stand on which the touch screen has been placed is oak. This can be painted to any desired colour if required, as this is dependent upon the institution in which it is placed. The decision can be made to cover the stand with another material if oak is not sufficient.

16.2.2. User Interface

In this section the user interface of the ground station will be defined. The screen described in the previous section will be used to provide a graphical user interface. This user interface will have to provide information regarding HORUS and should display requested information in a child friendly manner. The use of a touch screen was considered to be mandatory for the design. The size of this screen is presented in Figure 16.3 where a CATIA model of the terminal is presented.

In Figure 16.4 a prototype is presented. The figure shows the interface with a selected planet, in this case Neptune, and provides information. When one presses the 'Present Neptune' button on the right side of the screen, the UAV representing Neptune will present itself by lowering its altitude. If one was to press the "Present Neptune" button this would then become a "Switch back to Default" button to put HORUS back in default mode. One can select planets either from the live render on the left side of the screen or from the list provided on the right side of the screen.

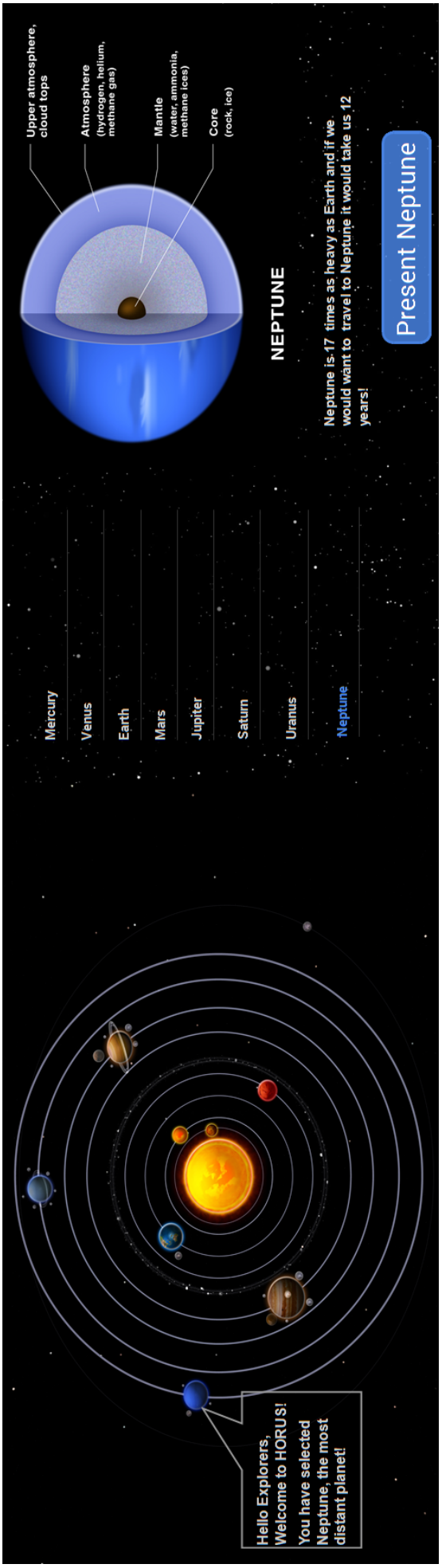


Figure 16.4: User Interface Lay-Out

16.2.3. Supervisor Interface

In Figure 16.5 the interface of the supervisor is depicted. An Ipad has been used to visualise the interface as this is the expected tool of use. The figure shows the "UAV" menu where each UAV present in HORUS can be selected and inspected. The live data is depicted on the right table, the data presented can be tailored to the needs of the supervisor. The "Modes" tab will be able to do the same as the terminals user interface which is then also depended upon the rights of the supervisor. As for the "Ground Station" tab this will be a similar tab as the UAVs tab with live data regarding the ground station and its status. Lastly, the "Hall" tab will give the supervisor control of the hall in which HORUS is operational. This shall include for instance dimming the lights.

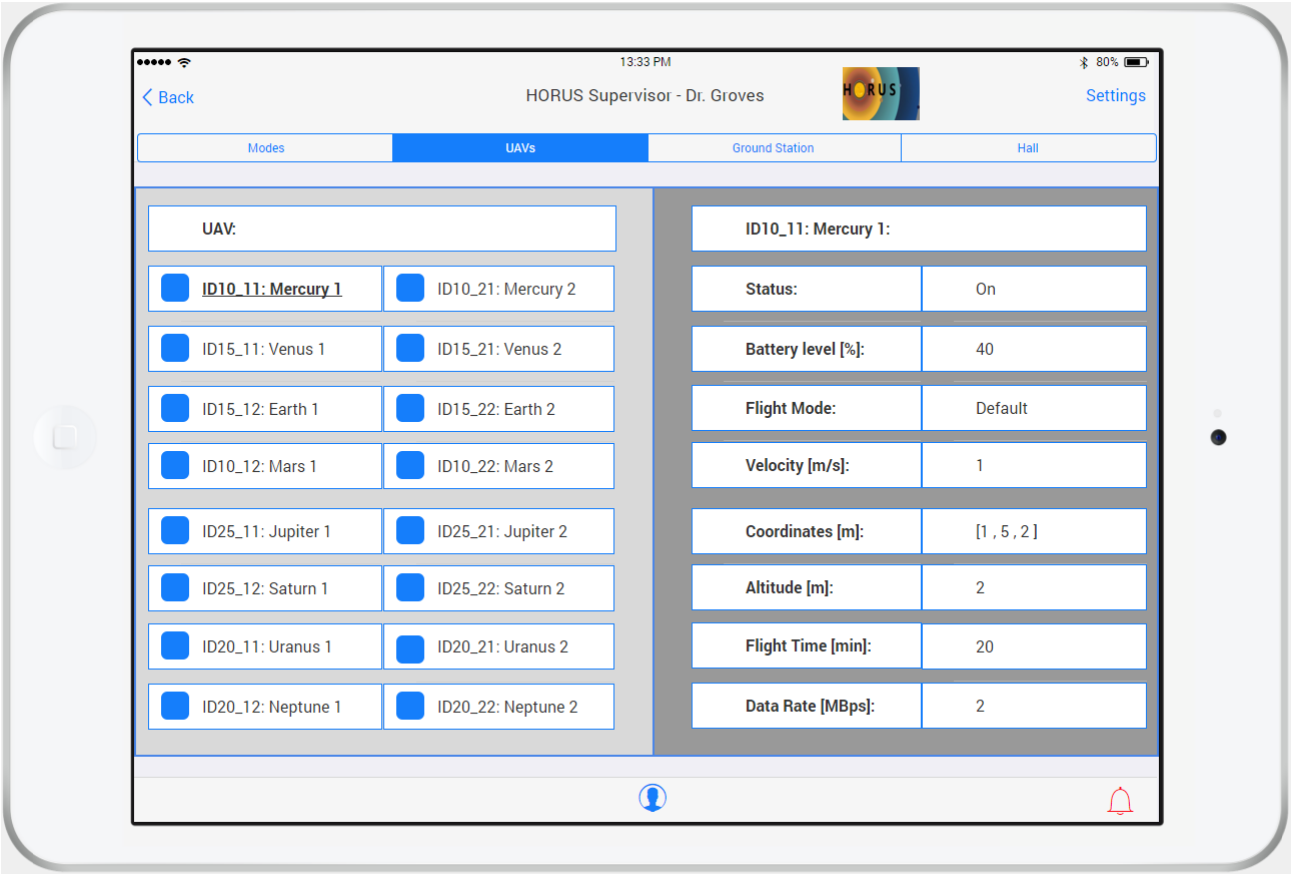


Figure 16.5: Supervisor Interface Lay-Out

16.3. Server

The server, as stated in [1], will be sending the provided modes to the UAVs as they are put in by supervisor or user. The data received from the UAV will be logged by a log system and handled via a processing unit. The data to be send will then be send to the UAVs via a Bluetooth signal. The server shall then provide feedback to the terminal and the supervisor interface. The UAVs can send signals back to the server, also via a Bluetooth Connection. The server has an antenna to receive and transmit signals. The server is located outside the flight envelope.

The server is a computer, which will be the Hp Elitebook as it provides all the required functionalities.[a] The server will have at least the following elements:

- AC to DC Converter: The server uses the mains connection via a wired connection to power its systems. The server will use DC to power the systems. Mains connection provides AC, so this should be converted to DC.

[a] HP EliteBook 8570w, URL `https://tweakers.net/pricewatch/322551/hp-elitebook-8570w-(ly556et).html`, accessed on 13 June 2016

- Motherboard: The motherboard of the server is the heart of the computer. It controls all the elements in the server and receives the power.
- Processor: Carries out the instructions of the computer program.
- Data Log: The Data of the UAVs is logged in the server and stored in a data log.
- Antenna: The server can send and receive Bluetooth signals through an antenna.

16.4. Requirements Compliance

In this chapter the following requirements have been met in their respective sections: Section 16.2.2 meets **Hor-Gnd-1** due to the fact that the interface is designed in a user-friendly manner. Then in Section 16.2.3, **Hor-Gnd-2** has been treated. In Section 16.3 is stated that the server has data storage capabilities and is located outside the flight envelope. This means that requirements **Hor-Gnd-3** and **Hor-Gnd-8** are met. Second to last, Section 16.2.1 **Hor-Gnd-4** and finally, in Section 16.1, requirements **Hor-Gnd-6** and **Hor-Gnd-9** have been met.

17 | Hardware

For the different functionalities of the UAV itself or the communication between UAV and server, hardware elements are needed. First the hardware elements that are chosen are summarized in Section 17.1. The Hardware Block Diagram (HBD) can be found in Section 17.2.

17.1. Hardware Design Choices

During the designing of HORUS, hardware elements are chosen which are compatible with the design goals. In Table 17.1 the hardware elements used in HORUS are summarised. The more detailed explanation of the elements is covered in the chapter discussing that subject.

A Raspberry Pi Model B (Generation 3) is chosen to use as the micro-computer. Important specifications are that it comes stock with Bluetooth 4.1 capabilities, multi-core processor and on-card memory of 1 GB.[a] It also has the highest clock speeds and bit architecture (x64). For collision avoidance, IR sensors are used. For the smallest UAV, 8 sensors are needed in order to detect other UAVs. For the largest, 14 sensors are needed. Every UAV is equipped with 2 TP4500-3SHV batteries. The batteries are custom shaped and have a mass of 490 g. The largest UAV has one additional TP450-3SM70J battery which provides the additional capacity. Due to the fact that the size of the UAV influences the required thrust, the propeller differs per UAV. The smallest uses the Zagi Carbom 5.1 x 4.9 propeller. The other UAVs all use APC E Propeller, only the sizes differ. For the same reason as the propellers, the motor differs per size UAV. The charge system requires DC. Due to the fact that the mains connection delivers AC, an AC to DC converter is needed in the charge station. Also, each battery will feature an internal monitor specifically selected for single battery charging. The hardware element inside the server are the motherboard, processor and antenna. A HP EliteBook 8570w computer will be used as server.

Table 17.1: Hardware Elements

Subject	Element	Product	Amount (Per)
Communication & Control	Micro-Computer	Raspberry Pi	1 (UAV)
	Collision Avoidance	IR Sensors	8-16 (UAV)
	Antenna	Bluetooth Antenna	1 (UAV)
	Bluetooth Beacons	-	5 (Hall)
	IMU	-	1 (UAV)
	Photo-resistors	-	4 (UAV)
Power	Battery	Li-Po	2-3 (UAV)
	Resistor	-	11-17 (UAV)
Propulsion	Motor	Suppo A2212-6 Gold, KEDA TR2830/14 Justgofly 450T2, E. Rocket 2215-3400	2 (UAV)
	Propeller	Zagi Carbon/APC E	2 (UAV)
Charge Station	Converter	AC to DC	1
	Battery Charger	TP820HC	1
Server	Motherboard	-	1
	Processor	-	1
	Antenna	Bluetooth Antenna	1

17.2. Hardware Block Diagram

The Hardware Block Diagram (HBD) shows the elements of the system and their interactions. In Figure 17.1 the HBD can be found of the entire HORUS system.

[a] Raspberry Pi 3 on sale, URL `https://www.raspberrypi.org/blog/raspberry-pi-3-on-sale/`, accessed on 1 June 2016

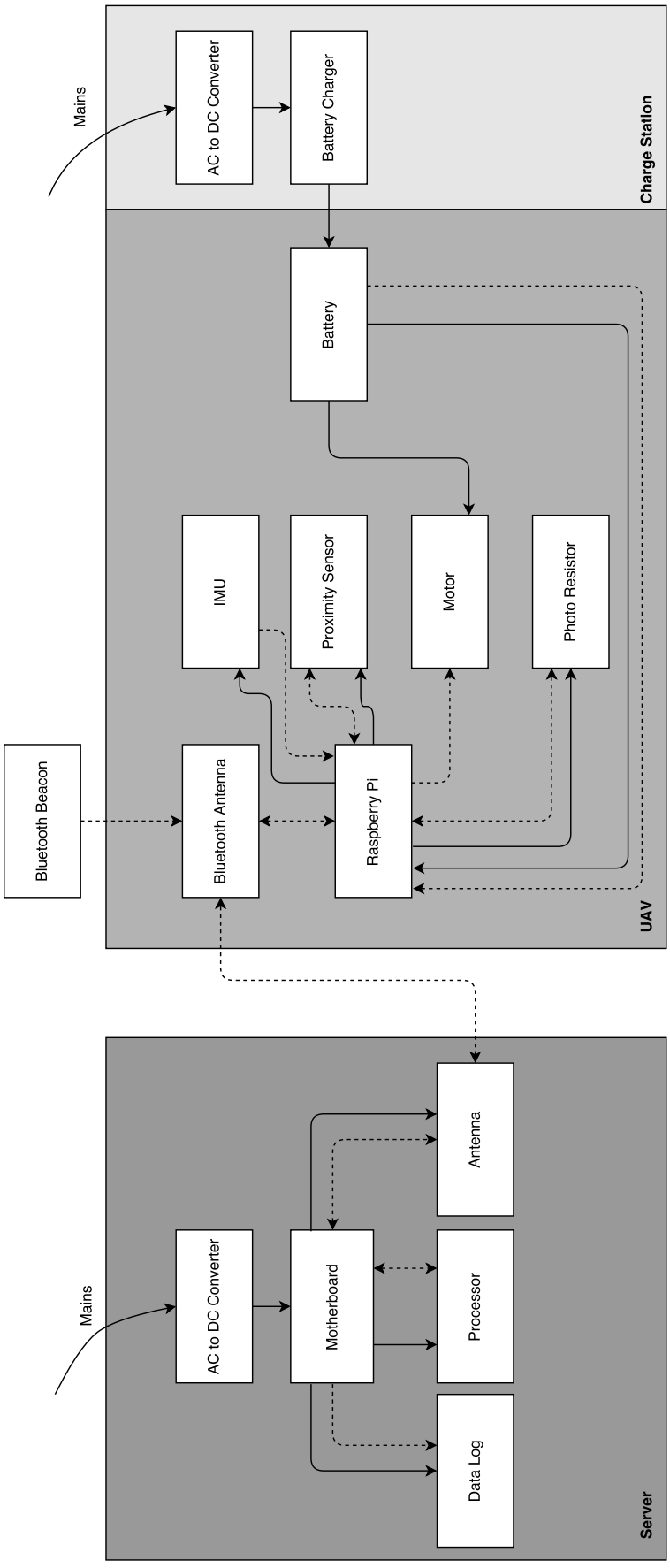


Figure 17.1: Hardware Block Diagram; the Solid Lines indicate Power Transfer, the Dotted Lines indicate Software Transfer

VI

Performance of HORUS

18 | Compliance Matrix

The requirements set by the stakeholder and the internal requirements need to be met for the project to be deemed a success. This chapter will introduce the compliance matrix and a feasibility study. The compliance matrix can be found in Table 18.1. The matrix lists all the identifiers of the set requirements with a check mark when the requirement is met, accompanied by the chapter in which they are deemed met. The list of requirements can be found in Appendix B.

Table 18.1: Compliance Matrix

Requirement	Compliance	Chapter	Requirement	Compliance	Chapter
Hor-1	✓	Chapter 4	**Hor-Ctrl-1**	✓	Chapter 14
Hor-2	✓	Section 3.5	**Hor-Ctrl-2**	✓	Sections 3.5 and 10.2
Hor-3	✓	Section 3.5	**Hor-Ctrl-5**	✓	Section 21.3
Hor-4	✓	Section 3.5	**Hor-Ctrl-6**	✓	Section 3.2.3
Hor-5	✓	Section 3.5	**Hor-Ctrl-7**	✓	Section 3.2.3
Hor-6	✓	Section 3.5	**Hor-Ctrl-8**	✓	Section 9.1
Hor-7	✓	Section 3.5	**Hor-Ctrl-9**	✓	Section 9.1
Hor-8	✓	Chapter 25	**Hor-Ctrl-10**	✓	Section 9.1
Hor-9	✓	Chapter 7 and Section 6.2	**Hor-Ctrl-11**	✓	Section 9.1
Hor-10	✓	Section 27.1	**Hor-Ctrl-12**	✓	Section 3.5
Hor-11	✓	Section 21.3	**Hor-Ctrl-13**	✓	Section 3.5
Hor-12	✓	Chapter 2	**Hor-Ctrl-14**	✓	Section 3.2
Hor-13	✓	Chapter 2	**Hor-Ctrl-15**	✓	Section 9.1
Hor-14	✓	Chapter 2 and Section 3.5	**Hor-Ctrl-16**	✓	Section 6.2
Hor-15	✓	Section 3.2	**Hor-Ctrl-17**	Not Met	-
Hor-16	✓	Section 15.2	**Hor-Ctrl-18**	Not Met	Section 6.1
Hor-17	✓	Chapter 2	**Hor-Ctrl-19**	Not Met	-
Hor-UAV-1	✓	Section 7.3	**Hor-Ctrl-20**	✓	Section 13.1
Hor-UAV-2	✓	Section 7.3	**Hor-Ctrl-21**	✓	Section 5.3
Hor-UAV-3	✓	Section 13.1	**Hor-Ctrl-22**	✓	Section 13.1
Hor-UAV-4	✓	Chapter 4	**Hor-Ctrl-23**	✓	Section 14.1
Hor-UAV-5	✓	Section 3.5	**Hor-Gnd-1**	✓	Section 16.2.2
Hor-UAV-7	✓	Section 7.5	**Hor-Gnd-2**	✓	Section 16.2
Hor-UAV-8	✓	Sections 7.5 and 7.6	**Hor-Gnd-3**	✓	Section 16.3
Hor-UAV-9	✓	Section 3.1	**Hor-Gnd-4**	✓	Section 16.2.1
Hor-UAV-10	✓	Section 13.3	**Hor-Gnd-6**	✓	Section 16.1
Hor-UAV-12	✓	Section 10.2	**Hor-Gnd-7**	✓	Section 6.2
Hor-UAV-13	✓	Section 19.2	**Hor-Gnd-8**	✓	Section 16.3
Hor-UAV-14	✓	Section 14.2	**Hor-Gnd-9**	✓	Section 16.1
Hor-UAV-17	✓	Section 3.1			
Hor-UAV-18	✓	Section 11.2			
Hor-UAV-19	✓	Section 11.2			
Hor-UAV-20	✓	Section 5.3			
Hor-UAV-21	✓	Section 5.3			
Hor-UAV-22	✓	Chapter 8			

18.1. Feasibility

The matrix shows that all but three requirements have been met. Starting with **Hor-Ctrl-18**, the power limit, which was set at 300 W, has been exceeded. The power required for the UAVs ranges from 340 – 520 W. Although the requirement is not met, the UAVs are deemed flyable with a slightly larger power consumption than preferred. **Hor-Ctrl-17** and **Hor-Ctrl-19** have also not been met. Contrary to the power requirement, these requirement have not been met due to the decision to exclude the e-paper as visual representation on the UAVs. The coating will use no power and therefore will not generate a thermal increase. Thus, the design only fails to meet three requirements of which zero are stakeholder requirements. These requirements do not have a big impact on any of the core abilities of the UAVs, thus the design team feels the design is feasible.

19 | Performance Analysis

An analysis regarding the performances of the UAVs will be provided in this chapter. First the flight profile diagram of the UAVs will be treated in Section 19.1. Then in Section 19.2 the noise characteristics will be discussed. First the noise propagation is treated and then a noise prediction is made.

19.1. Flight Profile Diagram

In this section the typical flight profile is outlined. It starts with the activation of all UAVs. For each pair of twin UAVs, one will take-off and move to its default starting position while the other UAV stays in the charge station. This starting position depends on the identity. When the UAV is in position it will start orbiting in default mode. When all UAVs are in position they will start to orbit in the default mode until a new mode is assigned to them. More information regarding these modes can be found in Chapter 3. The UAVs will continue to orbit until they have reached their minimal allowed state-of-charge. When the battery is drained to the set depth of discharge, the UAV will go to the charge station to swap with its twin. The twin will then continue orbiting in the assigned mode. When the twin has to charge, it will swap again with the first UAV, therefore constantly keeping one UAV of the pair in operation. More information regarding the capacity, charging time and discharge time of the batteries can be found in Section 6.2. When the system is shut down, the UAVs that are already in the charge station will shut down, while the orbiting UAVs will first dock and then shut down. The described flight profile is graphically depicted in Figure 19.1.

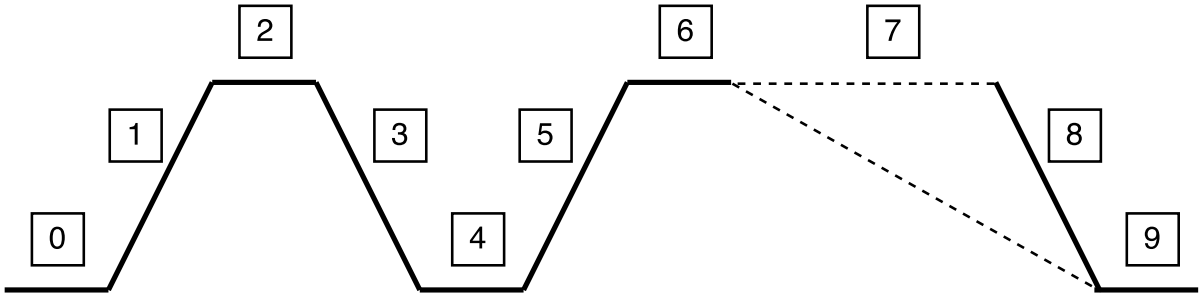


Figure 19.1: UAV Flight Profile Diagram

The functions designated by the numbers in Figure 19.1 correspond to the following activities:

- **0** Turn On.
- **1** Go to default mode starting position.
- **2** Perform assigned mode until low state-of-charge.
- **3** Move back to charge station and start charging.
- **4** Charge until full capacity and wait until twin UAV has to charge.
- **5** Move to the position determined by the current orbit mode.
- **6** Perform in designated mode until low state-of-charge.
- **7** Repeat steps 3, 4, 5 and 6 until told to shut down.
- **8** (Optional) Go to charge station and dock.
- **9** Shut Down.

19.2. Noise Characteristics

In this section the noise characteristics of HORUS will be discussed. First the noise propagation will be discussed in Section 19.2.1. Then in Section 19.2.2 the noise produced by the UAVs will be discussed and a conclusion regarding the mission requirements will be made.

19.2.1. Noise Propagation

Sound emitted by a noise source is not equal to the sound received by the observer. The sound propagation is subject to a number of different factors. In this section first the directivity will be discussed followed by the propagation of noise as it travels from the source.

Directivity

Directivity is the effectiveness of the sound radiation in each direction. There are generally three patterns considered, monopole, dipole and quadrupole. For HORUS the monopole distribution of noise has been assumed as this is the most intense way of noise propagation. The result of this decision will provide a directivity (D_H) of 1, which will be used in Equation (19.1), (19.2) and (19.3). The three different propagation patterns are shown in Figure 19.2, where the assumed directivity is the left-most picture.

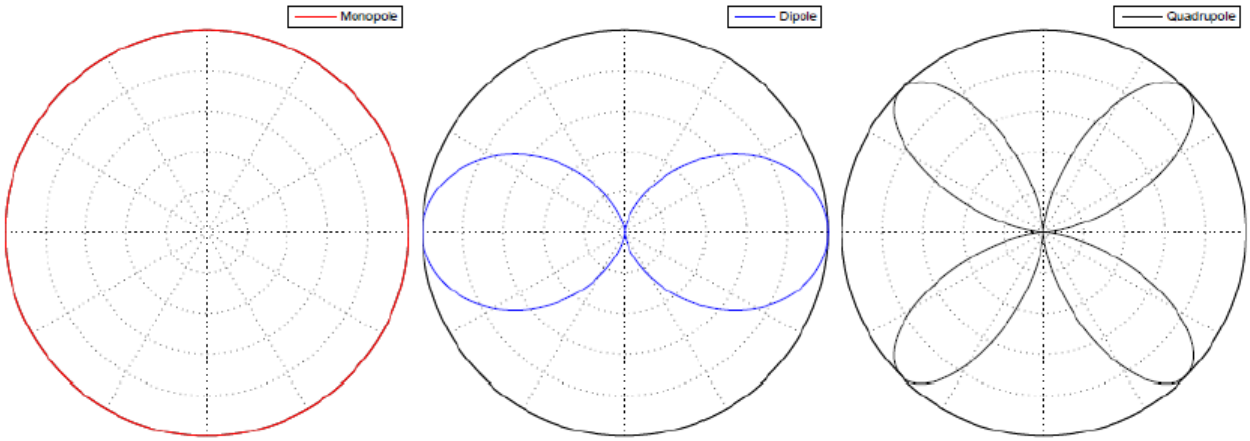


Figure 19.2: Monopole, Dipole and Quadrupole Radiation Pattern [10]

Propagation

In Equation (19.1), (19.2) and (19.3) the factor re is the distance from the source to the receiver. As stated in [1] the decibel levels needs to be evaluated at different distances. This factor allows the noise requirement **Hor-UAV-13** to be evaluated.

19.2.2. Noise Estimation of the UAV

First, the pitch generated by the UAVs will be shortly treated. Then the different noise levels for non zero angle of attack functions: the suction and pressure side of the propellers are calculated. After which the angle dependent noise is evaluated. A total noise calculation will be performed and the results will be discussed for noise close to the source as well as the noise level at the typical visitor distance.

Pitch

The frequency, or pitch, of the sound generated by the UAVs greatly affects the perceived noise level. Pitch will however not be discussed, as it is subjective and dependent on the radial propeller velocity. Because that velocity is set by the thrust requirement of the propulsion system, it will not be designed for a noise level.

Noise Prediction

The noise prediction shall be done using the BPM aero-acoustic model which uses three main functions to provide the Sound Power Level (SPL). The functions are as follows: the suction side equation, Equa-

tion (19.1), the pressure side, Equation (19.2), and finally the angle dependant noise equation, Equation (19.3). Suction and pressure side refer to the suction and pressure side of the propellers used by the UAVs. The conditions used for these computations are similar to those used in the aerodynamic model in Chapter 5.

Suction side:

$$SPL_s = 10 \cdot log(\frac{\delta_p M^5 s D_H}{re^2}) + A\frac{St_p}{St_1} + (K_1 - 3) + \Delta K1 \tag{19.1}$$

Pressure side:

$$SPL_p = 10 \cdot log(\frac{\delta_s M^5 s D_H}{re^2}) + A\frac{St_s}{St_1} + (K_1 - 3) \tag{19.2}$$

Angle dependent noise:

$$SPL_\alpha = 10 \cdot log(\frac{\delta_s M^5 s D_H}{re^2}) + B\frac{St_s}{St_2} + K_2 \tag{19.3}$$

St is the Strouhal number, which gives a correlation between flow rate and frequency. It can be calculated via Equation (19.4), the subscripts are p for the pressure side and s for the suction side. The boundary layer thickness $\delta_{p/s}$ is estimated from statistical data [4].

$$St = \frac{f\delta_{p/s}}{v_\infty} \tag{19.4}$$

St_1 for the zero degree angle of attack functions can be found by using the following equation:

$$St_1 = 0.02 \cdot M^{-0.6} \tag{19.5}$$

For the non zero angle of attack function(Equation (19.3)), St_2 for angles of attack higher than 12.5 ° the following holds[4]:

$$St_2 = St_1 \cdot 4.72 \tag{19.6}$$

First the spectral shape function A, and later B, can be found to find the first unknowns needed to calculate SPL_s, SPL_p, SPL_α:

$$A(a) = A_{min}(a) + A_R(a_0) \cdot [A_{max}(a) - A_{min}(a)] \tag{19.7}$$

Where,

$$a = |log(\frac{St_{p/s}}{0.5 \cdot (St_1 + St_2)})| \tag{19.8}$$

$$A_{min}(a) = -32.665a + 3.981 \tag{19.9}$$

$$A_{max}(a) = -4.669a^3 + 3.491a^2 - 16.699a + 1.149 \tag{19.10}$$

$$A_R(a_0) = \frac{-20 - A_{min}(a_0)}{A_{max}(a_0) - A_{min}(a_0)} \tag{19.11}$$

Where for Reynolds numbers lower than $8.57 * 10^5$ and higher than $9.52 * 10^4$, $a_0 = 0.6382$ which applies for HORUS as stated in Equation (19.12) [4]. Where a_0 is the absolute value of the logarithm of the ratio of Strouhal number. The absolute value is used because the spectral shape is modeled to be symmetric about $a = 0$.

$$Re = \frac{VL}{\mu} = 1.4015 \cdot 10^5 \tag{19.12}$$

Now for the shape function B:

$$B(b) = B_{min}(b) + B_R(b_0) \cdot [B_{max}(b) - B_{min}(b)] \tag{19.13}$$

$$b = |log(\frac{St_{p/s}}{0.5 \cdot (St_1 + St_2)})| \tag{19.14}$$

$$B_{min}(b) = -817.810b^3 + 355.210b^2 - 135.024b + 10.619 \tag{19.15}$$

$$B_{max}(b) = -80.541b^3 + 44.174b^2 - 39381b + 2.344 \tag{19.16}$$

$$B_R(b_0) = \frac{-20 - B_{min}(b_0)}{B_{max}(b_0) - B_{min}(b_0)} \tag{19.17}$$

Where for Reynolds numbers lower than $8.57 \cdot 10^5$ and higher than $9.52 \cdot 10^4$, $b_0 = 0.32$ which applies for HORUS as stated in Equation (19.12) [4]. In these functions the values a and b are the absolute value of the logarithm of the ratio of Strouhal numbers. The absolute value is used because the spectral shape is modelled to be symmetric about $b = 0$ [4].

Finally, the amplitude functions, K_1, K_2 and ΔK, required to find the SPLs. First K_1 and K_2 are calculated:

$$K_1(a) = 4.31 \cdot log(Re) + 156.3 \tag{19.18}$$

$$K_2 = K_1 + \sqrt{\beta^2 - (\frac{\beta}{\gamma}(\alpha - \gamma_0^2)} + \beta_0; \tag{19.19}$$

Where,

$$\gamma = 27.094M + 3.31 \tag{19.20}$$

$$\gamma_0 = 23.43M + 4.651 \tag{19.21}$$

$$\beta = 72.65M + 10.74 \tag{19.22}$$

$$\beta_0 = -34.19M + 13.82 \tag{19.23}$$

Now for ΔK, as the Re_{δ_p} is higher than 5000 the ΔK was found to be 0, the Reynolds number has been found using Equation (19.24) [10]:

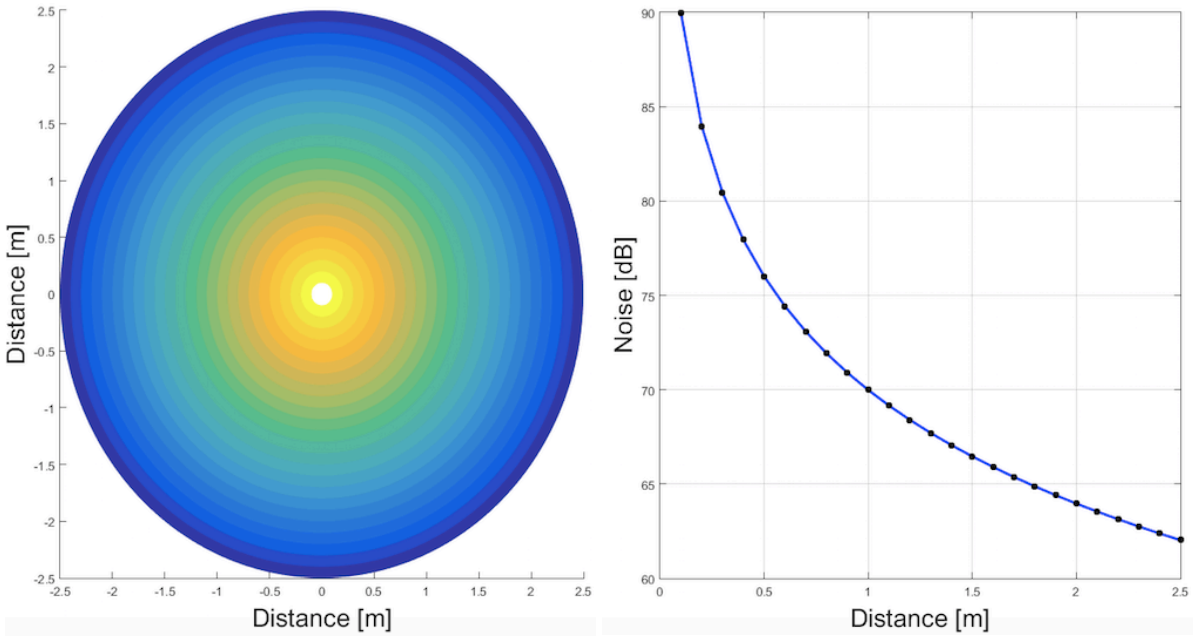
$$Re_{\delta_p} = \frac{v_\infty \delta_p}{\mu} \tag{19.24}$$

To find a total SPL which properly adds all the SPLs up Equation (19.25) has been used.

Total Noise:

$$SPL_{tot} = 10 \cdot log(10^{0.1SPL_p} + 10^{0.1SPL_s} + 10^{0.1SPL_\alpha} + 10^{0.1SPL_v}) \tag{19.25}$$

Implementing the 2.5 m specified in **Hor-UAV-13** the received sound power level is equal to 59.02 dB. At a distance of 0.1 m of the source the SPL will be 86.98 dB per propeller. Assuming the noise of second rotor will be the same as from the first, this would mean a total of 90 dB at 0.1 m from the source and 62.04 dB at 2.5 m from the source. This is well within the required decibel levels of 70 dB stated in **Hor-UAV-13**. In Figure 19.3 a visual representation has been made of the noise propagation of one UAV.



(a) Noise Intensity

(b) Noise Propagation

Figure 19.3: Noise Propagation for one UAV

As this is a noise prediction and thus somewhat inaccurate, the actual noise production will have to be measured during production. If these measurements show that the noise requirement will not be met, extra noise reducing material can be added to the UAVs.

20 | Operations & Logistic Concept

In order to provide an illustration of the proposed operational plan, an initial operations and logistics concept has been developed. First the system logistics from manufacturing to disposal is explained and illustrated in Section 20.1. Then the logistics associated with the day-to-day operations of the system are presented in Section 20.2.

20.1. System Logistics

The logistic flow diagram for the system is shown in Figure 20.1. The stages in this diagram are explained below.

Manufacturing
At the manufacturing stage the UAVs, ground station and server will be manufactured and assembled. The base materials and parts will all go through this stage. Most of them will be used in the system immediately, but some will be designated spare parts. The transport of these will be outsourced to commercial transporting systems.

Transporting The system as a whole will use specialised transporting, in order to assure the quality of the system when it arrives at the user. It will also allow for specialised packaging, which will make it more sustainable and prevent damage to the system. For customers close to the manufacturing station, a truck can be used. For far-away customers, ships or planes may be used.

Normal Operations At the exhibition space of the customer, the system will be installed and then normal operations will commence. Spare parts from manufacturing are used to replace irreparable components. A more detailed look at normal operations is shown in Figure 20.2.

Disposal At the end-of-life stage of the system, it will be disposed. This will be done by disassembling the system and disposing of each component in the most sustainable way. If any components are still usable, they will be designated spare parts and taken to the spare part storage.

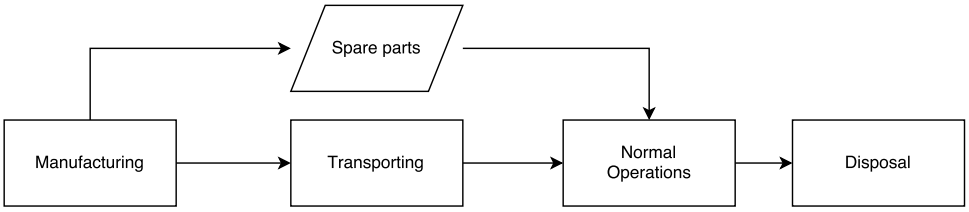


Figure 20.1: The Logistics Flow Diagram for the Manufacturing and Transportation of the System

20.2. Normal Operations

The system will be designed so that it is transportable and can be set up on location in less than a week. The basic maintenance of the system will be performed within a few hours, and routine checks will be performed once a week.

Because HORUS will perform each day in the same way, the day-to-day operations will be applicable for most of the general operations of HORUS, with the exception of manufacturing, transportation, set-up and disposal, as specified in Figure 20.1. The flow diagram for the day-to-day operations for horus/ is thus shown in Figure 20.2. The blocks in this diagram are explained below.

Turn On The turn on of the system, where the UAVs will go to the mode specified by the operator. This will in most cases be default mode, so at turn on the UAVs will fly to their orbits autonomously.

System Check A system check will be performed right after the system is turned on, this can be a visual check, a UAV self-check or a server check. If any of these checks fail, the UAV will go to Maintenance, otherwise the system will continue to normal operations.

Normal Operations The UAVs will fly according to the mode they are in, as specified by the operator. The different modes are explained in Section 3.2.

Charge station As stated in Section 19.1, when the battery of on UAV is running low, it will autonomously switch with the second UAV at the charge station. If it is charged, it will switch again. If it fails, it will go to maintenance.

Maintenance If the UAVs fail during the system check, normal operations or at the charge station, they will move to maintenance. There, they will be fixed by qualified personnel, either immediately or at a set time. In the meantime, a spare UAV can be used, or the UAV can be be kept out of operations.

Spare Part Storage If the UAV maintenance has parts that can not be fixed, they will be replaced. The replacement parts will have to be moved from the spare part storage.

Overnight Storage When the mode is set to turn off, all the UAVs will move from normal operations to their overnight storage.

Turn Off When all the UAVs are at the overnight storage, the whole system will turn off. They will stay at this state until they are turned on again.

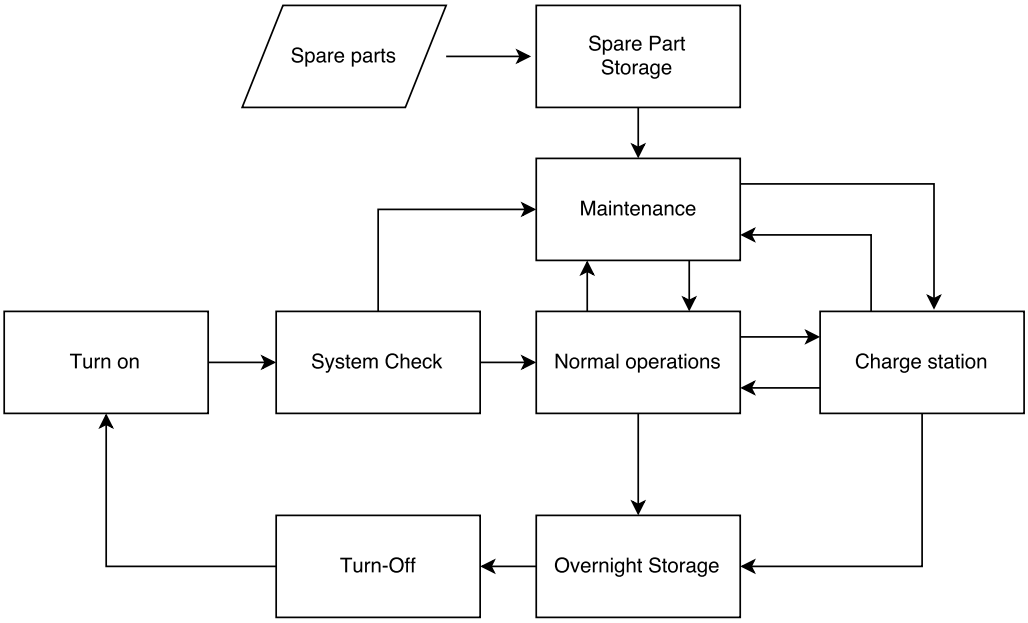


Figure 20.2: The Logistics Flow Diagram for the Day-to-Day Operation of HORUS, for the Movement of Individual UAVs

21 | RAMS Analysis

In this chapter the Reliability, Availability, Maintainability and Safety (RAMS) analysis will be presented to identify the factors that would lead to possible failure of the system. First, a fault tree analysis and hazard analysis will be performed in Section 21.1 and Section 21.2, respectively. Based on these analyses, the RAMS analysis can be performed and is described in Section 21.3.

21.1. Fault Tree Analysis

In this section a fault tree analysis will be performed. The goal of this analysis is to identify all possible failure modes (indicated by the circular boxes), which are tabulated and classified in Section 21.2. The decision has been made to focus on Loss of Propulsion, Loss of Control, Loss of Visualisation and Loss of Ground Segment depicted in Figure 21.1, 21.2, 21.3 and 21.4, respectively. Figure 21.5 and 21.6 are extensions of the fault tree analyses, indicated with a triangular box at the top of the trees.

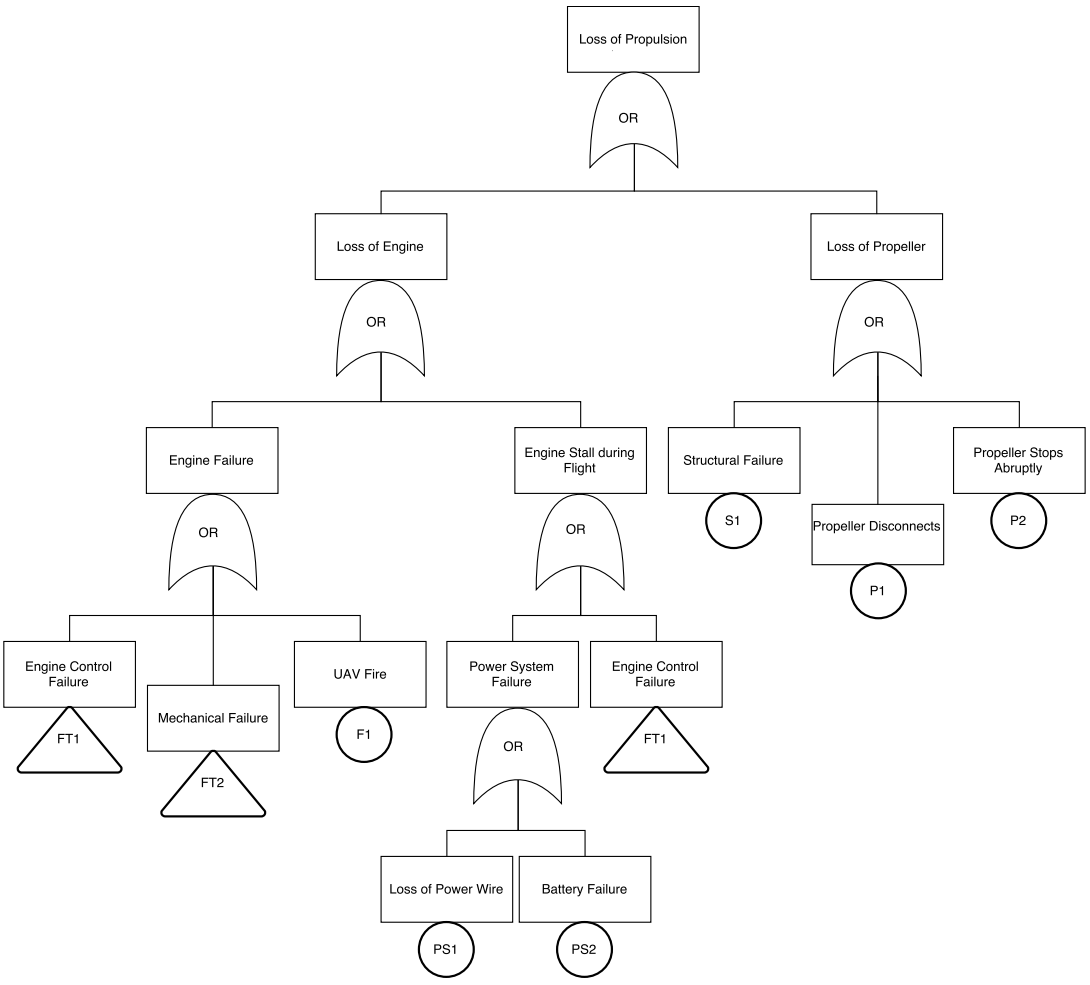


Figure 21.1: Fault Tree Analysis: Loss of Propulsion

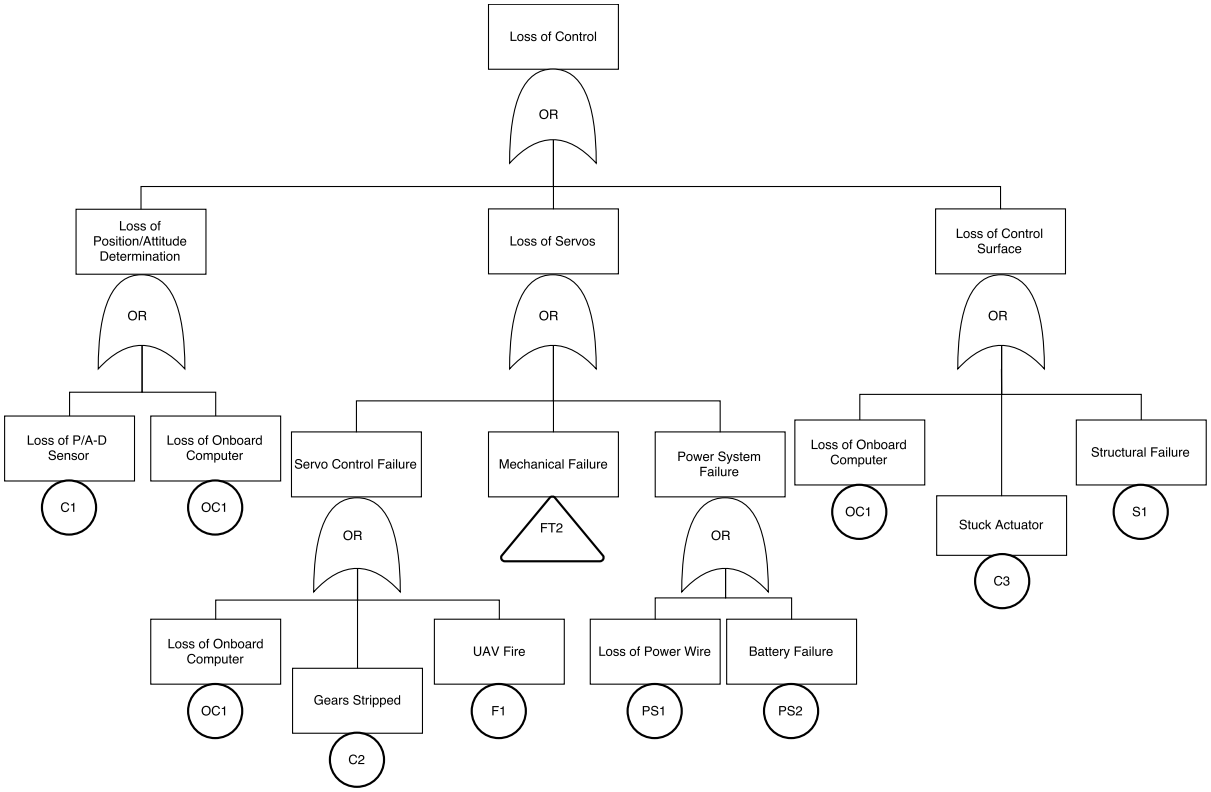


Figure 21.2: Fault Tree Analysis: Loss of Control

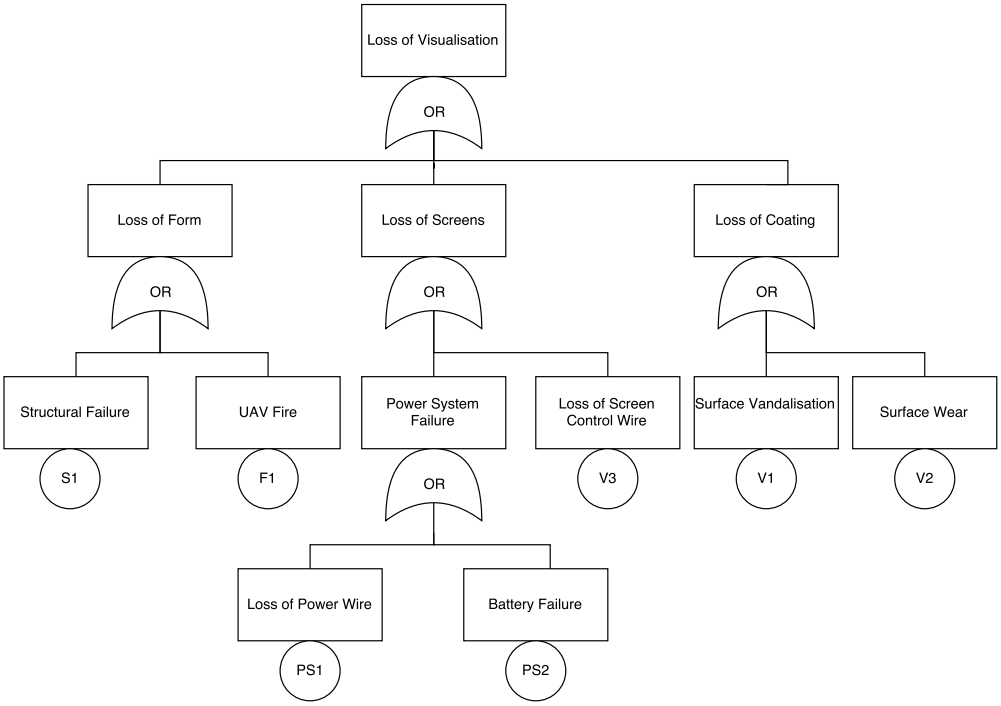


Figure 21.3: Fault Tree Analysis: Loss of Visualisation

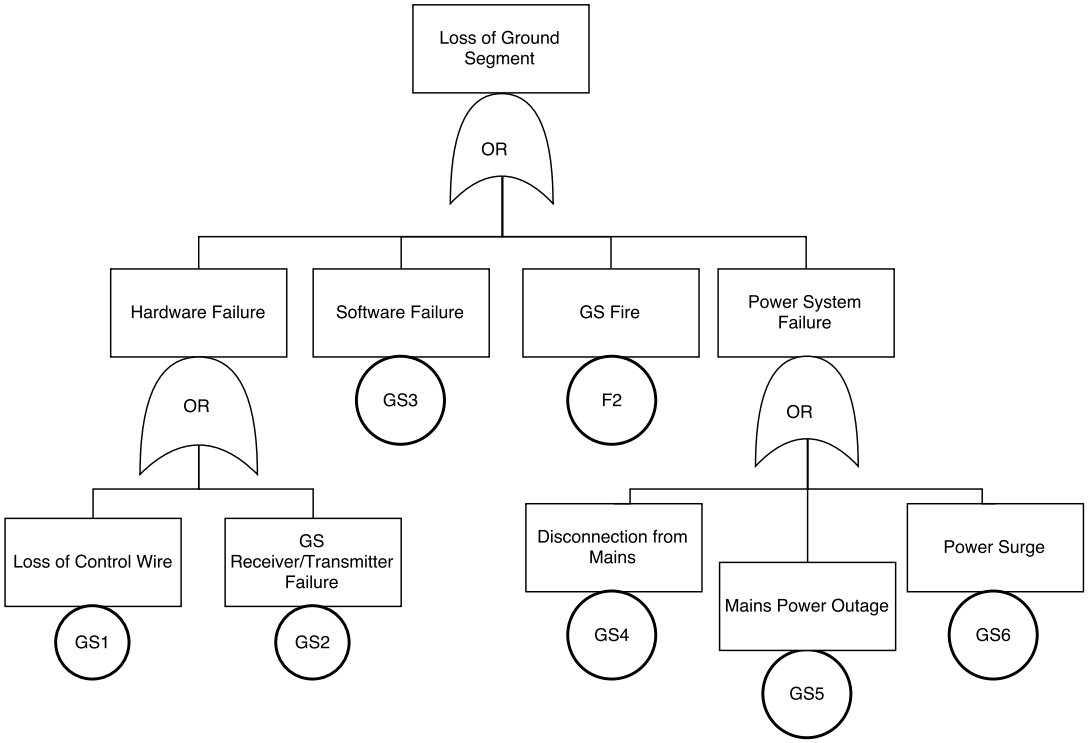


Figure 21.4: Fault Tree Analysis: Ground Station Failure

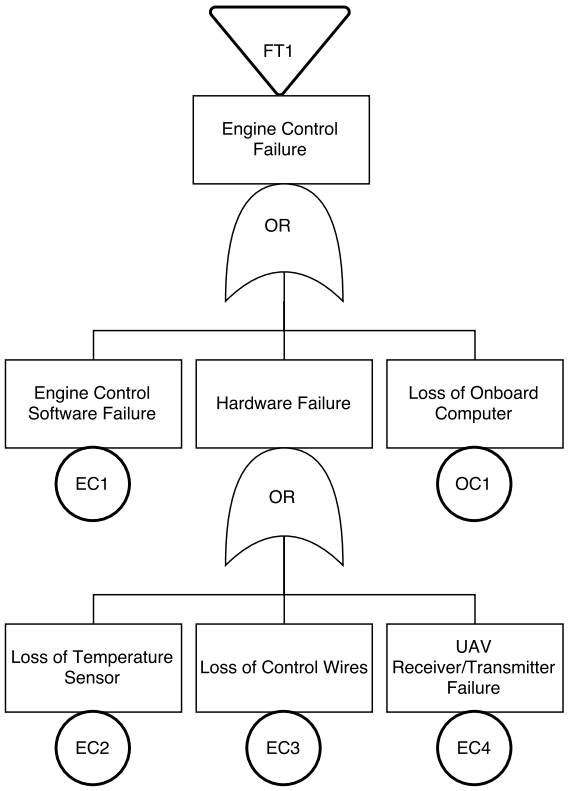


Figure 21.5: Fault Tree Analysis: Engine Control Failure

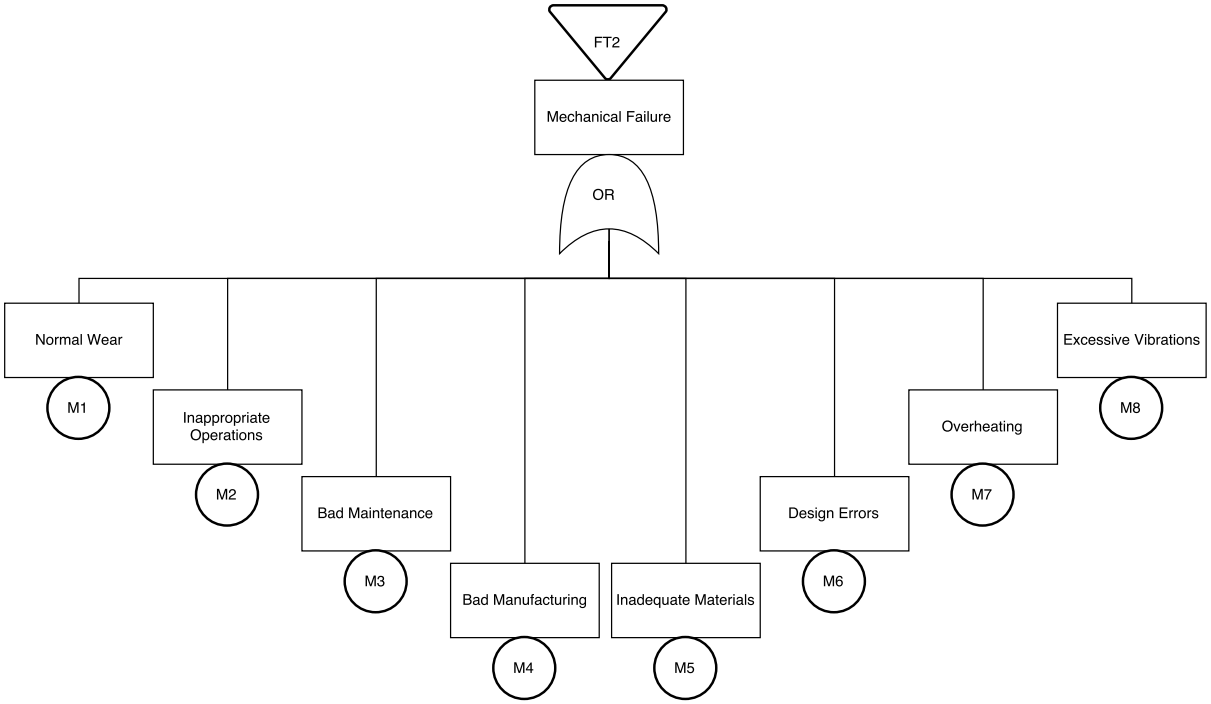


Figure 21.6: Fault Tree Analysis: Mechanical Failure

21.2. Hazard Analysis

In this section, the possible failure modes found in Section 21.1 are given a severity and occurrence classification. These classifications are depicted in Table 21.1 and Table 21.2, respectively. A criticality matrix is produced, shown in Table 21.3. In this matrix the failure modes and their classification from Table 21.4 are shown. The probability figures for the different failures were taken from [11], [14] and [27]. As can be seen in Figure 21.3, the loss of screens tree branch can be discarded since no screens used. It is still included in the hazard analysis for future reference, in case in e-paper is still used for visualisation in an improved UAV design. Risk PS1, PS2 and V3 can for this reason be discarded for this design.

Table 21.1: Severity Classifications

Category	Description	Definition
I	Catastrophic	A failure which may cause system or UAV loss
II	Critical	A failure which may cause critical damage to the system
III	Marginal	A failure which may cause marginal damage to the system
IV	Negligible	A failure which may cause negligible damage to the system, which does not have to be repaired immediately

Table 21.2: Occurence Classifications

Category	Occurrence	Probability
A	Frequent	$O > 0.2$
B	Probable	$0.1 > O > 0.2$
C	Occasional	$0.01 > O > 0.1$
D	Remote	$0.001 > O > 0.01$
E	Extremely Unlikely	$O < 0.001$

Table 21.3: Criticality Matrix

Probability of Occurrence → \ Severity of Occurrence →	IV	III	II	I
A				
B				
C		C3, M2		
D	GS4, V2	M1, P2	EC1, EC3, GS1, GS3, M3, M4, M5, M7, M8, PS2	C1, F2, M6
E	GS5, V3	V1	EC4, GS2, GS6	C2, EC2, F1, OC1, S1, P1, PS1

Table 21.4: Hazard Analysis

ID	Failure Mode	Occurence	Severity
C1	Loss of P/A-D Sensors	D	I
C2	Gears Stripped	E	I
C3	Stuck Actuator	C	III
EC1	Engine Control Software Failure	D	II
EC2	Loss of Temperature Sensor	E	I
EC3	Loss of Control Wires	D	II
EC4	UAV Receiver/Transmitter Failure	E	II
F1	UAV Fire	E	I
F2	GS Fire	D	I
GS1	Loss of Control Wire	D	II
GS2	GS Receiver/Transmitter Failure	E	II
GS3	GS Software Failure	D	II
GS4	Disconnection from Mains	D	IV
GS5	Mains Power Outage	E	IV
GS6	Power Surge	E	II
M1	Normal Wear	D	III
M2	Inappropriate Operations	C	III
M3	Bad Maintenance	D	II
M4	Bad Manufacturing	D	II
M5	Inadequate Materials	D	II
M6	Design Errors	D	I
M7	Overheating	D	II
M8	Excessive Vibrations	D	II
OC1	Loss of Onboard Computer	E	I
P1	Propeller Disconnects	E	I
P2	Propeller Stops Abruptly	D	III
PS1	Loss of Power Wire	E	I
PS2	Battery Failure	D	II
S1	Structural Failure	E	I
V1	Surface Vandalisation	E	III
V2	Surface Wear	D	IV
V3	Loss of Screen Control Wire	E	IV

21.3. RAMS Analysis

With the results gathered from the fault tree analysis and the criticality analysis, the RAMS analysis will be performed. The RAMS analysis consists of the elements Reliability, Maintainability, Availability and Safety.

21.3.1. Reliability

From the criticality matrix shown in Table 21.3 it can be concluded that the designed system will be reasonably reliable and has a low risk.

The two risks in the yellow band of the criticality matrix pose the highest threat to the system. The occurrence of these threats will have critical failure results for the system, likely resulting in either the replacement of certain parts of the UAV or ground station. However, their low chance of occurrence balances their potential threat to an acceptable risk. All in all, it can be concluded that the design system will have a reasonably high reliability.

21.3.2. Availability

Due to the low chance of failure, in combination with the short time required for maintenance and the design choice made to use multiple sets of UAVs, the availability of the system will be satisfactory. The multiple UAVs boost the availability of the system in two ways. When a UAV has to leave its orbit to recharge, its position can be taken by a second version, so that there only is a negligible amount of time

where a planet is missing. Secondly, when a UAV is unavailable due to maintenance, the other UAV can step in full time, most likely halving the time where a planet is missing from the system.

21.3.3. Maintainability

Due to the high reliability, or in other words, low chance of failure, maintenance will not be required very often. However, when maintenance is required, both planned and unplanned, the system is designed to be quickly accessible, in order to minimize the time spent on repairs and check-ups, as well as to limit the amount of specialised equipment required. This is done by designing the shell in such a way that it allows quick access to all components inside of the UAV. Since quite some use will be made of off-the-shelf components, replacement parts will be readily available if a component ceases to function. By the use of a modular and "plug-and-go" design, replacement parts can be easily removed and replaced. All components which will be bought off the shelf with no custom modifications to them, such as the batteries, are readily available from suppliers. Thus no reserve has to be kept as they can be ordered when needed. For the batteries a small reserve will be kept since these have to be custom ordered if they break which could lead to expensive delays. Additionally, a break in the internal structure cannot be easily repaired, and thus a new structure will have to be manufactured. However, the internal structure has been designed in such a way that any event leading to a break in the structure would be highly improbable, and would lead to damage to the entire UAV. Also the Sun is easily accessible, because the height is regulated using a motor and can be lowered to the ground. The light bulbs can be taken out of the Perspex Shell easily, by just pulling on the cables at the opening at the top of the shell. The light bulbs are commercially available, and can just be bought from most mainstream suppliers.

21.3.4. Safety

During the design of the lay-out of the UAVs, continuous attention has been paid to the safety of HORUS. Any hard or sharp edges have been avoided to make sure playground regulations are met. Furthermore, access to dangerous parts of the UAV, such as the propeller, has been limited as much as possible, in order to prevent possible injury. The mesh and shaft cover the propellers during operation, where the mesh sizing has accounted for the size of a children's finger. This makes sure children cannot get close to the propeller.

Regarding the chance of injury due to a UAV of HORUS during free-fall, the impact of a UAV on the head of a child is investigated. The worst case scenario of directly hitting a child on the head with the load carrying structure has been considered. This is due to the fact that any other part of the UAV is either covered with a softer material, such as the shell or mesh, or not directly on the surface of the UAV. The Head Injury Criterion (HIC) will be used to determine the consequence of impact. This can be calculated with Equation (21.1) [7].

$$HIC = \left\{(t_2 - t_1)\left[\frac{1}{t_2 - t_1}\int_{t_1}^{t_2} a(t)dt\right]^{2.5}\right\} \tag{21.1}$$

In this equation, $a(t)$ is the acceleration expressed in g's (standard gravitational acceleration) and $(t_2 - t_1)$ is the time interval in which the acceleration acts, with a maximum of 15 ms [18]. The longer the duration of the acceleration, the larger the consequence of the impact. Taking the time interval at its maximum would therefore yield a conservative value. The acceleration in this case is approximated as constant, and is obtained by dividing the impact force by the average mass of a human head. Since the acceleration is constant, the $1/(t_2 - t_1)$ disappears from the equation, and it can be seen that the larger the time interval, the higher the HIC value. The impact force is calculated in the same manner as in Section 7.1. For this calculation, the heaviest UAV is considered, since this will result in the highest impact force. The maximum flying height of a UAV is 5 m, and the average height of a 11-year old child is 1.5 m.[a] This means that the maximum height the UAV falls before it hits the child's head is 3.5 m. The impact force is then equal to 3515 N. Taking the average weight of a human head as 5 kg [b] and assuming $(t_2 - t_1)$ to be 15 ms, the HIC value equals 652. For a six-year-old child, the threshold for serious injury is 700 [18]. Since the calculated HIC-value is below the threshold, requirement **Hor-11** and **Hor-Crtl-5** is met. As **Hor-Crtl-5** can be considered as a scenario for which the UAV would fall during operation. Furthermore, the mesh on the bottom will also act as a crumple zone, which would decrease the impact force and thus the consequences of a crash.

[a] 0-18 jaar, URL `http://kindexpert.nl/element/55/1/20/3/groei/groei/0-18-jaar`, accessed on 15 June 2016

[b] Mass of a Human Head, URL `http://hypertextbook.com/facts/2006/DmitriyGekhman.shtml`, accessed on 15 June 2016

22 | Sensitivity Analysis

When a design parameter is changed, it may influence other parameters as well. In this chapter this sensitivity of the design will be investigated. The sensitivity analysis is also used to determine the robustness and feasibility of the design. A division is made between system parameters in Section 22.1 and UAV parameters in Section 22.2. All requirements referenced in this chapter can be found in Appendix B.

22.1. System Parameters

System parameters influence the entire system and are general aspects relating to HORUS. The analysed system parameters are operating time/availability, educational factor, flight envelope, location and operational limit.

Operating Time/Availability

The operating time of the system is set at eight hours a day, seven days a week, see **Hor-9**. If the daily operating hours are decreased, the energy requirement for the system can be relaxed, because the number of charging cycles that the UAVs are subjected to each day is also decreased. The same reasoning applies for operating less days each week. This will also increase the life-time of the system, assuming that the number of charge cycles until end-of-life stays unchanged. Increasing the operation time will increase the number of charging cycles, reducing the operational life-time of the system.

Educational Factor

Removing **Hor-4**, the educational requirement, will result in a lot more freedom in the design. The relative size of the planets and their orbital path could then be set arbitrarily. The flight formation modes of the system can be omitted or changed as well. For example, formation modes can be included where the planets fly in a figure 8 configuration. Without the educational requirement, it is also no longer necessary to represent the planets accurately. This will allow for more options for visualisation.

Flight Envelope

The flight envelope of the system, **Hor-6**, has a large influence on the orbital model of the system. Increasing the flight envelope means more space to realistically represent the planet orbits, including eccentricity and longitudinal nodes. The size of the UAVs can also be increased and more planets can be added.

If the minimal size of the UAVs is increased, **Hor-UAV-9**, the smallest UAV will have more surface area for visualisation. But this increase means a decrease in the space between the planets if they are in a straight line which increases the requirements on the collision avoidance system. The difference in sizes between the planets would also be less visible. Increasing the maximum size of the UAVs, **Hor-UAV-17**, will allow for larger differences in planet sizes, which makes the model more accurate. There will also be a larger surface for visualisation. However, the mass of the UAV will increase, which might violate **Hor-UAV-22**. Decreasing the maximum size of the UAV will have the opposite effects on the system.

Location

In **Hor-12**, it is stated that the system will function indoors. If the system needs to function outside as well, weather conditions such as wind gusts or rain need to be taken into account when designing HORUS. Though, it would allow for changes regarding power generation. When the system is functioning outdoors, solar panels might become a feasible option to provide power to the server or the UAVs.

Operational Limit

The requirement is set that the system will be functional in 2017, see **Hor-10**. If the functional date is postponed, more recent technical advancements can be implemented into the design. The design and manufacturing time frame can also be increased, which allows a more detailed look at specific subsystems. More time consuming manufacturing methods can also be employed. If the functional date is brought forward, time might become strained for the design or manufacturing, increasing the possibility that bad design choices will be made and that defects will be overlooked.

Cost
The cost constrain of the prototype system is set in **Hor-8**. An increase in funding will not change the current design. Because the budget available was already more than sufficient, design choices were not limited due to this factor. However, a decrease in funding will reduce the number of possible material options and manufacturing methods. High-quality COTS parts might not be affordable, which will lead to less reliable or heavier components in the system. On the other hand, a decrease in cost would also decrease the selling price which will make the design more marketable, as the initial investment for potential customers is then also lower.

22.2. UAV Parameters

UAV-specific parameters are analysed in this section. The parameters investigated are speed, mass, interaction, level of autonomy and noise.

Speed
In **Hor-UAV-5**, it is stated that the UAV shall not reach a speed higher than 4 km/h. If the maximum speed is increased, a smaller orbital period is possible. This might benefit the visual representation of the system, because the speed of the outer planets can then be increased as well, while keeping the scale of the system. The opposite can be said for a decrease in maximum speed, which will hurt the visual representation of the system. However, the lower speed might mean that the flight time will increase as the amount of power used for propulsion is decreased.

Mass
In **Hor-UAV-22** the mass requirement of 2 kg is stated. If the maximum allowable mass of the UAVs is increased, heavier elements can be chosen. When the power system is allowed to weigh more, different or more batteries can be used. In this situation the amount of power and lift needed to perform flight will also increase, which will mean a longer maximum flight time. If the maximum size of the UAV becomes a restricting factor before the mass does, an increase of the maximum allowable mass will have no influence on the UAV design.

If the maximum mass is decreased, this will likely become the restricting factor on the design. Because the main contribution to the mass are the batteries, a decrease in mass will lead to less batteries, which will lead to less flight time. This will also decrease the needed propulsion and thrust, as well as the structural strength of the system.

Interaction
In requirement **Hor-5** the interactive capability of the system is stated, which affects the visualisation of the UAVs and the required flight modes. If the interactive capabilities are decreased some of the flight modes can be removed, such as the identification mode stated in Chapter 3.

Level of Autonomy
In **Hor-Ctrl-1**, **Hor-Ctrl-2** and **Hor-UAV-12** the autonomy requirements are stated. If the level of autonomy is decreased, the amount of onboard calculations will be significantly lower. This means that the onboard computer might be changed to a micro-controller. The more the autonomy is decreased, the more the system will have to be designed to be operated by one server, which increases the requirements on the server.

Increasing the degree of autonomy means that the onboard computer capabilities need to be increased. This might mean a micro-computer is no longer a feasible option, and that an FPGA or a custom system will have to be used.

Noise
In **Hor-UAV-13**, it is stated that the UAVs shall not produce more than 70 dB of noise. If this is increased, bigger engines and/or bigger propellers with a higher RPM are possible. If the maximum size is then exceeded, the increase of allowable noise has no influence on the UAV design. If the maximum allowable noise was to be decreased, this would greatly affect the design of the UAVs. As stated earlier, this will likely mean a decrease in size of the propellers which would decrease the generated lift and thus the maximum mass and flight time of the UAVs.

23 | Sustainable Development Strategy

The Sustainable Development Strategy will be presented in this chapter. Having a sustainable design is important for the minimisation of the effect on people, the environment and to ensure that resources will remain available for future generations. This will be discussed in Section 23.1. The impact of HORUS on social and economical aspects are discussed in Section 23.2 and Section 23.3, respectively.

23.1. Environmental

Both the sustainability of during the design phase and during operations are considered. Furthermore, the sustainability of the design when it has reached its end-of-life will also be discussed, as this is an equally important part.

23.1.1. During Design Phase

During the design phase, the material selection and resource consumption are analysed for sustainability. The materials of the design will be evaluated on the sustainability of production and the environmental cost of transportation. For materials processing and the manufacturing of the design, commercial off-the-shelf products or other sustainable methods are preferred.

In order to minimize the use of resources, the structure should achieve the minimum required strength with as little material as possible. By using a hemispherical frame to carry the loads the material used for support is already minimised. By using materials with high specific weights, the weight of the UAV is as low as possible. The low weight decreased the amount of thrust required to be generated by the propeller, resulting in a lower power usage, and thus a lower battery mass required. An additional benefit is the decrease in energy consumption due to this. For the visualisation of the planets, the UAVs should have a durable, long-lasting and environmentally friendly coating.

23.1.2. During Operations

While HORUS is in use, the main environmental cost will be the energy consumption. Effort was therefore made to develop an energy efficient design. This was done by minimising the power consumed and selecting an engine and battery for the UAV with a high efficiency. Although the propulsion system of HORUS is as efficient as possible, the chosen co-axial propeller configuration and the battery discharging is not 100% efficient. The second propeller will encounter a disturbed flow, thus decreasing its efficiency.

In order to minimise the environmental cost of packaging and transport, the system will be produced locally, eliminating most transportation cost during production. Furthermore, a smart method of packaging was thought up, minimising the space required to transport the UAVs. This will allow minimal packaging and thus less space will be occupied by HORUS during transport, which will minimise transport costs.

The maintenance costs are also taken into account by making the design of the UAV accessible and repairable. The accessibility is ensured by taking a shell structure that easily opens and an open internal structure. This means that for part failure, it is not necessary to replace the entire UAV. Instead, the shell can be opened, and the part can be replaced or repaired within the UAV. When multiple components are stored in advance, the time the UAV is out of order is minimised.

23.1.3. End-of-Life

When the system reaches the end of its life, it will need to be recycled, reused or be disposed off. The design therefore allows itself to be disassembled into individual components that can either be resold, recycled or disposed independently. The money gained from reselling certain parts of the system will add to the revenue gained.

23.2. Societal

The aim of HORUS is to inspire the new generation, to create interest in the space engineering field and to improve awareness of sustainability and global warming. The design can contribute to sustainability through education. Apart from information regarding the planets, additional information can be provided concerning subjects such as global warming, management of resources or environmental impact. Interest in the sustainable engineering field can also be increased by HORUS, as well as inspiring visitors to take action to increase sustainability in their everyday lives.

23.3. Economical

From an economical point of view HORUS can benefit society in two ways. Similar to most products, HORUS will create jobs. During production, employees will be required to manufacture, assemble and test the system. After each exhibition, HORUS will have to be transported and every new location the HORUS will have to be set up and later uninstalled. While HORUS is in use, a periodic system check will be advised and a specialist is needed for possible repairs. Besides creating extra jobs, HORUS will attract more visitors to the exhibitions it is installed in and would thus generate extra revenue for the host location.

VII

Commercial Aspect of HORUS

24 | Market Analysis

An initial market analysis and a calculation regarding the return on investment will be shown in this chapter. The market analysis will contain the goal of HORUS, the target market, a competitive analysis and finally a Strength, Weaknesses, Opportunities and Threats (SWOT) analysis. Then the challenges that the HORUS project might face will be discussed, after which the return on investment calculations will follow.

24.1. Goal

Since the design phase has been completed, the next goal of the project is to build a working prototype of HORUS. When that is finished, the new goal is to be fully operational by the end of 2017. The main objective will then be to pay off the incurred debts from the production of the first HORUS. These cost can also be seen in Chapter 27. In order to end 2017 in positive numbers, one system will need to be sold and one exhibition booked.

24.2. Target Market

Because of the educational quality of HORUS, institutions such as science centres, high schools and universities can be approached. But the private sector can be considered a potential market as well. Renting out HORUS can be done to fairs, parties or temporary exhibitions, among others. These will not only generate revenue, but also market HORUS to potential customers.

Because UAVs are a popular modern commodity, and the HORUS design is unique, there are more opportunities to explore. These include marketing the individual UAVs to children, where they can be easily customised and tailored to the needs of the customer. Their child-friendly and robust design will mean that they can be used for a plethora of reasons.

24.3. Competitive Analysis

The HORUS design is unique, but there are a couple of comparable drones currently in production. The most notable are:

- Fleye, autonomous personal robot[a]
- Elios, captures images of confined spaces[b]
- Spherical Flight Vehicle produced by the Korea Aerospace University[c]

The latter is commissioned by the Korean military and will thus not be used for commercial purposes. The Fleye and Elios share the commercial market that the individual UAVs would be marketed in. But a similar orrery system has never been attempted, which makes HORUS unique. Museums that already have a large-scale inbuilt orrery might be less likely to purchase a HORUS, as they have already invested in a different system.

24.4. SWOT Analysis

In this section a SWOT Analysis will be performed on HORUS to identify the strength, weaknesses, opportunities and threats of HORUS. This will involve a external and internal analysis. The SWOT analysis is graphically shown in Figure 24.1.

[a] Fleye, URL `http://gofleye.com/`, accessed on 6 June 2016
[b] Flyability, URL `http://www.flyability.com/`, accessed on 6 June 2016
[c] Spherical Flying Drone, URL `https://www.youtube.com/watch?v=55d5ppwQBQ4`, accessed on 6 June 2016

24.4.1. Internal Analysis

The internal analysis consists of the strengths and weakness analysis of HORUS. These analyses are focussed on the system itself. The software, hardware and concept are considered to be the main internal aspects of HORUS.

The idea to combine an orrery with UAVs is unique, which makes it interesting for clients. Because almost all system parts are independently movable and located on the ground, HORUS will be easily transportable and installable. The aim is to sell HORUS as a complete system, but the three components of control, orbit and UAV design might be sold separately. It has happened before that a separate module or physical element turned out to be interesting for other applications.

24.4.2. External Analysis

In the Netherlands there are currently five science centres, over 35 universities and a number of other science oriented facilities, such as Space Expo. Only the Dutch market is considered at this moment, but the international market can also be explored in a later stage. The network of the TU Delft can be used to reach potential customers. For example, Space Expo already showed interest in HORUS.

The system is highly customisable, as planets can easily be added or removed due to the autonomous properties of HORUS. Single planets can also be ordered and custom made, which makes it especially interesting to the private sector. Another option is that only one single planet can be ordered and custom made, which is especially interesting for the private sector.

However, existing threats need to be evaluated. Because visitors can analyse the system carefully, it is a possibility that HORUS will be copied. Furthermore, due to the main focus on institutions it will be hard to find a good marketing strategy, as well as the limited number of institutions that are present in the Netherlands. The clients might find HORUS less interesting due to other existing orrery systems, although no other UAV orrery exists.[d] HORUS uses a large flight envelope, for which can form a problem for smaller institutions might not have the space necessary to facilitate HORUS.

	Helpful	Harmful
Internal	Strengths: - Unique Idea - Unique Service - Easily Installed - Multi-applicable - Safety	Weaknesses: - Software Based - Expensive - Niche Product - Large Flight Envelope
External	Opportunities: - Scalability - Network of TU Delft	Threats: - Copyable - Marketing - Existing Orreries - Limited Number of Institutions

Figure 24.1: SWOT Analysis

24.5. Challenges

As the project is currently only at its design phase, the design team will likely face challenges during the upcoming phases. Any delays during the upcoming phases will demand more resources to be spend, but the estimated resources are still below budget, so this will not likely become a problem. However, if the resources spent increase, the profit per sold HORUS decrease. Logically, this is not a favourable outcome

[d] Planetarium Eise Eisinga, URL `http://www.planetarium-friesland.nl/en`, accessed on 6 June 2016

and must be avoided as much as possible. The marketing aspect might also provide problems. Because this is a unique and expensive product, there is a big step between being interested in and actually buying a HORUS.

24.6. Return on Investment

As can be seen in Chapter 25 and Section 27.2, the design and production of HORUS results in a total cost of € 150.000, if the initial investment is more than this, it can be invested in marketing, improvement of the design or in making the design more sustainable. The same goes for profits made. The production cost can be earned back through the sale of one HORUS for € 200.000 and the rent of € 10.000 per week. For an institute such as NEMO, which gets 500.000 visitors a year, € 1 per visitor is needed to finance the exposition [21]. It is expected that one HORUS will be sold and one rented for ten weeks in the first year. The above has been summarised in Table 24.1.

Table 24.1: Return on Investment

Component	Amount [€]	Expected Orders per Year	Return per year [€]
Standard Selling Price	200,000	1	200,000
Standard Renting Price	10,000/week	10 weeks	100,000
Total Return per year			300,000

This price includes the complete HORUS system; two sets of eight planets, the Sun, the interface terminal, supervisor tablets and the Charge System. Software, installation and maintenance will be included as well, and a training day for the personnel of the institution. The training will include how HORUS works and how to operate it. Furthermore, there will be a customer service to assist in case of problems. This will also be available if HORUS is rented. In this case maintenance activities are included as well as a supervisor. The expected production costs will be earned back completely with each sale of HORUS, while an initial investment of 150.000 €will thus be earned back in one year. This leads to a Return of Investment of 100%, as calculated using Equation (24.1).

$$ROI = \frac{Gain\ on\ Investment - Cost\ of\ Investment}{Cost\ of\ Investment} = \frac{300.000 - 150.000}{150.000} = 100\% \qquad (24.1)$$

25 | Cost Estimation

In order to verify that the price of the proposed system still falls within the budget allowed for the project, a cost estimation of the system is produced. This cost estimation will take into account the price of the materials and components used to create the system, but not the overhead costs which can be expected from processing, assembly, maintenance, and disposal. These last items will be discussed later in Chapter 27, together with any other cost which can be expected following the design phase. This means that the price listed in Table 25.1 for the materials is only the cost of acquiring the raw material volume required to make the product. However since it is estimated that even during careful processing material will be lost, a factor of 1.2 is used on the calculated required volume to ensure that no unexpected material cost will occur. In order to estimate the cost of the system, specific components were found from retailers. The retailers of the components which were found are listen in the footnotes of this chapter. As a reminder, the budget allocated to the project was € 1,000,000, however, a contingency factor of 2.5% was put in place to ensure the feasibility of the project when it comes to budget. The effective budget is therefore € 975,000.

From the table it can be concluded that the material and component cost of HORUS are well within the imposed budget constraint, complying with requirement **Hor-8**. With the portion of the budget which goes unused during the acquisition of the components and materials the manpower, facilities, machinery, and other overhead costs during the later stages in the life of HORUS can be covered. Additionally, the remaining budget could be put towards investigating more efficient power storage components, motors or propellers, and even materials with better specific properties. More recommendations on what can be done with the remaining budget are given in Section 27.2.

---

[a] PRI 3 (2016) Model B, URL https://www.newit.co.uk/shop/all-raspberry-pi/raspberry_pi_3/raspberry_pi3, accessed on 13 June 2016

[b] Cytron IR-01 IR Sensor, URL http://www.cytron.com.my/p-sn-irs-01, accessed on 13 June 2016

[c] SparkFun Triple Axis Accelerometer Breakout - ADXL345, URL https://www.sparkfun.com/products/9836, accessed on 13 June 2016

[d] Thunder Power RC TP4500-6SHV, URL http://www.thunderpowerrc.com/Products/4500-mAh/TP4500-6SHV, accessed on 13 June 2016

[e] Thunder Power RC TP450-3SM70J, URL http://www.thunderpowerrc.com/Products/450magna/TP450-3SM70J, accessed on 17 June 2016

[f] Suppo 2212/6 2200kv Brushless Motor, URL http://www.altitudehobbies.com/brushless-motor-400-28-30-2200kv-suppo-2212-6, accessed on 13 June 2016

[g] KEDA TR2830/14-750kv Brushless Motor, URL http://rctimer.com/product-1186.html, accessed on 13 June 2016

[h] JustGoFly Order Info, URL http://www.justgofly.com/Order%20Form.htm#450_Motors, accessed on 13 June 2016

[i] Exceed RC Rocket 2215-3400, URL http://www.hobbypartz.com/86ma09-2215-920kv.html, accessed on 13 June 2016

[j] Zagi Carbon Prop, URL http://www.zagi.com/motors-props/zagi-carbon-prop-w-adapter-for-most-brushless-motors, accessed on 13 June 2016

[k] APC 7x6 Thin Electric Propeller, URL http://www3.towerhobbies.com/cgi-bin/wti0001p?&I=LXBHPC&P=M, accessed on 13 June 2016

[l] APC 8x4 Thin Electric Propeller, URL http://www3.towerhobbies.com/cgi-bin/wti0001p?&I=LXZK92&P=M, accessed on 13 June 2016

[m] APC 10x5 Thin Electric Propeller, URL http://www3.towerhobbies.com/cgi-bin/wti0001p?&I=LXZK95&P=M, accessed on 13 June 2016

[n] Feit BR40 Dimmable LED, 250W Equivalent, 3000K, URL https://www.amazon.com/Feit-BR40-DM-2500-LED/dp/B00JXPPPOY, accessed on 13 June 2016

[o] Pulley system, URL http://www.tractorsupply.com/tsc/product/marathon-electric-farm-duty-shop-motor-3-4-hp, accessed on 16 June 2016

[p] Poured Clear Perspex, URL http://www.stoutperspex.nl/kunststof-producten/perspex-helder-000/7, accessed on 17 June 2016

[q] HP EliteBook 8570w, URL https://tweakers.net/pricewatch/322551/hp-elitebook-8570w-(ly556et).html, accessed on 13 June 2016

[r] PN-L8028/PN, URL https://www.megekko.nl/product/0/759205/Sharp-PN-L802B-touch-screen-monitor, accessed on 13 June 2016

[s] BlueSense Beacon v3, URL http://bluesensenetworks.com/product/bluebar-beacon/, accessed on 16 June 2016

Table 25.1: Components and Materials Cost of Two Sets fo Eight UAVs, the Sun and the Ground Station

Component	Specific Product	Quantity	Part Cost [€]	Total Cost [€]
	UAV			
Microcomputer	Raspberry Pi 3 Model B [a]	16	45	720
IR Sensors	Cytron IRS-01 [b]	176	20	4160
IMU	Sparkfun - ADXL345 [c]	16	18	288
Batteries	TP4500-6SHV [d]	32	130	4120
	TP450-3SM70J [e]	4	25	100
Engines	Suppo 2210/6 2200kv [f]	8	15	120
	KEDA TR2830/14 [g]	8	11	88
	Justgofly 450T2 [h]	8	50	400
	Exceed - Rocket 2215-3400 [i]	8	15	120
Propellers	Zagi Carbon 5.1x4.9 [j]	8	10	80
	APC E 7x6 [k]	8	2.5	20
	APC E 8x4 [l]	8	3	24
	APC E 10x5 [m]	8	4	32
UAV Shell Foam	Low Density Polyurethane Foam	0.01 m^2	750 per m^2	7.56
UAV Shell Plastic	Polypropylene (PP-H)	0.01 m^3	1,500 per m^3	15.48
UAV Structure Material	HM Carbon Fibre Epoxy Matrix	0.0005244 m^2	400,000 per m^2	209.76
UAV Honeycomb Mesh	Over-expanded Aramid Honeycomb Sheet	16	150	2400
	Sun			
Sun Light Source	Feit Light Bulbs [n]	4	40	160
Sun Custom Sticker	Custom designed sticker	1	60	60
Pulley System	Marathon Electric 3/4HP Motor [o]	1	200	200
Sun Shell Material	Clear Perspex [p]	3.141592654 m^2	61.70 per m^2	193.84
Sun Structural Material	HM Carbon Fibre Epoxy Matrix	54.72 cm^3	400,000 per m^3	21.9
	Ground Station			
Ground station server	HP EliteBook 8570w [q]	1	1,500	1,500
User Interface	Sharp PN-L802B [r]	1	13,500	13,500
	External			
Bluetooth Beacons	BlueSense Beacon v3 [s]	5	25	125
Total				28665.52

VIII

The Future of HORUS

26 | Risk Assessment and Mitigation

In this chapter, the risks that may occur during the operation of HORUS will be assessed. These risks have to do with the design of the HORUS system, but occur after the final design choices have been made. These risks can form a threat for the schedule, cost or performance of the design. First of all, the possible situations that form a risk are identified. Then, the probability of occurrence and severity of impact of the risks will be elaborated, after which they will put in a risk map. Thereafter, a risk mitigation plan is made for the highest risks.

26.1. Risk Assessment

Table 26.1 shows the assessed risks and their assigned numbers, which are divided into five groups: UAV, Ground Station, Communication and Control, Orbital Mechanics and Sun. These risks are mapped in Table 26.2. The vertical axis indicates the probability of occurrence, ranging from highly improbable to highly probable. The vertical axis shows the consequence on the performance. This is divided into four levels of severity. Negligible means that the consequence has no impact on the operation or is only an inconvenience. Marginal indicates a small reduction in performance of the system, while critical denotes a large reduction in performance, which might lead to failure. A catastrophic consequence means failure of the design or great reduction in performance, such that primary functions cannot be achieved.

The propellers not providing enough thrust (Risk 1.1) would have a catastrophic consequence for the performance of HORUS, since the UAVs would not be able to fly. However, a high contingency allowance has been taken into account while designing the propellers, so that the chance of this event to occur is very low (highly improbable).

In case that the variable pitch control is not able to perform the required manoeuvres (Risk 1.2), the performance is reduced significantly, but the system can still operate. The orbit representation could then be less accurate because the path planning cannot be executed properly. It could also mean that the UAVs do not avoid each other entirely, but slide against each other or experience a soft collision. This adds wear to the surface coating, but this increase is minor. It is highly improbable that this will happen.

The visualisation technique (coating) has been proven to provide accurate representations, so the probability of Risk 1.3 is very low (highly improbable). Also, since most of the planets do no need a very accurate representation (for example, Uranus is mainly blue with shades), the consequence is marginal.

The structure has been designed in such a way that it can handle slightly higher loads then anticipated. Furthermore, the loads are calculated for a 5 m fall, a height which a UAV is not usually at. Thus, it is improbable that it is not able to cope with the loads (Risk 1.4). If this does happen to be the case, the consequence is marginal, because in the event of a fall followed by failure, the structure can be replaced.

Since the current estimation of the costs for HORUS are considerably lower than the assigned budget, it is highly improbable the manufacturing becomes too expensive (Risk 1.5). The consequence is critical, because if the budget is not enough, extra investors would have to be found. Furthermore, HORUS would be a more expensive acquisition, making it less marketable.

Due to high tolerances during manufacturing, the components might not fit in the UAV (Risk 1.6). The probability of this event is however very low, since the tolerances nowadays are extremely small and there is a little margin taken into account during packaging of the UAV. The consequence is critical since it would not be possible to assemble then UAV in such event.

It is improbable that the motor cannot handle the pitch control (Risk 1.7), but the consequence is critical on the performance. This is for the same reasons as Risk 1.2.

If the software for the user and supervisor interface does not work as required (Risk 2.1), the consequence is marginal. This is because the interface would be a little more inconvenient to work with, but will still does its job. The probability of this happening is relatively high, since software usually is a relatively less predictable.

If the landing accuracy required by the charging system cannot be provided by the UAV positioning system (Risk 2.2), the consequence is catastrophic, since the UAVs would not be able to charge and thus not be able to operate. This is an improbable event to happen, because the design allows for relative high inaccuracy for making contact for charging.

When the communication system cannot handle the complexity of the system (Risk 3.1), it means that delays occur for sending the modes to the UAV or the server crashes. Another option is that the server cannot connect to eight UAVs, and one or more UAVs are not able to change modes. This is clearly an undesired situation, but the UAVs can continue flying.

The risk that the desired accuracy of the proximity sensors is not reached is low (Risk 3.2). The same counts for the desired accuracy of 3 cm for the Bluetooth system (Risk 3.3)[29]. However, they are assigned critical performance consequences, because when the accuracy is too low, the UAV can slide along each other or even collide.

When the range of the communication system does not include the entire flight envelope (Risk 3.4), this means that when a UAV flies to far away from a certain point (beacon, sever or other UAVs), it cannot be controlled any more. This would be a catastrophic situation. However, the bluetooth pulses are expected to have a range of over 100 m, so this is highly improbable. Risk 3.5 is located at 'marginal', because when a problem occurs related to the software, this can be fixed by reprogramming the system. When the software is not compatible with the hardware (Risk 3.6), the UAVs cannot be controlled by the server or control themselves, and the system is unable to function.

When the spacing between to UAVs is too low (Risk 4.1), the UAVs will collide. Expected is that if this happens, it is a soft collision and the UAVs are able to calculate a new path and keep flying.

Risk 5.1 and 5.2 are both highly improbable and have a marginal performance consequence. The lights bulbs used are extremely strong and will give very bright light. When no shadows are visable, this will reduce the educational value, but the HORUS sytem will still work. The sun is suspended using three points on the ceiling, and connected to the shell with rod ends which only have one degree of freedom. Still will be a stable system. Also, there is 7 cm space between the Sun and the closest UAV.

26.2. Risk Mitigation

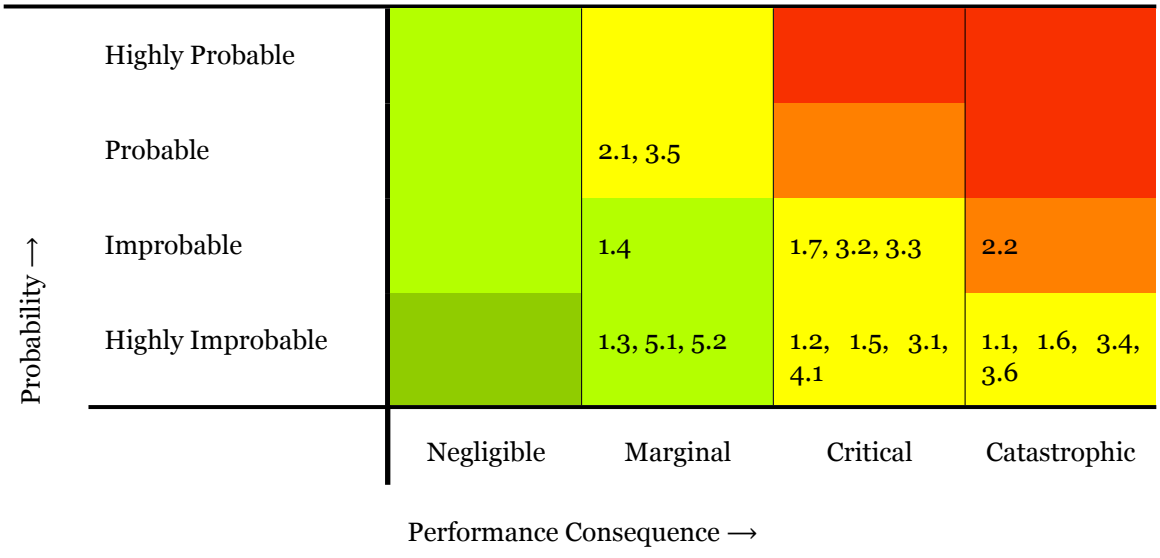
According to the risk map (Table 26.2), Risk 2.2 is the highest risk. Only for this risk, a mitigation plan will be made. The other risks are not an immediate danger to the success of the design.

It is improbable that the conductive charging system requires a too high accuracy from the UAV positioning system. To mitigate this risk, it is chosen to implement an alarm system on the supervisor interface. The server knows when the UAV is at the charging station. When it turns out that the UAV is not charging while at the station, the supervisor is alarmed and can check on the UAV. By implementing this risk mitigation plan, the consequence becomes marginal, since the UAVs will still be able to charge after the supervisors action. The risk is then reduced and moved back to the green area.

Table 26.1: Assessed Risks and Their Respective Numbers

Category	Risk No.	Description
1. UAV	1.1	The co-axial propellers do not provide enough thrust
	1.2	The required manoeuvres cannot be achieved by the variable pitch control
	1.3	An accurate representation cannot be achieved with the visualisation technique
	1.4	The structure cannot handle the predicted design loads
	1.5	The budget does not allow for the manufacturing of the UAVs
	1.6	Due to tolerances during manufacturing, the components do not fit in the UAV
	1.7	The motor cannot handle pitch control
2. Ground Station	2.1	The software for user and supervisor interface does not work as required
	2.2	The conductive charging system requires a too high accuracy from the UAV positioning system
3. Communication and Control	3.1	The communication system cannot handle the complexity of the system
	3.2	The accuracy of the sensors is not enough to make sure the UAVs do not collide
	3.3	The Bluetooth system does not provide the desired accuracy
	3.4	The range of the communication system does not include the entire flight envelope
	3.5	The software algorithms do not work as required
	3.6	The software is not compatible with the hardware
4. Orbital Mechanics	4.1	The spacing between two orbits is too low, causing conflicts between UAVs
5. Sun	5.1	The Sun does not emit sufficient light to cast shadows on the other planets
	5.2	The Sun is not suspended in a stable manner

Table 26.2: Risk Map of Technical Performance Elements



27 | Product Design & Development Logic

In this chapter the the Project Design & Development (PD&D) logic is shown. From this the Cost Breakdown Structure (CBS) follows. These will serve as a guideline for the realisation of the system after the final design of HORUS is finished.

27.1. Product Design & Development Logic

The Product Design & Development (PD&D) Logic shows the logical order of activities to be executed after the design phase of the project is finished. It provides a description and the order of activities to be executed after finalising the design. The PD&D proposed for HORUS can be seen in Figure 27.1. All of these activities are scheduled to be finished before the end of 2017, complying with the requirement **Hor-10**.

Producing a working prototype of the UAV will take a considerable amount of time. Starting with the printable plastic components, the printing of the part is estimated to take a few days. Four shell, two battery , two shaft and the Raspberry Pi components need to be printed, estimating at least an hour for each component. Producing the foam shells will take a few days as well to get the right curvatures. Producing the internal structure will take the longest amount of time. For the structure, a mould will have to be produced. Once the mould is produced, the laying for the fibres will take a day. The curing of the fibres and epoxy will take a few days. Ordering of the required raw materials will take a considerable amount of time and depends on the delivery time of the companies where the raw material is ordered. Finally, the ordering and production of the custom battery will take quite a while, where the estimated production time is solely dependent on the manufacturer of the battery.

Assembling all the system components and getting everything to fit will take a few days to a week, operating under the assumption that even though the part are designed in CATIA to fit exactly, while the actual produced components may differ slightly and have to be altered slightly to fit correctly.

Testing of the prototype of the UAV will consume the largest amount of time. Setting up test environments for the internal structure, noise, pitch control and collision avoidance will cost a considerable amount of time and manpower, estimating a minimum of 20 days to a month. Writing the operating software for the UAV will take some time and is preferably outsourced to a professional which experience in creating operating software of UAVs.

Assuming a time span of a month to acquire all the raw and off-the-shelf components, the overall estimated time to produce a working prototype of the UAV will be two and half months.

27.2. Cost Breakdown Structure

The Cost Breakdown Structure (CBS) contains all elements which add expense to the project after the design stage. The CBS is an *AND* Tree and serves to identify all the elements that contribute to the overall development and production costs. This can be used to compare the forecast and actual costs during the project. The CBS made for HORUS can be seen in Figure 27.2. Because there was limited information available regarding the precise amounts of the post-design cost of the project, a total of 120,000 € was estimated to be required for the production in addition to the cost of purchasing all the components and materials.

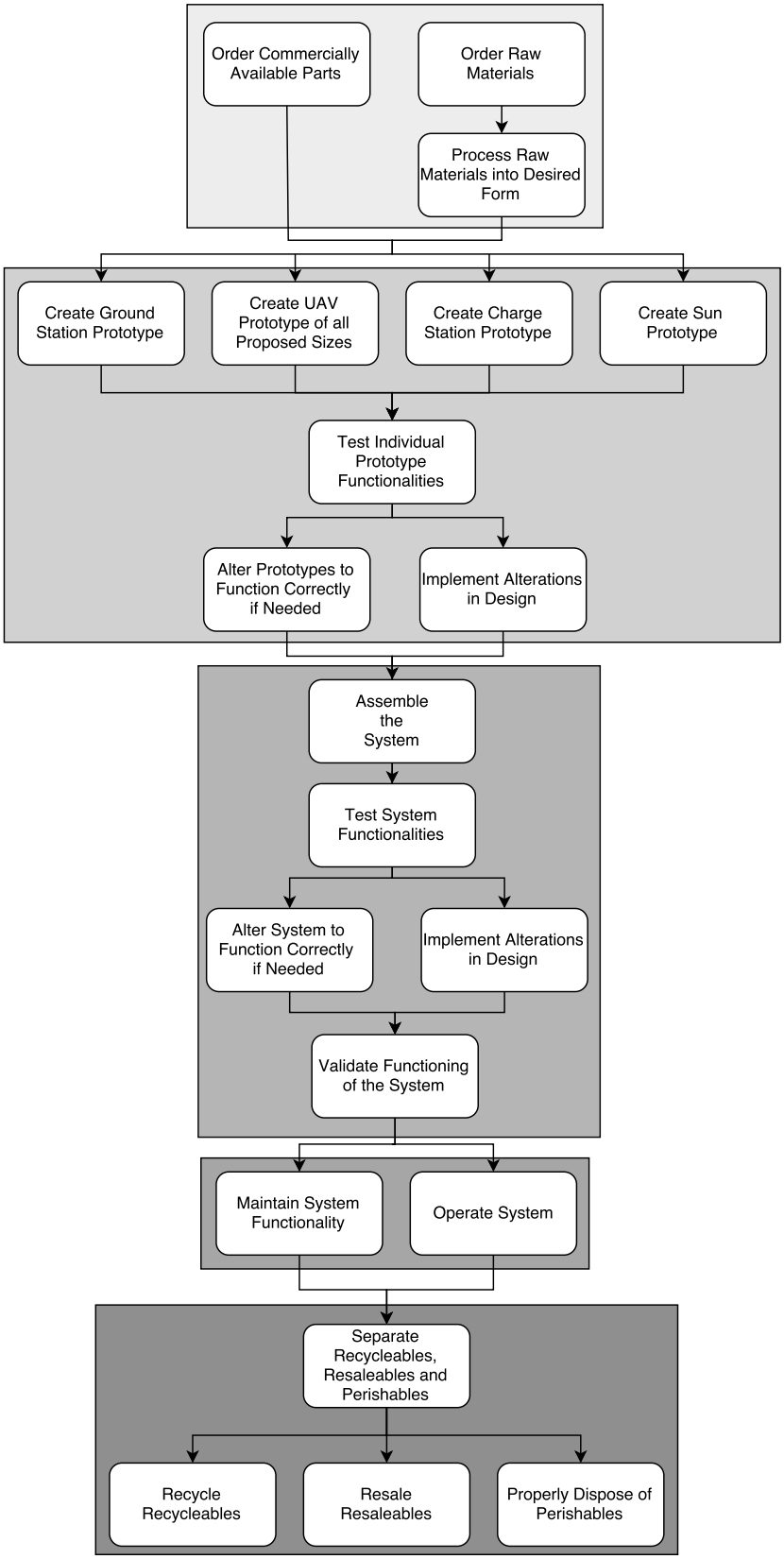


Figure 27.1: The Project Design and Development Logic for HORUS

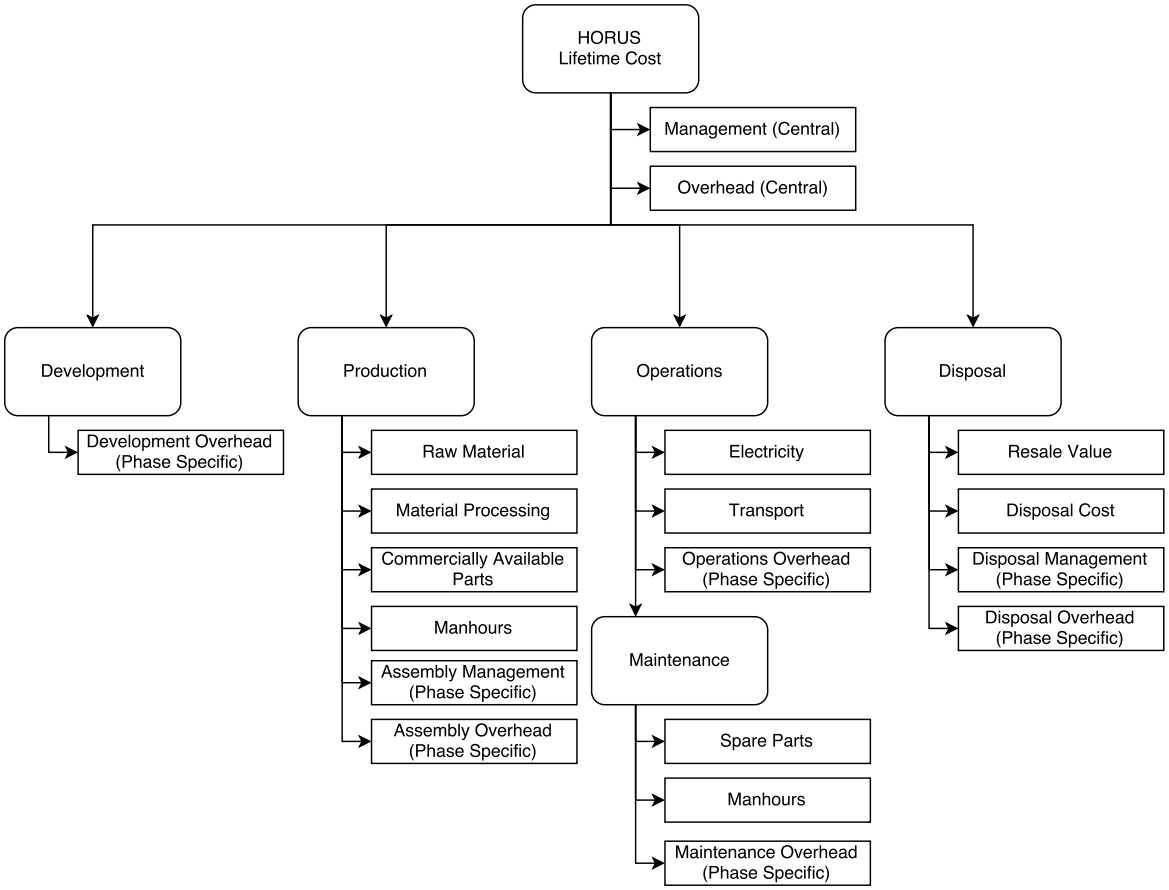


Figure 27.2: The Cost Breakdown Structure which followed from the Product Design and Development Logic for HORUS

Conclusion

The introduction of UAVs has allowed the reimagining of the age-old orrery. HORUS thus aims to represent the planetary motions by designing spherical UAVs flying in formation, which shall be operational by 2017. The design phase of the project will be performed by 9 students within 11 weeks and will have a budget of €1 M.

This orrery will allow human interaction, by allowing users to navigate between the planets, remove planets from their orbits or block the light from the sun, amongst other interactive capabilities. It will therefore also be crash-resistant, have a maximum speed comparable to walking speed and have other precautions to prevent any injury or damage to the observers and the UAVs.

The top level functions will be to turn on, perform a systems check and run in default mode until the shut down sequence is initiated. Then it will turn off. This is reflected in the flight profile.

Orbital Model

The system will be able to function in different modes. These are divided into four categories. User modes are default mode and identification mode. They can be activated by any user of the system. Modes that can only be activated by a supervisor include interaction and presentation mode. Autonomous modes are the charging and emergency mode, while administrator mode allows for powerful customisation and troubleshooting of the system.

UAV Design

E-paper was chosen for visualising the planets, but due to the mass of $110 - 660\ g$ for respectively the small and the extra large UAVs, this is not considered feasible. Therefore, a coating will be used for the graphic representation of the planets. This coating is coloured according to the dominant colours of the planets. The rings of Saturn will be from a light foam board around the UAV.

For the aerodynamic calculations, the CFD model was too inaccurate. Therefore the thrust has been determined using experimental data and research. The upper one of the two coaxial propellers will generate 57% and the lower propeller 43%. The optimal propeller spacing is $h/r = 0.446$, at which point the steadiness will be highest while the fluctuation will be minimal. A mesh will be placed between the propellers to further optimise performance. The propellers will have a diameter of $0.13 - 0.25\ m$. The motors will provide a torque of $0.06 - 0.19\ Nm$. Therefore, different motors will be used for the different sizes of UAVs. Two motors and two propellers are used for each UAV, which leads to a total power use of $340 - 520\ W$. Therefore, the motors an the propellers will have a total mass of $108 - 132\ g$, and generate enough thrust to provide hover and manaeuvring capabilities to all of the UAVs.

The UAVs will use Lithium Polymer batteries, which will have a capacity of 102.6 – 115.44 Wh. The other systems on the UAV will add an additional 5 W of power use. In order to fly 10-15 minutes, the three smallest sizes of UAVs will need two batteries of 490 g, while the biggest UAV needs a small additional battery of 44 g. The batteries will have a depth of discharge of 0.9.

The inductive charging system is infeasible due to its total mass of 1200 g and therefore a conductive charging system will be used. This system will use an integrated circuit to optimize battery charging.

The in flight forces are not expected to exceed 30 N, but impact force in case of a crash may be much higher. The design forces will thus range from $5.89 - 8.04\ kN$. The load carrying structure will be a two rings connected by semi-circles. A carbin fibre tube will be used. The beams will be attached to each other using carbon fibre tow. The shell will be made from light weight foam. This leads to a total structural mass of $70 - 330\ g$, and a total mass of $1300 - 2700\ kg$.

Communications and Control

In order to facilitate communications, each of the transmitted items is given a class. The data stream starts at the user, after which the data is transmitted to the server, which then sends the value for each class over Bluetooth to the UAVs. A total of 80 bits need to be sent to the UAV, which sends 136 bits back at 100 Hz. This requires much less than the total available Bluetooth data rate of 24 Mbps. The Raspberry Pi model 3 B will be used for data handling.

The use of five beacons and the new Bluetooth Low Energy protocol allows for positioning with a precision of 3 cm. For the short range system, four photo-resistors will be used. These will provide sufficient accuracy when landing on the charge station, which will have four matching LEDs.

The control forces will be generated using propellers with angled hinges. This is a much lighter system than conventional systems, and it will be able to generate the necessary moment to tilt the UAV to fly a path or an orbit.

When the UAVs are orbiting, the orbit model uses the actual radius and speed as feedback. This leads to a damped system that is able to react to disturbances. The path tracking mode will be used when the UAV needs to move around the space, such as to the charging station.

For collision avoidance IR Sensors will be used. Eight sensors with a field of vision of 30°will be required for the smallest UAV and 14 sensors for the biggest. Distance estimation will be done using signal strength. If it is determined a neighbouring UAV is too close, an autonomous collision avoidance manoeuvre will be started.

Contributing Elements

For the Sun, four LED bulbs will be used. These will use 144 W, to provide full light coverage. It will hang by three suspension cables with pulleys. The outside will be transparent Perspex covered by a sun-coloured sticker.

A terminal will be used to provide a graphical user interface. A planet can be selected, which will then present itself while the terminal will provide more detailed information. A supervisor interface can be operated from a personal device.

The server will send modes and parameters to the UAVs and log the data received from the UAVs. The charge station will have individual pedestals for each UAV, located in a transparent display cabinet that will be 1.8 by 1.8 by 7.2 m.

Performance

The external requirements set for HORUS have all been met, and of the internal one only the 2 kg requirement is not met for the biggest UAVs. Since the thrust generated is still considerably more than the weight, this is not considered a problem.

The UAVs will generate a predicted 62 dB of noise at 2.5 m, which is within the required levels. Actual levels will need to be measured during production.

The system will be reasonably reliable and most of the failures will result in nothing more than unscheduled maintenance. The highest two risks will be gears stripped and design errors. The system will have satisfactory availability due to the use of two sets of UAVs.

During design, material and resource selection will be analysed for sustainability. To minimise the environmental footprint of operations, energy efficient design and transport will be used. At end-of-life, the system will be disassembled into individually recyclable parts. HORUS will aim to inspire a new generation to pursue aerospace engineering and to improve awareness of sustainability.

The Commercial Aspect

The market goal of HORUS is to sell at least one system and to rent the system out for one exhibition. Dutch institutions and private buyers are considered. Potential risks are copy-cats, the small Dutch market and the size of the flight envelope. The main challenge is production delays. HORUS will be sold for € 200,000 and rented for € 10,000 /week, which leads to an expected € 300,000 of return each year. The total design and production cost of HORUS will be approximately € 29,000. The remaining budget can be used for more efficient power storage, motors or propellers and lighter materials.

The Future of HORUS

The highest risk to the system is that the conductive charging system needs a too high accuracy from the landing system, and an alarm will be introduced that informs the supervisor that there is a UAV at the charging station but not charging. The other risks are all either very improbable or they do not result in system failure.

In order to build HORUS, first materials and parts will be ordered, after which the various prototypes will be created. Then the whole system will be assembled and validated. At that point, the system can be sold. It will then be maintained until end-of-life, after which it will be recycled and disposed.

Recommendations

The Design of HORUS has been completed and HORUS will perform according to the set requirements by the stakeholders and the group members. Given the time constraint posed during the DSE, some aspects of the design have not been designed in further detail. Some improvements in the design can still be identified and will be listed in this chapter.

The control system can be optimised to function better on a Raspberry Pi and a new trade-off can be performed to see if a micro-controller such as the Arduino can be used instead. Because this is a lighter, less-power consuming system, it might be a better choice. The collision avoidance and positioning sub-systems might be improved by experimental hardware testing, to see if they can not be used more optimally than the data-sheets for the various hardware components dictate. The orbit model feedback might be improved as well when the true properties of the components are known.

The inner structure can be optimised in several ways. First the upper ring can be optimised to withstand the forces experienced during a crash on the side of the UAV. This can be done via the same iterative method as performed in Chapter 7.This however would most likely increase the overall weight of the UAVs as the dimensions would increase due to the lower number of beams on which the force is distributed. It is not expected to have the UAV crash on its side however. Regarding the optimisation of the current design, local weakpoints can be improved by for instance introducing taper. Furthermore, the manufacturing methods can be perfected by trial and error or new production methods. In case connectors are necessary, these can then be custom made to minimise the weight and maximise the structural properties.

All in all, the batteries were sized to fit the required power and capacity of each sub-system without over designing. Improved batteries might be on the market in the future, which could enhance the battery mass, capacity and charge time. It might also be possible to discard the second set of UAVs if the battery performance in increased.

The e-paper visualisation system can improve the HORUS design in many ways. It adds an extra dimension to the orrery with a dynamic visualisation and also allows for some degree of interactivity with the users. With a decreased coverage of the e-paper on the shell the visualisation is feasible with the thrust generated. However for the extra large UAV size requirement **Hor-UAV-22**, which states the mass of the UAV can not exceed 2 kg, will not be met. If this requirement is discarded, it is possible to add e-paper to the UAV design in the future.

For the propulsion system the design could be improved by getting a better understanding on the effects of all the parts on the thrust being generated. This can first be done by setting up a proper CFD model for analysis and later on during production with wind tunnel testing.

Bibliography

[1] Atmopawiro, N. R., Donders, I. J., Durrant, R. A., Douwes, D. A. J., Engelsma, J., Ionescu, L., Krol, A. M., Stikkelorum, S., and de Vrij, X., *Mid-Term Report of Design Synthesis Exercise*, TU Delft, Delft, the Netherlands, 2016.

[2] Bauer, P., Ferreira, J., Koeners, J., van der Merwe, J., Olsthoorn, H., Popovic, J., Velzeboer, T., and Weskin, M., *EE2E11: Electrical Energy Conversion*, TU Delft, Delft, the Netherlands, 2015 - 2016.

[3] *Specification of the Bluetooth System - Specification Volume 1*, Bluetooth SIG, Inc., Kirkland, WA, 2010, Page 17.

[4] Brooks, T. F., Pope, D. S., and Marcolini, M. A., "Airfoil Self-Noise and Prediction", 89N25673, NASA Langley Research Centre, Hampton, VA, 1989.

[5] Callister Jr, W. D., *Materials Science and Engineering, an Introduction*, John Wiley & Sons, Inc., Department of Metallurgical Engineering, The University of Utah, 2007.

[6] Carlisle, R., *Scientific American Inventions and Discoveries*, John Wiley & Sons, Ltd, Toronto, Ontario, Canada, 2005.

[7] Chichester, C., Bass, C., Boggess, B., Davis, M., Sanderson, E., and Di Marco, G., A Test Methodology for Assessing Demining Personal Protective Equipment (PPE), Technical report, US Army, University of Virginia, 2001.

[8] Chung, D., Continuous Carbon Fiber Polymer-Matrix Composites and Their Joints, Studied by Electrical Measurements, *Polymer Composites*, 22(2):250–270, 2011.

[9] Deters, R. W., Ananda, G. K., and Selig, M. S., "Reynolds Number Effects on the Performance of Small-Scale Propellers", In *32nd AIAA Applied Aerodynamics Conference*, Atlanta, GA, 2014. AIAA Aviation.

[10] Dijkstra, P., "Rotor Noise and Aero-Acoustic Optimization of Wind Turbine Airfoils", Master's thesis, TU Delft, Delft, The Netherlands, 2015.

[11] Fortna Murtha, J., "An Evidence Theoretic Approach to Design of Reliable Low-Cost UAVs", Master's thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2009.

[12] Gundlach, J., *Designing Unmanned Aircraft Systems: A Comprehensive Approach*, AIAA Education Series. AIAA, Reston, VA, 2012.

[13] IIT Kharagpur, Analysis of Statically Indeterminate Structures by the Displacement Method, Lecture Notes on NPTEL, 2016.

[14] Juliana de Oliveira Martins Franco, B. and Carlos Sandoval Góes, L., "Failure Analysis Methods in Unmanned Aerial Vehicles (UAV) Applications", In *19th International Congress of Mechanical Engineering*, Brasília, Brazil, 2007.

[15] Lakshminarayan, V. K. and Baeder, J. D., Computational Investigation of Microscale Coaxial-Rotor Aerodynamics in Hover, *Journal of Aircraft*, 47(3):940–955, 2010.

[16] Larson, W. and Wertz, J., *Space Mission Analysis and Design*, Space Technology Library. Microcosm Press and Kluwer Academic Publishers, El Segundo, CA and Dordrecht, the Netherlands/-Boston, MA/London, United Kingdom, 3 edition, 2005.

[17] Locke, E., Engineering Descriptive Geometry & Sheet-metal Design, Module 6A, Master's thesis, California State University Los Angeles, Los Angeles, CA, 2007.

[18] McHenry, B., Head Injury Criterion and the ATB, Technical report, McHenry Software, Inc., 2004.

[19] Mehta, R. D. and Bradshaw, P., Design rules for small wind speed tunnels, *The Aeronautical Journal of the Royal Aeronautical Society*, 83(827):443–449, November 1979.

[20] Munson, B. R., Young, D. F., Okiishi, T. H., and Huebsch, W. W., *Fundamentals of Fluid Mechanics*, chapter "Dimensional Analysis, Similitude, and Modeling", John Wiley & Sons, Ltd, Toronto, Ontario, Canada, 2009.

[21] Nationaal Centrum voor Wetenschap & Technologie, NCWT Jaarverslag 2012, Report from NEMO, 2012.

[22] Park, C., Choi, W., and Lee, J., Effects of Hardness and Thickness of Polyurethane Foam Midsoles on Bending Properties of the Footwear, *Fibers and Polymers*, 8(2):192–197, 2007.

[23] Paulos, J. and Yim, M., An Underactuated Propeller for Attitude Control in Micro Air Vehicles, In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1374 – 1379, Tokyo, Japan, 2013. IEEE.

[24] Paulos, J. and Yim, M., Flight Performance of a Swashplateless Micro Air Vehicle, In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5284 – 5289, Washington State Convention Center, Seattle, WA, 2015. IEEE.

[25] Pereira, J. L., *Hover and wind-tunnel testing of shrouded rotors for improved micro air vehicle design*, PhD thesis, University of Maryland, Maryland, WA, 2008.

[26] Pusca, A., Bobancu, S., and Duta, A., Mechanical Properties of Rubber - An Overview, *Bulletin of the Transilvania University of Brasov*, 3(52):107–114, 2010.

[27] Reimann, S., Amos, J., Bergquist, E., Cole, J., Phillips, J., and Shuster, S., "UAV for Reliability", AEM 4331 – Aerospace Vehicle Design, University of Minnesota, Minneapolis, MN, 2013.

[28] The George Washington University, School of Engineering and Applied Science, Current Carrying Capacity of Copper Conductors, Lecture Notes.

[29] Wang, Y., Yang, X., Zhao, Y., Liu, Y., and Cuthbert, L., Bluetooth Positioning using RSSI and Triangulation Methods, In *2013 IEEE 10th Consumer Communications and Networking Conference (CCNC)*, pages 837 – 842, Las Vegas, NV, 2013. IEEE.

A | Gantt Charts

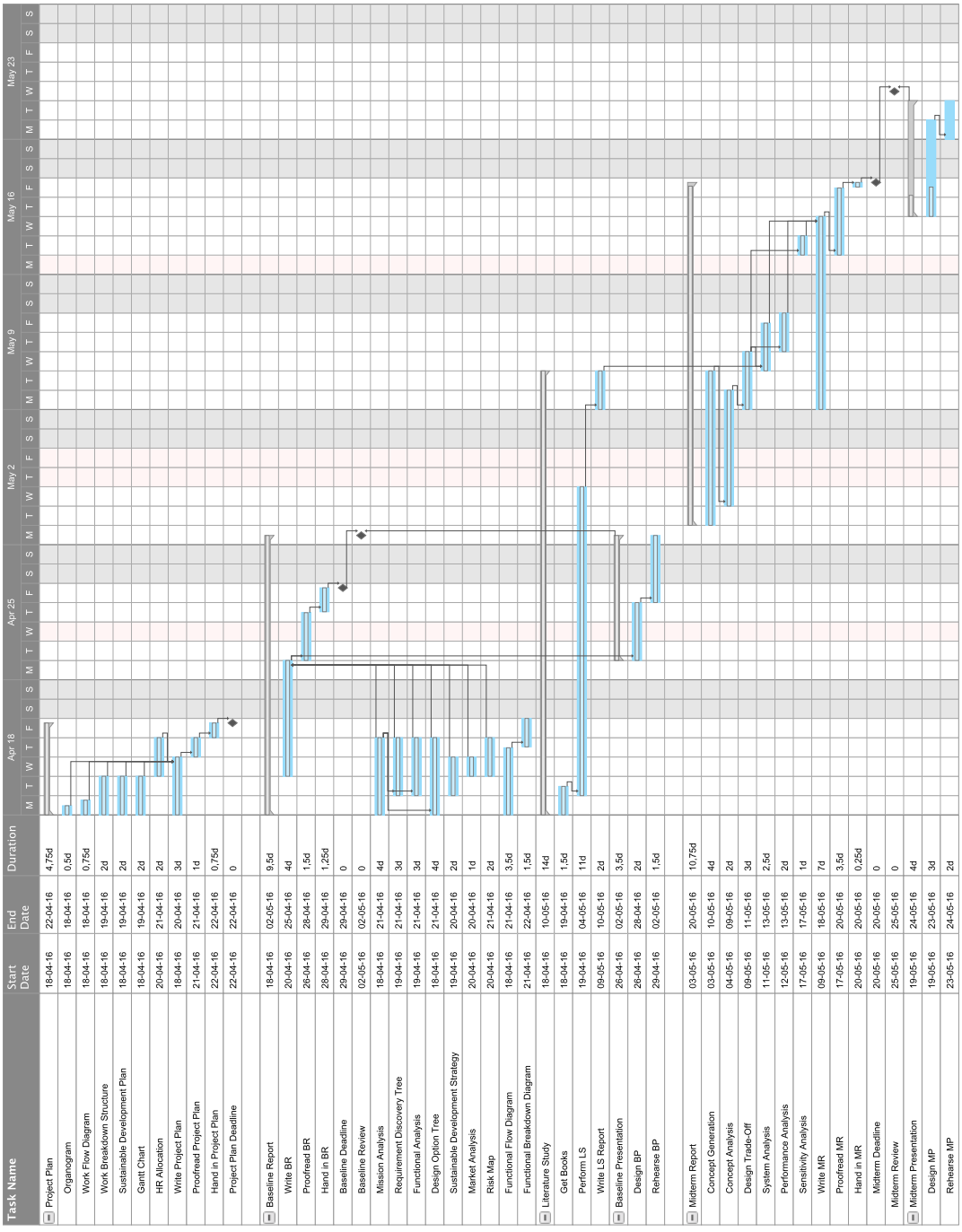


Figure A.1: Gantt Chart Up Until the Mid-Term Presentation

Figure A.2: Gantt Chart of the Final Report and Symposium

Task Name	Duration	Start	Finish
Team Assembly	45d	04-07-16	02-09-16
Design Evaluation	20d	04-07-16	29-07-16
Personnel Assessment	5d	01-08-16	05-08-16
Personnel Selection	20d	08-08-16	02-09-16
Client Identification	40d	08-08-16	30-09-16
Prototype Production	65d	15-08-16	11-11-16
Production Planning	5d	15-08-16	19-08-16
Manufacturing Planning	10d	22-08-16	02-09-16
Partner Idenification	15d	15-08-16	02-09-16
Component Specification	15d	05-09-16	23-09-16
Design Finalisation	15d	12-09-16	30-09-16
Preliminary Manufacturing	20d	03-10-16	28-10-16
Prototype Testing & Tuning	10d	31-10-16	11-11-16
Product Optimization	35d	07-11-16	23-12-16
Design Iteration	10d	07-11-16	18-11-16
Component Optimisation	10d	07-11-16	18-11-16
Final Design Production	15d	21-11-16	09-12-16
Final Design Verification & Validatio	10d	12-12-16	23-12-16
Christmas Holiday	11,25d	24-12-16	09-01-17
System Assembly	65d	21-11-16	17-02-17
Overhead Hardware Specification	10d	21-11-16	02-12-16
Hardware Production	15d	05-12-16	23-12-16
Hardware Testing & Optimization	20d	09-01-17	03-02-17
Full-Scale Production	20d	09-01-17	03-02-17
Full-Scale System Testing & Tuning	10d	06-02-17	17-02-17
System Preparation	60d	09-01-17	31-03-17
Client System Approval	5d	20-02-17	24-02-17
System Implementation & Opening F	25d	27-02-17	31-03-17
Client-Serving Marketing	60d	09-01-17	31-03-17
Opening Deadline	0	03-04-17	03-04-17
Operation	62,75d	03-04-17	28-06-17

Figure A.3: Gantt Chart of the Post-DSE Phase up to Operation Start

B | List of Requirements

B.1. Stakeholder Requirements

The stakeholder requirements are set by the client.

General Requirements

Hor-1 The system shall consist of UAVs representing the planets within the solar system.

Hor-2 The system shall represent the orbital motion of the planets within the solar system.

Hor-3 The orbital motions shall occur in a non-linear coordinate system.

Hor-4 The system shall function as an educational tool.

Hor-5 The system shall have interactive capability.

Hor-6 The system shall function within a flight envelope of 10 by 10 by 5 meters.

Hor-7 The system shall facilitate human movement between planets.

Hor-8 The system prototype shall have a maximum cost of €1,000,000.

Hor-9 The system shall function 8 hours per day for 7 days per week for a minimum of a 1 year.

UAV Requirements

Hor-UAV-1 The UAV shall be scalable.

Hor-UAV-2 The UAV shall have a crash-resistant structure.

Hor-UAV-3 The UAV shall be able to take-off after crash landing.

Hor-UAV-4 The UAV shall mimic a specific planet in an identifiable manner.

Hor-UAV-5 The UAV shall not reach a speed higher than 4 km/h.

Hor-UAV-7 The UAV exterior shall comply with playground regulations.

Hor-UAV-8 The UAV shall have a spherical shape.

Control Requirements

Hor-Ctrl-1 The UAV shall have a collision avoidance system.

Hor-Ctrl-2 The UAV shall recharge autonomously.

Hor-Ctrl-5 The UAV shall land without causing injury when loss of power occurs.

Hor-Ctrl-6 The UAV shall land without causing injury when loss of data communication with base station occurs.

B.2. Internal Requirements

The internal requirements are set by the design team.

General Requirements

Hor-10 The system shall be functional in 2017.

Hor-11 The system shall be child friendly.

Hor-12 The system shall function indoors.

Hor-13 The system shall consist of at least 2 sets of UAVs.

Hor-14 During operation time, there shall be 8 UAVs functioning at any given time.

Hor-15 In the event that one of the planets malfunctions or needs to charge, its twin shall take its place.

Hor-16 The height of the Sun shall be adjustable.

Hor-17 A set of UAVs shall consist of at least the 8 planets.

UAV Requirements

Hor-UAV-9 The UAV exterior shall have a minimum diameter of 10 centimeters.

Hor-UAV-10 The UAV shall have hover capability.

Hor-UAV-12 The UAV shall be able to land autonomously.

Hor-UAV-13 The UAV shall produce a maximum of 70 decibels of noise at a distance of 2.5 meters.

Hor-UAV-14 The UAV shall plan an unobstructed trajectory to the charging unit.

Hor-UAV-17 The UAV exterior shall have a maximum radius of 50 centimeter.

Hor-UAV-18 The UAV shall be able to translate along the three body axes.

Hor-UAV-19 The UAV shall be able to rotate about the three body axes.

Hor-UAV-20 The UAV shall be able to take-off vertically.

Hor-UAV-21 The UAV shall be able to land vertically.

Hor-UAV-22 The UAV shall individually have a maximum weight of 2 kg.

Control Requirements

Hor-Ctrl-7 The UAV shall land without causing injury when an emergency situation occurs.

Hor-Ctrl-8 The UAV shall have an altitude determination frequency of 1 Hz.

Hor-Ctrl-9 The UAV shall have a position determination frequency of 10 Hz.

Hor-Ctrl-10 The UAV shall have a heading determination frequency of 10 Hz.

Hor-Ctrl-11 The UAV shall have a orientation determination frequency of 10 Hz.

Hor-Ctrl-12 There shall be a standard operating mode.

Hor-Ctrl-13 When in standard operating mode, the UAVs shall function above a height of 2.5 meters.

Hor-Ctrl-14 There shall be a presentation mode.

Hor-Ctrl-15 The server will set the UAV flight mode.

Hor-Ctrl-16 The temperature of the battery shall not exceed 60 degrees Celsius.

Hor-Ctrl-17 The temperature of the screen shall not exceed 60 degrees Celsius.

Hor-Ctrl-18 The UAV shall use less than 300 Watt during operation time.

Hor-Ctrl-19 The Visualisation of the UAV shall use less than 40 Watt.

Hor-Ctrl-20 The UAV shall have autonomous path planning capabilities.

Hor-Ctrl-21 The propulsion subsystem shall be able to deliver a control force of at least 0.1 N.

Hor-Ctrl-22 The UAV shall be able to manoeuvre to a specified set of coordinates with an accuracy of 2.5 cm.

Hor-Ctrl-23 The UAV shall be able to detect objects with a radius bigger than 0.1 m at a minimum distance of 0.5 m.

Ground Station Requirements

Hor-Gnd-1 The ground station software shall have a user-friendly interface.

Hor-Gnd-2 The ground station shall have a manual control interface.

Hor-Gnd-3 The ground station shall have a data storage capability.

Hor-Gnd-4 The ground station shall have a screen.

Hor-Gnd-6 The charge station shall charge multiple UAVs simultaneously.

Hor-Gnd-7 The charging time shall be equal to or less than the operation time.

Hor-Gnd-8 The operating system shall be located outside of the flight envelope.

Hor-Gnd-9 The charging station shall have storage space for 2 sets of UAVs.

C | Propeller Characteristics

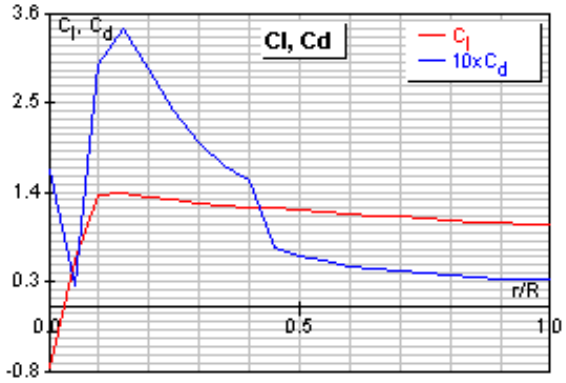


Figure C.1: Drag Polar of the Zagi Carbon 5.1x4.9 Propeller generated by JavaProp with a Free Stream Velocity of 1.00 m/s and Reynolds and Mach Number Distribution as Shown in Figure C.2

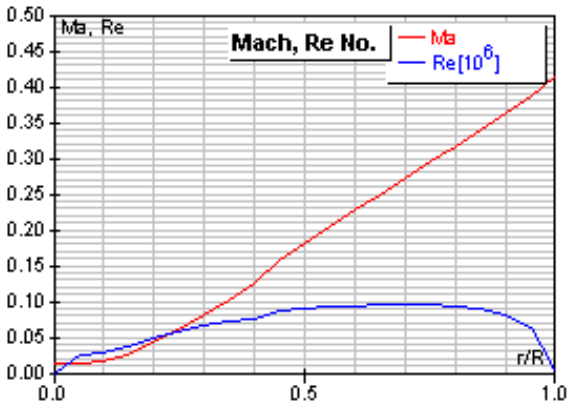


Figure C.2: Reynolds and Mach Number Distribution of the Zagi Carbon 5.1x4.9 Propeller as generated by JavaProp with a Free Stream Velocity of 1.00 m/s at Sea-level Conditions

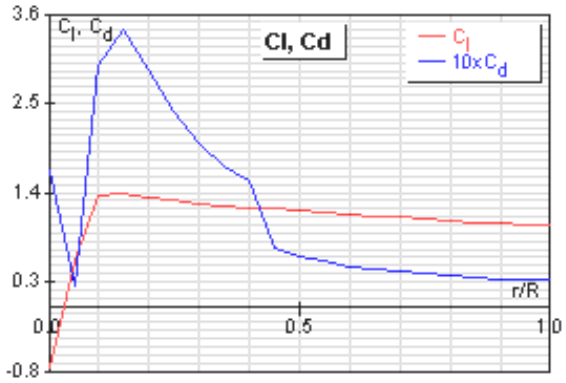


Figure C.3: Drag Polar of the APC E 7x6 Propeller generated by JavaProp with a Free Stream Velocity of 1.00 m/s and Reynolds and Mach Number Distribution as Shown in Figure C.4

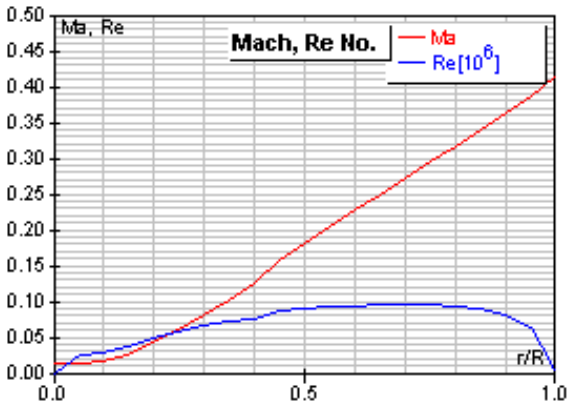


Figure C.4: Reynolds and Mach Number Distribution of the APC E 7x6 Propeller as generated by JavaProp with a Free Stream Velocity of 1.00 m/s at Sea-level Conditions

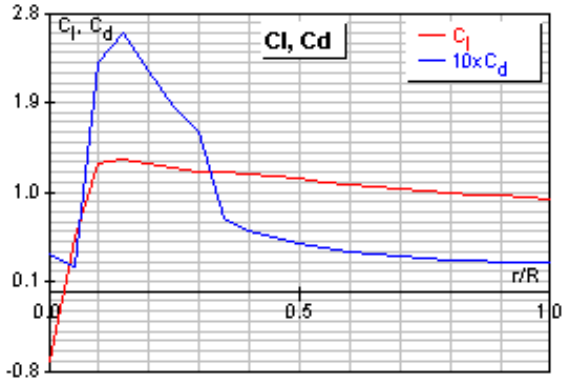


Figure C.5: Drag Polar of the APC E 8x4 Propeller generated by JavaProp with a Free Stream Velocity of 1.00 m/s and Reynolds and Mach Number Distribution as Shown in Figure C.6

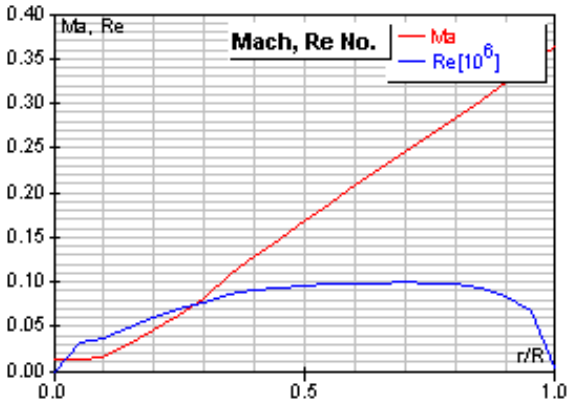


Figure C.6: Reynolds and Mach Number Distribution of the APC E 8x4 Propeller as generated by JavaProp with a Free Stream Velocity of 1.00 m/s at Sea-level Conditions

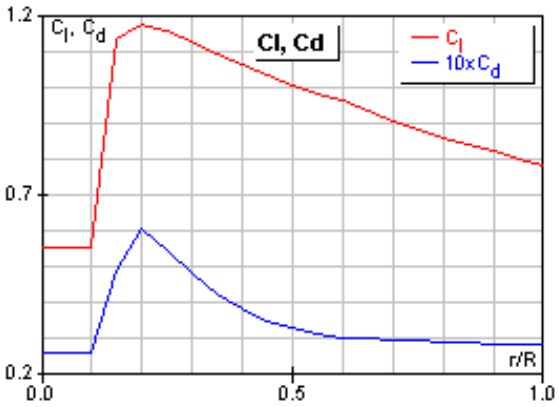


Figure C.7: Drag Polar of the APC 10x5 Propeller generated by JavaProp with a Free Stream Velocity of 2.00 m/s and Reynolds and Mach Number Distribution as Shown in Figure C.8

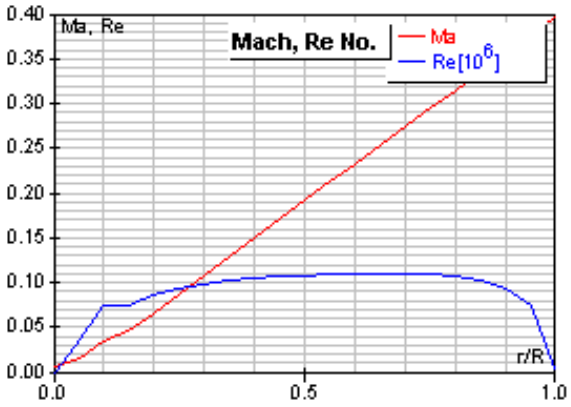


Figure C.8: Reynolds and Mach Number Distribution of the APC E 10x5 Propeller as generated by JavaProp with a Free Stream Velocity of 2.00 m/s at Sea-level Conditions