Design Synthesis Exercise Final Report

DRIVE-FC: Drone-based Responsive Intra-warehouse Vehicle Ecosystem for Fulfilment Centers

Group 15

Delft University of Technology

TU Delft

This page is left blank intentionally

Design Synthesis Exercise Final Report

DRIVE-FC: Drone-based Responsive Intra-warehouse Vehicle Ecosystem for Fulfilment Centers

by

Group 15

Berend de Vries	5988152
Bram Boekema	5723175
Cristian Savin	5981492
Joost Couwenberg	5567718
Luca van Workum	5696763
Mats Macke	5910846
Maxim Shcherbina	5803489
Pradyumn Mishra	5934559
Shiyu Takamura	5979625
Tim Schaeffer	5931037
Zeynep Azra Özel	5962544

Mentor:	Alessandro Bombelli
Coaches:	Christopher Lankhof, Olivier Moriaux
Submitted on:	Tuesday 23rd June, 2026
Faculty:	Faculty of Aerospace Engineering, Delft

Cover:	Drone renders created by the authors Background generated with ChatGPT
Style:	TU Delft Report Style, with modifications by Daan Zwaneveld

TU Delft

Executive Overview

Grocery delivery services have experienced a large increase in users during recent years[35]. In order to function, these services require warehouses to store items before being distributed. In many cases, intra-warehouse transportation is being handled by human workers. This project aims to design an autonomous intra-warehouse logistic systems using drones. The system is specifically tailored to the use case of grocery delivery fulfilment centres. The role of the system in these warehouses will be to bring items from stocks to the extraction area. The need and objective of the Drone-based Responsive Intra-warehouse Vehicle Ecosystem for Fulfilment Centers (DRIVE-FC) project can be summarized as follows:

Mission Need Statement

Provide an autonomous efficient intra-warehouse grocery transport system.

Project Objective Statement

Design an autonomous integrated drone-based intra-warehouse transport system for an e-grocery fulfilment centre.

The final design entails a fleet of 15 simultaneously operating hexacopter drones. The drone can carry payloads up to 1.0 kg and together and the fleet moves nearly 25,000 items per day. Further details on the design are discussed in this document. This overview provides a summary of the design Group 15 has produced during the ten week design phase. This project was overseen by our principal tutor Alessandro Bombelli, and our coaches Olivier Moriaux and Christopher Lankhof.

Market analysis

The e-grocery and, by extension, the warehouse automation market has shown strong growth trends, and is expected to grow due to the increasing need for efficient operations, and a broader demand for products, including specialised and niche items [44, 35]. An analysis of competitors reveals an opportunity for innovation, as a small number of major players dominate the market of Guided Vehicles, conveyors and Automated storage and retrieval systems solutions[55]. Indoor aerial drone fleets are only implemented for inspection purposes with little to no interaction with items [52, 16], leaving a niche for DRIVE-FC to thrive within the market and making it a one-of-a-kind solution.

From the analysis of the stakeholders it was found that the most interested are the warehouse workers and operator. The most influential stakeholders are identified as the client and the government, as the government has the power to stop development of the project if regulations are not respected. For the client, which is the most influential and and interested stakeholder, the main need was found to be to reduce operating costs, giving rise to requirements that constrain the project budget. From a SWOT analysis the strengths, weaknesses, opportunities, and threats were identified. The main strength found was the flexibility that an aerial drone system provides, and the main weakness: the complexity and cost of the system along with its limited capacity. The main opportunities are the advancements of the e-grocery market and drone technology, and the main threats are regulations that would prohibit flying within warehouses, as well as the competition from other warehouse automation products.

Project Management and Systems Engineering

The primary system functions are to operate autonomously, maintain a human-friendly working environment, and provide smooth and efficient item transportation within the warehouse. At the highest level, these functions describe the purpose and intended operation of the system. However, the project is fundamentally driven by stakeholder and user needs, as these define the expectations and constraints within which the system must operate. Therefore, these needs form the basis of the design process and are translated into a set of top-level stakeholder requirements.

A comprehensive Verification and Validation (V&V) plan was developed according to the systems engineering

V-model to ensure that stakeholder needs are correctly translated into system requirements and ultimately reflected in the final design. To support requirement validation activities, the required resources and testing infrastructure were identified, resulting in an estimated V&V cost of €44,270. Further execution of the V&V plan involves analysing developed models (S2M2 routing algorithm, flight dynamics simulation and CAD models) and tools (sizing tool, sensitivity analysis and warehouse optimisation tool). Finally, the overall system design is validated through mission-oriented simulations, stress testing, and operational readiness demonstrations within comparable warehouse environments.

To help inform the design, a technical risk analysis has been conducted. Risks have been divided into several categories, including operations, manufacturing and integration. Each risk has an associated likelihood and impact. From this combination, the risk is calculated. Significant risks have been assigned both a mitigation and contingency strategy, reducing their impact to more manageable levels.

Logistics and Warehouse

For an efficient system, a fitting logistical flow and warehouse layout was created. For this system, mainly the picking stage is modified compared to current typical operations. The warehouse layout can be seen in Figure 1. Products are first classified as either drone-pickable or non-drone-pickable before being stored in the appropriate ambient or chilled storage zone. This separation minimises worker-drone interaction and thereby improves operational safety.

The warehouse layout was optimised for space utilisation, SKU placement, and suitability for drone swarm operations. Each drone picking zone measures 42.1 m in width, 16.8 m in depth, and 6 m in height. The final configuration consists of six aisles with 2.5 m spacing between shelves. Shelves are arranged in three layers, ensuring 1.5 m of free operating space above the stored products for drone operations.

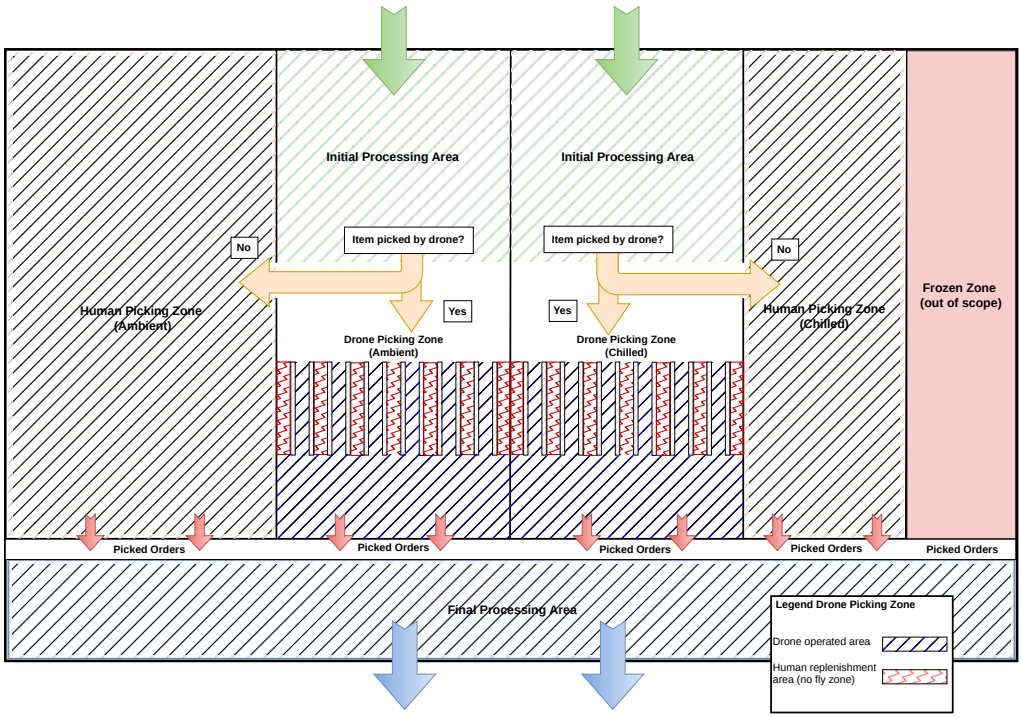


Figure 1: Warehouse layout and logistical flow

Propulsion and Noise

For propulsion, propellers were chosen based on the results from custom adjusted Blade Element Momentum Theory (BEMT). This theory designs a propeller for a given point design. Consequently, the remaining flight envelope was calculated for the given propeller geometry for the full velocity range. From this, parameters such as efficiency, mass and vertical separation distance were defined. This resulted in the final choice of the MID 118 propeller. After modelling the MID 118 in CATIA, a toroidal loop was added due to its potential aero-acoustic noise reduction.

Power and Energy Replenishment

The power sources for the drones were selected to be LFP batteries due to their long lifespan, stability and recyclability. This choice reduces operating costs, improves safety, and supports the project's sustainability goals. These batteries contain 24 cells, of which three groups of eight are posed in series. These three series are then connected in parallel. One cell has a capacity of 8.58 Wh, a voltage of 3.3 V and they are of type 26650. Furthermore, the batteries are swappable which introduces additional complexity and cost but also ensures higher system efficiency by minimising drone downtime. When the energy of the drones is nearly depleted, the drones move toward the Energy Replenishment Station (ERS), where it aligns itself with the structure before the battery is exchanged. The ERS is designed such that the depleted battery is removed from one side while a fully charged battery is inserted from the other, reducing the overall swap time. The discharged batteries are subsequently transported upwards within the ERS to the charging section, which has capacity for 15 batteries. A single ERS measures 4.55 m $\times$ 1.70 m $\times$ 1.48 m. Two ERSs are stacked vertically, resulting in a total height of 2.94 m. Both the ambient and chilled warehouse zones are equipped with one such ERS stack. Due to the high utilisation of the batteries, they require regular replacement, resulting in an estimated annual battery consumption of 204 batteries.

Grabber

The grabber subsystem is used to pick up, transport and deposit the payload. For these purposes, a bio-inspired tentacle based concept was chosen. It is cheap, lightweight, easy to control, and is able to lift a large variety of items. There are two actuation servos to control the mechanism, and a third one to control the arm it is mounted on. It is capable of picking up a hypothetical worst-case payload of a 50 cm x 25 cm x 3 cm cuboid. The whole system weighs roughly 400 g, and uses 70 W during peak operation.

The mechanism itself is at the end of an arm that is capable of folding away during emergency landings and so the legs can be designed shorter. In order to minimise the moving mass, all servos are mounted to the drone body, and the moving cables are fed through the arm to the tentacles.

Drone Body

For the drone body subsystem, a modular design was selected, prioritizing maintainability, low production cost, and sustainability over a fully integrated, more efficient drone structure. The mass of the body resulted to be 2.1 kg. The physical assembly features carbon fibre arms clamped at both ends connecting the drone body to the motors, independent aluminium landing legs, and four identical body plates to reduce unique part counts. To isolate high-frequency motor vibrations and ensure reliable measurements from the arm-mounted mm-wave sensors, an o-ring damping configuration was integrated. Structural verification was performed in two stages. Firstly, the core body, arms, and legs were evaluated using conservative, simplified analytical formulas under static hover loads. This was scaled by a factor of two, to account for the maximum thrust to weight ratio of two. While manual calculations yield stresses well below the tensile limit for all the evaluated parts, the compliance status remains open. Full verification requires post-DSE implementation of an advanced Finite Element Method (FEM) simulation, a dynamic fatigue model, and sensor vibration testing to deliver a complete product.

Localisation

The localisation subsystem determines the drone's position in the warehouse, feeding directly into control. After comparing on-board SLAM, external motion-capture, and anchor-based options, an anchor-based Ultra Wide Band solution was selected for its cost, robustness, and proven indoor use. Position is found by Time Difference of Arrival (TDoA), solved as a nonlinear least-squares problem, with the DW1000 UWB transceiver, an MPU-9250 IMU, and an upward-facing mmWave radar acting as a vertical time-of-flight altitude aid, all fused into a loosely-coupled Extended Kalman Filter (EKF) running at 250 Hz. The estimator was verified against the real-world UTIL UWB-TDoA dataset, achieving $\sim$ 4 cm RMSE in line-of-sight (within the 10 cm requirement) but degrading to $\sim$ 31 cm under a heavy four-anchor no-line-of-sight (NLOS) case. The 1 cm pickup/drop-off accuracy cannot be met by UWB alone, so close-range mmWave ranging is left to future work.

Collision Avoidance

DRIVE-FC's collision avoidance subsystem fuses nine mmWave radars into a DBSCAN-clustered point cloud, treating each obstacle as a spherical Collision Safety Boundary (cluster radius plus 0.9 m clearance). Its modified collision cone adds a secondary projection sphere ($R_2 = R + kD$, $k = 0.2$), pushing aiming points further out and smoothing heading changes. In the single-obstacle test, average heading change fell to 3.05°

versus 5.43° for the standard collision cone, with peaks staying under 10°, at the cost of greater clearance (1.808 m vs 1.554 m) and path deviation (3.453 m vs 2.886 m). The multi-obstacle case showed an even stronger improvement: average heading change of 1.44° versus 5.45° and maximal heading change of 31.5° versus 53.2°, with closest distance effectively tied ($\sim$ 1.26 m) and again larger deviation (3.277 m vs 2.885 m). Of ten requirements, four were verifiable: the mmWave hardware operates within the -40 °C to 85 °C range and far exceeds the readout frequency requirement, while clearance was only partially compliant since kinematic constraints were not implemented. Key caveats are that the system is purely reactive, was tested only on idealized sensor data, and uses a static k value.

Communication

The communication subsystem handles data exchange between the drones, central computer, and the warehouse inventory and ERS systems. Rather than a dedicated radio, it reuses the localisation UWB link, the multifunctionality of sharing ranging and data handling hardware was the deciding factor over Wi-Fi or a wired option. A data-budget analysis shows that the high-rate streams stays local, leaving only low-rate routing data on the 6.8 Mbps UWB link, which meets the 1 Mbps bandwidth requirement.

Routing

The routing subsystem is structured around a warehouse database which stores static objects, a scheduler which assigns tasks and a routing algorithm which generates paths for individual drones whilst avoiding static obstacles and other drones. A separate charge cycle balancer regulates the timing of drone battery replacement to prevent ERS stations from being overloaded. The routing algorithm uses a Multi Agent Pathfinding algorithm called S2M2; a continuous-space algorithm based on Priority-Based-Search. The current simulation implementation lacks the Large Neighbourhood Search optimization which would prevent unnecessary recalculation of paths. As such, to compensate: a lower accuracy collision sampling and coarser time-step meshing is used.

To test the performance of both the routing subsystem and total throughput, a logistical simulation was created. This simulation generates incoming orders and emulates drones' responses to path commands in order to test how the whole system would operate as a whole. As a result, an average throughput of 24003 products delivered per day was recorded.

Control

For the control subsystem, a cascaded PID was selected, resulting in twelve configurable non-zero PIDs controlling the drone dynamics. Flight modes were created to maintain desirable behaviour across different configurations, situations or tasks. These include cruise, stable and One Engine Inoperative (OEI). To ensure stable flight, the drone was limited in velocity, acceleration and attitude depending on the flight mode. For cruise this entails a maximum cruise velocity of 3.0 m/s in both horizontal and vertical directions, and a maximum tilt angle of 30.0 degrees.

The control system was tuned in two stages. Firstly, the dynamics were linearised around hover such that initial estimates could be made. These were then used in the second stage, where the Hooke Jeeves optimisation method was used to tune the 36 gains. The resulting behaviour complies with the set requirements and is able to follow trajectories closely.

Final Design

The final integrated design of the DRIVE-FC drone entails a hexacopter drone with asymmetric looped propellers. The drones carry a flight computer, IMU and UWB tag, six motors, six mmWave sensors, an electronic speed controller, a motor controlled arm, the tentacle grabber, its two motors and a swappable battery. The grabber is connected to an arm such that it can be rotated and does not interfere with landing procedures. Complete hardware, software, electrical and data architectures were developed, with the flight computer acting as the central processing unit and the battery as the power source. The drone communicates with the warehouse through UWB, with built-in redundancy through multiple anchors.

The final mass of the drone is 6.89 kg, which is below the budgeted mass. Worst case power consumption is 1.23 kW, again below the allocated budget of 1.46 kW. Only the Raspberry Pi 5 exceeded its power budget but was still chosen for its simple implementation and performance. The fleet counts 17 drones of which 15 are operational at any given time and two are stationed at the ERS. Together, they move 24003 items a day, taking on average 38.7 s to deliver an item. The cruise speed is set at 3 m/s with a tracking accuracy of

0.15 m. The drones can be localised with 0.04 m accuracy with no obstacles, and 0.31 m in the worst case scenario. Sensitivity analysis showed that fleet size could be reduced to an absolute minimum of eight whilst still meeting the 10,000 item throughput requirement, however, this is less financially efficient. The final drone design is visualised in Figure 2.

Figure 2: DRIVE-FC drone layout

Cost analysis

DRIVE-FC's cost analysis confirms strong economic viability against critical budget requirements RQ-BDT-01 and RQ-BDT-02. The per-unit drone cost totals €1,792.18, far below the €10,000 cap (about 82% headroom), with the propeller motors as the dominant driver at roughly €630.58; battery and manufacturing labour costs are instead folded into CAPEX/OPEX. The cash flow analysis assumes a 4.3% Dutch wage growth rate, an 8% WACC, a 10-year lifespan, and a 20% CAPEX contingency that brings total CAPEX to €1,169,360.45 (just under the RQ-BDT-01 limit). This yields a net present value of about €11.89 million, a discounted payback period of two years, and 27.01% savings in total operational costs, enabled by $\sim$ 25,000 items/day throughput that replaces 27 full-time employees with a 17-drone fleet. The sensitivity analysis found payback period to be relatively insensitive to CAPEX, as payback only reaches three years above €1,450,000. The sensitivity to throughput is surprisingly weak, with payback staying at two years even near a 25% increase, while OPEX is the dominant risk. A €250,000 rise pushes payback to three years and cuts savings by 3.81%, and the project becomes unprofitable if OPEX exceeds €5,500,000.

Manufacturing and Integration

As the ten week scope of the project came to an end, a manufacturing, assembly, and integration plan was created for future use. Manufacturing is organised into four parallel assembly streams: the drone, the ERS, the UWB localisation network, and the central computer. The fastener and predominantly off-the-shelf component based drone design enables assembly, disassembly, and maintenance by a single technician using standard hand tools, whilst interchangeable components simplify manufacturing and spare-part management. During integration, the ERSs are installed within the warehouse, the UWB anchors are positioned to provide full localisation coverage, and the central computer is connected to coordinate communication, localisation, and fleet operations, resulting in a fully integrated autonomous fulfilment system.

Sustainability

Sustainability has primarily guided the design by three UN Sustainable Development goals: Decent Work and Economic Growth (SDG 8), Industry, Innovation and Infrastructure (SDG 9), and Responsible Consumption and Production (SDG 12). The use of recycled materials, minimising worker-drone interactions, and ensuring long-term economic viability where incorporated in the design process. A life cycle assessment has been conducted in which a full cycle CO_2 analysis identified operational energy consumption as the largest contributor to environmental impact, followed by battery production and replacement. Over its 10-year

lifetime, DRIVE-FC emits 228,160 kg CO_2, 27,043 kg more than the equivalent human picking system. Future recommendations include reducing the drone energy consumption and battery replacements.

RAMS Analysis

The Reliability, Availability, Maintainability and Safety (RAMS) analysis showed that for reliability, the risks are mitigated by incorporating redundancies in propulsion, sensors, infrastructure and operationally. In the end, the multi-agent pathfinding algorithm serves as the primary bottleneck of reliability, having one conflict every 2.4 minutes while 15 drones are operative. Availability is estimated at 99.8%, with downtime primarily caused by occasional human intervention in drone operating zones. Since interchangeable off-the-shelf components are used in the design, maintenance is done rapidly with simple hand tools. Preventive maintenance includes bi-annual inspections, major maintenance every 2.5 years, a complete system overhaul after 5 years, and decommissioning after 10 years. Worker safety is ensured through physical separation of worker and drone zones, onboard obstacle detection, and a contingency procedure for out-of-control drones, providing multiple layers of protection.

Project Future

The current design forms the basis of DRIVE-FC, with the next phase focusing on further detailed design, verification, validation, prototyping and iterative procedures before the project can be taken to full-scale manufacturing. The project timeline targets design finalisation in September 2026, prototype manufacturing in October 2026, prototype testing in December 2026, system manufacturing in February 2027, final integration in March 2027, and start of operations in March 2027. The system is designed to have a 10-year operational lifetime, supported by maintenance every 6 months, major inspections after 2.5 and 7.5 years, and a complete system overhaul after 5 years. The projected end of the systems operational life is in March 2037, when the system will be decommissioned.

Contents

List of Abbreviations

AFS Automatic Fulfilment Systems
AGV Automated Guided Vehicle
AI Artificial Intelligence
ALARP As Low as Reasonably Practicable
AoA Angle of Attack
AS/RS Automated storage and retrieval systems

BEMT Blade Element Momentum Theory
BLDC Brushless Direct Current
BPF Blade Passing Frequency
BVI Blade Vortex Interactions

CAD Computer-Aided Design
CAGR Compound Annual Growth Rate
CAPEX Capital Expenditure
CFD Computational Fluid Dynamics
CO_2 Carbondioxide
CoG Centre of Gravity

dBA A-Weighted Decibels
DBSCAN Density Based Spatial Clustering of Applications with Noise
DoD Depth of Discharge
DoF Degrees of Freedom
DRIVE-FC Drone-based Responsive Intra-warehouse Vehicle Ecosystem for Fulfilment Centers
DSE Design Synthesis Exercise

EKF Extended Kalman Filter
ERS Energy Replenishment Station
ESC Electronic Speed Controller

FEM Finite Element Method
FoM Figure of Merit

IMU Inertial Measurement Unit

LCA Life-Cycle Assessment
LFP Lithium Iron Phosphate
LiDAR Light Detection and Ranging
Linear MPC Linear Model Predictive Control
LM Levenberg-Marquardt
LNS Large Neighbourhood Search
LOS Line of Sight

MAI Manufacturing, Assembly and Integration
MAPF Multi-Agent Pathfinding
MIT Massachusetts Institute of Technology
ML Machine Learning
MTTR Mean Time To Repair

NL Netherlands
NLOS No Line of Sight
NPV Net Present Value

OASPL Overall Sound Pressure Level
OASPL-A A-weighted Overall Sound Pressure Level
OEI One-Engine-Inoperative
OPEX Operational Expenditure
ORCA Optimal Reciprocal Collision Avoidance
OTS Off-The-Shelf

PID Proportional-Integral-Derivative

RAMS Reliability, Availability, Maintainability and Safety
RMSE Root Mean Square Error
RPM Revolutions Per Minute

S2M2 Scalable and Safe Multi-Agent Motion Planning with Non-linear Dynamics and Bounded Disturbances
SAMC Semi-annual Maintenance Check
SF Shaft Frequency
SKU Stock Keeping Unit
SLAM Simultaneous Localisation and Mapping
SWOT Strengths, Weaknesses, Opportunities and Threats

TBD To be decided
TDoA Time Difference of Arrival
ToF Time of Flight
TPU Thermoplastic PolyUrethane
TWR Two Way Ranging

UAV Unmanned Aerial Vehicle
UHMWPE Ultra High Molecular Weight Polyethylene
UN United Nations
USD United States Dollar
UWB Ultra Wideband

V&V Verification and Validation
VALID Verifiable, Achievable, Logical, Integral, and Definitive

WACC Weighted Average Cost of Capital

List of Symbols

$\mathbf{a}$ Anchor position vector	m
$\boldsymbol{a}$ Acceleration	m/s^2
$\boldsymbol{\alpha_i}$ Induced angle of attack	rad
$\boldsymbol{\alpha_{design}}$ Design angle of attack	rad
$\boldsymbol{c}$ Speed of sound	m/s
$\boldsymbol{C_d}$ Drag coefficient	[-]
$\boldsymbol{C_l}$ Lift coefficient	[-]
$\boldsymbol{c_{max}}$ Maximum chord length	m
$\boldsymbol{\delta}$ Characteristic length	m
$\boldsymbol{D_{skin}}$ Skin friction drag	N
$\boldsymbol{E}$ Young's modulus	MPa
$\boldsymbol{e}$ Span efficiency factor	[-]
$\boldsymbol{\eta}$ Aerodynamic efficiency	[-]
$\boldsymbol{F}$ Force	N
$\boldsymbol{f_{vortex}}$ Vortex frequency	Hz
$\boldsymbol{g}$ Gravitational constant	9.81 m/s^2
γ Separation angle between the blades	[-]
$\boldsymbol{h}$ Height	m
$\boldsymbol{I}$ Moment of inertia	kg m^2
$\boldsymbol{J}$ Advance ratio	[-]
$\boldsymbol{K}$ Scaling factor	[-]
$\boldsymbol{l}$ Length	m
$\boldsymbol{M}$ Moment	Nm
$\boldsymbol{m}$ Drone mass	kg
$\boldsymbol{\omega_b}$ Angular velocity	rad/s
$\boldsymbol{\dot{\omega}_b}$ Angular acceleration	rad/s^2
$\mathbf{p}$ Position vector	m
$\boldsymbol{p}$ Roll rate	rad/s
$\boldsymbol{\phi}$ Inflow angle	rad
$\boldsymbol{\phi}$ Angle between subsequent tentacles	rad
$\boldsymbol{q}$ Pitch rate	rad/s
$\boldsymbol{Re}$ Reynolds number	[-]
$\boldsymbol{R}$ Radius	m
$\boldsymbol{r}$ Range difference	m
$\boldsymbol{r}$ Yaw rate	rad/s
$\boldsymbol{r_{prop}}$ Propeller radius	m
$\boldsymbol{St}$ Strouhal number	[-]
$\boldsymbol{\sigma}$ Stress	N/m^2
$\boldsymbol{t}$ Time	s
$\boldsymbol{T_{actual}}$ Actual thrust	N
$\boldsymbol{T_{req}}$ Required thrust	N
$\boldsymbol{\tau_b}$ Torque	Nm
$\boldsymbol{\theta}$ Pitching angle	rad
$\boldsymbol{\theta_{lim}}$ Spiral Tentacle Angle Parameter	rad
$\boldsymbol{v}$ Velocity	m/s
$\boldsymbol{V}$ Flow velocity	m/s
$\boldsymbol{V_{tan}}$ Tangential velocity	m/s
$\boldsymbol{V_{fwd,n}}$ Forward normal velocity	m/s
$\boldsymbol{w}$ Width	m

1

Introduction

In recent years, the prevalence of app-based services has been on the rise. Specifically in the grocery sector, there is an increasing demand for convenient delivery services. Novel e-grocery companies, like Picnic, exclusively offer app-based delivery services. However, even more established brands like Albert Heijn and Jumbo offer online grocery delivery services. The e-grocery sector is quickly emerging as one of the world's leading commerce industries, reaching a market value of United States Dollar (USD) 21.84 billion (as of 2025) and projected to grow at a Compound Annual Growth Rate (CAGR) of 15.93% between 2026 and 2033.

Conventional warehouse operations depend heavily on manual labour for product handling, which limits scalability and creates a gap between increasing demand and available supply. This gap has created an opportunity for the warehouse automation sector. Many warehouses already use conveyor belts and Automated Guided Vehicle (AGV)s. However, these solutions present issues with picking item from taller shelves and transporting individual items from storage boxes directly to delivery crates. A drone-based warehouse ecosystem can alleviate these bottlenecks, enabling an adaptable system that operates autonomously 24/7.

This report outlines the design of Drone-based Responsive Intra-warehouse Vehicle Ecosystem for Fulfilment Centers (DRIVE-FC), a novel, autonomous, drone-based system for warehouse logistics. This system would allow for further expansion with less additional hiring being necessary in a supply-constrained labour market. The drones would fill delivery crates with requested items from an ordering system. The system is fully integrated in the warehouse, meaning that warehouse workers are left as undisturbed as possible to continue their regular operations.

The report starts with project objectives and market analysis. Then moves to project management and systems engineering. After which, the design concept and details are presented. Finishing with a critical analysis of project's feasibility. Since the project's time-frame is quite limited, the results will mostly be preliminary and will require more iterations to achieve a working real-world system.

2

Project Objectives

This chapter presents the objectives of the project. It frames the project within the context of the growing e-grocery sector and the operational pressures it places on fulfilment warehouses, before formalising the project's purpose through a Mission Need Statement and a Project Objective Statement.

In recent years, online grocery ordering has become widespread, with customers even being able to purchase just a few items for home delivery. To support this demand, extensive logistical networks have been developed, particularly in the form of centralized warehouses where order picking is performed. As the popularity of e-groceries is expected to grow [45], the operational demand on these warehouses will increase accordingly. This has led to significant research into warehouse automation, including within the e-grocery sector. For instance, companies such as Picnic are actively exploring automation solutions for their facilities.

Warehouse environments present several constraints that must be considered in the design of automated systems. These include requirements for low noise emissions, reliable navigation and route tracking, effective obstacle avoidance, and the ability to handle a wide variety of products. At the same time, human workers remain essential to warehouse operations, necessitating safe and efficient interaction between automated systems and personnel.

To support and relieve human workers, this project aims to develop an automated system that reduces workload and increases operational efficiency. Specifically, the objective is to design a swarm of Unmanned Aerial Vehicle (UAV)s capable of transporting individual items within the warehouse. These UAVs will move products from central storage locations to drop-off stations. The UAV will then deposit these items into a box in the combination given by the order and then the box can be sent to the customer.

This can all be summarized by the following Mission Need Statement and Project Objective Statement:

Mission Need Statement

Provide an autonomous efficient intra-warehouse grocery transport system

Project Objective Statement

Design an autonomous integrated drone-based intra-warehouse transport system for an e-grocery fulfilment centre

3

Market Analysis

At times it may be that a technically successful design does not see the same success commercially. Having a well established understanding of the market in which the design will compete in allows the design to be tailored for market opportunities that help ensure commercial success. This chapter presents the market analysis conducted prior to the development of DRIVE-FC for the autonomous fulfilment industry, online grocery industry, and the significance of these findings on the scope of this project.

3.1. Market Overview & Trends

With the global shift towards automation accelerating rapidly, the market of Automatic Fulfilment Systems (AFS) has seen a significant increase in market size, with analysts hypothesizing further increase over the next few years [44]. Though estimates vary per source due to differences in definition, the directional consensus is abundantly clear, with the current warehouse automation market valued at USD 21.84 billion in 2025, and expectations of reaching as high as USD 71.25 billion by 2033, translating to a CAGR of 15.93% over the forecast period of 2026-2033 [44]. This rapid expansion is primarily driven by technological advancements and a rising demand for efficient, scalable logistics and supply chain operations. Notably, the growing accessibility of Artificial Intelligence (AI) and Machine Learning (ML) have enabled more sophisticated data analysis, predictive maintenance, and optimized decision making processes, making this growth in market size a natural progression.

Simultaneously, the online grocery consumer market is also a growing market of its own, recording a market valuation of USD 67.6 billion in 2024 with an unprecedented forecasted CAGR of 36.8% for the 2025-2033 period[35]. According to Ground View Research, the primary catalysts for this growth are the increasing penetration of smartphones and internet connectivity across urban and sub-urban regions. Specifically, within the European market, growth is seen both in sub-urban regions as well as high-density urban environments, heavily influenced by demographic shifts. For countries such as Germany, with growing ageing populations and a rise in dual-income households, convenient home delivery services and bulk purchasing of cooking essentials have become an irreplaceable piece of daily infrastructure for many.[35]

Additional insights into product categories have shown that the 'staples & cooking essentials' segment has dominated the online grocery industry, accounting for 29.1% of the market size in 2024. However, as extrapolated from the analysis provided by [35], product assortments are diversifying to accommodate specialized niches such as those tailored to local cuisines, organic, gluten-free, and health-focused staples which are also growing in demand. With one of the appeal points for online grocery over traditional physical stores being the more easily accessible lower popularity products, it highlights the need for services to be able to provide a wide range of products in a single order.

3.2. Competitor Analysis

Currently, popularized systems vary from AGVs, conveyors, and Automated storage and retrieval systems (AS/RS). These systems offer significant improvements in speed, accuracy and efficiency of warehouses, making it essential for meeting the high demands of modern logistics. In terms of competition, the AFS market currently exists in a highly concentrated state, with the ten largest vendors holding roughly half of global revenue [55].These include the Ocado Group, Autostore, and Symbotic, many of which operate within

ground based, rail based, or cube-grid systems. Indeed, Ocado and Autostore both sell cubic-grid systems with vehicles operating over the grid for fully automated fulfilment or replenishment[36, 58]. In the more specific case of indoor drone fleet usage within warehouses, names such as Verity AG and Corvus Robotics are beginning to rise. As of today, both companies primarily focus on warehouse inspection using drone fleets, with little to no direct interactions with stored items [16, 52]. In regards to direct competition, as of this moment no company is operating commercially with a drone fleet system that is capable of autonomous item transportation, making DRIVE-FC a one-of-a-kind solution within this highly competitive landscape.

It is also worth noting that projected life spans for general fulfilment systems observed from recent history tends to be in the range of 15 years. For instance, the AGV, 'DU 1000', developed by Kiva in 2004 saw its debut for Amazon warehouses in 2005 [39, 46]. Though the system saw small scale upgrades along the way with models like the Hercules in 2012 [24], the first significant change in architecture was seen in 2018, 13 years after the original was released, with the Pegasus X-sort Drive[5]. Likewise, Autostore, a pioneer in cube-grid systems first deployed their product in 2005 at Elotec in Norway [57]. The system is still in operation today having seen its first major overhaul in 2019 with the introduction of black-line-robotics. Since then Autostore has continued to release upgrades for their products, the most significant one being the introduction of frozen storage capability in 2025 [57]. Based on these and other references, typical fulfilment systems see significant changes in 10-15 year intervals with major upgrades at around 6-8 year intervals.

Taking a conservative approach considering factors such as the preliminary nature of DRIVE-FC, and the rapid rate of technological developments in drones, an estimated total project lifespan of 10 years with major upgrades at the 5 year mark may be set as base values for further analysis.

3.3. SWOT Analysis

To contextualize DRIVE-FC's position within this landscape, a Strengths, Weaknesses, Opportunities and Threats (SWOT) analysis is presented in Table 3.1. Despite operating in an area without direct competitors, the system faces internal constraints and external pressures that shape it's development trajectory and commercial viability. Based on the SWOT analysis, concerns surrounding complexity and high maintenance frequency along with regulatory restrictions pose serious hurdles for DRIVE-FC to be realised as a system. However, with the added versatility and scalability in contrast to human-centred fulfilment ecosystems and with the tailwinds of technological advancements, DRIVE-FC is positioned as a highly promising project.

Table 3.1: SWOT Market Analysis

	Helpful	Harmful
Internal	• Transportation is more flexible compared to other methods. • Drones are more agile compared to other methods. • The infrastructure is adaptable to different environments. • The infrastructure is scalable. • Use of multiple drones makes the system resilient. • The drone can operate 24/7. • The drones reduce human contact with food, resulting in a more sanitary solution.	• System can only transport limited payload ($\sim$ 1kg) • Complex system • High initial setup cost • Regular maintenance required due to 24/7 operation
External	• Expanding e-groceries/rapid delivery Market • Advancements in drone technology • Advancements in battery technology	• Changes (or no changes) in regulations can stop drones in warehouses from becoming a reality. • Competition in warehouse technology can make it more difficult to sell the product to warehouse companies • Drones will likely reduce the need for workers, this could cause a backlash from worker unions.

3.4. Stakeholder analysis

Where the SWOT analysis illustrated the overall context surrounding DRIVE-FC, the stakeholder analysis builds on the contextual analysis, assessing the influence and interest levels of external stakeholders. This furthers the analysis by identifying key stakeholder groups whose needs directly translate to stakeholder requirements, driving the course and direction of the project overall. The constructed stakeholder map can be seen in Figure 3.1.

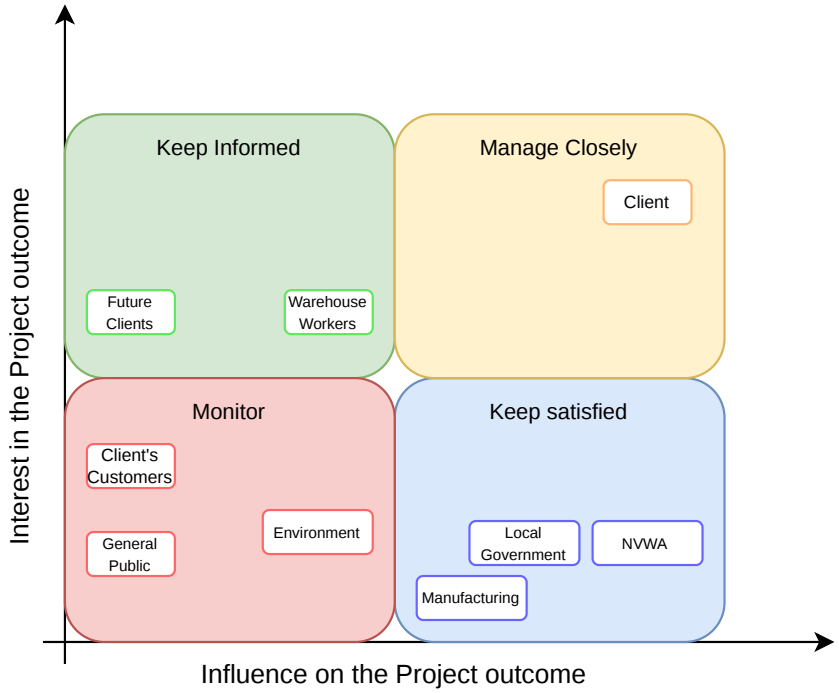


Figure 3.1: Stakeholder map showing influence and interest levels of external stakeholders

The stakeholder analysis identifies direct clients and their workforce as primary stakeholders whose needs are central to project viability. Secondary stakeholders, such as local governments and manufacturers, must be accommodated to ensure overall project feasibility. Table 3.2 synthesizes these stakeholder requirements and their implications for the project.

Table 3.2: Requirements derived from analysis

Requirement ID	Requirement	Relevant Stakeholder(s)	Justification
RQ-BAG	The project shall be within the budget	Client	The client is the party paying for the project. They expect the project to stay within budget
RQ-QLT	The system shall ensure quality of product and service	Client	The client needs groceries to not arrive damaged, as this will increase clean-up, and reduce their service quality.
RQ-WOS	The system shall ensure worker safety during operations	Warehouse Workers	For their work and safety they can not be harmed by the system.
RQ-GOV	The system shall obey local laws	Local Government, NVWA	The system must comply with laws, otherwise it may not operate
RQ-MFC	The system shall be manufacturable	Manufacturers	Without willing manufacturers the product will not be able to be made.

4

Project Management and Systems Engineering

To structure the project development, project management and systems engineering techniques are used. This chapter establishes a comprehensive Verification and Validation (V&V) plan in Section 4.1 to test every requirement and verify the technical integrity of all developed models and tools. To mitigate uncertainty, in Section 4.2 a risk management framework was used to prepare the team for potential project disruptions. Finally, Section 4.3 presents the functional breakdown structure and functional flow diagram for the project.

4.1. Verification and Validation Plan

In order to ensure the success and reliability of the final design product, an extensive V&V procedure must be carried out throughout the project. This process encompasses the verification and validation of requirements, models, software tools, and the final system design. All requirements were created using the Verifiable, Achievable, Logical, Integral, and Definitive (VALID) method.

Within this section, the planned V&V approach is presented, including the methods that will be used to establish confidence in the developed models, analyses, and resulting design decisions, as well as to demonstrate that the final system fulfils its intended operational purpose and stakeholder expectations. This is based on the V model for project management and systems engineering [49]. This model ensures that the requirements are met and that they fulfil the stakeholders needs and wants.

4.1.1. Requirement Validation

As stakeholders and users are essential to the success of the project, it is important to ensure that their needs are correctly represented within the system definition. Stakeholder needs from Chapter 3 and user needs were translated into top-level requirements. The complete set of stakeholder requirements was reviewed to assess whether it sufficiently captures the intended functionality, operational context, and stakeholder expectations of the system.

4.1.2. Product Verification

To prove that the system works as intended and according to the requirements set, multiple verification methods will be used; these are (**I**) inspection, (**A**) analysis, (**D**) demonstration, and (**T**) testing. Demonstration is considered a less formal test, where no additional resources are needed while for tests there is generally a more formal test setup. The verification method For every system and subsystem requirement in Table 4.1 is given in the method (**M**) column. An asterisk (*) is added for requirements that cannot be verified at this stage of the project, due to a lack of prototype hardware, accurate models, or detail in the design. The method is then further explained for that specific requirement. The requirements are structured in a hierarchical tree format, such that the relationship between stakeholder, system, and lower-level subsystem requirements is clearly visible.

Table 4.1: Verification plan for requirements.

Requirement ID	Requirement	M	Method description
RQ-BDT	**The project shall be within the budget.**		
RQ-BDT-01	The total Capital Expenditure (CAPEX) shall not exceed €1,200,000	A	Analyse the price of all of the components and manufacturing, and ensure these stay within the budget.
RQ-BDT-02	The unit production cost per drone shall not exceed €10,000.	A	Analyse the price of all of the components and manufacturing steps of each drone to ensure they do not exceed the budget.
RQ-ENV	**The system shall be environmentally friendly.**		
RQ-ENV-01	At least 70% of the drone mass shall either be reusable, recyclable or made from recycled materials.	A	Analyse all subsystems and add recyclable parts to obtain overall mass percentage.
RQ-ENV-01-COM-01	The communication subsystem shall use reusable hardware.	I	Check properties of communication hardware.
RQ-ENV-01-SNS-01	The sensor subsystem shall use reusable hardware.	I	Check properties of sensor hardware.
RQ-ENV-01-PRP-01	The propulsion subsystem shall use recyclable or recycled materials.	I	Check properties and origin of propulsion material.
RQ-ENV-01-GRB-01	The grabber subsystem shall use reusable hardware.	I	Check properties of grabber hardware.
RQ-ENV-01-PWR-01	The power subsystem shall use recyclable hardware.	A	Determine recyclability through literature.
RQ-ENV-02	The system shall be fully electrically powered.	I	Check that each component is electrically powered.
RQ-ENV-02-PWR-01	The power subsystem shall be fully electric.	I	Check that the power subsystem only delivers electrical energy.
RQ-QLT	**The system shall ensure quality of the product and service**		
RQ-QLT-01	The transport system shall leave no visible damage to the transported payload.	D	Show that no visible damage is left on vulnerable products due to the operation of the transport system.
RQ-QLT-01-GRB-01	The grabber shall exert no more than 30 N of force on any carried item.	T*	Test exertion force with compression load cells.
RQ-QLT-01-RTG-01	The routing subsystem shall ensure drop-off height of no more than 30 cm.	D	Through simulation it can be shown that the drone does not exceed the maximum drop-off height during drop-off.
RQ-WOS	**The system shall ensure worker well-being.**		
RQ-WOS-01	The system shall ensure worker safety during operations.	A	Analyse whether combined subsystem requirements sufficiently ensure workers safety during operations.
RQ-WOS-01-PRP-01	The propulsion subsystem shall not produce noise levels higher than 75 dBA at one meter distance from the drone.	T*	Measure sound level at one meter distance from drone.

Continued on next page

Table 4.1 – Continued from previous page

Requirement ID	Requirement	M	Method description
RQ-WOS-01-SNS-01	The sensor subsystem shall locate human workers with 1 m accuracy.	T*	Measure accuracy of human worker detection.
RQ-WOS-01-RTG-01	The routing subsystem shall ensure a minimum 30 cm separation between drones and human workers.	T	Test that drone keeps distance from human workers during motion.
RQ-GOV	**The system shall obey local laws.**		
RQ-GOV-01	The system shall comply with Dutch regulations.	A	Compare system to relevant regulations.
RQ-GOV-01-COM-01	The communication subsystem shall comply with Dutch communication regulations.	I*	Check communication system certification.
RQ-GOV-01-SNS-01	The sensor subsystem shall comply with Dutch privacy regulations.	A	Compare subsystem to relevant regulations.
RQ-GWH	**The system shall be operational in an e-grocery warehouse.**		
RQ-GWH-01	Drone operations shall be feasible in Cold-storage zones (2 ± 2 °C).	T	Test drone operation in cold storage zone.
RQ-GWH-01-COM-01	The communication subsystem's hardware shall be functional in Cold-storage conditions.	I	Check hardware operation temperature range.
RQ-GWH-01-SNS-01	The sensor subsystem's hardware shall be functional in Cold-storage conditions.	I	Check hardware operation temperature range.
RQ-GWH-01-PWR-01	The power subsystem's hardware shall be functional in Cold-storage conditions.	T*	Test power subsystem functioning in cold storage zone.
RQ-GWH-01-PRP-01	The propulsion subsystem shall be functional in Cold-storage conditions.	T*	Test propulsion subsystem functioning in cold storage zone.
RQ-GWH-01-GRB-01	The grabber subsystem shall be functional in Cold-storage conditions.	T*	Test grabber subsystem functioning in cold storage zone.
RQ-GWH-02	Drone operations shall be feasible in Ambient zones (18 ± 2 °C).	D	Demonstrate drone operation in ambient storage zone.
RQ-GWH-02-COM-01	The communication subsystem's hardware shall be functional in Ambient conditions.	I	Check hardware operation temperature range.
RQ-GWH-02-SNS-01	The sensor subsystem's hardware shall be functional in Ambient conditions.	I	Check hardware operation temperature range.
RQ-GWH-02-PWR-01	The power subsystem's hardware shall be functional in Ambient conditions.	D	Demonstrate power subsystem functioning in ambient storage zone.
RQ-GWH-02-PRP-01	The propulsion subsystem shall be functional in Ambient conditions.	D*	Demonstrate propulsion subsystem functioning in ambient storage zone.
RQ-GWH-02-GRB-01	The grabber subsystem shall be functional in Ambient conditions.	D*	Demonstrate grabber subsystem functioning in ambient storage zone.
RQ-GWH-03	The system shall be capable of transporting payloads up to 1.0 kg.	D	Demonstrate the drone can fly with 1 kg of payload.
RQ-GWH-03-PRP-01	The propulsion subsystem shall be able to provide a thrust-to-weight ratio of at least 2.0.	A	Check the thrust-to-weight ratio by analysing mass and thrust limits.
RQ-GWH-03-GRB-01	Each tentacle of the grabber shall be able to withstand a maximal tensile load of 20 N.	T*	Test maximal tensile load by mean of a tensile-stress test.

Continued on next page

Table 4.1 – Continued from previous page

Requirement ID	Requirement	M	Method description
RQ-GWH-03-CTR-01	The control subsystem shall be able to accurately follow path within 0.15 m lateral deviation when carrying payload.	A	Analyse through means of a simulation.
RQ-GWH-03-CTR-02	The control subsystem shall be able to accurately follow path within 0.15 m vertical deviation when carrying payload.	A	Analyse through means of a simulation.
RQ-GWH-04	The system shall be able to lift all payloads out of the stock pallets	T*	Test the lifting capabilities in test-case scenarios to ensure compliance.
RQ-GWH-04-GRB-01	THe grabber shall be able to lift objects of size 400x250x50mm	D*	Demonstrate grabber lifting this size of box.
RQ-GWH-04-GRB-02	The grabber shall be able to access all points of the crate at the pickup location	D*	Demonstrate picking up when payload positioned at random positions in crate.
RQ-RLB	**The system shall be reliable.**		
RQ-RLB-01	The system shall be operational for at least 10 continuous years.	A	Analyse degradation of drone components, and compare with similar applications.
RQ-RLB-01-BDY-01	The drone body subsystem shall be capable of operating for at least 5 continuous years.	A*	Analyse degradation of drone body subsystem, and compare with similar applications.
RQ-RLB-01-GRB-01	The grabber subsystem shall be capable of operating for at least 10 continuous years.	A*	Analyse degradation of grabber subsystem, and compare with similar applications.
RQ-RLB-01-PRP-01	The propulsion subsystem shall be capable of operating for at least one continuous year.	A	Analyse degradation of propulsion subsystem, and compare with similar applications.
RQ-RLB-02	The system shall have mid-air collision avoidance protocols.	I	Check presence of mid-air collision avoidance protocols.
RQ-RLB-02-SNS-01	The sensor subsystem shall detect obstacles with 10 cm positional accuracy at 8 m distance.	D	Demonstrate that the sensor subsystem can detect obstacles with sufficient accuracy at varying distances.
RQ-RLB-02-RTG-01	The routing subsystem shall plan collision avoidance paths maintaining a 30 cm distance between known obstacles and drone.	A	Simulate collision situation to analyse avoidance behaviour.
RQ-RLB-02-CTR-01	The control subsystem shall safely handle unknown obstacle avoidance.	A	Simulate collision situation to analyse avoidance behaviour.
RQ-RLB-03	The system shall have protocols for single drone failures.	I	Check presence of single drone failure protocols.
RQ-RLB-03-CTR-01	The control subsystem shall maintain stability in case of single motor failure.	A	Analyse flight dynamics of single motor failure.
RQ-RLB-03-RTG-01	The routing subsystem shall have emergency landing protocol in-case of single motor failure.	I	Check presence of emergency landing protocol in case of single motor failure.
RQ-RLB-03-GRB-01	The grabber subsystem shall not impede emergency landings.	D*	Demonstrate an emergency landing.
RQ-RLB-04	The system shall have protocols for mid-air collisions.	I	Check presence of mid-air collision protocols.

Continued on next page

Table 4.1 – Continued from previous page

Requirement ID	Requirement	M	Method description
RQ-RLB-04-CTR-01	The control subsystem shall enact forced shutdown upon collision detection.	D	Demonstrate that control subsystem shutsdown upon collision.
RQ-RLB-04-COM-01	The communication subsystem shall communicate collision to the routing subsystem.	I*	Check that communication subsystem communicates collision.
RQ-RLB-04-SNS-01	The sensor subsystem shall be capable of detecting collisions.	I	Check manufacturer specification to determine whether selected sensors are capable of detecting collisions.
RQ-EFF	**The system shall be efficient.**		
RQ-EFF-01	The system shall achieve a daily volume of at least 10,000 successful item transfers per 24 hours.	A	Analyse through usage of simulations whether the drone system is able to achieve a throughput of 10,000 items.
RQ-EFF-01-PWR-01	The power subsystem shall provide sufficient energy to ensure continuous operation for at least 15 minutes per drone charge.	A	Verify through calculation of both power draw and battery capacity.
RQ-EFF-01-ERS-01	The energy replenishment stations shall provide enough charging and swapping capacity to continuously fly up to 10 drones in either temperature zone.	D	Fly the required number of drones simultaneously and verify their batteries never deplete due to ERS unavailability.
RQ-EFF-01-RTG-01	The routing subsystem shall ensure a maximal average time per item of 2 minutes.	A	Analyse through usage of simulations whether the drone system is able to achieve an average time per item of 2 minutes.
RQ-ATM	**The system shall be autonomous.**		
RQ-ATM-01	The system shall plan drone actions.	I	Check that the procedures are in place.
RQ-ATM-01-RTG-01	The routing subsystem shall ensure a minimum of 30 cm separation between drones and static infrastructure.	D	Simulate whether drone keeps distance from static infrastructure.
RQ-ATM-01-RTG-02	The routing subsystem shall schedule Energy Replenishment Station (ERS) routing when drone energy is below 10%.	I	Check that drone automatically goes to ERS when energy is below 10%.
RQ-ATM-01-RTG-03	The routing subsystem shall schedule maintenance path.	D	Show that routing subsystem schedules maintenance.
RQ-ATM-02	The system shall communicate system status.	D	Show that system status is communicated.
RQ-ATM-02-COM-01	The communication subsystem shall have a minimal bandwidth of 1 Mbps between all communicating modules.	I	Check the bandwidth between all communicating modules.
RQ-ATM-02-COM-02	The communication subsystem shall guarantee a minimal inter-drone communication frequency of 1 Hz.	I*	Check the minimal frequency that is used between drones.
RQ-ATM-02-COM-03	The communication subsystem shall communicate with a expected data loss rate of no more than 10^{-6} at 10 m range.	T*	Measure data loss rate between two communicating bodies by the ratio of bits between sending and receiving.

Continued on next page

Table 4.1 – Continued from previous page

Requirement ID	Requirement	M	Method description
RQ-ATM-02-SNS-01	The sensor subsystem shall measure drone system status at a minimum frequency of 100 Hz.	I	Inspect the frequency of the sensor readout.
RQ-ATM-03	The system shall be able to replenish drone energy.	D	Replenish drone energy by demonstration.
RQ-ATM-03-ERS-01	Energy replenishment procedures shall take no longer than 2 minutes.	D*	Demonstrate that the energy replenishment procedures do not take longer than required.
RQ-ATM-03-ERS-02	The energy replacement shall correctly perform at least 99% of battery swaps correctly.	T*	Repeatedly test battery swaps and verify they are done correctly.
RQ-ATM-03-ERS-03	The energy replenishment stations shall function without human intervention outside of unexpected circumstances.	D*	Demonstrate that the ERS can perform battery swaps with no human intervention.
RQ-ATM-04	All drones shall be able to follow planned actions	D	Drone will demonstrate flying a planned action.
RQ-ATM-04-SNS-01	During item pickup, the sensor subsystem shall measure drone position with an accuracy of 1 cm in each direction at 1 m from the item.	T*	Test that the drone measures the position within an accuracy of 1 cm.
RQ-ATM-04-SNS-02	During item drop-off, the sensor subsystem shall measure drone position with an accuracy of 1 cm in each direction at 1 m from the drop-off location.	T*	Test that the drone measures the position within an accuracy of 1 cm.
RQ-ATM-04-SNS-03	The sensor subsystem shall measure drone position with an accuracy of 1 cm in each direction at 1 m from the ERS.	T*	Test that the drone measures the position within an accuracy of 1 cm.
RQ-ATM-04-SNS-04	The sensor subsystem shall measure drone position with a deviation of maximum 20 cm at any given point in time.	T	Test sensor accuracy used for position determination.
RQ-ATM-04-CTR-01	The drone shall follow given path to destination within 0.15 m margin in altitude.	A	Simulate and show that the drone flies at the predetermined height.
RQ-ATM-04-CTR-02	The drone shall follow given path to destination within 0.15 m margin laterally.	A	Simulate and show that the drone flies at the predetermined position.
RQ-MFC	**The system shall be manufacturable.**		
RQ-MFC-01	All Off-The-Shelf components shall be readily available.	I	Check with manufactures and distributors whether components are available.
RQ-MFC-01-SNS-01	All off-the-shelf sensor subsystem components shall have a maximum verified supplier lead time of 4 weeks.	I*	Check with manufactures and distributors whether sensor subsystems are available within time frame.
RQ-MFC-01-COM-01	All off-the-shelf communication subsystem components shall have a maximum verified supplier lead time of 4 weeks.	I*	Check with manufactures and distributors whether communication subsystem components are available within time frame.
RQ-MFC-01-PRP-01	All off-the-shelf propulsion subsystem components shall have a maximum verified supplier lead time of 4 weeks.	I	Check with manufactures and distributors whether propulsion subsystems are available within time frame.

Continued on next page

Table 4.1 – Continued from previous page

Requirement ID	Requirement	M	Method description
RQ-MFC-01-POW-01	All off-the-shelf power source subsystem components shall have a maximum verified supplier lead time of 4 weeks.	I*	Check with manufactures and distributors whether power subsystems are available within time frame.
RQ-MFC-02	The system shall be designed such that a single trained technician can fully assemble the drone in under 10 hours.	D	Show that it is possible for a technician to assemble the drone in time frame.
RQ-MFC-02-BDY-01	The drone body subsystem shall be designed such that a single trained technician can fully assemble the drone in under 3 hours.	D*	Trained technician shows that drone body can be fully assembled within time frame.
RQ-MFC-02-SNS-01	The sensor subsystem shall be designed such that a single trained technician can fully integrate onto the drone in under 2 hours.	D*	Trained technician shows that sensor subsystem can be fully integrated within time frame.
RQ-MFC-02-GRB-01	The grabber subsystem shall be designed such that a single trained technician can fully integrate onto the drone in under 1 hour.	D*	Trained technician shows that the grabber subsystem can be fully assembled within time frame.
RQ-DCM	**The system shall be decommissionable.**		
RQ-DCM-01	The system shall be designed such that it can be disassembled by a single technician in under 2 hours, using standard hand tools.	D	Trained technician shows that drone can be fully disassembled within time frame.
RQ-DCM-01-BDY-01	Drone body shall be designed in such a way that it can be disassembled by a single technician in under 45 minutes, using standard hand tools.	D*	Trained technician shows that drone body can be disassembled with standard hand tools within the time frame.
RQ-DCM-01-SNS-01	Sensor subsystem shall be integrated into design in such way that it can be disassembled by a single technician in under 30 minutes, using standard hand tools.	D*	Trained technician shows that drone sensors can be disassembled with standard hand tools within the time frame.
RQ-DCM-01-GRB-01	Grabber subsystem shall be integrated into design in such way that it can be disassembled by a single technician in under 45 minutes, using standard hand tools.	D*	Trained technician shows that drone grabber subsystem can be disassembled with standard hand tools within the time frame.
RQ-MNT	**The system shall be maintainable during operations.**		
RQ-MNT-01	The drone shall be designed such that the Mean Time To Repair Mean Time To Repair (MTTR) is less than 3 hours by a single technician.	A	The most common problems and their repair time are analysed to ensure that the MTTR is acceptable.

As can be seen from the verification table, a large portion of the requirements can be verified through analysis and inspection. These verification methods can generally be executed without requiring significant additional resources. Demonstrations can largely be performed using the facilities available at the Cyber Zoo of TU Delft [1], and are therefore are not expected to introduce substantial costs.

However, requirements verified through physical testing may require additional equipment, prototype development, or measurement systems, resulting in associated costs. As V&V activities are essential to ensuring the reliability and credibility of the final design, these activities cannot be omitted from the project scope.

[1] https://www.tudelft.nl/lr/organisatie/afdelingen/control-and-operations/facilities/drone-facilities/cyberzoo

Therefore, an estimate of the expected costs associated with executing the V&V plan is made. The following list provides an overview of the anticipated costs related to the activities.

- **RQ-QLT-01-GRB-01**: small compression load cells are sufficient, estimated at €120, using four cells of approximately €30 by RS Netherlands.
- **RQ-WOS-01-PRP-01**: a midrange decibel-meter costs around €80.
- **RQ-WOS-01-SNS-01**: repeated testing must be done in a test facility to investigate the detection of human workers. Due to safety considerations, a human dummy will be used resulting in an estimated cost of €60.
- **RQ-WOS-01-RTG-01**: the same dummy can be used as for RQ-WOS-01-SNS-01, no additional cost is necessary.
- **RQ-GWH-01**: to test the drone and it's subsystems, a cooled container needs to be rented. As many tests must be concluded, the testing time-line is estimated to last a month, resulting in a renting cost of €450.
- **RQ-GWH-03-GRB-01**: a tensile-stress test is to be conducted, resulting in a cost of €650 by using the manual test bench of Vetek.
- **RQ-ATM-02-COM-03**: data loss can be calculated from the difference in bits between sent signal and received signal. No additional costs are needed.
- **RQ-ATM-03-ERS-02**: Repeated testing of ERS (dis)engagement is needed to gain insight in the certainty. As the ERS will be developed regardless, the costs are not allocated to the V&V plan.
- **RQ-ATM-04-SNS-01, RQ-ATM-04-SNS-02, RQ-ATM-04-SNS-03**: Repeated testing is necessary to ensure that sensors accurately determine position. For precise drone position determination, a visual tracking system can be used, mainly two Prime^{X}13 which cost €3,000 in total. Two are needed to ensure both altitude and flat plane position are measured.

The estimated costs associated with the above mentioned requirements to €4,270. In addition, many of the planned tests and demonstrations require the availability of physical drone prototypes. The cost of a single prototype drone is estimated at €10,000, corresponding to the maximum allowable unit production cost defined for the final system.

It is expected that a total of four prototype drones will be required throughout the V&V process. This includes prototypes intended for collision testing, compensation for unforeseen component failures or crash damage during testing, and the availability of spare drones to ensure final product validation can be conducted. This results in a cost estimation of €44,270 to execute the requirement verification tests.

4.1.3. Model and Tool Verification and Validation

To ensure that the final product is of high quality, the models and tools used in the development must also be of high quality. Therefore, each model and tool that is developed and used must be critically verified and validated, to ensure they work as intended and do not negatively impact the final product. This process is described in this section.

Model Verification and Validation

For this project, several models are used throughout the design and development process. For the routing system, a simulation environment is being created in which the warehouse layout is implemented and drones autonomously pick up payloads and deliver them to designated drop-off locations. This simulation consists of a pathfinding algorithm and a warehouse layout model. The pathfinding algorithm is based on an open-source, widely used, A* Python package, adapted to the specific requirements of the project. Another considered algorithm is the Scalable and Safe Multi-Agent Motion Planning with Non-linear Dynamics and Bounded Disturbances (S2M2) based on the 2021 paper from Chen et al. [12], which still has to be validated through means of simulation.

For complete model verification of the routing system, a hierarchical approach will be used: code units are tested first, followed by subsystems, and finally the entire model. As subsystems are connected, integration testing is performed. Once everything is connected, extreme value tests are performed. This method ensures that the fundamental components of the design are fully verified before transitioning to more complex systems, making it easier to isolate and trace issues. Validation of the routing model is performed through analysis by

evaluating whether the generated drone trajectories, routing logic, and conflict-resolution behaviour satisfy the operational requirements and constraints of the warehouse environment.

Additionally, a software simulation model describing the physical behaviour and flight dynamics of the drones is used. Again, unit testing as well as system testing will be executed to verify the correctness of the model. This model can also be validated through comparison with data and behaviour observed in existing drone systems and experimental data, however, this falls outside of the time constraints of this design project. Furthermore, graphical Computer-Aided Design (CAD) models are developed in CATIA, including detailed subsystem models such as the grabber and a complete drone assembly model. These graphical models are validated through analysis by assessing geometric consistency and subsystem integration, using CATIA's built in conflict and clash tools.

Tool Verification and Validation
Several software tools were developed to support design-space exploration and subsystem sizing. As these tools directly influence design decisions, dedicated V&V activities are required to ensure the reliability and correctness of their outputs.

One such tool is the iterative drone sizing model, which is used to estimate subsystem masses, voltage demand, and overall vehicle sizing during conceptual and preliminary design stages. Verification of this tool is performed through inspection of the implemented equations, comparison with hand calculations and literature methods, and consistency checks on intermediate and final outputs. It is currently not possible to say what the precision or accuracy is or needs to be, however, the tool is based on literature and component contenders. The tool is mainly used as a starting point for an iterative design process. Furthermore, assumptions made during early-stage sizing are continuously reassessed and validated as the design progresses into greater detail.

In addition, sensitivity analysis tools were developed to evaluate the influence of varying design parameters on system performance. These tools are verified through analysis of output trends and consistency checks to determine whether parameter variations result in logically expected behaviour. Unit testing is additionally performed on individual functions and computational modules to reduce the likelihood of implementation errors. This was done by printing outputs of the tool during the development and comparing to manual calculations.

Lastly, an optimisation tool was developed to determine the optimal number of aisles, shelves, and shelf depth. This tool was developed in-house and preliminarily validated by analysing the generated results. In all evaluated cases, the resulting configurations complied with realistic physical dimensions and practical warehouse layouts. The tool has not yet been formally verified and is therefore still subject to verification through unit testing.

Together, these V&V activities aim to establish confidence in the developed design tools and ensure that the resulting design decisions are based on reliable and sufficiently accurate analyses. At present, there are no plans to develop additional tools in the future. However, it is likely that the need for new tools will arise during the project. In such cases, the objective will be to establish the same level of confidence in the validity of these tools by performing V&V in accordance with the procedures described in this plan.

4.1.4. Product Validation
At the end of the design stage, the product can be validated using mission scenario tests. In these tests, realistic operational scenarios are simulated to demonstrate that the system performs according to the intended requirements and operational expectations. In addition, stress testing will be performed within the simulation environment to evaluate the robustness of the system under abnormal or high-demand conditions. These stress scenarios subject the system to unexpected operating conditions and fault situations in order to assess its resilience and fault tolerance. For example, a situation may arise in which a large portion of the drone fleet is simultaneously tasked with retrieving items from the same location. This creates increased traffic density and potential routing conflicts, challenging the routing and collision-avoidance algorithms to safely and efficiently resolve possible drone interactions.

Ultimately, operational readiness tests will be conducted to demonstrate that all system segments are capable of successfully performing the intended mission and that the system is ready for implementation. For these tests, a full-scale prototype drone will be constructed and evaluated within a mock warehouse environment. Although the realization of this prototype falls outside the scope and timeline of the current project phase, its development and testing are expected to require significant resources. The mock warehouse testing environment could potentially be done within the Cyber Zoo facilities at TU Delft. As mentioned in Section 4.1.2, the construction cost of the prototype drone is estimated at approximately €10,000, as this corresponds to the

maximum allowable unit production cost defined for the final operational system. This cost is included in the cost estimate for the requirement verification costs.

4.2. Technical Risk Analysis

To ensure proper functioning of the system it is key to understand the possible risks which could limit the effectiveness of the system. Although risks can never be completely eliminated, by assessing the possible risks and their driving causes, it is possible to reduce the risks and form contingency plans to limit the probability and impact of known risks.

In Table 4.4 the foreseen risks are listed and considered. Each risk is assigned a unique ID for reference throughout the project, the naming follows the following rules: **TR** describes the fact that it is a technical risk. This is followed by a dash and an abbreviation which relates the risk to a function, as indicated in the function column. Furthermore, each risk is described and the main drivers are given. Each risk is then evaluated for likelihood (L), the impact (I), and the risk (R) which is the product of the likelihood and the impact. Both the likelihood and impact are scored on a scale of 1 to 5, the scales are explained in Table 4.2 and Table 4.3 respectively. A risk score of 25 is therefore the maximum possible risk. A justification for the scoring is given in the justification column with p indicating the justification for the likelihood scoring and I for the justification of the impact. Where applicable, a mitigation and contingency strategy is given to decrease the likelihood and/or the impact. These new likelihoods, impacts and risks (after the mitigation and contingency plan) are given in the columns **L***, **I***, and **R*** respectively.

The risks are then assessed using the risk assessment diagram shown in Figure 4.1. Here the risks are classed in 4 different categories, risks that fall under the green area are situations with low risk and impact, and can be considered acceptable risks. The yellow range have a higher risk and impact, meaning that these risks have to be considered and mitigated in the design process but are not deemed critical. The orange region are risks with a higher impact and likelihood than the yellow region. This means that these risks have priority to mitigate compared to the risks in the yellow region. Lastly, the risks in the red region are high likelihood and high impact risks, meaning that it is imperative that these risks are mitigated as they are critical to the mission's success. The regions the risks are located in are also colour-coded accordingly in Table 4.4.

Table 4.2: Probability scoring

Score	Probability Range
1	$0 \leq p < 0.0625$
2	$0.0625 \leq p < 0.125$
3	$0.125 \leq p < 0.25$
4	$0.25 \leq p < 0.5$
5	$0.5 \leq p \leq 1$

Table 4.3: Impact scoring

Score	Impact Description
1	**Negligible:** No functional impact. Resolved during routine maintenance.
2	**Minor:** Mission stays on track but requires minimal extra resources.
3	**Considerable:** Significant repair required. Causes noticeable delay.
4	**Critical:** Major failure. Significant cost overruns or missed milestones.
5	**Catastrophic:** Total system loss or failure.

Table 4.4: Technical Risk Analysis

ID	Function	Description	Driver	L	I	R	Justification	Mitigation Strategy	Contingency Strategy	L*	I*	R*
TR-MAN-01	Manufacture system	Inability to find suitable supplier	Prices prohibitively high. Difficult to produce components.	1	4	4	P: Drones are a mature market, a large number of parts are available. I: A critical bottleneck in getting product to market. Serious redesign necessary.	Design with manufacturability in mind, this will make the availability of resources better (more common materials/parts are easier to obtain).	Redesign the product.	1	4	4
TR-MAN-02	Manufacture system	Preferred materials unavailable	Geopolitical tensions, Export controls, raw material shortages, price prohibitively high	2	4	8	P: Drones are a mature market, a large number of materials are available. Current geopolitical climate is uncertain I: A critical bottleneck in getting product to market. Serious redesign necessary.	Design with manufacturability in mind, this will make the availability of materials better (more common materials/parts are easier to obtain).	Redesign the product.	1	4	4
TR-MAN-03	Manufacture system	Unable to find suitable manufacturer	Manufacturers ask for high price due to large competition in drone market. Manufacturers unwilling/prohibited to manufacture the design	1	4	4	P: Drones are a mature market, a large number of manufacturers are available I: A critical bottleneck in getting product to market. Serious redesign necessary.	Design with manufacturability in mind, this will make the availability of manufacturers better (more common manufacturing allows for a larger pool of suitable manufacturers).	Redesign the product.	1	4	4
TR-MAN-04	Manufacture system	Manufacturing system breaks down	Machine in factory breaks. Workers on strike	2	2	4	P: Most manufacturing companies operate at 60% operating efficiency, where some efficiency is lost due to unplanned downtime (i.e. machine breaks). Aerospace industry standards are much higher. I: Delay in production	Implement predictive maintenance on manufacturing machines. Design a flexible supply chain. Set margins in the production time	-	1	2	2
TR-MAN-05	Manufacture system	Quality does not meet standards	Machine makes error. Factory does not control quality enough.	2	4	8	P: Modern machine manufacturing is mostly reliable and precise. I: This can lead to the system not working as intended.	Build in redundancies. Perform due diligence on manufacturers	Find new manufacturer, more suitable to needs.	1	3	3
TR-MAN-06	Manufacture system	Inspection discovers fault in manufactured batch	Subpar quality from manufacturer. Unconventional designs (for props).	1	2	2	P: Modern machine manufacturing is mostly reliable and precies. I: This can lead to the need to reproduce a batch, adding cost and time.	-	Reproduce the batch	1	2	2
TR-AIT-01	Assemble, Integrate & Test system	System fails functional tests	Design does not translate well into physical world	3	5	15	P: Without proper verification and validation this is likely to happen. I: This will pull the entire mission into question as complete redesign is necessary.	Perform simulations to ensure system functional compliance	Redesign the product.	2	5	10
TR-AIT-02	Assemble, Integrate & Test system	System fails software tests	Bugs	3	4	12	P: Without proper verfication and validation this is likely to happen. I: The entire software may need to be redevelopd	Perform thorough verfication during design	Perform bug fixes.	2	4	8

Continued on next page

Table 4.4 – continued from previous page

ID	Function	Description	Driver	P	I	R	Justification	Mitigation Strategy	Contingency Strategy	P*	I*	R*
TR-AIT-03	Assemble, Integrate & Test system	System fails environmental tests	Design does not translate well into physical world	2	4	8	P: Without proper verfication and validation this may happen. I: The entire drone hardware may need to be redesigned.	Perform thorough testing on prototypes	Redesign the product.	1	4	4
TR-AIT-04	Assemble, Integrate & Test system	System fails acceptance test	Design does not translate well into physical world	2	5	10	P: Without proper verfication and validation this may happen. I: The entire drone and system needs to be redeveloped.	Validate the system	Redesign the product.	1	5	5
TR-OPS-01	Operate	Drone fails measure status properly	Sensor issue. Sensor readout issue	1	1	1	P: This would require a sensor to fail, this is reletively unlikely. A single drone sensor failing is fairly inconsequential	Perform regular maintenance	Remove drone from fleet and perform mainenance.	1	1	1
TR-OPS-02	Operate	Communication with drone fails	Wireless connection fails. Antenna in drone fails	3	2	6	P: Wireless connections fail commonly. I: If this is for a limited time this can cause problems but is mostly inconsequential	Perform regular maintenance, add redundancies	Remove drone from fleet and perform mainenance.	2	2	4
TR-OPS-03	Operate	Drone unable to find suitable path	Short term environment changes. Obstacles in path. Drone has callibration issue. Busy airspace	1	4	4	P: Drone has 3 dimensions it can manuever and in general wait for obstacle to be removed. I: If no suitable path can be found it may cause a collision into infrastructer or a worker.	-	Find root cause, and act accordingly.	1	3	3
TR-OPS-04	Operate	Drone collides with other drone/environment	Short term environment changes. Obstacles in path. Drone has callibration issue. Busy airspace	2	3	6	P: With drones constantly moving a system miscalculation may result in a collision. A collision between drones and environment does not pose an inmiddiate danger to workers or the mission.	Add system which allows drone to avoid collisions based on sensors.	Take drones in for repair.	1	3	3
TR-OPS-05	Operate	Drone fails to pick up item/drops item	Grabber failure. Sudden movement. Turbulence	3	3	9	P: Without proper edge case considerations the drone grabber may not work as expected. I: This defeats the point of using drones.	Design with edge cases in mind. Perform maintenance.	Send human to resolve issue.	2	2	4
TR-OPS-06	Operate	Drone picks up/drops item off at wrong location	Calibration issue. Object recognition issue	1	2	2	P: Would require an issue in the calibration I: This requires a human to manual correct the mistake.		Send human to resolve issue.	1	1	1
TR-OPS-07	Operate	Drone fails to take off	Propulsion system fails. Charge failure. Obstacle in the way.	1	3	3	P: Multiple failures need to take place for this to happen. I: A drone less in the fleet will impact the performance. I human needs to collect the drone.	Perform maintenance	Remove drone from fleet and perform maintenance.	1	3	3

Continued on next page

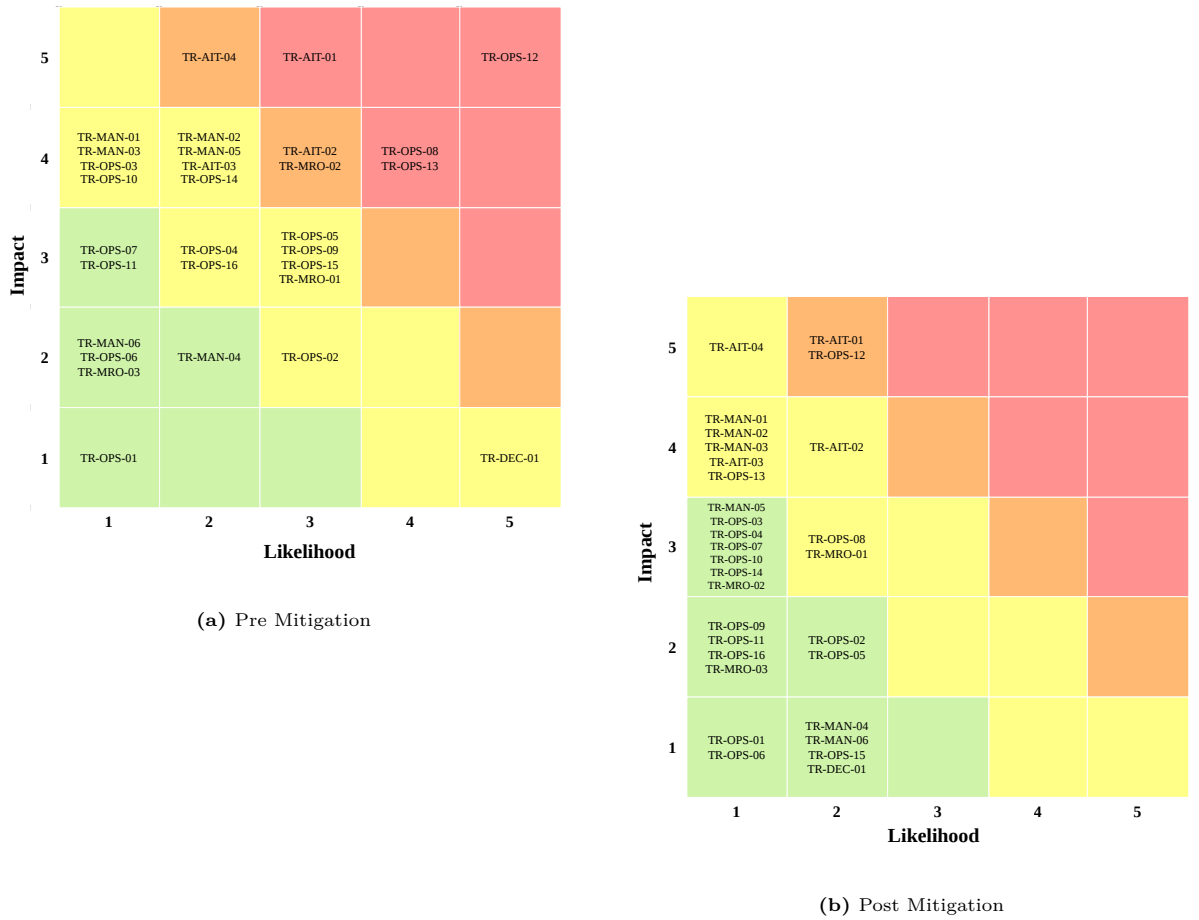
Table 4.4 – continued from previous page

ID	Function	Description	Driver	P	I	R	Justification	Mitigation Strategy	Contingency Strategy	P*	I*	R*
TR-OPS-08	Operate	Drone responds unexpectedly to control input.	Picked up item causes cg shift (ex. sloshing), Calibration issue. Drone wake. Grabber drops item. Disturbances in warehouse.	4	4	16	P: With large simplifications made in the drone control this is likely to happen. I: This will make it difficult for drones to perform as intended.	Design with multiple use cases in mind	Implement "Drone out-of-control" plan (drone kill switch, remove workers from drone vacinity)	2	3	6
TR-OPS-09	Operate	Drone does not land at ERS	Calibration issue. ERS not sensed Drone battery dies prematurely.	3	3	9	P: Multiple points of failures possible to cause the same issue. I: A drone less in the fleet will impact the performance. A human needs to collect the drone.	Design ERS knowing that drone can be unprecise	Check condition of drone. If it is a 1 time issue manual place drone at ERS.	1	2	2
TR-OPS-10	Operate	ERS does not charge drone	ERS malfunction. Drone battery malfunction. Alignment issue	1	4	4	P: Battery and charging technology is mature, minimizing this risk. I: An ERS failure will be an issue for the entire fleet, having a system wide imapct	maintenance of ERS	Check condition of ERS. If it is a 1 time issue manually place drone at ERS. Otherwise act accordingly	1	3	3
TR-OPS-11	Operate	Drone battery cell dies	Malfunction in battery. Battery not charged properly	1	3	3	P: Battery and charging technology is mature, minimizing this risk. I: A drone less in the fleet will impact the performance	Send drone to charging station well before failure	Manually place drone at ERS	1	2	2
TR-OPS-12	Operate	Drone exceeds noise limit	Multiple drones in close proximity.	5	5	25	P: Drones currently in market exceed warehouse noise limit. I: This is a hard requirement.	Take noise into account during routing. Perform noise studies during design phase. Soundproof warehouse layout	-	2	5	10
TR-OPS-13	Operate	Large part of fleet needs to be charged simultaneously	Large influx of orders.	4	4	16	P: People generally order products at the same time, without proper order management this is likely a common issue. I: Will make it difficult to fulfil orders.	High level planning to spread load on drone system	Add extra human workers to work alongside the drones	1	4	4
TR-OPS-14	Operate	Drone fails to follow set path	Unpredictability in route	2	4	8	P: likelihood of obstacles and unpredictability in warehouse high but drone can fly multiple preset routes to get to its destination. I: system wide offset in routing, flying off route can increase risk of collusion	Ensure algorithm takes into account the physical constraints of the drone	Attempt to return to set path. Else return to maintenance for inspection.	1	3	3
TR-OPS-15	Operate	Items are not available for pickup.	Restocking failure. Items are not where they should be.	3	3	9	P: Likely there are human errors during placing of items. I: Effectively stops drone from doing anything.	Track stock to constantly reflect stock size and position	If item pickup failure occurs reschedule drone such that it does not get stuck	2	1	2
TR-OPS-16	Operate	Dynamic avoidance creates unsafe situation.	While avoiding object drone goes into path of another drone	2	3	6	P: Drone will be aware of most obstacles so dynamic avoidance will be minimal. I: Causes a collision.	Map warehouse accurately.	Remove and repair drones in collision	1	2	2
TR-MRO-01	Maintenance, Repair & Overhaul	Maintenance inspection does not detect faults/system wear	Negligence and human error	3	3	9	P: Without proper inspection methods this will likely happen. I: This can cause an issue within the sytem reducing effectiveness.	Implement data driven inspection methods	Remove drone from fleet and perform mainenance again.	2	3	6

Continued on next page

Table 4.4 – continued from previous page

ID	Function	Description	Driver	P	I	R	Justification	Mitigation Strategy	Contingency Strategy	P*	I*	R*
TR-MRO-02	Maintenance, Repair & Overhaul	Large part of fleet needs to be maintained simultaneously	Similar lifespans of drones which are deployed at the same time	3	4	12	P: Without proper planning the similar lifespans of drones will conincide. I: This will reduce size of the effective fleet.	High level planning to spread load on drone system	Add extra human workers to maintenance crew.	1	3	3
TR-MRO-03	Maintenance, Repair & Overhaul	Drone fails system test after maintenance	Human error	1	2	2	P: A human maintance worker must make a mistake and not notice. I: The drone will be removed from the fleet for a longer than necessary time.	-	Redo maintenance	1	2	2
TR-DEC-01	Decommission	Materials handled incorrectly after decommision	Improper handeling of materials	5	1	5	P: Without a good plan materials will be disposed of incorrectly. I: This will require fixes but the drones will continue to operate.	Create constructive recycling plan for decommission	-	2	1	2



(a) Pre Mitigation

(b) Post Mitigation

Figure 4.1: Risk Assessment diagrams

As Figure 4.1 shows, the mitigation and contingency plans reduce the risks to less critical levels. However two risks remain in the orange region: **TR-AIT-01** and **TR-OPS-12**, which cover the functional tests and the noise operations. These will require extra attention throughout the project to manage the high impact of their occurrence.

4.3. Functional Breakdown Structure and Functional Flow Diagram

The functional breakdown structure and the functional flow diagram are presented in the following pages. These diagrams detail the all of the functions of the DRIVE-FC.

DRIVE-FC

DEV
Develop system
DEV-1 SYS-1
Define requirements
DEV-1.1 SYS-.1.1
Formulate stakeholder needs
DEV-1.2 SYS-1.1
Define stakeholder requirements
DEV-1.3 SYS-1.1
Translate into system requirements
DEV-2 SYS-2
Define system architecture
DEV-2.1 SYS-2.1
Define possible system architectures
DEV-2.2 SYS-2.1
Define possible subsystem architectures
DEV-2.3 SYS-2.1
Perform trade-off studies
DEV-3 SYS-2
Design system
DEV-3.1 SYS-2.1
Generate preliminary design
DEV-3.2 SYS-2.1
Generate detailed design
DEV-3.3 SYS-2.1
Finalize design
DEV-4 SYS-2
Verify and validate design
DEV-4.1 SYS-2.1
Check with requirements
DEV-4.2 SYS-2.1
Verify the design
DEV-4.3 SYS-2.1
Validate the design
DEV-5 SYS-1
Prepare for production and decommissioning
DEV-5.1 SYS-1.1
Prepare production plan
DEV-5.2 SYS-1.1
Prepare decommissioning plan

MAN
Manufacture system
MAN-1 SYS-3
Source materials and off-the-shelf components
MAN-1.1 SYS-3.1
Identify suppliers
MAN-1.2 SYS-3.1
Obtain materials
MAN-1.3 SYS-3.1
Obtain off-the-shelf components
MAN-2 SYS-3
Manufacture subsystems
MAN-2.1 SYS-3.1
Manufacture airframe
MAN-2.2 SYS-3.1
Manufacture grabbing tool
MAN-2.3 SYS-3.1
Manufacture propulsion subsystem
MAN-2.4 SYS-3.1
Fabricate electronics hardware
MAN-3 SYS-3
Check quality
MAN-3.1 SYS-3.4
Inspect subsystems and components
MAN-3.2 SYS-3.4
Prepare component for integration
MAN-3.3 SYS-3.4
Discard component

AIT
Assemble, Integrate, and Test system
AIT-1 SYS-3
Assemble full system
AIT-1.1 SYS-3.2
Combine all subsystems into final product
AIT-1.2 SYS-3.2
Integrate sensors and electronics
AIT-2 SYS-3
Test full system
AIT-2.1 SYS-3.3
Perform functional tests
AIT-2.2 SYS-3.3
Perform software tests
AIT-2.3 SYS-3.3
Perform environmental tests
AIT-3 SYS-3
Verify system readiness
AIT-3.1 SYS-3.3
Perform final acceptance test
AIT-3.2 SYS-3.3
Prepare product for operations
AIT-3.3 SYS-3.3
Initial starting sequence

MRO
Maintain, Repair and Overhaul
MRO-1 SYS-5
Monitor system health status
MRO-1.1 SYS-5.1
Measure system drone health
MRO-1.2 SYS-5.2
Detect faults
MRO-2 SYS-4
Perform routine maintenance
MRO-2.1 SYS-4.1
Perform routine cleaning
MRO-2.2 SYS-4.1
Visually inspect the system
MRO-2.3 SYS-4.2
Perform repairs
MRO-2.4 SYS-4.2
Replace components
MRO-2.5 SYS-4.2
Overhaul system
MRO-3 SYS-4
Update software or firmware
MRO-3.1 SYS-4.3
Apply updates
MRO-3.2 SYS-4.3
Test system after update

DEC
Decomission
DEC-1 SYS-4
Deactivate system
DEC-1.1 SYS-5.2
Remove sensitive data
DEC-1.2 SYS-4.4
Turn-off power
DEC-2 SYS-4
Handle waste material
DEC-2.1 SYS-4.4
Recycle recyclable components
DEC-2.2 SYS-4.4
Dispose hazardous materials safely
DEC-3 SYS-1
Document process
DEC-3.1 SYS-1.1
Archive documentation

OPS
Operate

Communication & Computation Module

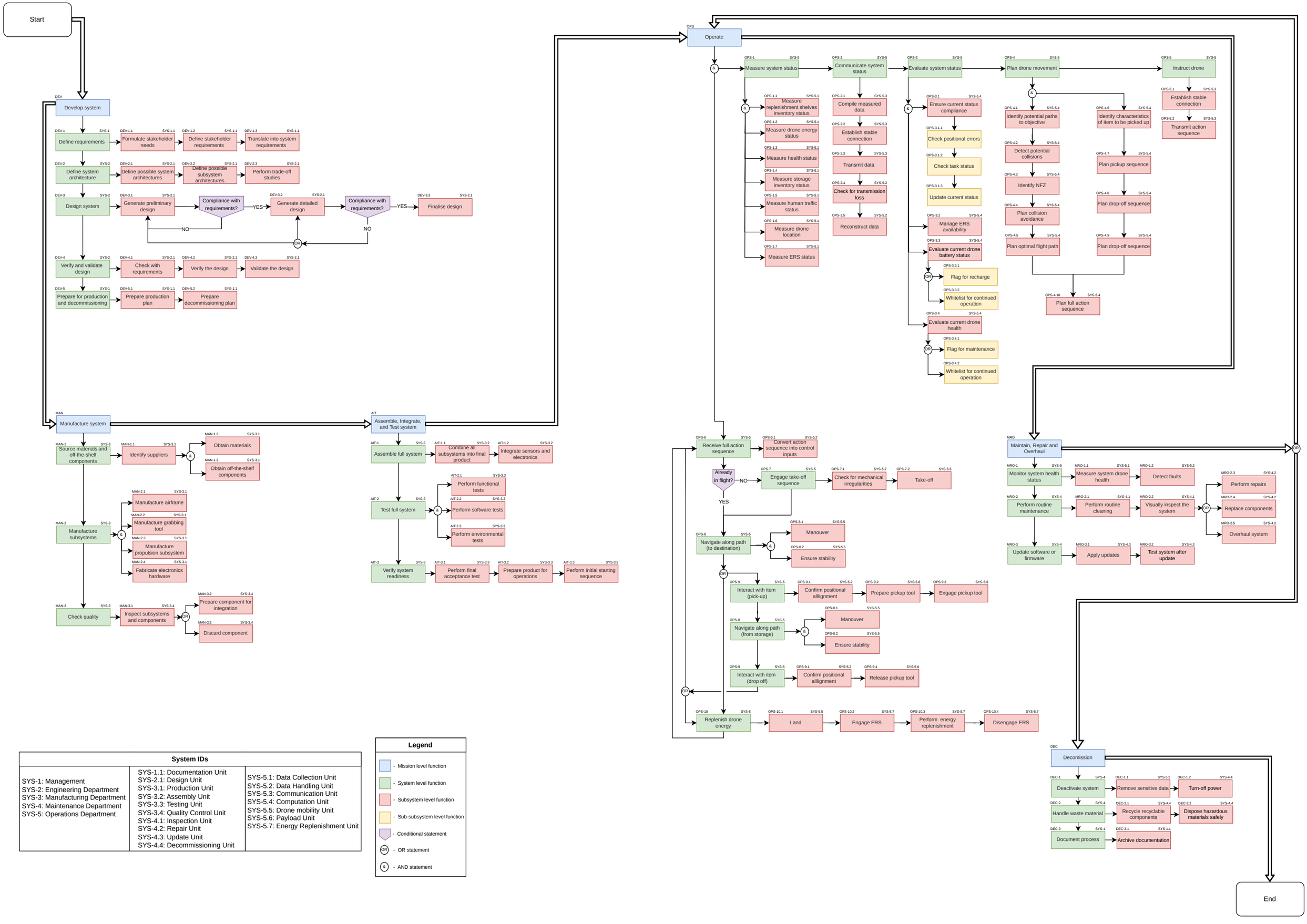
OPS-1 SYS-5
Measure system status
OPS-1.1 SYS-5.1
Measure replenishment shelves inventory status
OPS-1.2 SYS-5.1
Measure drone energy status
OPS-1.3 SYS-5.1
Measure health status
OPS-1.4 SYS-5.1
Measure storage inventory status
OPS-1.5 SYS-5.1
Measure human traffic status
OPS-1.6 SYS-5.1
Measure drone location
OPS-1.7 SYS-5.1
Measure ERS status
OPS-2 SYS-5
Communicate system status
OPS-2.1 SYS-5.3
Compile measured data
OPS-2.2 SYS-5.3
Establish stable connection
OPS-2.3 SYS-5.3
Transmit data
OPS-2.4 SYS-5.2
Check for transmission loss
OPS-2.5 SYS-5.2
Reconstruct data
OPS-3 SYS-5
Evaluate system status
OPS-3.1 SYS-5.4
Ensure current status compliance
OPS-3.1.1-Check positional errors
OPS-3.1.2-Check task status
OPS-3.1.3-Update current status
OPS-3.2 SYS-5.4
Manage ERS availability
OPS-3.3 SYS-5.4
Evaluate current drone battery status
OPS-3.3.1-Flag for recharge
OPS-3.3.2-Whitelist for continued operations
OPS-3.4 SYS-5.4
Evaluate current drone health
OPS-3.4.1-Flag for recharge
OPS-3.4.2-Whitelist for continued operations
OPS-4 SYS-5
Plan drone movement
OPS-4.1 SYS-5.4
Identify potential paths to objective
OPS-4.2 SYS-5.4
Detect potential collisions
OPS-4.3 SYS-5.4
Identify NFZ
OPS-4.4 SYS-5.4
Plan collision avoidance
OPS-4.5 SYS-5.4
Plan optimal flight path
OPS-4.10 SYS-5.4
Plan full action sequence
OPS-4.6 SYS-5.4
Identify characteristics of item to be picked up
OPS-4.7 SYS-5.4
Plan pickup sequence
OPS-4.8 SYS-5.4
Plan drop-off sequence
OPS-4.9 SYS-5.4
Plan drop-off sequence
OPS-5 SYS-5
Instruct drone
OPS-5.1 SYS-5.3
Establish stable connection
OPS-5.2 SYS-5.3
Transmit action sequence

Drone Module

OPS-6 SYS-5
Receive full action sequence
OPS-6.1 SYS-5.2
Convert action sequence into control inputs
OPS-7 SYS-5
Engage take-off sequence
OPS-7.1 SYS-5.2
Check for mechanical irregularities
OPS-7.2 SYS-5.5
Take-off
OPS-8 SYS-5
Navigate along path
OPS-8.1 SYS-5.5
Manouver
OPS-8.2 SYS-5.5
Ensure stability
OPS-9 SYS-5
Interact with item
OPS-9.1 SYS-5.2
Confirm positional alllignment
OPS-9.2 SYS-5.6
Prepare pickup tool
OPS-9.3 SYS-5.6
Engage pickup tool
OPS-9.4 SYS-5.6
Release pickup tool
OPS-10 SYS-5
Replenish drone energy
OPS-10.1 SYS-5.5
Land
OPS-10.2 SYS-5.7
Engage ERS
OPS-10.3 SYS-5.7
Perform energy replenishment
OPS-10.4 SYS-5.7
Disengage ERS

System ID		
SYS-1: Management SYS-2: Engineering Department SYS-3: Manufacturing Department SYS-4: Maintenance Department SYS-5: Operations Department	SYS-1.1: Documentation Unit SYS-2.1: Design Unit SYS-3.1: Production Unit SYS-3.2: Assembly Unit SYS-3.3: Testing Unit SYS-3.4: Quality Control Unit SYS-4.1: Inspection Unit SYS-4.2: Repair Unit SYS-4.3: Update Unit SYS-4.4: Decommissioning Unit	SYS-5.1: Data Collection Unit SYS-5.2: Data Handling Unit SYS-5.3: Communication Unit SYS-5.4: Computation Unit SYS-5.5: Drone mobility Unit SYS-5.6: Payload Unit SYS-5.7: Energy Replenishment Unit

Legend
- Mission level function
- System level function
- Subsystem level function



System IDs		
SYS-1: Management SYS-2: Engineering Department SYS-3: Manufacturing Department SYS-4: Maintenance Department SYS-5: Operations Department	SYS-1.1: Documentation Unit SYS-2.1: Design Unit SYS-3.1: Production Unit SYS-3.2: Assembly Unit SYS-3.3: Testing Unit SYS-3.4: Quality Control Unit SYS-4.1: Inspection Unit SYS-4.2: Repair Unit SYS-4.3: Update Unit SYS-4.4: Decommissioning Unit	SYS-5.1: Data Collection Unit SYS-5.2: Data Handling Unit SYS-5.3: Communication Unit SYS-5.4: Computation Unit SYS-5.5: Drone mobility Unit SYS-5.6: Payload Unit SYS-5.7: Energy Replenishment Unit

Legend

- Mission level function
- System level function
- Subsystem level function
- Sub-subsystem level function
- Conditional statement
- OR statement
- AND statement

5

Logistics concept & Warehouse layout

In this chapter, the layout of the warehouse and operations concept is discussed. Section 5.1 includes the functions and requirements that the warehouse must adhere to. Section 5.2 presents the data provided by the client. Section 5.3 covers the top level logistical flow in the warehouse and presents the final concept for integrating a drone based system in the warehouse. With the concept decided, Section 5.4 shows the operational process in the warehouse. Lastly, based on the data and some assumptions, in Section 5.5, the drone picking area is sized, which will be used in routing and control simulations.

5.1. Requirements

The warehouse is the base for the logistical processes in e-grocery operations. As such it is an important factor when designing a drone based system aiming to improve efficiency and has to be designed accordingly to ensure the drone system operates effectively and other processes in the warehouse are disturbed as little as possible. From the list of requirements, the following are considered in the design process for the warehouse:

- **RQ-WOS-01**: The system shall ensure worker safety during operations.
- **RQ-EFF-01**; The system shall achieve a daily volume of at least 10,000 successful item transfers per 24 hours.

5.2. Warehouse Specifications

An external expert has provided data on warehouse information relevant for the design process (W. Möllmann, personal communication, May 2026). These are listed below.

- **Warehouse layout**
  - Typical warehouse dimensions are 110×182 meters
  - Warehouse is sectioned in temperature zones: Ambient ($\approx 18^oC$), Chilled ($\approx 2^oC$) and Frozen ($\approx -20^oC$).
- **Warehouse items**
  - An average warehouse stores 10,000 different kinds of Stock Keeping Unit (SKU).
  - 50% of SKU's are stored in the ambient section, 40% in the chilled section, and 10% in the frozen section.
  - 65% of SKU's weigh less than 1 kg.
  - SKU's arrive either in boxes ranging from $50 \times 50 \times 50$ mm to $600 \times 400 \times 400$ mm or in plastic totes ranging in size from $600 \times 400 \times 70$ mm to $600 \times 400 \times 230$ mm.
  - The average daily throughput of a warehouse with the above given size is 200,000 items/day.

5.3. Logistical Flow & Layout Concept

In general, the logistical flow within a drone-based picking system warehouse is relatively unchanged compared to current flows. SKU's arrive in the boxes or totes via truck loads at several moments during the day. They are then processed and sent to fulfilment racks. There, human pickers gather the items place them in crates with the customer's order. These crates are then transported to other trucks which bring the orders to a distribution centre, where they are shipped out to the customer. Figure 5.1 shows this process in a block diagram.

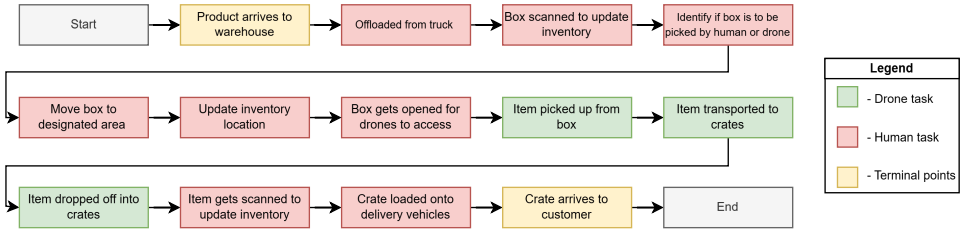


Figure 5.1: Logistical flow for a single product

The big difference with a drone based system is in the picking stage. A portion of items is now proposed to be collected and transported by drones, highlighted by the green boxes in Figure 5.1. This means a section of the warehouse must be allocated to accommodate this. The chosen concept sections off this area physically to minimise worker-drone interactions and possible safety risks (see RQ-WOS-01). Regarding the configuration of the drone operating zone, a simple multi aisle shelf system is adopted. This configuration allows for optimal space utilisation compared to an enclosed drone area with shelves forming the border. Within the operating zone, picking drones and the humans replenishing shelves are kept separate by means of no-fly zones programmed into the drones. This method was chosen mainly due to its ease of implementation into current warehouse systems, compared to designing new shelves that physically block human-drone interaction. Both human-picked and drone-picked items will be placed in crates in 'Picked Orders'. Given these design choices, the new warehouse layout and logistical flow is presented in Figure 5.2.

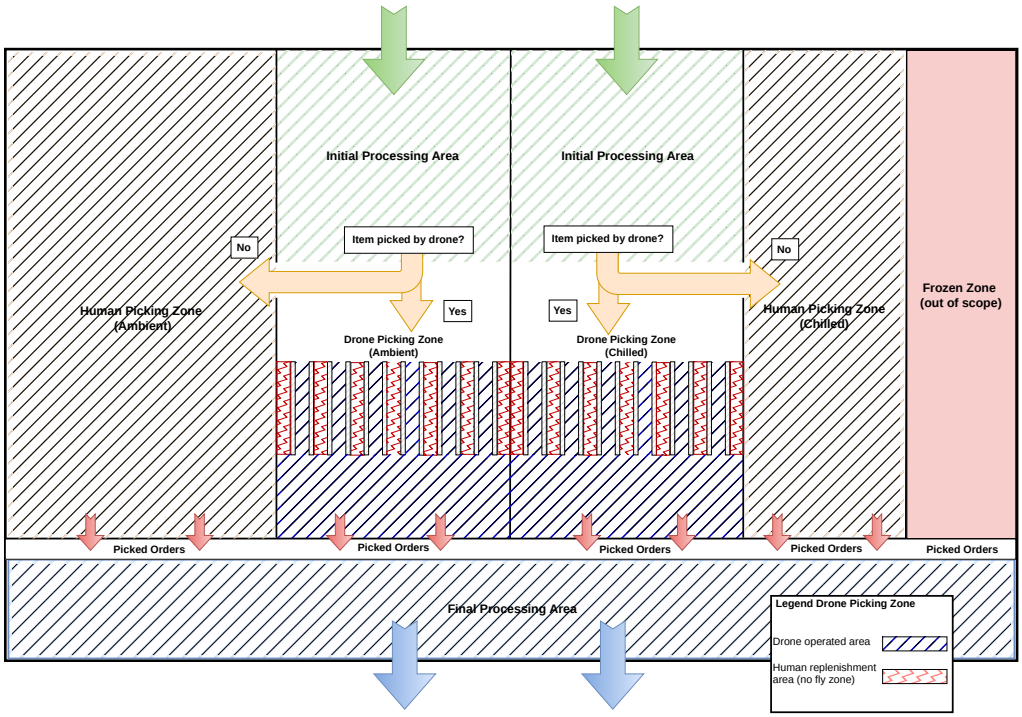


Figure 5.2: Warehouse layout and logistical flow

5.4. Operations Concept

As shown in Figure 5.2, the warehouse is spatially divided such that each stage of the logistical process is allocated a dedicated zone. SKU's first arrive at the identified *Initial processing area*. Here, they are, offloaded from the trucks, removed from from the roll carts and placed on an automated system (like a conveyor belt) that identifies if the product is sent to the drone picking zone or the human picking zone. If the destination is the drone picking zone, the box will be unloaded from the automated system, placed on a pallet and transported to the shelves using a vertical pallet stacker (like the Mitsubishi AXiA ES stacker [48]) and opened for the drones to access. The steps are illustrated in Figure 5.3

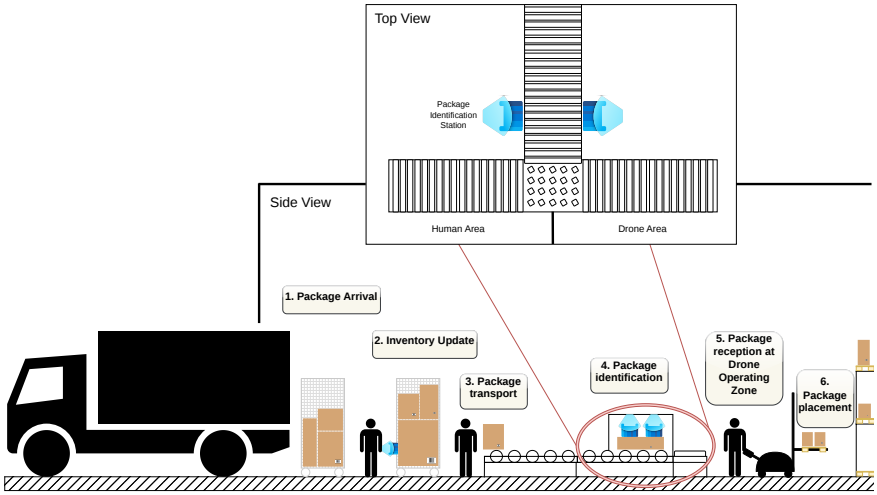


Figure 5.3: Initial processing area operations

Once the SKU's have been placed in the shelves, the drones pickup the items based on incoming orders. While the drone's flight mechanics and environmental interactions are detailed in later sections, the operational sequence can be summarised as five key steps. The first is path planning, where the drone receives the calculated optimal path to the requested item from the central computer. The drone then picks up this item from the boxes in the shelves, and flies with this item to the order crate. The drone then drops off the item in the order crate. Lastly, if the drone is low on power, it will fly to the ERS to recharge. These steps are visualised in Figure 5.4.

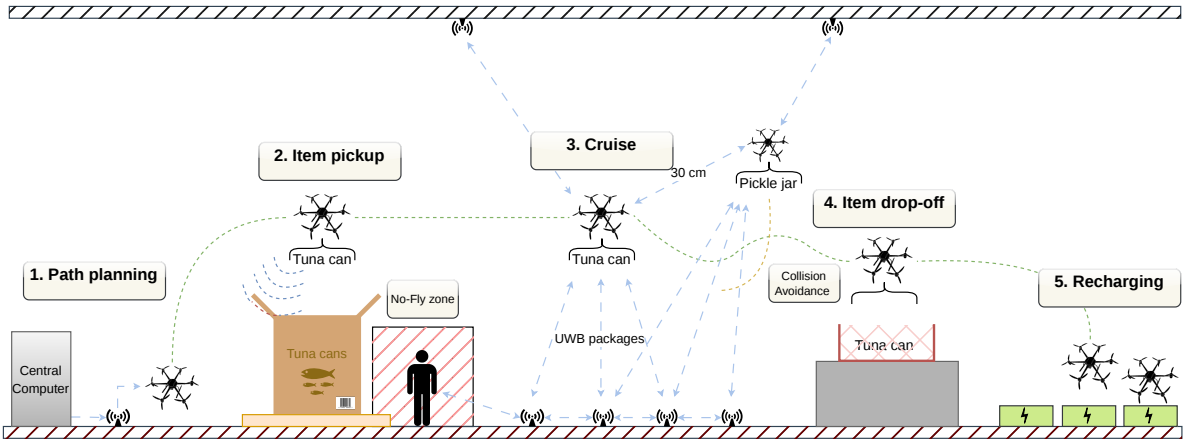


Figure 5.4: Drone area operations

The final step in the operation is transferring the order crates from the the drone operating area to the trucks that are sent out to the distribution centres. These operations are unchanged from current operations. A visualisation is presented in Figure 5.5.

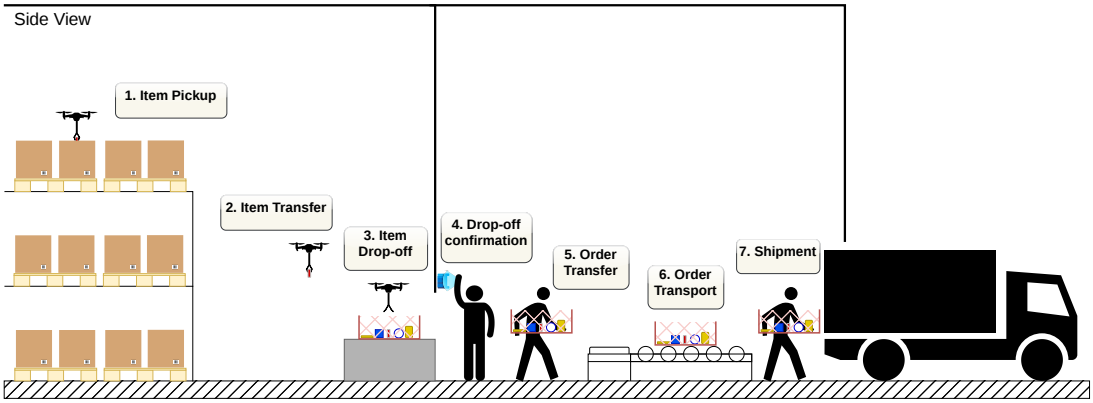


Figure 5.5: Final processing area operations

5.5. Drone Picking Area

To meet requirement RQ-EFF-01, the drone picking area must be sized accordingly to determine the number of SKU's placed in the drone picking area, and the resulting size and layout which will be used in the routing simulation, which will verify if the requirement is met.

5.5.1. Assumptions

Before the drone area is sized, a number of assumptions have been made. They are listed below:

- Aisle width is 2.5 m. This is to accommodate the pallet stacker restocking the shelves.
- Between the box and the next shelf level, 1.5 m margin is taken to ensure the drone can fly into the shelf without running into propeller wake issues.
- Warehouse height is taken to be 6 m. This is to allow the UWB sensors in the warehouse to be mounted on the roof, as they become less effective at long distances.
- The number of shelf levels then becomes three, as this is the maximum allowable within the 6 m height constraint.
- The pallets used in the warehouse are European standard size, which is 80×120 cm. Given the maximum box and tote dimensions from Section 5.2, a maximum of four boxes can be placed on a pallet, shown below.

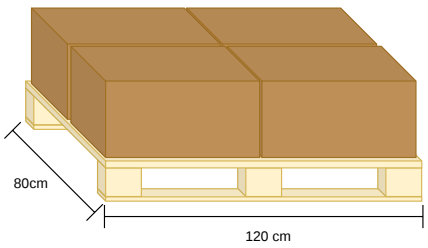


Figure 5.6: Warehouse storage method

5.5.2. Warehouse Layout Optimisation

Considering that the drone operations will only take up a fraction of the total warehouse operations, it is essential that the space taken up is minimised. To achieve this, an optimisation program is written that finds a number of solutions with different aisle configurations and the corresponding optimised area given a number of SKU's stored in the area. RQ-EFF-01 states that a daily throughput of 10,000 SKU's per day must be achieved. This is 5% of the 200,000 SKU's per day as stated in Section 5.2. As a result, it is assumed 500 SKU's must be stored in the drone picking zone.

Using the assumptions in Section 5.5.1, the aisle width in the area is set to 2.5 m wide. For the shelves, it

is assumed that the dimensions of the shelf are fully dependant on the pallet dimensions. Considering a minimum of four boxes can be placed on a pallet simultaneously (see Figure 5.6), it is deemed that one pallet per SKU in the drone picking zone is sufficient. Given a vertical pallet stacker picks up pallets from the wide end, it is assumed that the pallets will be placed in the shelves in this orientation. As a result, the shelf depth is determined to be 0.8 m and the pallet width of 1.2 m is taken into account when determining the aisle length.

The shelf height is not explicitly taken into account in the optimisation however the number of shelves is. Given the margin of 1.5 m determined in the assumption and a maximum box height of 0.4 m, the shelf height is 1.9 m. Then using the assumed warehouse height of 6 m, an optimal point of three levels of shelves is determined and this value is taken into account in the optimisation.

The optimisation program runs through all possible aisle configurations and determines the minimum required dimensions to fit 500 pallets. The chosen final configuration is presented in Figure 5.7.

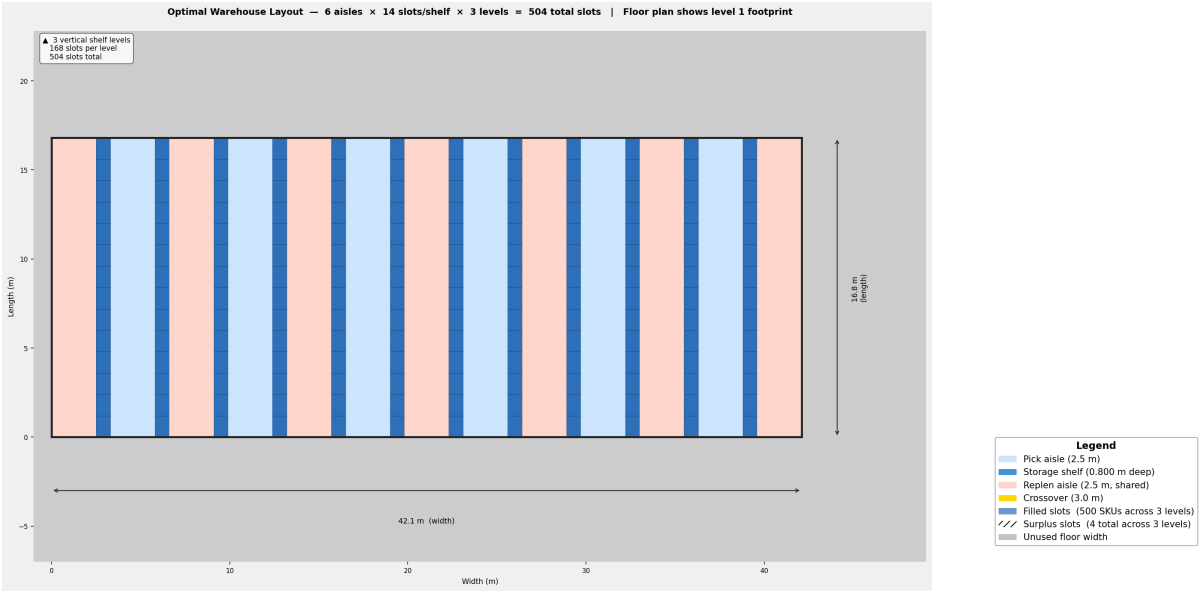


Figure 5.7: Drone picking zone configuration

The final configuration represents a sweet spot between number of aisles, where a higher number is beneficial for a multi drone operation while also minimising the warehouse area utilisation in the concept presented in Section 5.3. The final dimensions are as follows:

- **Drone picking zone width:** 42.1 meters
- **Aisle length:** 16.8 meters
- **Number of aisles:** 6

5.6. Verification and Validation

As the warehouse forms the base of the operations, there is little that can be specifically verified and validated with respect to DRIVE-FC's requirements. The two top level requirements that have been identified in Section 5.1 are restated in the compliance matrix in Table 5.1.

Table 5.1: Compliance matrix for the communication subsystem requirements. Status: C = compliant, PC = partially compliant, NC = non-compliant, O = open/TBD.

Req ID	Method	Status	Remarks
RQ-WOS-01	A	C	Physical barrier between drone and human picking operations. Human restockers and drones separated in drone picking zone using no fly zones.
RQ-EFF-01	A	O	Will be tested in Chapter 13 and Chapter 14.

5.7. Limitations and Recommendations

The initial design concept for the warehouse presents a solution for integrating a drone picking operations. The warehouse has been sized to optimally fit the predetermined number of SKU's in a minimum area given some constraints identified by an external source. Currently the focus has been on the drone picking zone itself, which is due to DRIVE-FC's focus on the drone operation aspect of the project. However future developments should also elaborate on the pre- and post processing area in the warehouse. Section 5.4 currently presents a concept for how this could look like in a warehouse however this should be elaborated upon in order to get an accurate estimate of item replenishment and further improve the accuracy of the simulation presented in Chapter 13. Further discussions with a potential client about how to integrate such a system properly is also something that has to be set up before any physical testing can take place. Because DRIVE-FC currently only covers 5% of the total warehouse throughput, it is not financially sensible to construct an entire warehouse from scratch with the concepts presented in this chapter. Instead solutions must be found to properly accommodate the DRIVE-FC operation into an existing warehouse, and this is something that can only be achieved in collaboration with the client.

6

Propulsion and Noise

This chapter details the design of the drone's propulsion subsystem, focusing heavily on aero-acoustic noise reduction. First, core requirements are defined in Section 6.1, followed by a detailed literature study of potential design options in Section 6.2. Next, Section 6.3 outlines the selected conceptual approach before detailing the implementation of a custom Blade Element Momentum Theory (BEMT) script used to iteratively size, optimise, and finalise the propeller geometry. Finally, Section 6.4 evaluates the completed subsystem's compliance against the initial requirements, while Section 6.5 concludes the chapter by exploring methodological limitations alongside recommendations for future design iterations.

6.1. Requirements

Several key requirements have been established in Table 4.1, with the main ones driving the propulsion system given in Table 9.1. These requirements and functions will largely drive the design of the subsystem, with a special emphasis on noise.

Table 6.1: Verification plan for propulsion requirements

Requirement ID	Requirement	M	Method description
RQ-ENV-01-PRP-01	The propulsion subsystem shall use recyclable or recycled materials.	I	Check properties and origin of propulsion material.
RQ-WOS-01-PRP-01	The propulsion subsystem shall not produce noise levels higher than 75 dBA at one meter distance from the drone.	T*	Measure sound level at one meter distance from drone.
RQ-GWH-01-PRP-01	The propulsion subsystem shall be functional in Cold-storage conditions.	T*	Test propulsion subsystem functioning in cold storage zone.
RQ-GWH-02-PRP-01	The propulsion subsystem shall be functional in Ambient conditions.	D*	Demonstrate propulsion subsystem functioning in ambient storage zone.
RQ-GWH-03-PRP-01	The propulsion subsystem shall be able to provide a thrust-to-weight ratio of at least 2.0.	A	Check the thrust-to-weight ratio by analysing mass and thrust limits.
RQ-RLB-01-PRP-01	The propulsion subsystem shall be capable of operating for at least one continuous year.	A	Analyse degradation of propulsion subsystem, and compare with similar applications.
RQ-MFC-01-PRP-01	All off-the-shelf propulsion subsystem components shall have a maximum verified supplier lead time of 4 weeks.	I	Check with manufacturers and distributors whether propulsion subsystems are available within the time frame.

While material and environmental requirements will be solved through manufacturing choices, the geometric design of the propeller is driven entirely by the noise (RQ-WOS-01-PRP-01) and thrust (RQ-GWH-03-PRP-01) requirements, the latter of which will be explored in more detail in Section 6.2.

6.2. Design Options

As established in the preliminary design stage, the main driving criteria for the propeller design are those enabling noise reduction while providing aerodynamic efficiency. Since there is no singular best way to achieve noise reduction, two design options, namely uneven blade spacing and toroidal propellers, are considered going into the detailed design phase. While serrated edges proved interesting at first, the latest research shows that they can cause an increase in high frequency components to which humans are most sensitive, and hence were eliminated as an option. Given these final design options, a more detailed literature study is performed to ensure that all characteristics are well understood - for this, a number of papers are considered and their findings listed below.

6.2.1. Unevenly Spaced Propellers

While being widely used in modern-day applications such as low-noise *fans* for radiators in cars, research on unevenly spaced blades/propellers for propulsive use is rather sparse, with only a handful of authors actively writing related papers. Of these, Usov et al. [51] provides the most useful insight relevant to this project and hence will serve as main source. Additionally, Arystanbekov et al. [7] also provides interesting insights but applies them to symmetric unevenly spaced blades. The main takeaways from these papers are listed below:

- **Noise reduction:** In traditional propellers, where the twin-blades are spaced 180° apart, blade passage happens twice per motor-shaft-revolution, i.e., $BPF = 2SF$, where BPF is the Blade Passing Frequency and SF the Shaft frequency. At every passage, the underlying aerodynamics result in noise emissions, creating distinct noise-peaks at integer multiples of the BPF - this component is called harmonic/tonal noise and is the most irritating to humans since it occurs at high frequencies we are most sensitive to. By shifting the blades to an asymmetric configuration (e.g. $< 180°$) these noise emissions only repeat once per shaft rotation, meaning the fundamental frequency now is only half of the original BPF. As such, the acoustic energy is now spread out over a denser spectrum, resulting in more, but smaller spikes at 0.5, 1, 1.5,... BPF. This lowers the amplitude of the highest tonal peaks, and spreads the energy into lower frequency domains to which humans are less sensitive, translating to a considerable reduction in annoyance [23].

- **Overall Sound Pressure Level (OASPL):** The OASPL is the integral of all the acoustic energy across the entire frequency spectrum and is a powerful tool in noise analysis. Following the explanation above, a logical conclusion is that the OASPL remains constant. This is because uneven spacing simply takes the energy from the main BPF peaks and redistributes it into the sub-harmonics while the total area under the curve remains about the same, or in some extreme cases, slightly increases due to aerodynamic penalties [51][7]. Hence an important takeaway is that uneven rotors aim to lower human annoyance (quantified by the A-weighted decibel scale, OASPL-A), not to lower the acoustic energy [60].

- **Directivity:** In aeroacoustics, directivity refers to the 3D shape of the noise footprint (e.g., whether it is louder directly below the drone or off to the sides). In both symmetric and asymmetric rotors, the noise directivity forms a dipole shape [51] (i.e., loud directly below the drone and more quiet to the sides).

- **Efficiency:** The papers highlight aerodynamic penalties in uneven blade spacing, mainly from blade-wake interaction [51][7]. This is due to propeller blades leaving behind a low-pressure, turbulent, swirling wake. As such, when blades are closely spaced the second blade moves through turbulent air, resulting in a substantial loss in efficiency. This effect is worsened as the spacing angle (γ) is reduced since there is less time for the wake to dissipate before the next propeller encounters it. This effect is also present in 180° blade spacing but less pronounced due to increased spacing. Specifically, studies found angles below 130° to introduce substantial efficiency losses as well as increased broadband noise due to the turbulent wake.

Limitations and Considerations

Although the findings from Section 6.2.1 are quite complete, several gaps have been noted going into the detailed design phase. First, it seems logical that blade-wake interactions would worsen upon a reduction in

blade spacing. The problem here is that existing research on *asymmetric* unevenly spaced propellers sticks to the traditional ($\gamma = 180°$) idea of blades situated in the same plane; however, this configuration will *not* work for closely unevenly spaced propellers. As a result, the findings of a reduction in efficiency could stem from the inherent limitations of a co-planar design rather than a flawed concept. This co-planar limitation is partly tackled by Arystanbekov et al., who provide critical insight into *symmetric* unevenly spaced blades - a positive finding here was that the propeller is more efficient if the upper blade trails the lower blade, because a leading upper blade would create a wake that the lower blade would directly travel into. However, a critical finding is that the 30° four-bladed propeller was vastly noisier and more inefficient than a simple 90°, four-bladed propeller.

While these findings seem to eliminate the uneven blade spacing as a viable option, it is hypothesised that these limitations are limited to the symmetric layout. In fact, by using a *symmetric*, 30° uneven blade spacing propeller with vertical offset, the distance between two non-paired blades reduces to 150°. Due to the vertical offset, the vortices created by pair-1's elevated, lagging blade are likely to hit the next, pair-2's leading, lower blade, causing a massive loss in efficiency and noise due to Blade Vortex Interactions (BVI). Furthermore, the paper ran the propellers at 15.000 RPM, which is way above the design point considered for this design - this high RPM in combination with a reduced separation angle and vertical offset makes it impossible for any vortices to clear the trailing blade, increasing noise and decreasing efficiency. Since the propeller designed as part of this project will be a two-bladed asymmetric one, assuming a separation angle of 30°, a lagging upper blade and low operational RPM, the vortices created by the upper blade should have sufficient time to travel downstream before the lower blade enters that region, giving both blades relatively clean air to operate in. Despite this, two main aerodynamic penalties remain - the lower blade is continuously operating in the upper blade's slipstream as well as induced velocity. This, however, can partially be compensated for by pitching the blade, if deemed necessary.

Moreover, research papers on unevenly spaced fans have highlighted that although tonal annoyance drops, tonal roughness may increase. Here, tonal roughness is caused by rapid frequency modulations, usually at low frequencies between 20 Hz and 300 Hz. Assuming that the drone parameters from the preliminary sizing stay about the same, the BPF is expected to fall around 100 Hz for conventional propellers, or multiples of 50 Hz for the unevenly spaced one. Furthermore, assuming a separation angle of 30°, the individual propeller blades pass at a frequency of 600 Hz, which lies well outside the critical 'roughness range' that humans are most sensitive to. However these parameters will be refined towards the end of the chapter, and the critical range will have to be re-evaluated. Overall, noise reduction using unevenly spaced blades is a balancing act between minimizing tonal peaks and managing this roughness penalty.

Finally, a main drawback in asymmetric uneven propellers is that of added weight - in order to maintain proper equilibrium, parasitic weight needs to be added to the side opposing the blades. This adds dead weight as well as potential aerodynamic penalties due to the leverage arm needed to balance it. Furthermore, increased propeller weight will reduce controllability due to a higher moment of inertia, making it more difficult for the motors to execute the rapid micro-RPM adjustments sent by the flight controller.

6.2.2. Toroidal Propellers

Toroidal propellers were initially developed by the Massachusetts Institute of Technology (MIT), taking inspiration from toroidal boat propellers and utilising it in air applications. Since initial results were promising, an abundance of papers covering toroidal propellers exist and analyse its performance. From these, the main takeaways are listed below:

- **Noise reduction:** Toroidal propellers aim to reduce high-energy tip vortices originating from a high-pressure bottom and a low-pressure top. These vortices typically generate high-frequency broadband noise which is highly aggravating to humans. Toroidal propellers instead shed this vortex continuously along the curved outer loop, creating a much weaker and diffuse vortex structure. Resultantly, high-frequency noise is suppressed, while low-frequency broadband noise slightly increases. However, given human hearing sensitivity, this simply shifts the noise to a frequency we are less sensitive to, thereby reducing perceived noise considerably [13].
- **Overall Sound Pressure Level (OASPL):** Unlike uneven spacing, toroidal propellers can actually reduce the physical acoustic energy leaving the drone. Literature shows that a true reduction in OASPL, typically in the range of 1 to 3 dB compared to a standard propeller generating the exact same thrust, can be achieved for a carefully designed loop [56]. However, as noted above, because human hearing is more sensitive to high frequencies, the reduction in the A-weighted scale (dBA) is much more pronounced.

- **Directivity:** Contrary to traditional propellers which have a dipole distribution, as explained in Section 6.2.1, toroidal propellers generally have a more isotropic (even in all directions) distribution, meaning the noise signature on the ground does not spike as much as it would normally [56].
- **Efficiency:** By eliminating the tip vortex, toroidal propellers significantly reduce induced drag - following the same principle as winglets on an airplane. However, given an increase in physical surface area (wetted area) to cover the same diameter as a standard blade, skin friction increases considerably, translating to possibly higher torque and hence current requirements.

Limitations and considerations
Smearing the high-frequency vortices into lower-frequency, distributed vortices is only beneficial in a certain working range. As an example, if a vortex creates noise around 20 000 Hz then it is already inaudible to an average adult human, hence lowering its frequency would actually shift it into a range that humans are sensitive to, resulting in a net increase in dBA. Conversely, if the base vortex noise frequency is around 3000 Hz, i.e., the peak hearing sensitivity of humans, then lowering the frequency to less sensitive ranges would be worthwhile due to a large reduction in dBA. A rough estimate of the vortex frequency can be found by using the following equation:

$$f_{\text{vortex}} = \frac{StV}{\delta}\,, \tag{6.1}$$

Here, St is the Strouhal number which is found empirically based on the Reynolds number, but typically is 0.1 to 0.2. V is the speed of the flow and δ is the characteristic length of the object causing the vortex. Since the exact frequency depends on all these inputs, which in turn depend on the propeller geometry and flow conditions that are not known yet, the equation will be evaluated once the base-propeller design is completed and only the loop remains to be added - if the frequency ends up in a range of 1000 Hz to 4000 Hz, the toroidal loop will be employed as it would reduce noise emissions. If the frequency lies beyond that range then adding it would likely cause an increase in human-perceived-noise and the loop is not employed.

As mentioned in Section 6.2.2, another consideration is that of the added wetted surface - although the exact increase in surface area depends on the propeller geometry, a rough estimate would put it around a 20-30% increase. This increase in wetted area directly translates to an increase in skin friction which in turn decreases with increasing Reynolds number ($D_{\text{skin}} \propto 1/Re^{0.2}$), for turbulent flow. Reynolds number increases with increasing chord length; however, given it is raised to the power of 0.2 whereas surface area scales linearly with the chord length, the objective is to keep the loop's chord length to a minimum as to add as little skin friction as possible. However, while a small chord is beneficial for skin friction, shrinking it down too much would also prove problematic since operating in a low Reynolds number regime dramatically increases the likelihood of laminar separation bubbles - hence this has to be taken into account when setting the loop chord length. Furthermore, since the loop essentially acts like a winglet, induced drag could decrease by about 30% [18], potentially outweighing the increase in skin friction and perhaps even achieving a net lower torque requirement, which is beneficial to keep the battery weight down. This is reinforced by the fact that induced drag is dominant at low speeds whereas skin friction drag is dominant at high speeds - since the propeller will be designed for low RPM (to keep the BPF down), the reduction in induced drag is more likely to outweigh the increase in skin friction drag.

6.3. Detailed Design

Having established a theoretical background in Section 6.2, this section explores the methodology used to attain the design. A key aspect of this is the custom Blade Element Momentum Theory (BEMT) script that was written to help with the design - although powerful tools are readily available, it was decided to write a custom program as it gives vastly more freedom in the design. This freedom is needed since the considered design is very atypical and requires special considerations in the calculation of various parameters and generally allows one to define variables as needed without external limitations. Additionally, elaborate optimisation loops can be defined to find the best design out of countless options - for this a vast dataset spanning 6351 airfoils by Michael Quayle from BigFoil [1] was used for the optimisation loop to identify the optimal aerodynamic propeller configuration. As for the theoretical background, McCormick [9] served as reference, given the elaborate explanations provided.

Finally, the program is sectioned into three main branches - first, the propeller is designed for a specific operating point, then attributes are calculated for the full flight envelope and once that is done, the output is filtered and ranked, yielding the final propeller design.

[1] Quayle, Michael. 2026. "BigFoil.com Airfoil Database" https://www.bigfoil.com - Accessed 31.05.2026

6.3.1. Considered Design

Going into the detailed design phase, the design option considered is that of a closed loop, vertically offset, two-bladed, *asymmetric*, unevenly spaced blade propeller - here *closed loop* implies the existence of a toroidal loop inspired structure connecting the lower and upper blade. This loop aims to weaken/eliminate tip vortices in order to reduce tip vortex noise and, if designed correctly, can also generate thrust [13]. Although some vertically offset blades have been found to generate more noise than a traditionally designed propeller, by use of a carefully calculated vertical offset in combination with a toroidal loop, we hypothesise that the BVI can be largely, if not fully eliminated, leading to an overall extremely quiet and efficient system. Balancing the vertical offset means adjusting the separation such that the upper blade does not operate within the lower blade's potential flow disturbance, while keeping the offset to a minimum to ensure the upper blade's vortices clear the lower blade. Because potential flow disturbance effects decay exponentially with distance and become aerodynamically negligible within about half a chord length, this minimum separation distance is set to $0.8c_{max}$, where c_{max} is the maximum chord length. Furthermore, to minimise aerodynamic interactions, the leading blade (the one that hits the flow first rotationally) is positioned at the bottom, and the trailing blade is positioned at the top. In other words, the first (lower) blade leads the second (upper) blade in the rotational direction. With these considerations in mind, it becomes clear why this solution could prove powerful: It both shifts frequencies into lower-frequency domains to which humans are less sensitive and potentially increases efficiency by letting the blades operate in relatively clean air as well as potentially reducing torque requirements.

Note on the number of blades

Generally, two-bladed propellers are used for applications that require maximum efficiency and flight time per battery. Using more than two blades is increasingly inefficient and requires more current draw, resulting in a snowballing effect in terms of weight due to added battery mass, both because of increased current and voltage sag. Furthermore, adding more blades inherently shifts the frequency to higher tones which human hearing is most sensitive to as the SF is multiplied by a larger number. Hence, a two-bladed configuration is the logical choice for this project.

6.3.2. Operating Point Geometric Sizing

First, a design needs to be established - for this, a given operational point is defined that will drive the propeller geometry and flight characteristics. Given that the preliminary sizing found the drone to weigh about 6.33 kg, a conservative weight of 6.5 kg was chosen as design point weight. Furthermore, hover flight RPM was found to be 3900 RPM - since advanced flight is expected to have higher RPM but the design point already considers a conservative weight, the design RPM is set to a flat 4000 RPM - from this, the tangential velocity V_{tan} is determined. The size and layout from the preliminary sizing are kept constant, resulting in a design propeller size of 18.75 cm. Finally, the design speed is taken to be 30 km/h. This value was selected to represent a high cruising velocity for a logistics drone, striking a balance between short delivery times while keeping parasitic drag down. Because the BEMT script is fully parametric, these baseline assumptions can be easily updated in future design iterations - whether adjusting the cruise speed to match control limits, refining the mass based on system-level integration, or tuning the RPM to shift noise emissions away from sensitive frequency bands.

With the design parameters defined, the forward normal velocity, $V_{\text{fwd, n}}$ (i.e., the forward flight velocity component normal to the propeller rotational plane) as well as the pitching angle θ and inflow angle, ϕ can be found by equating the forward thrust and body drag. Here, the inflow angle is the angle at which the incoming air approaches a stationary object, i.e., the propeller blade, and varies along the span. Since the drag depends on the pitching angle, these values have to be solved for iteratively. One main assumption here was the effective surface area and C_d - while both values are estimations, the general trend of propeller performance as a function of selected airfoil is expected to stay the same. In a next step, the chord needs to be defined - in theory, the most optimal shape is the one that distributes the lift equally along the whole blade, i.e., is elliptical. This is similar to $e = 1$ in airplane wings; however, since a propeller blade spins, the velocity components vary a lot along the span, resulting in an excessively large propeller root which was deemed unfeasible for the given design. As such, a more typical shape was chosen that does not provide uniform lift distribution - while this may not sound ideal, it ensures that the design is actually manufacturable and keeps the chord length down to reasonable values. For the definition of this chord, various commercial propellers served as reference and the spanwise chord distribution was recreated using 7 anchor points and SciPy's cubic spline tool[2]. The baseline was defined to be 28 mm wide at its maximum extent, along with a 11 mm blade tip - these values will be iteratively scaled in subsequent steps, according to thrust needs. Moreover, the aerodynamic profile of the blade is set to start at 40mm from the axis of rotation, or at about

[2]SciPy documentation. 2026. https://docs.scipy.org/doc/scipy/reference/generated/scipy.interpolate.CubicSpline.html

20% of the propeller radius - this is based on these same commercial references and strikes a balance between lift and drag generation of the propeller. Furthermore, Blade Element Momentum (BEM) theory is utilised to calculate the local induction. This yields the induced velocity of the air pulled through the rotor disc - a value required to determine the local velocity vector for the spanwise Reynolds number evaluation. Directly related to the induced velocity, the induced angle of attack α_i is also determined and represents the angle between the free-stream air and local airflow (downwash) around the blade. An important consideration here is that the induced velocity calculation assumes a small vertical separation distance, as if the vertical separation were large enough, far stream acceleration of the lower blade induced velocity would result in a much elevated velocity component for the upper blade, changing its aerodynamic properties. However given that this effect is only pronounced multiple chord length distances from the blade, it can be safely neglected.

With all velocity components, as well as a preliminary chord size known, the Reynolds Number can straightforwardly be determined for any given point along the span of the blade. Next, for a given Reynolds Number, the BigFoil dataset is queried to find Re closest to the specified spanwise Re - this discretises the data further but is deemed acceptable given minor differences between airfoil characteristics for the considered Re range. Next, based on the specified Reynolds Number as well as considered airfoil, the ideal operating point needs to be determined. This can be done in many ways by use of a given Figure of Merit (FoM) - for aircraft, typically C_l/C_d is taken for maximal range and $C_l^{1.5}/C_d$ for maximal endurance. Although the logic varies slightly for propellers, C_l/C_d was chosen as the driving Figure of Merit to find the optimal operating point. While a different FoM may have yielded better results, this requires choosing it such that it actually does, which requires expert knowledge and is deemed out of scope. Following the determination of the FoM, the program scans the given airfoil data at the specified Reynolds Number to find the angle of attack (AoA) associated with the maximum C_l/C_d ratio - this angle of attack is defined as the design AoA, α_{design}.

Finally, with α_i, α_{design} and ϕ known, the spanwise twist of the blade can straightforwardly be determined by summing the 3 contributions. Hence, propeller twist determined as part of this calculation ensures that the airfoil operates at the maximal C_l/C_d for every point along the span of the blade.

However, as pointed out above, this calculation is not a simple one-way calculation as various parameters depend on each other - hence, the full loop has to be run iteratively. First, the total thrust is calculated for the initial guess of the spanwise chord distribution - since the required thrust can be determined from the drone weight and θ, the ratio T_{actual}/T_{req} is defined. If this ratio is below 1.0 it means that the chord is under-sized and needs to be enlarged, and if it is above 1.0 it is over-sized and needs to be reduced in size. While the spanwise chord length can be directly scaled with this ratio, this can lead to a "ping-pong" effect of over- and under-sizing the chord, resulting in excessively long computation times. Hence, a damping factor is introduced to gradually introduce the new chord length and ensure convergence - for this, 15% of the old value and 85% of the new value proved to be most computationally efficient. With the new chord known, the various angles and propulsive coefficients are re-calculated and fed back into the program - this is done up to convergence of all values to within 1% and typically takes 6 to 9 epochs, or 0.5 s. At the end of this iterative loop, the propeller geometry is fully defined and the design is concluded - this is done for all 6351 airfoils.

6.3.3. Flight Envelope Sweep

Having defined the propeller geometry based on its operational point, its propulsive characteristics for a given flight envelope need to be determined. This is done as to quantify its performance over a broader spectrum, which will in turn drive the selection of the final propeller configuration. As such, for each airfoil the geometry found in Section 6.3.2 is kept constant and evaluated at varying forward velocities - this forms the flight envelope and is defined from 0 to 50 km/h. The lower bound represents hover flight whereas the upper bound of 50 km/h aims to evaluate the propeller's performance at speeds vastly exceeding the operating speed to check for effects such as windmilling, which is when the flow hits the blade at such a shallow angle that the thrust becomes negative and the drone crashes. Ideally, windmilling would not occur within the flight envelope, and if it does, the corresponding airfoil should be either rejected or the propeller geometry adjusted.

By varying the speed at which the configuration is tested, several key parameters are affected - the main ones being the thrust, pitching angle and RPM. As these parameters change, the resulting inflow angle and induced AoA change - given that the blade's twist is fixed according to its design point in Section 6.3.2, the effective spanwise AoA can straightforwardly be determined. Based on this AoA, the lift and drag can be determined which, integrated over the blade, yield the thrust and power coefficients.

Since the RPM depends on the latter two values, a new RPM is calculated and the loop repeated until convergence of the RPM to within 0.1%. Similarly to what was done in Section 6.3.2, a damping factor of 0.33 was defined, meaning the new RPM is made up of 33% the old one and 67% the new one. While lower

damping values would have been computationally faster, convergence was not guaranteed, hence the value of 0.33 was taken.

Finally, this loop is applied to a given amount of points in the considered velocity range - for comparison's sake, 25 steps (i.e., one every 2 km/h) were deemed sufficient to make accurate decisions in the later design process.

6.3.4. Designing for Blade Vortex Interactions

With the full operational envelope defined for all airfoils, a first rejection criterion is defined before moving to the culling phase, namely that of BVI. As discussed in Section 6.2.1, avoiding BVI is essential when creating a low noise propeller, hence this is a killing criterion, meaning that if a blade has BVI then it will be rejected.

In order to predict BVI, a simple model is employed - given a thrust level and corresponding forward velocity, all velocity components normal to the propeller plane of rotation can be computed. Since the upper blade is lagging, the point of interest is if the vortices from the upper blade clear the lower blade before it returns to that position, i.e., travels 330°. Since the vortices travel down in the direction of the accelerated air at the total normal velocity seen by the propellers, the BVI avoidance calculation is simply checking if the air travels downstream far enough such that the distance covered exceeds that of the vertical blade separation distance in the given period of time before the lower blade travels the specified $360 - \gamma$, or 330° for $\gamma = 30°$.

This condition is checked for both the design point velocity of 30 km/h and hover condition. Although the RPM is lower for hover conditions, this is generally the most critical condition since the forward normal velocity component is 0 m/s so the vortices are travelling downstream slower than at advanced flight. Given the vertical separation distance between the blades was set to $0.8c_{max}$ in Section 6.3.1, the required vortex travel distance varies depending on the airfoil used. Furthermore, low RPM at hover flight is desirable as it essentially gives the vortex more time to cover the same distance.

6.3.5. Output Filtering and Final Design Criteria

Following the methodology from the sections above, the propeller geometry can be determined for any of the 6351 different airfoils, along with a critical BVI interaction condition. Given that the flight envelope sweep is approximately four times more computationally expensive than the actual propeller sizing script, it makes sense to try and define first rejection mechanisms between these stages in order to skip the sweep if the propeller is deemed out of bounds. In order to do this, a condition was implemented that extracts the maximum chord length from the converged propeller profile - if c_{max} ended up scaled by more than 80% compared to the reference defined in Section 6.3.2, the design point is rejected and the next case computed. This corresponds to a maximum chord length of 50 mm. This stage saw a bit over 55% of the airfoils rejected, leaving 2820 valid options for which the full flight envelope sweep from Section 6.3.3 is performed.

Given the 2820 valid design points, further rejection is needed - since some propeller configurations did not fully converge in the sweep, the next logical step is to eliminate those that did not. This is done by checking if, between hover and operational flight speed, at least 80% of the points converged - if it is below the threshold, the option is eliminated as it would lead to inaccurate weighting criteria in the final step. Additionally, the efficiency threshold at the operational velocity of 30 km/h was set to 50%, meaning that any airfoil with efficiency below 50% is rejected. Finally, some minor manual culling is performed on results that have discontinuities.

With all of the steps above implemented, 485 high-quality design options remain. Given the sheer number of options remaining, a weighting criterion is defined in order to help with the final choice. For this, 4 criteria and their respective weights are defined:

- **Blade mass (35):** Blade mass drives the total mass of the drone, but more importantly has a huge impact on controllability, as discussed in Section 6.2.1. Furthermore, an increase in blade weight requires more counterweights, resulting in a snowballing effect - hence this criterion was given the highest weighting, with lower mass being better.
- **Average efficiency (30):** The second highest weighted criterion is that of average efficiency, taken from 0 to 30 km/h. This is especially important as it characterises the efficiency not only at the operational speed but on the full flight envelope, with higher efficiency being better.
- **Design efficiency (20):** As most time will be spent at the design point velocity, this specific efficiency is given a weight of 20.

- **Hub height (15):** Directly linked to the maximal chord size, the hub height was given a relatively low weight of 15 as it is not critical, but helpful in keeping the propeller size down as well as generally ensure less BVI.

With this weighting formula defined, the final design can be easily determined and is presented in Section 6.3.6. Moreover, a positive sign regarding correct implementation of the method is that all top-weighted airfoils look extremely similar, meaning that a specific profile is always coming out on top, namely thin and high camber ones. Given the starting point of 6351 airfoils, this is believed to be the optimum for the considered design.

6.3.6. Final Design

The final propeller design uses the MID 118 airfoil, a thin, high-camber airfoil that excels in relatively low Re regimes. As expected, the airfoil strikes a balance between all weighting criteria, overall scoring the highest with a weighted rank of 15.3. An overview of the weighting parameters and associated rank is given in the table below:

Table 6.2: Weighting parameters and global rank overview for the MID 118 airfoil

	Design ETA (20%)	Average ETA (30%)	Blade Mass (35%)	Hub Height (15%)
Value	0.6237	0.1737	5.8 g	30.4 mm
Rank	8	8	11	50

Note that for this phase of the design, the upper and lower blade are given the same ideal 'clean-air-pitch', resulting from the simplified α_i assumption in Section 6.3.2. While a fully optimised vertically offset propeller would have different pitching angles for each blade, quantifying the slipstream effect in order to accurately determine this ideal pitch is deemed out of scope as it requires advanced knowledge about all interaction effects involved, which are simply not known at this stage given the innovative design. More on this in Section 6.5.

Finally, an overview of various propulsive characteristics for MID 118 is given in the comprehensive multi-plot figure below Figure 6.1, along with the lift and drag as well as C_l/C_d curves for $Re = 100\,000$, as provided by BigFoil.com Figure 6.2.

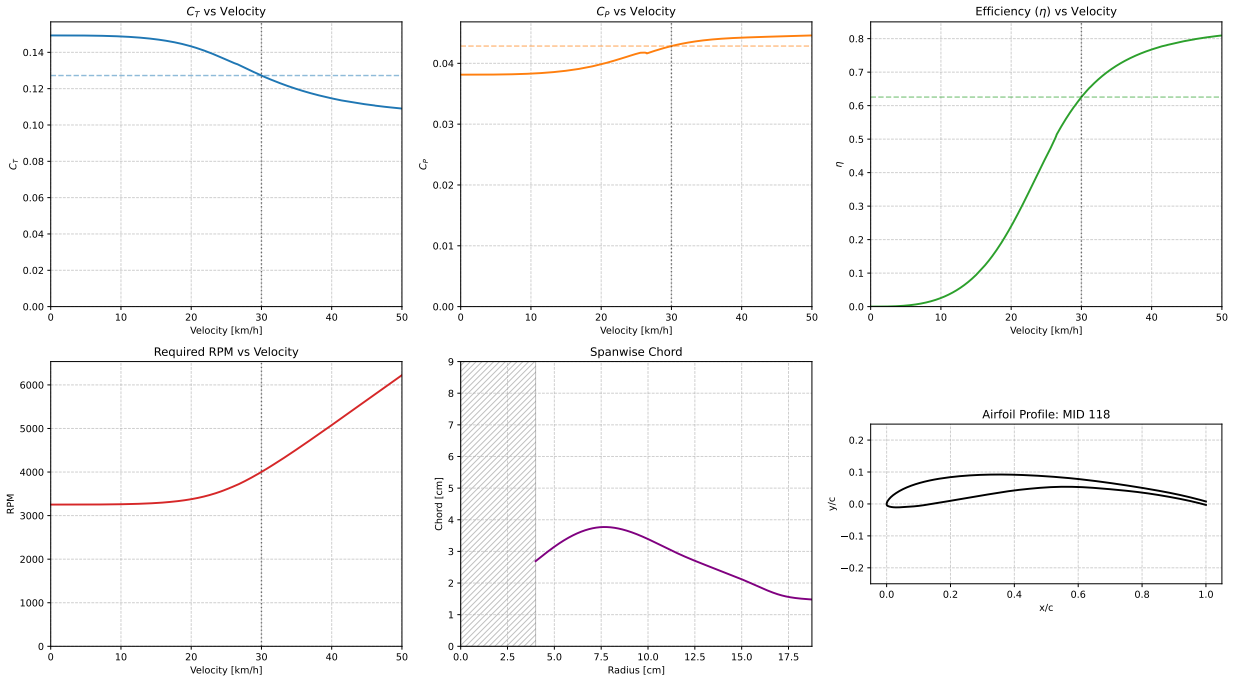


Figure 6.1: Overview of propulsive characteristics of the MID 118 propeller

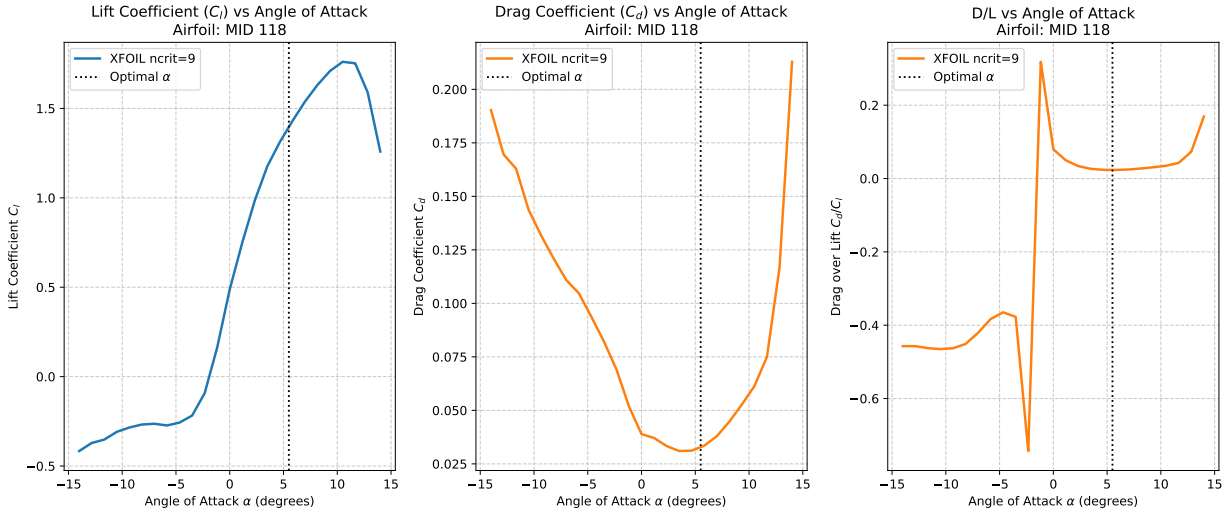


Figure 6.2: Lift, Drag and Lift/Drag curves for MID 118

Finally, a table with the key geometric parameters is given below. As defined earlier, the starting point is 40 mm from the rotation axis - with the first 40 mm being purely structural support - and the propeller is 187.5 mm in radius. The table splits the aerodynamic profile into 10 segments and lists the key parameters, i.e., the spanwise position, chord length and pitch. As the pitch is traditionally defined at $0.75r_{prop}$, this propeller is roughly the equivalent of a 14.8 × 7.0 inch propeller.

Table 6.3: Geometric definition of the MID 118 propeller blade at 10 equally spaced spanwise stations

Station	Radius [mm]	Chord [mm]	Twist [deg]
Sec 1	40.0	26.9	27.4
Sec 2	56.4	33.9	23.0
Sec 3	72.8	37.5	17.7
Sec 4	89.2	36.5	15.7
Sec 5	105.6	32.3	14.0
Sec 6	121.9	27.8	12.6
Sec 7	138.3	23.9	11.6
Sec 8	154.7	20.0	10.7
Sec 9	171.1	16.1	9.9
Sec 10	187.5	14.8	9.4
Calculated minimum vertical offset (hub height): 30.4 mm			

Next, the propeller needs to be balanced - this is done using Centre of Gravity (CoG) calculations extracted from CAD and adding an adequate counterweight as to shift the CoG to the axis of rotation. As balancing requires a leverage arm on the opposite side of the propeller as the blades, some vortex interactions are expected for an excessively long arm that protrudes into the aerodynamically defined spanwise blade position, i.e., ≥ 40 mm. Hence, the objective is to keep the leverage arm below that threshold while also not adding excessive weight, striking a balance between controllability and propulsive efficiency. Furthermore, the counterweight arm should ideally be aerodynamically shaped in order to minimise parasitic drag as well as perturbations that could lead to BVI. With this in mind, for the considered propeller geometry and material, balancing the blade resulted in a 20 g counterweight placed at 32 mm from the rotational axis. This, according to CATIA brings the propeller's total mass from 65 g to 85 g, with the hub contributing most of the mass.

Finally, two key conditions from Section 6.2.1 and Section 6.2.2 need to be checked for compliance. The first condition is that of roughness in uneven propellers, which typically manifests when multiple frequencies simultaneously fall within the roughness range of 20 Hz to 300 Hz. For the most critical condition, hovering at 3200 RPM, the fundamental pair-blade passing frequency lies around 53 Hz. However, for a blade spacing angle $\gamma = 30°$, the individual blade passing frequency is shifted to 640 Hz. Because this secondary frequency

is pushed well outside the critical 20 Hz to 300 Hz interval, the two frequencies do not overlap within the roughness band, successfully avoiding the irritating acoustic interaction. The second condition is that of whether adding a loop is aero-acoustically beneficial or not - for this, ideally the earlier defined formula would be used. However, given large ranges of possible Strouhal values depending on the exact operating condition [19], this is deemed not feasible at the current design stage as it would require detailed CFD analysis as well as experimental testing of the specific airfoil used. Instead, an experimental route is identified as the most efficient way of verifying the loop's noise reduction capabilities and will be conducted at the end of the detailed design phase. If deemed impractical or wildly inefficient, the loop is simply removed and a non-closed-loop propeller geometry implemented.

With all these considerations in mind, a 3D model of the propeller can be created, which is shown below and merges all of the ideas explored in this chapter.

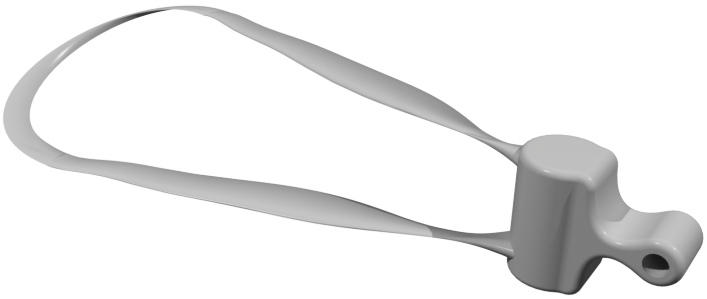


Figure 6.3: 3D model of the designed propeller

As defined in the preliminary design phase, the propeller is made from carbon-fibre-filled nylon for which generally a higher carbon-fibre content is desirable as it mitigates nylon's hygroscopic properties. The exact carbon-fibre content is left to be determined in the next design iteration, but is expected to lie in the 20% to 30% range. Furthermore, recent research conducted by F Bandinelli et al.[8] explored the recyclability of carbon-fibre reinforced nylon and found it to be fully recyclable, validating compliance with RQ-ENV-01-PRP-01.

6.4. Verification and Validation

Table 6.4: Compliance matrix for the propulsion subsystem requirements. Status: C = compliant, PC = partially compliant, NC = non-compliant, O = open/TBD.

Requirement ID	Method	Status	Evidence and Remarks
RQ-ENV-01-PRP-01	I	C	The chosen material for the propellers is carbon-fibre-filled nylon, which literature confirms is fully recyclable.
RQ-WOS-01-PRP-01	T*	O	Cannot be physically tested without a manufactured prototype. Literature suggests reduced tonal noise, but precise dBA levels require anechoic chamber testing.
RQ-GWH-01-PRP-01	T*	O	Cannot be tested without a physical prototype inside a cold-storage chamber, though motor data sheets indicate normal functionality.
RQ-GWH-02-PRP-01	D*	O	Requires a manufactured prototype for physical demonstration.
RQ-GWH-03-PRP-01	A	C	The BEMT sizing script confirms that the subsystem can successfully provide a thrust-to-weight ratio of at least 2.0.
RQ-RLB-01-PRP-01	A	C	The required continuous operation timeframe is 1 year (8760 hours). While specific manufacturer data is unavailable, degradation analysis based on comparable industry-standard BLDC motors indicates an expected bearing lifespan of approximately 10 000 hours, satisfying the requirement.
RQ-MFC-01-PRP-01	I	C	All selected off-the-shelf components (e.g., motors) have verified supplier lead times of under 4 weeks. Custom propellers will be manufactured in-house.

6.5. Limitations and Recommendations

The custom BEMT methodology successfully designs and sizes a highly-optimised, high-efficiency propeller for the specified operational conditions. However, due to the atypical, asymmetric closed-loop nature of the design, several aerodynamic simplifications were made. The limitations of the current design iteration, along with corresponding recommendations for future development, are outlined below:

- **Simplified Slipstream Interactions and Constant Pitch:** Currently, both the upper and lower blades are assigned the same ideal 'clean-air' pitch distribution. This relies on a simplified induced angle of attack (α_i) assumption that neglects far-stream acceleration, while also ignoring the slipstream interaction between the two blades. In reality, the lower blade constantly operates within the accelerated, turbulent slipstream of the upper blade, and the upper blade operates within the accelerated induced velocity stream of the lower blade as well as its potential flow - effects which can greatly alter the local dynamic pressure and effective angle of attack.
  Recommendation: Future iterations should utilise Computational Fluid Dynamics (CFD) or advanced free-wake vortex models to accurately quantify the aerodynamic interactions between the blades. The BEMT script should then be updated to generate a differential pitch distribution, where especially the lower blade's twist is optimised for the 'dirty' air it operates in.

- **Acoustic Verification of the Closed-Loop Geometry:** While the vertical offset mitigates BVI and the uneven spacing shifts the blade passage frequency out of the human sensitivity range, the exact aero-acoustic benefit of the closed tip-loop remains theoretical. Analytically verifying this noise reduction using Strouhal numbers was deemed unfeasible at this stage, given the wide variation of Strouhal values across different operating conditions and complex geometries.
  Recommendation: As opposed to attempting highly complex acoustic CFD simulations, experimental verification is recommended as the most logical path forward. Prototypes of the MID 118 design should be manufactured and tested in an anechoic chamber to quantify the actual noise footprint. If the loop proves aerodynamically or acoustically inefficient in practice, it should be removed in favour of a traditional, non-closed-loop asymmetric configuration.

- **Far-Stream Acceleration Assumptions:** The induced velocity calculations assume a minimal vertical separation distance between the blades. If structural or acoustic testing dictates that the vertical hub offset must be significantly increased, far-stream acceleration of the air could result in an elevated velocity component for the upper blade, invalidating the current aerodynamic properties.
  Recommendation: If the required hub height increases significantly beyond the currently calculated 30.4 mm, the BEMT script must be modified to include stream-tube contraction effects between the upper and lower rotor planes.

- **Body Drag and System-Level Kinematics:** To iteratively solve for the advance ratio, forward normal velocity, and pitching angle, the drone's effective surface area and drag coefficient (C_d) were treated as static estimates. Furthermore, the 30 km/h design speed, while aerodynamically sound, may exceed the dynamic capabilities of the control subsystem.
  Recommendation: Once the first design iteration is completed, exact C_d profiles mapped against pitch attitude should be extracted via wind tunnel testing or CFD. These profiles should be fed back into the BEMT script. Because the script is fully parametric, a new optimal propeller geometry can directly be generated once all design departments agree on a realistic operational top speed.

- **Figure of Merit:** The design optimises for maximum aerodynamic efficiency in forward flight by targeting the maximum C_l/C_d ratio. While this generally ensures the greatest distance travelled per unit of energy, alternative Figures of Merit (such as $C_l^{1.5}/C_d$ for maximum endurance) were excluded to bound the design scope.
  Recommendation: If mission requirements pivot from a "range-critical" profile to a "hover-critical" profile (e.g., spending more time stationary during payload pickup/dropoff) because of routing limitations or prolonged pickup sequences, the BEMT script's optimisation criterion should be updated to target $C_l^{1.5}/C_d$ to better align with the mission profile.

- **Parasitic Drag of the Counterweight:** Following the analysis in Section 6.3.6, a 20 g counterweight must be placed on the side opposing the blades. The current BEMT script strictly evaluates the aerodynamic efficiency of the lifting blades and does not mathematically simulate the parasitic profile drag generated by this counterweight spinning at high RPM.
  Recommendation: While the counterweight is shaped to minimise its wake, future detailed design iterations should use CFD to quantify its exact drag penalty. This parasitic torque requirement should

then be added to the BEMT performance sweep to yield a more accurate efficiency profile for the entire rotating assembly.

- **Dynamic and Aerodynamic Imbalance of Asymmetric Rotors:** While the static Centre of Gravity (CoG) was successfully aligned with the axis of rotation using a 20 g counterweight at 32 mm, this does not constitute a fully dynamically balanced system. Because the moment of inertia scales with the square of the radius ($I \propto mr^2$), the extended blades possess a vastly larger rotational inertia than the compact, hub-mounted counterweight, resulting in principal axes of inertia that do not align with the rotational shaft. Furthermore, the lifting blades generate significant aerodynamic thrust on one side of the disc, while the opposing counterweight generates none. At high operational speeds (4000 RPM), these uncompensated inertial and aerodynamic moments will introduce severe, high-frequency oscillatory bending loads to the motor bearings and structural arms.

  Recommendation: Future iterations must execute a full dynamic balancing (two-plane balancing) in CAD by redistributing the counterweight mass further outward or altering its geometry to align the principal axes of inertia. Additionally, a dynamic structural simulation must be performed to quantify the vibratory loads induced by the asymmetric aerodynamic thrust, dictating the necessary stiffness and damping requirements for the drone arms and motor mounts.

7

Power and Energy Replenishment

The power and energy replenishment subsystem is essential for the proper operation of the DRIVE-FC system. It is a critical system, since the batteries are one of the main factors in determining the drone's flight performance. In previous phases of the project, the power source was selected to be batteries swapped into and out of drones. The concept of powering drones with cables was discarded due to technical risk and operational limitations. With the other remaining option of batteries attached to drones more permanently, drones would be out of service whenever their batteries are charging, leading to a large idle time and a larger number of drones and ERS being necessary. This decision does, however, carry some additional complexity, cost, and technical risk. Mainly, the ERS represents a major risk in the functioning of the system (TR-AIT-01), as existing examples of autonomous battery swapping are few and limited. This section will describe the design of the power subsystem, as well as the ERS.

7.1. Requirements

The function of the power and energy replenishment system is to ensure the drones are provided with sufficient energy and power to fulfill their mission. Furthermore, to comply with sustainability requirements, this needs to be done fully electrically. The subsystem requirements imposed on this subsystem are shown in Table 7.1.

Table 7.1: Verification plan for power subsystem requirements

Requirement ID	Requirement	M	Method description
RQ-ENV-01-PWR-01	The power subsystem shall use recyclable hardware.	A	Determine recyclability of materials through literature.
RQ-ENV-02-PWR-01	The power subsystem shall be fully electric.	I	Check that the power subsystem only delivers electrical energy.
RQ-GWH-01-PWR-01	The power subsystem's hardware shall be functional in Cold-storage conditions.	T*	Test power subsystem functioning in cold storage zone.
RQ-GWH-02-PWR-01	The power subsystem's hardware shall be functional in Ambient conditions.	D*	Demonstrate power subsystem functioning in ambient storage zone.
RQ-EFF-01-PWR-01	The power subsystem shall provide sufficient energy to ensure continuous operation for at least 15 minutes per drone charge.	A	Verify through calculation of both power draw and battery capacity.
RQ-EFF-01-ERS-01	The ERS shall provide enough charging and swapping capacity to continously fly up to 10 drones in either temperature zone.	D	Fly the required number of drones simultaneously and verify their batteries never deplete due to ERS unavailability.

Continued on next page

Table 7.1 – Continued from previous page

Requirement ID	Requirement	M	Method description
RQ-ATM-03-ERS-01	Energy replenishment procedures shall take no longer than 2 minutes.	D*	Demonstrate that the energy replenishment procedures do not take longer than required.
RQ-ATM-03-ERS-02	The energy replacement shall correctly perform at least 99% of battery swaps correctly.	T*	Repeatedly test battery swaps and verify they are done correctly.
RQ-ATM-03-ERS-03	The ERS shall function without human intervention outside of unexpected circumstances.	D*	Demonstrate that the ERS can perform battery swaps with no human intervention.
RQ-MFC-01-POW-01	All off-the-shelf power source subsystem components shall have a maximum verified supplier lead time of 4 weeks.	I*	Check with manufactures and distributors whether power subsystems are available within time frame.

7.2. Battery

As decided in the preliminary design phase, the batteries to be used are Lithium Iron Phosphate (LFP) batteries. These were chosen, since their significantly longer lifespan makes them advantageous over other battery chemistries from a cost perspective. Furthermore, they are very stable compared to other battery chemistries, minimising risk of a fire starting in the warehouse [38]. Finally, this chemistry has significantly lower emissions, uses less harmful and scarce materials in its construction, and is recyclable [38, 59] . This is particularly important since the large number of batteries used by the system makes the batteries a major source of sustainability concerns.

Following from the preliminary sizing, the batteries will consist of 24 cells, with 8 cells placed in series and 3 in parallel. The physical arrangement of the cells has changed from the preliminary design to reduce the height of the drone battery, and thereby make more efficient use of space in the drone body. The optimised layout is shown in Figure 7.1.

(a) Original layout **(b)** Space-saving layout

Figure 7.1: Comparison between original and optimised battery layouts. Not to scale.

7.3. Energy Replenishment Stations

To enable the swapped-battery concept for powering the drones, ERS are necessary to perform the battery swaps. The ERS are also responsible for charging batteries.

7.3.1. Mechanisms

Due to RQ-ATM-03-ERS-03, the ERS needs to be capable of autonomously swapping the drone's battery. To achieve this, the following steps need to be taken:

1. Line up drone with ERS mechanisms
2. Remove old battery from drone
3. Move old battery into charger

4. Remove new battery from charger
5. Put new battery into drone

Drone Alignment

The drone is aligned with the ERS by landing into a tapered surface that forces the drone to be in the correct position and orientation when its legs reach the landing surface. This is illustrated in Figure 7.2. The landing leg receptacles are mounted on two separate pieces to prevent a collision between the grabber and the ERS structures.

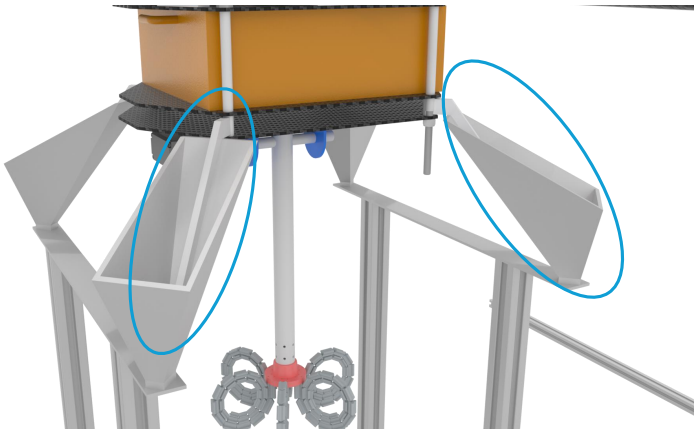
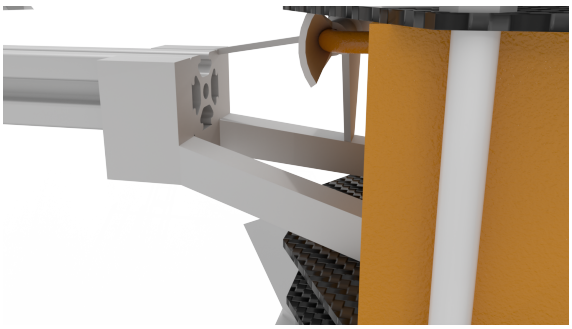


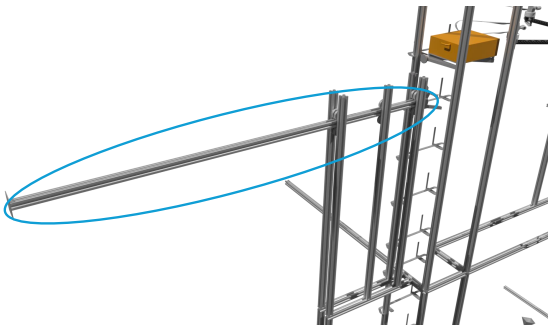
Figure 7.2: A drone aligned with an ERS

Battery removal and insertion

To insert and remove batteries, an arm is used that can extend to the drone and attach to or detach from the battery. This arm has a moving piece on the end that slides onto a pin on the battery to attach it to the end of the arm. Two other parts contact the battery to provide three points of contact that can together hold the battery safely.



(a) Picking arm attached to a battery in a drone

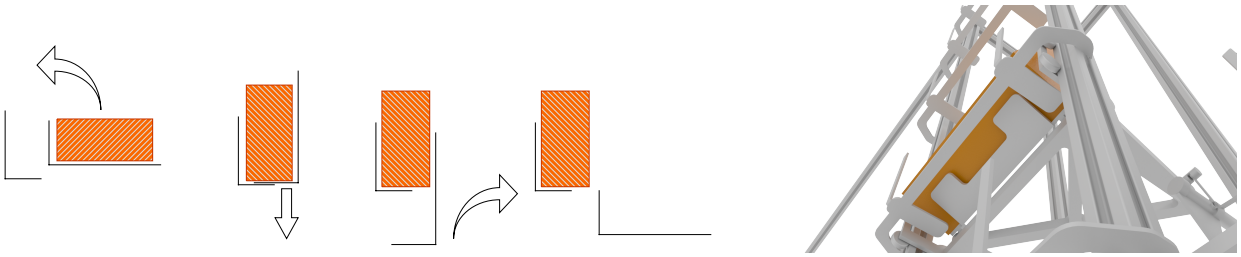


(b) Battery picking arm

Figure 7.3: Mechanism for moving battery in and out of drone.

Placement in Chargers

To place the batteries into chargers, a mobile platform is used that can transport batteries to chargers and insert them into the chargers. This is achieved by moving vertically and tilting the platform the battery is on. Once the battery picking arm has placed a battery on the platform, the platform is moved upwards so that when it tilts, the battery will be placed on the charger platform. These are shown with a drone landed in Figure 7.4.



(a) Illustration of how the moving and tilting platform moves a battery into a charger.

(b) Render of a battery being tilted into a charger

Figure 7.4: Illustration of mechanism for moving batteries into chargers.

7.3.2. Structure and Layout

The structure of the ERS consists mostly of T-slotted aluminium extrusions. These were chosen, since their design allows for very simple manufacturing and due to their broad availability. This will help reduce the cost of the ERS, and make redesigns much easier, which will be especially useful in the prototyping design phase. The profiles used are 20 mm × 20 mm, although it may be necessary to use larger profiles in the final design if loads end up being too much for this profile to handle.

Each ERS will use two identical swapping mechanisms, located on either side of the drone. The advantages of this are twofold. Firstly, this system can remove the old battery with one of the mechanisms, and put in the new battery from the other side, significantly speeding up swaps. It also allows for the ERS to be much less tall, since much less batteries need to be stored on top of one another.

In operation, one charging slot will always need to be kept free. This is because when the old battery is removed from the drone and needs to be put into a charger, there needs to be a free slot for it to be put into. With identical swapping mechanisms on both sides, only one of these needs an empty slot. The swapper without an empty slot will provide the new battery, and the swapper with an empty slot will remove the old battery. These roles will switch after each swap, as the previously empty slot will be filled and a slot on the other side will be empty.

7.3.3. Sizing

There are several main parameters of the ERS to determine to ensure it both meets operational needs and fits within the warehouse space.

- **Outer dimensions** need to be large enough to allow drones to enter, batteries to be stored during charging, and the battery swap operation to be performed. However, these should also be minimised to optimise warehouse space usage.
- **Number of ERS** needs to be sufficient to swap batteries at a rate that provides continuous power to all drones.
- **Number of batteries** needs to be sufficient to provide continuous power to the drone fleet.

Outer Dimensions

From the previously discussed design of the ERS, the outer dimensions of an individual ERS can be determined. The horizontal footprint is 4.55 m × 1.7 m, with a height of 1.48 m. Due to the large footprint, vertically stacking ERS to reduce the footprint of the system was deemed to be worthwhile. Theoretically, three ERS could be stacked within the height of the warehouse. However, a stack this high is not used as there are no more than two ERS needed in any given space. This is also beneficial as keeping mechanisms and batteries closer to the ground makes access for maintenance easier. The dimensions of a stacked ERS configuration are 4.55 m × 1.7 m × 2.94 m. The two ERS as will be present in both zones of the warehouse is shown in Figure 7.5.

Number of ERS

From analysis of results of the logistical simulation described in Chapter 13, it was determined that two ERS are needed for the full fleet of 15 drones. To enable the operational flexibility to allocate drones between the ambient and cooled zones of the warehouse, the decision was made to place two ERS in each zone. This will also allow for maintenance to be done to one or two ERS, depending on drone distribution. They will be placed against the wall of the warehouse in the extension of a human aisle, where they will not impede traffic.

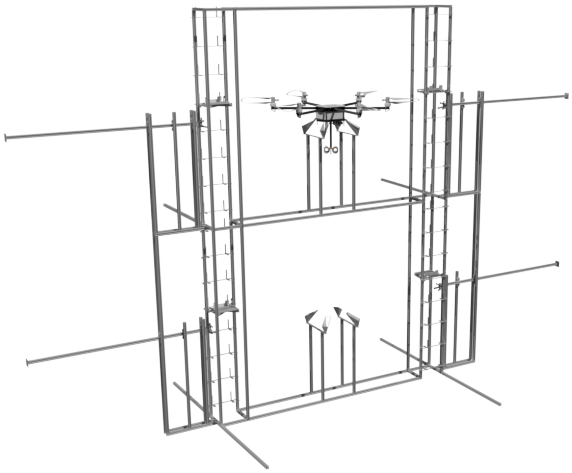


Figure 7.5: Final ERS layout to be used in the warehouse, consisting of two stacked ERS.

This location is also fairly close to the drop-off stations, such that drones can replenish their energy after completing a delivery.

Number of Batteries
The main advantage of the swapped battery power concept is the decoupling of number of drones and number of batteries, allowing batteries to charge without a drone needing to be down whilst they are doing so. However, the number of batteries is still related to the number of drones. For each drone there need to be 2.94 batteries charging at any time to be able to provide continuous power, due to charging being slower than discharging [27]. With 15 drones in operation, this leads to a total of 45 batteries charging. However, to account for operation of 10 drones on either side to enable operational flexibility, 30 batteries per side are needed for a total of 60. These are spread across four ERS, meaning there will be 15 batteries per ERS. An additional charging slot needs to be available as explained previously, meaning each ERS needs 16 charging slots, which will be divided into eight slots per side.

7.4. Verification and Validation

With the detailed design of the power and energy replenishment subsystem, verification of compliance with several of the requirements can be performed. Any requirments that require real-world testing or demonstrations cannot be performed at this stage, since no prototype hardware has been built. Where possible, simulation results have been used to verify requirements. Compliance with these requirements should be verified again once sufficient hardware has been built to perform real-world testing or demonstrations.

Table 7.2: Compliance matrix for the power subsystem requirements for localization. Status: C = compliant, PC = partially compliant, NC = non-compliant, O = open/TBD.

Requirement ID	Method	Status	Evidence and Remarks
RQ-ENV-01-PWR-01	A	C	All components are either reusable or recyclable.
RQ-ENV-02-PWR-01	I	C	No non-electric power sources have been used.
RQ-GWH-01-PWR-01	T*	O	Needs to be verified after building prototype hardware.
RQ-GWH-02-PWR-01	T*	O	Needs to be verified after building prototype hardware.
RQ-EFF-01-PWR-01	T*	C	At an average power draw of 427 W and a battery capacity of 123.6 Wh at 60% depth of discharge (DoD), flight time is 17.37 minutes. A DoD lower than this is not expected to be used as it has no significant benefit, so this is the worst-case scenario.
RQ-EFF-01-ERS-01	T*	C	Logistical simulation has verified that the number and performance of ERS are sufficient to swap batteries.
RQ-ATM-03-ERS-01	D*	O	A model of the ERS has shown that this can be achieved, although it still needs to be demonstrated with real hardware.
RQ-ATM-03-ERS-02	T*	O	Needs to be verified after building prototype hardware.
RQ-MFC-01-POW-01	I*	C/O	Verified for all components that have been chosen at this point in the design, but needs to be verified for the remaining components to be selected in further design work.

7.5. Limitations and Recommendations

There are still several elements of the power subsystems that need to be designed in full detail, and some complicating factors to account for.

The current ERS geometry does not consider the effects of deflection of the battery picking arm. This would result in the battery colliding with the bottom panel of the drone. To address this, the drone should land at a lower location in the ERS, such that the battery will end up slightly above the plate it should sit on. It should then be verified that the funnel for the battery pin tapers enough to allow the battery to be placed into the drone gently. Furthermore, the funnel should taper enough to ensure the arm can engage with the battery pin with a height difference caused by the arm not being deflected much without a battery attached.

To complete the final design of the power subsystem, several electrical circuits also need to be designed and integrated with software. On the drone side, the circuits that manage power draw from the battery and balance the load over cells of the battery needs to be designed. Furthermore, the electrical components of the ERS need to be designed in detail. This includes the battery chargers themselves, as well as the motors, sensors, and circuit responsible for the actuation of the ERS' mechanisms. The software to control the ERS' mechanisms also needs to be written.

Finally, the remaining open requirements need to be verified. The requirements that still require verification work are indicated in Table 7.2, along with the steps needed to complete verification of these requirements.

8

Grabber

A key component of the overall system is the ability to pick up objects. This chapter contains the relevant design ideas that were traded-off for the grabber subsystem, and then the more detailed aspects with more accurate sizing. In order to effectively assess the trade-offs and motivate the design, the relevant subsystem requirements are presented in Section 8.1. Next, the considered design options after the trade-off are presented in Section 8.2. Then the detailed design of the chosen subsystem is presented in Section 8.3. Lastly, the verification and validation, and the limitations and recommendations are presented in Section 8.4 and Section 8.5 respectively.

8.1. Requirements

The functions of the grabber are threefold. First, it must securely pick up the payload; second, it needs to protect the payload in transit, and lastly, to safely deposit the item at the drop off point. These functions are central to the functioning of the system as a whole. Based on these functions and other, larger requirements, the requirements for this subsystem are tabulated in Table 8.1.

Table 8.1: Verification plan for grabber requirements

Requirement ID	Requirement	M	Method description
RQ-GWH-04-GRB-01	The grabber shall be able to lift objects of size 400x250x50 mm.	D*	Demonstrate grabber lifting this size of box.
RQ-GWH-03-GRB-01	Each tentacle of the grabber shall be able to withstand a maximal tensile load of 20 N.	T*	Test maximal tensile load by means of a tensile-stress test.
RQ-RLB-03-GRB-01	The grabber subsystem shall not impede emergency landings.	D*	Demonstrate an emergency landing.
RQ-GWH-04-GRB-02	The grabber shall be able to access all points of the crate at the pickup location.	D*	Demonstrate picking up when payload positioned at random positions in crate.
RQ-QLT-01-GRB-01	The grabber shall exert no more than 30N of force on any carried item.	T	Test exertion force with compression load cells.
RQ-ENV-01-GRB-01	The grabber subsystem shall use reusable hardware.	I	Check properties of grabber hardware.
RQ-GWH-01-GRB-01	The grabber subsystem shall be functional in Cold-Storage conditions.	T*	Test graber subsystem functioning in cold storage zone.
RQ-GWH-02-GRB-01	The grabber subsystem shall be functional in Ambient conditions.	D*	Demonstrate grabber functioning in ambient storage zone.

Continued on next page

Table 8.1 – Continued from previous page

Requirement ID	Requirement	M	Method description
RQ-RLB-01-GRB-01	The grabber subsystem shall be capable of operating for at least 5 continuous years.	A*	Analyze degradation of grabber subsystem, and compare with similar applications.
RQ-MFC-01-GRB-01	All Off-The-Shelf grabber subsystem components shal have a maximum verified supplier lead time of four weeks.	I	Check with manufacturer and distributors whether components are available within time frame.
RQ-MFC-02-GRB-01	The grabber subsystem shall be designed such that a single trained technician can fully integrate onto the drone in under one hour.	D*	Trained technician demonstrates that the grabber subsystem can be fully assembled within time frame.
RQ-DCM-01-GRB-01	The grabber subsystem shall be integrated into design such a way that it can be disassembled by a single technician in under 45 minutes, using standard hand tools.	D*	Trained technician shows that the drone grabber subsystem can be disassembled with standard hand tools within the time frame.

8.2. Design Options

Based on the requirements in Table 8.1, many viable solutions were thought of during preliminary design. A trade-off was conducted to decide which idea to follow through on. The key ideas to be traded off were the main mechanism to carry the payload, and how that is attached to the drone.

Main Mechanism

For the main mechanism, the key criteria for the trade-off were its weight, payload diversity, and speed of interaction. These criteria were decided based on the requirements for the subsystem, and the dependencies on other subsystems, such as weight on propulsion. The most viable options after this were cabled tentacles, magnets, and a flexible claw. Each option is explained further below.

Cabled Tentacles

The cabled tentacles refers to using a 3-D printed limb actuated by 3 cables, as outlined in Wang et al. [54]. The proposed configuration is to use multiple arms actuated by two motors. As a whole, this idea is simple, lightweight, and cheap. It is also the most capable of lifting the largest variety of items.

Magnets

Magnets perform exceptionally well across all criteria. It is very fast at picking up items, and also reliable. It has a small footprint and is lightweight. The only (and very important) drawback is that it only works on some metallic objects. This could still be feasible considering many grocery items have such surfaces, like cans and jar lids, but it would require a significant reduction in scope.

Flexible Claw

A flexible claw is inspired by the mechanism behind a claw machine at an arcade, but with a modified 'finger' structure to be softer, and more form-fitting. Similar to the tentacles, it performs well for a large variety of payloads, but it is more expensive to realise. It is also predicted to be heavier. There are no notable benefits that would outweigh these deficits compared to the tentacles.

8.3. Detailed Design

After choosing the ideas to follow through on, the grabber system needs to sized more accurately for its purposes. The first step in doing so is determining how long the tentacles need to be to carry certain items. Then, the rest of the geometry can be designed around it. It is important to note that the tentacles are inspired by logarithmic spirals [54], so a lot of its parameters are calculated by numerical approximation, as the relations in literature are non-linear and not easily reversible to isolate those parameters.

Tentacle Length

The length is determined by first choosing the worst geometry the grabber may need to pick up, and then the worst case scenario for picking up that object. The length of tentacle required for those conditions are chosen in order to be conservative in the design, and allow for errors in numerical calculations.

The tentacles excel at picking up uneven surfaces as they can make use of the grooves and irregularities to increasing the effective surface to grip [54]. As a result, most organic shapes are very easy to pick up. Approximating objects using basic 3-D shapes, a large flat cuboid would perform the poorest by the tentacles. There is a large flat surface with no usability, so the tentacles must be longer to reach over the edge to even begin to grasp the object.

The dimensions of this shape can be assigned the dimensions l, w, and h for length, width and, respectively. To simplify the calculations, it is assumed that for a stable pick up, at least 2 tentacles need to grasp opposing sides. This immediately eliminates any solutions with four or less tentacles as they cannot meet this criteria. An implied consequence of this is that the length of the cuboid is irrelevant for this length estimate, and can be considered to be significantly larger than the width, $l >> w$. As a consequence, any solution that optimises for a more square cross-section is neglected.

For the worst picking scenario, one arm will be perpendicular to the edge of the payload as illustrated in Figure 8.1. If the grabber is rotated any amount, the second arm required for a stable grip will not need to be as long, so it is a conservative estimate. The perpendicular arm's required length will increase, but the neighbouring arm can be shorter (imagine a counter-clockwise rotation in Figure 8.1, the arm on the other side is not visualised). This scenario assumes that there is no knowledge of the relative rotational position of the grabber to the payload. From this, the equation for the shortest length of tentacle is as follows:

$$L = k \cdot \frac{w}{2\cos(\phi)} + h \tag{8.1}$$

where k is a multiplying factor for the length to the edge to allow for a stable grip underneath, and ϕ is the angle between the tentacles. For a conservative estimate of how much length underneath the box is required, k is assumed to be 1.1.

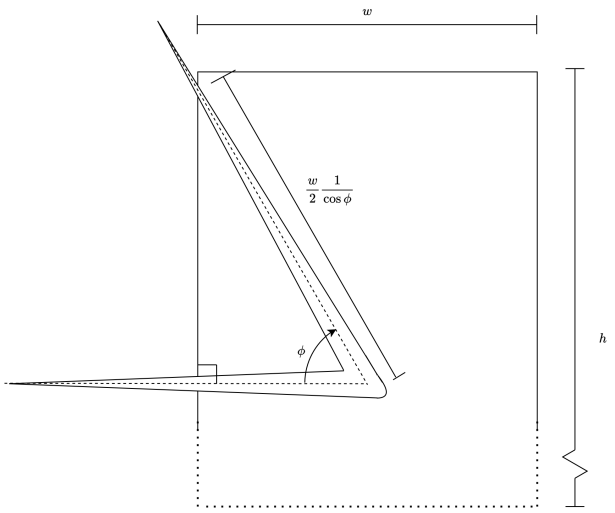


Figure 8.1: Illustration of the worst-case picking scenario. ϕ is the angle between adjacent tentacles and w, h are the dimensions of the payload.

Tentacle Parameters

The tentacles are defined by three parameters, a, b, and θ_{lim} [54]. The first two are arbitrary parameters, representing mathematical constants that describe a logarithmic spiral as $r = ae^{b\theta}$, while θ_{lim} is the maximum angle the spiral will extend to. Based on these parameters, the tentacle is split as a spiral at 30° intervals and 'uncurled' for the core design of the tentacles [54], refer to Figure 8.2 for visualisation. To limit the number of segments to the tentacles, θ_{lim} was limited to 5π, so there will be 30 sections per tentacle.

Now that there are only two variables, two relations can constrain them. The first is the length of the tentacle,

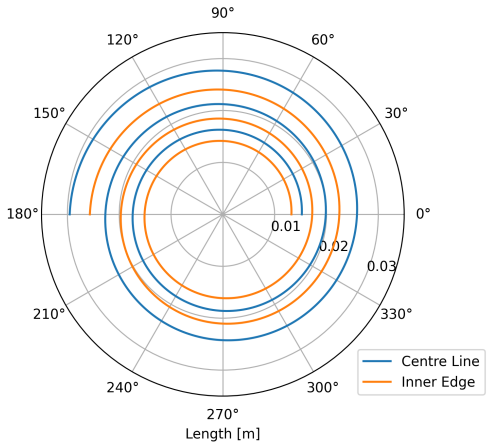


Figure 8.2: Mathematical definition of the tentacle, with angle marks where the design is split into rotating segments. The centre of the tentacle is marked in blue, while the inner edge is marked in orange. The 'uncurled' position is attained with the blue line is perpendicular to the reference.

which is defined by Wang et al. [54]:

$$L(a, b, \theta_{lim}) = \frac{a\sqrt{b^2+1}}{b}(e^{b\theta_{lim}} - 1) \tag{8.2}$$

The second constraint was decided to be the width of the tip. Initially, the idea was to use the yielding limit of Thermoplastic PolyUrethane (TPU) to size the width. TPU is the chosen material as it is highly elastic and large deformations are required from the tentacles [54]. For a conservative estimate, it was assumed that half the tentacles are not load-bearing, and the tip of the tentacle (the smallest cross-section) would be carrying the force through it. In reality, the load-bearing elements are the cables that actuate the tentacles, so this would drastically overestimate the required thickness. Despite this, the required thickness was far too low to be manufacturable. Therefore, the constraint was set to a 5 mm thickness at the tip to ensure manufacturability in a 3-D printer, rather than the yielding point of the material.

To keep track of the mass budget the tentacles are responsible for, an estimate for its volume was required. This volume is approximated by integrating the swept circle of the radial difference along the major axis of the tentacle (see Figure 8.3). This is a conservative estimate as the real tentacle would have cutouts that reduce its mass, but this estimate scales well with the spiral parameters, thus it is a very useful indicator of its mass.

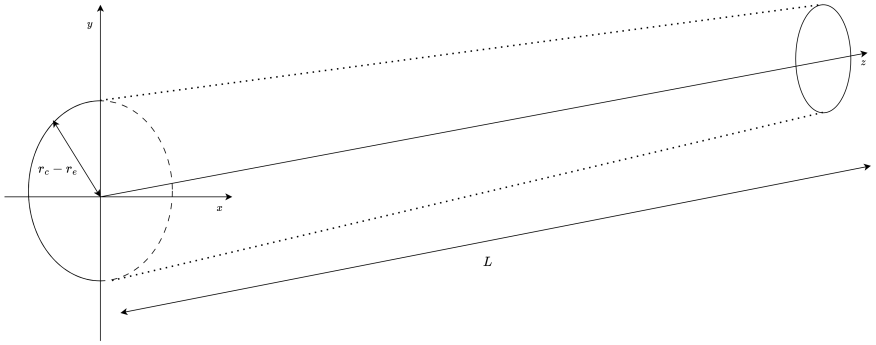


Figure 8.3: Visualisation of integral used to estimate the volume of each tentacles. r_c and r_e refer to the curves that describe the centre and edge of the spiral, as in Figure 8.2

It was noticed that very minor changes in tentacle length resulted in drastic changes in the tentacle mass, ranging from around 10 g to 1.5 kg. In order to avoid having unnecessarily heavy tentacles, a new algorithm was set up (based on the trust-region constrained [41] to optimize the weight of the tentacles, while imposing the constraint on the tip width and tentacle length, such that it can be longer or wider than required if it results in a lower mass. After this optimisation, the values for the parameters of a and b were decided. With these parameters fixed, a 3-D CAD model of the tentacle was created. A visual is present towards the end of the section in Figure 8.4

Grabber Interface

The interface itself was not the focus of this report, but it was still designed to be functional. An arm of length 20 cm before the tentacles is necessary to allow the grabber to reach the bottom of the crates the items will be in. Due to this length, the arm needs to be able to fold away for emergency landings, as the landing legs would be a hazard for mobility if they were that long.

A carbon fibre tube was selected as the arm for its strength to weight efficiency, and was checked for compliance in bending loads. To allow a degree of rotation, this tube is drilled to allow an 8 mm aluminium dowel to fit in. This dowel is then connected to a servo which is able to rotate the arm. A suitable servo was found based on the torque requirements of holding the arm stretched out while also carrying the heaviest payload: placing a load of 1 kg at a distance of 25 cm yields a torque of 0.25g Nm, so this lightweight servo [42] is suitable.

To allow this structure to be mounted, the dowel is placed in between two bent aluminium sheets with holes for both the dowel, and some bolts to the main structure of the drone. The thickness of aluminium sheet is decided to be 2 mm, to still allow tight radii in bending. The bolts used are 4 M4x10 bolts; these bolts are chosen as it is a standard size to use, and also capable of transferring the loads[31].

At the other end of the arm, there needs to be a structure to attach the tentacles to. To reduce the mass of the moving items, the motors controlling the tentacles are kept onboard the drone body, with the cables feeding through the tube. A few more smaller holes are drilled to allow the cables to leave the arm, and a small 3-D printed adapter is designed to allow the tentacles to be mounted and the cables to easily pass through.

Additional Components

This solution still requires electrical actuation, so its supporting components must be sourced. The specific items required are suitable servos, and a suitable cable to transmit the loads.

It is difficult to estimate the required force (and by extension, the motor torque) required for the tentacles to be actuated. As a conservative estimate, the same servo used for actuating the arm is chosen to actuate the tentacles. It will likely be too performant, but it has a low weight, so it does not negatively impact the functioning of the system. This was possible due to overestimating the mass requirement of the grabber subsystem earlier in the design phase, so choosing these motors will still be compliant to the requirements.

The only requirement on the cable is that it needs to be a 0.5 mm diameter cable, to be compliant with the scheme proposed by Wang et al. [54]. For this purpose, an Ultra High Molecular Weight Polyethylene (UHMWPE) cable was sought after. It has a high strength and low friction coefficient [14], which means each fibre can hold roughly 2-3x the mass of maximum payload individually. This may seem overdesigned, but there are significant moments these cables need to account for, so tensile loads will exceed beyond what each cable would carry 'evenly split.' The property of low-friction is also desired as the cable will need to slide through the holes in the tentacle regularly.

Figure 8.4: Final renders of grabber subsystem, (a) when tentacles are fully curled up, (b) when tentacles are fully extended.

8.4. Verification and Validation

As the grabber subsystem is largely a physical system, the majority of the official verification methods involve some prototype for testing. However, preliminary analytical tests have been conducted to ensure the design

is on the right track. These include simplified stress calculations and rendering in simulations to check for physical interferences. For all structural components, the internal stresses were calculated with large conservative margins

Furthermore, during the design phase, a sensitivity analysis was conducted before solidifying on the tentacle design parameters. The inputs were the dimensions of the payload, and the output was the mass of all the tentacles. In Figure 8.5, the height of the box is less relevant than the width, as an increasing width leads to a critical point where the mass skyrockets. There are also some spots with odd behaviour, and this is most likely due to the optimization algorithm faltering. For certain geometries, the initial guess for the spiral parameters are likely far from the solution so the algorithm diverges, and yields the dark spots in the figure. Based on this, the requirement for the geometry of the worst payload were slightly changed to have a less drastic impact on the tentacle weight, the required width was reduced from 30 cm to 25 cm.

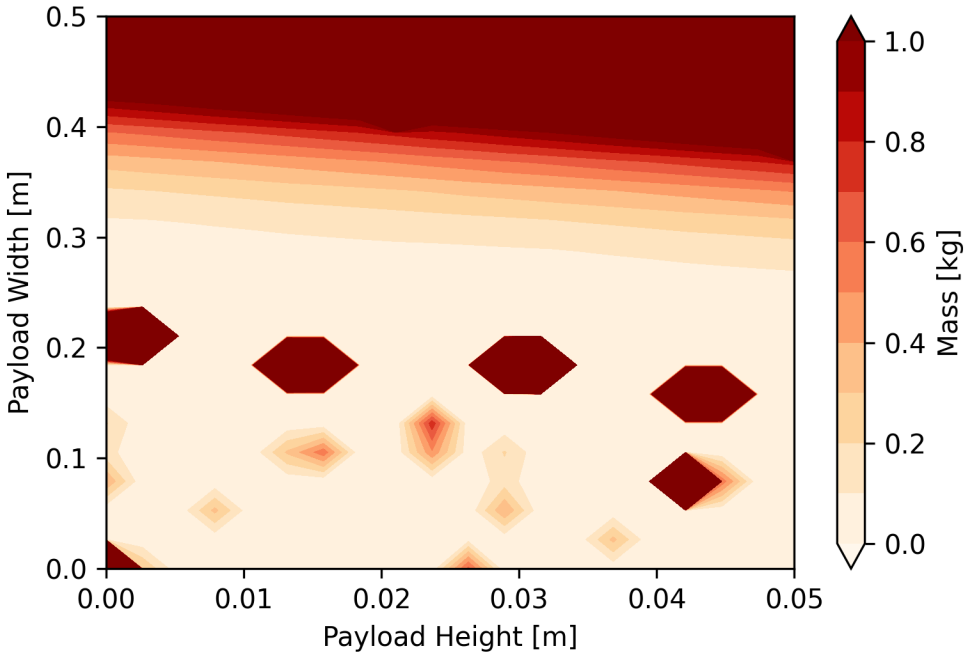


Figure 8.5: Optimised mass of differently sized payloads. Dark spots are due to the inner mechanism of the optimisation algorithm.

The total mass of the system is around 400 g. The tentacles themselves weigh 80 g (combined, based on the volume estimation earlier and the density of TPU), the three servos weigh 64 g each, with an additional 24 g mounting weight [42]. The arm weighs 40 g [43], with additional support weights of 50 g (mainly the pieces of bent aluminium at the base). In total, this is 386 g, which is rounded to 400 g to have a small buffer for inaccurate tentacle masses, or overlooked elements in the interface.

The only contributors to the power are the three servos. They have a designed voltage of 7.4 V, with a peak current of 3 A [42]. This means at the worst case, it is drawing up to 22.2 W. At idle, the current draw is 200 mA, so the idle power is 1.48 W.

Table 8.2: Compliance matrix for the grabber subsystem requirements for localization. Status: C = compliant, PC = partially compliant, NC = non-compliant, O = open/TBD.

Requirement ID	Method	Status	Evidence and Remarks
RQ-GWH-04-GRB-01	D*	O	Has been sized with this as a driving requirement, but requires a prototype for testing.
RQ-GWH-03-GRB-01	T*	O	Requires physical prototype for testing.
RQ-RLB-03-GRB-01	D*	O	Requires physical prototype for demonstration.
RQ-GWH-04-GRB-02	D*	O	Has been sized with this as a driving requirement, but requires a prototype for testing.
RQ-QLT-01-GRB-01	T*	O	Requires physical prototype for testing.
RQ-ENV-01-GRB-01	I	C	All components are reusable, CFRP will be chosen such that the matrix and fibres are separable.
RQ-GWH-01-GRB-01	T*	O	Requires physical prototype for testing.
RQ-GWH-02-GRB-01	D*	O	Requires physical prototype for demonstration.
RQ-RLB-01-GRB-01	A*	O	
RQ-MFC-01-GRB-01	I	C	All components have been sourced, and have the required lead time
RQ-MFC-02-GRB-01	D*	O	Requires physical prototype for demonstration.
RQ-DCM-01-GRB-01	D*	O	Requires physical prototype for testing.

8.5. Limitations and Recommendations

This design as-it-is has quite a few limitations, and recommendations that stem from those limitations. Many assumptions were made, and some elements were more simplified than others, in order to prioritise the more critical aspects of the subsystem.

Firstly, the grabber interface design is still quite simple. It has not been optimised based on its loading and functions, rather only fulfilling its requirements. A more detailed trade-off between different options for fulfilling the function of the grabber should be conducted.

Second, the tentacles have not been sized with how much surface is required to ensure there is enough friction to pick up various payloads, assuming the tentacles cannot get underneath the payload. In the future, more analysis should be done on the impact of the surface texture of the tentacles, and an additional unit test to check its friction.

Additionally, the tentacle length was sized assuming a large aspect ratio for the sake of being conservative. In the future, a more detailed analysis needs to be done on the various types and dimensions of payload items, beyond the label of grocery items, to further optimise the design for specific shapes.

Moreover, one parameter was fixed to start with, θ_{lim}. The effect of altering the number of rotations the tentacle makes on the optimal mass of the tentacle also needs to be investigated. Increasing the angle will likely result in a lighter tentacle, but a thorough trade-off needs to be done on based on its consequences.

Lastly, it is recommended to continue with building a prototype to be able to test the concept more thoroughly. It is also required to verify the subsystem requirements of the grabber.

9

Drone Body

This chapter will present the drone body. Section 9.1 redefines the requirements which dictate the drone's functions and form. In Section 9.2 the main design options are highlighted which must be chosen from, followed by Section 9.3 which discusses the full detailed design and presents the drone body. To ensure the design satisfies the requirements, verification and validation is performed for the drone body in Section 9.4 and finally, as no design is perfect, especially ones that have not been built and tested, the limitations and the recommendations for the current design are discussed in Section 9.5.

9.1. Requirements

To offer a roadmap and to ensure that the design satisfies the requirements, the relevant requirements are repeated. These requirements define what the design must contain, and be capable of performing.

Table 9.1: Verification plan for drone body requirements

Requirement ID	Requirement	M	Method description
RQ-RLB-01-BDY-01	The drone body subsystem shall be capable of operating for at least 5 continuous years.	A*	Analyse degradation of drone body subsystem, and compare with similar applications.
RQ-MFC-02-BDY-01	The drone body subsystem shall be designed such that a single trained technician can fully assemble the drone in under 7 hours.	D*	Trained technician shows that drone body can be fully assembled within time frame.
RQ-DCM-01-BDY-01	Drone body shall be designed in such a way that it can be disassembled by a single technician in under 45 minutes, using standard hand tools.	D*	Trained technician shows that drone body can be disassembled with standard hand tools within the time frame.

9.2. Design Options

The design philosophy of the drone body lay on a spectrum. On one side of the spectrum was a fully modular design which had simple interchangeable parts, and the other side of the spectrum was a fully integrated body with many custom made components which prioritizes mass and aerodynamic efficiency at the cost of manufacturability and maintainability. As the maintainability and production cost is important to the sucess of the product it was decided to design closer to the former, more maintainable format. A more modular design has the added advantage that it is more sustainable, as a local component failure is easily interchangeable and reduces the need to replace the entire drone. This philosophy has been a driving force in the design choices made in this section. The choice for the legs and arms are elaborated further in the sections.

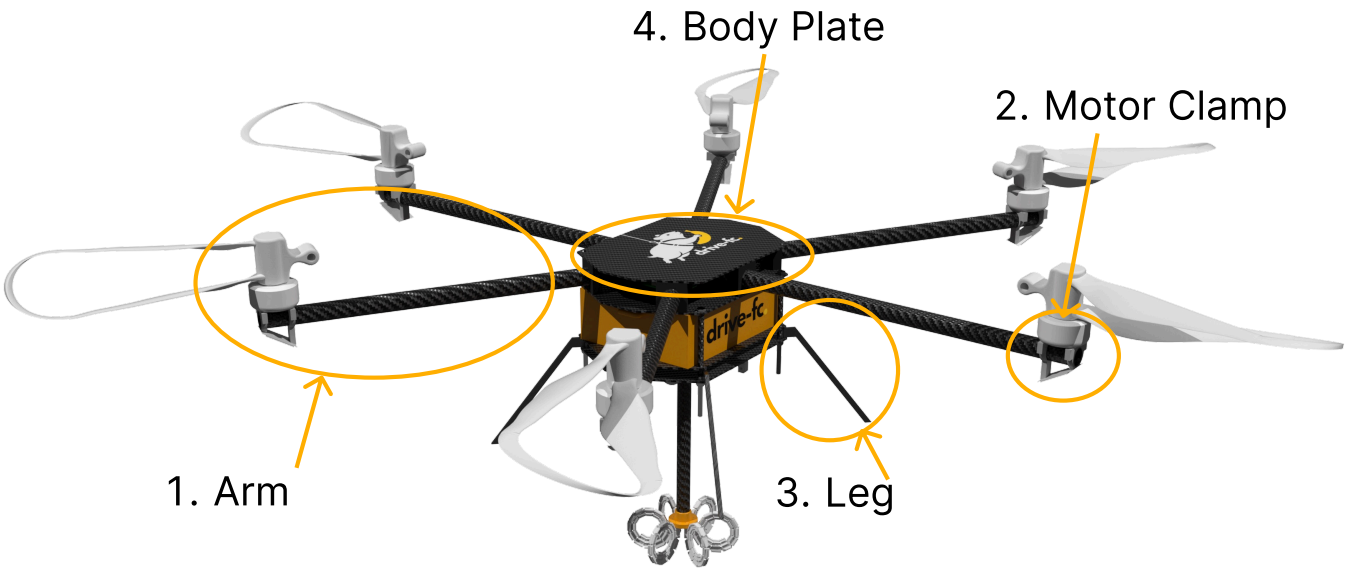


Figure 9.1: Annotated drone body model

Legs

The purpose of the legs of the drone is to allow for precise landing at the ERS such that the battery is positioned as expected. The secondary purpose of the legs is to accommodate for emergency landings without the drone sustaining damage.

The design choice for legs is between the legs being fully integrated into the drone body or as a separate component attached through a bolt or other fastening mechanism, such that there is no permanent connection. As mentioned there is a preference for a modular, removable design to ensure requirements **RQ-MFC-02-BDY-01** and **RQ-DCM-01-BDY-01** are easily met. For this reason the non-permanent solution is chosen.

Arms

The purpose of the drone arms is to hold the propeller motors and sensors. The arms of the drone must be sufficiently spaced to ensure the propellers have enough clearance. The secondary function to hold mm-wave sensors at some distance from the body to allow for a larger field of view as the body, arms or propellers do not obstruct the sensors.

The design choices of the drone arms are similar to that of the legs described in Section 9.2. Here the choices consist of clamping the arms to the body or making the arms and body as a unibody construction. At the outer end of the arms the same choice is also possible, whether the motor hub is clamped to the arm or a unibody construction. Any combination of the two is possible and as such there are 4 different design concepts to be chosen from. To favour manufacturability and modularity the ”clamp-at-both-sides” is chosen.

9.3. Detailed Design

Using the philosophy and design choices described in Section 9.2 the detailed design can be created. As the design is highly dependent on other subsystems, most of the design choices are driven by the other subsystem when possible. The drone body is shown in Figure 9.1.

The arm of the drone consists of a carbon fibre tube clamped at both ends as described in Section 9.2. It is hollow inside to allow wiring from the sensor and the motor to go through to the flight computer.

The motor clamp is clamped onto the arm and holds the motor and the mm-wave sensor. Small bolts that fit into the threaded holes of the motor. An o-ring damper fits between the clamps and the motor to reduce vibrations, which reduce the total noise made by the drone, and allows for improved measurements of the mm-wave sensor.

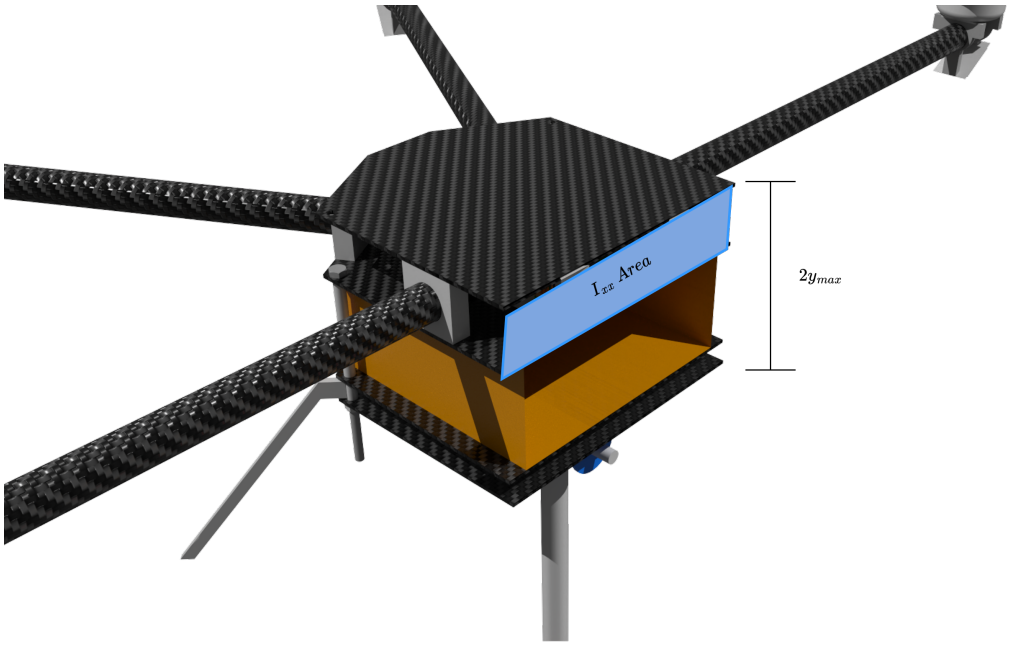


Figure 9.2: Drone body cross section highlighting chosen geometry values for stress calculations.

The legs are independent aluminium components. The legs are secured by the drone's primary vertical bolts, which provides the necessary clamping force to hold the leg in position. This allows the legs to be easily removed and replaced if necessary.

The drones main structural components are 4 body plates, which are each identical. The drone has 4 vertical bolts that hold the plates in place. It was chosen to to keep each of the body plates identical, thus reducing the number of unique parts necessary to service the drone.

9.4. Verification and Validation

The bulk of the verification of the drone body needs to be done after the drone has been built as a prototype, however, some preliminary calculations have been done to predict compliance for the key structural elements. The compliance matrix is presented in Table 9.2.

The first check was to ensure the core of the drone body does not break or deform. The drone is approximated as a rectangular tube with a constant cross section. As the drone geometry is more complex than that, some conservative assumptions are made. The stress in the material is estimated by

$$\sigma = \frac{My}{I_{xx}} \tag{9.1}$$

[30], where M is the internal moment in the cross-section, I_{xx} is the area moment of inertia, and y is the distance from the neutral axis. The I_{xx} of the cross section is approximated only by the tube enclosed by the closest plates (see Figure 9.2), but the furthest y is half the distance between the furthest two plates. This will drastically overestimate the stress within the structure. This can provide some margin for neglecting other sources of stress, such as shear loading. The only contributors to I_{xx} are the two plates, the bolts between them are neglected as they are not present throughout the cross-section.

The moment itself is calculated by approximating all the propeller thrusts to be introduced to the body as point loads, the weight of the grabber with a payload as a point mass in the centre, and the rest of the weight of the drone uniformly distributed over the length of the body. The propeller thrusts and masses are assumed to be double the value in the standard 'hovering' procedure, to predict the stress at the highest thrust-to-weight setting, but still as a static analysis. These forces were related to the internal moments via the integration relation $\int V(z)dz = M(z)$[30]. The largest moment found along the drone body was the one used for the calculation. The resulting stress in the plates was 4.3 MPa, which is still well below the tensile limit [2], and still has a factor of 9 above the critical buckling scenario (calculated using Equation (9.2) [30], k is a tabulated parameter depending on the edge conditions, t is the part thickness, and a is the length of the plate).

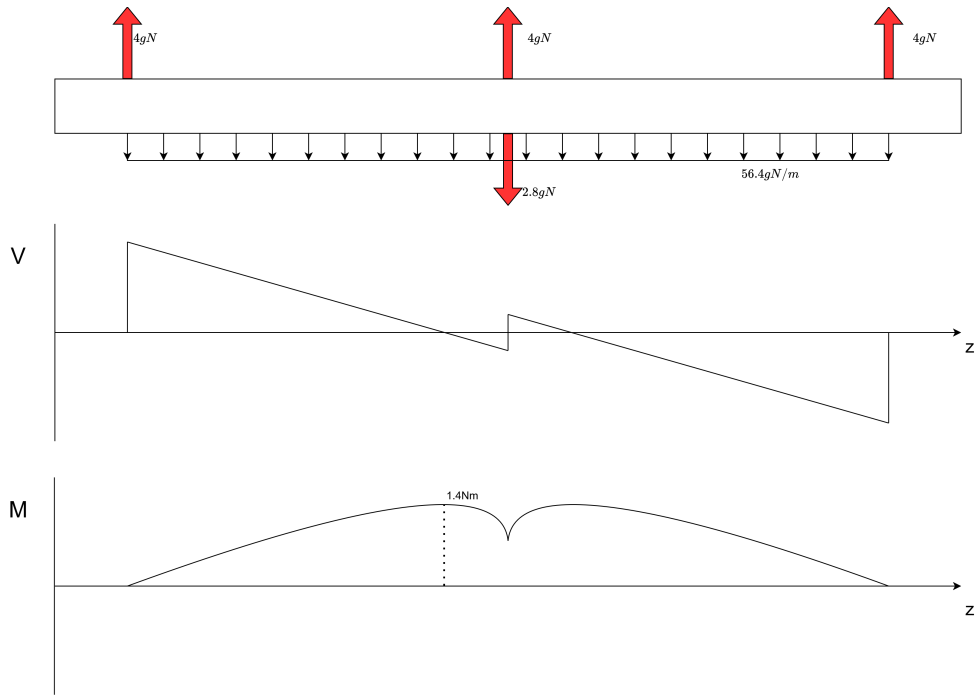


Figure 9.3: Free body diagram of a simplified, loaded drone drone with internal shear and moments for stress calculation. The peak moment is labelled.

$$\sigma_{cr} = k\frac{E\pi^2}{12(1-\nu^2)}\left(\frac{t}{a}\right)^2 \tag{9.2}$$

The second check was to ensure the arms of the drone do not fail. For this, Equation (9.1) was used again. The I_{xx} was of the circular cross-section of the tube (making use of the thin wall formula $I_{xx} = \pi r^3 t$[30]) and the y is the radius of it. The highest moment is naturally at the base of the arm, which was calculated by the thrust of a propeller multiplied by the length of the arm. The maximum stress here is 25 MPa. This value is still below the compressive strength of CFRP [2].

The last structural element to be checked are the legs. The key failure here is buckling during landing. The legs are approximated as two-force members, as they are quite slender. Assuming all four legs take the weight of the drone equally, each leg will take up 37.1 N. This also accounts for the angle the legs are at. The buckling formula column is:

$$F_{crit} = \frac{\pi^2 EI}{KL^2} \tag{9.3}$$

where K is 2 for this case [30] (one end free). Therefore, the F_{crit} is 1877.7 N (E from [2]). This is at least 2 orders of magnitude larger than the experienced load, so it is a very conservative design, and will likely also be reliable in dynamic cases.

Table 9.2: Compliance matrix for the drone body subsystem requirements. Status: C = compliant, PC = partially compliant, NC = non-compliant, O = open/TBD.

Req ID	Method	Status	Evidence / Remarks
RQ-RLB-01-BDY-01	A*	O	Need to generate a fatigue model to test against.
RQ-MFC-02-BDY-01	D*	O	Requires physical prototype for demonstration.
RQ-DCM-01-BDY-01	D*	O	Requires physical prototype for demonstration.

9.5. Limitations and Recommendations

At this stage of the design process, several limitations exist in the current model that must be addressed post-Design Synthesis Exercise (DSE) to produce a complete product. The primary limitations consist of the missing battery holder design, the lack of a FEM analysis to verify structural stress limits, and the absence of fatigue and vibration analyses. Furthermore, the landing legs are currently too short to accommodate a malfunction in the grabber pivoting mechanism.

The battery holder must still be designed while keeping the integration with the ERS in mind. The holder should reliably secure the battery and ensure electrical contact, but it must not grip it so tightly that the ERS arm is unable to safely remove it. This mechanism could be designed as an interlocking rail drill mechanism as found on power tools.

The structural analyses performed thus far are simplified and conservative. A more advanced FEM analysis is absolutely necessary to ensure complete regulatory compliance and product reliability. More specifically, there should more dynamic analysis on the structure; this is to ensure reliability under vibrations and accelerations.

Because the components will be used extensively day in and day out, a fatigue analysis must be performed across the entire structure. Particular focus should be placed on the grabber, as it is a novel solution. The arms also deserve careful analysis as these endure constantly changing dynamic loads, which can lead to fatigue failures.

Since the mm-wave sensors mounted beneath the motors are responsible for collision avoidance, they must be to ensure sensor data is not compromised by motor vibrations, which would significantly reduce the safety of the drone. As such it is recommended to find the impact of the motor vibrations on the performance of the mm-wave sensors. This analysis will make apparent what dampers are needed.

Finally, although the grabber is designed to pivot upward during landing, the legs are not long enough to guarantee a safe landing if the grabber mechanism malfunctions. It is recommended to implement a structural fail-safe; for example, adding secondary landing feet to the ends of the arms so the drone can land safely without damaging critical or delicate components.

10

Localisation

This chapter will present the localisation subsystem. Section 10.1 will explain the functions and relevant requirements for this subsystem. Section 10.2 will touch on the considered options during the conceptual design phase. Section 10.3 shows the Ultra-wideband standard and the chosen localisation algorithm, followed by Section 10.4 where the chosen data fusion method will be presented. In this section the Inertial Measurement Unit (IMU) and vertical Time of Flight (ToF) sensors are introduced. Finally Verification & Validation and Final Recommendations are presented in Section 10.5 and Section 10.6, respectively.

10.1. Requirements

The localisation subsystem needs to accurately determine the drone's position in the warehouse. Establishing a robust and reliable framework is of great importance as localisation flows directly into the control subsystem. The localisation subsystem also has to determine human positions in the warehouse. The following requirements drive the design of the subsystem:

Table 10.1: Localisation subsystem requirements and verification plan

Requirement ID	Requirement	M	Method description
RQ-ENV-01-SNS-01	The sensor subsystem shall use reusable hardware.	I	Check properties of sensor hardware.
RQ-WOS-01-SNS-01	The sensor subsystem shall locate human workers with 1 cm accuracy.	T*	Measure accuracy of human worker detection.
RQ-GWH-01-SNS-01	The sensor subsystem's hardware shall be functional in Cold-storage conditions.	I	Check hardware operation temperature range.
RQ-GWH-02-SNS-01	The sensor subsystem's hardware shall be functional in Ambient conditions.	I	Check hardware operation temperature range.
RQ-ATM-02-SNS-01	The sensor subsystem shall measure drone system status at a minimum frequency of 100 Hz.	I	Inspect the frequency of the sensor readout.
RQ-ATM-04-SNS-01	During item pickup, the sensor subsystem shall measure drone position with an accuracy of 1 cm in each direction at 1 m from the item.	T*	Test that the drone measures the position within an accuracy of 1 cm.
RQ-ATM-04-SNS-02	During item drop-off, the sensor subsystem shall measure drone position with an accuracy of 1 cm in each direction at 1 m from the drop-off location.	T*	Test that the drone measures the position within an accuracy of 1 cm.
RQ-ATM-04-SNS-03	The sensor subsystem shall measure drone position with an accuracy of 1 cm in each direction at 1 m from the ERS.	T*	Test that the drone measures the position within an accuracy of 1 cm.

Continued on next page

Table 10.1 – Continued from previous page

Requirement ID	Requirement	M	Method description
RQ-ATM-04-SNS-04	The sensor subsystem shall measure drone position with an average error not exceeding 10 cm.	T	Test sensor accuracy used for position determination.
RQ-MFC-01-SNS-01	All off-the-shelf sensor subsystem components shall have a maximum verified supplier lead time of 4 weeks.	I*	Check with manufactures and distributors whether sensor subsystems are available within time frame.
RQ-MFC-02-SNS-01	The sensor subsystem shall be designed such that a single trained technician can fully integrate onto the drone in under 2 hours.	D*	Trained technician shows that sensor subsystem can be fully integrated within time frame.
RQ-DCM-01-SNS-01	Sensor subsystem shall be integrated into design in such way that it can be disassembled by a single technician in under 30 minutes, using standard hand tools.	D*	Trained technician shows that drone sensors can be disassembled with standard hand tools within the time frame.

10.2. Design Options

As part of the design process, several localisation methods have been considered. This was approached by analysing sensor placement and dividing the solution space into three branches: On-board, anchor-based and external tracking.

On-board

For on-board hardware, Simultaneous Localisation and Mapping (SLAM) was considered by looking at sensors such as Light Detection and Ranging (LiDAR), mmWave radar or Depth Cameras. Utilizing ToF data, these sensors can position the drone with respect to mapped objects around it. While highly accurate, the methods have technical risk with SLAM algorithms being complex to implement.

External Tracking

External tracking promises the best performance from the three configurations, but requires significant infrastructure. Frameworks like OptiTrack also scale badly in terms of cost and full warehouse coverage proved to be well outside the budget.

Anchor-based

The final design landed on anchor-based solutions which were more fit for indoor applications. Apart from Ultra Wideband (UWB), which was selected for the final design, the trade-off also took into consideration Ultrasound beacons. These methods scale well while also providing accurate and robust localisation. UWB and Ultrasound infrastructure can also be used for human tracking and communications, offering great multifunctionality. While Ultrasound technology offers better positional accuracy, UWB was deemed the better option mainly due to cost, and because it is a proven concept in localisation applications. It can also easily be used to track human positions using tags.

10.3. Ultra-wideband Ranging

UWB-based localisation has recently emerged as a promising solution for indoor environments. It benefits from minimal hardware, relatively high performance and robustness. As a wireless positioning technology based on IEEE 802.15. standard, it offers centimetre level ranging capabilities and a substantially large bandwidth that also allows for communication.

10.3.1. Background

UWB localisation, similar to other Anchor-based methods, utilizes the drone's relative position to beacons to triangulate its 3D position. ToF is the underlying principle for this method. The on-board antenna receives pulses from the beacons, the pulse includes the time it was sent, thus it can find the time of flight and compute the distance from the tag to that specific beacon. The main issue arising from this is that the clocks of the beacons and the drone need to be perfectly synchronized. This can be done by re-calibrating at the ERS stations but that induces unnecessary complexity. The following two methods work without clock

synchronization for the drone. Two Way Ranging (TWR) does not require any type of synchronization, net even between beacons. It works by sending a signal from the drone, then receiving and sending one back from the anchor, timing everything. While this method is robust to clock drift, it scales badly because TWR uses one-to-one signals. This means that the anchor network has to send separate signals for all drones. Combining this with the fact that each link consists of three messages **drone** → **beacon** → **drone**, this will increase the number of signals exponentially, draining batteries faster and overcrowding the network.

10.3.2. Time Difference of Arrival

Time Difference of Arrival (TDoA) benefits from the best of both worlds. The method does not rely on drone clock synchronization and uses one-to-many pulses that any drone can pick up on. This method does however require anchor synchronization which can be implemented in indoor environments.

Optimization problem

The sensors measure t_1 and t_2 per anchor pair on the local clock. The range difference r between the target and the two beacons is

$$r_{1,2} = c(t_1 - t_2),$$

where c is the speed of light. The predicted range difference for a candidate target position $\mathbf{p}$ is $\|\mathbf{p} - \mathbf{a}_1\| - \|\mathbf{p} - \mathbf{a}_2\|$, with $\mathbf{a}_1, \mathbf{a}_2$ the known anchor positions. For a single measurement this draws a hyperboloid with foci at the two anchors. However, a single pair leaves the position undetermined, with a possibility of the drone lying anywhere on the hyperboloid. Therefore a batch of m measurement from different pairs are stacked. Thus for a measurement k, the following residual can be formulated:

$$r_k(\mathbf{p}) = \Big(\|\mathbf{p} - \mathbf{a}_1^k\| - \|\mathbf{p} - \mathbf{a}_2^k\|\Big) - r_{1,2}^k$$

the position is thus obtained as the nonlinear least-squares estimate:

$$\hat{\mathbf{p}} = \arg\min_{\mathbf{p}\in\mathbb{R}^3} \ \frac{1}{2}\sum_{k=1}^{m} r_k(\mathbf{p})^2. \tag{10.1}$$

The objective is minimised iteratively. Several methods are used in literature, here the Levenberg-Marquardt (LM) method [29] is chosen as it is robust to poor initial guesses. Its derivation and convergence properties are standard and beyond the scope of this work.

10.3.3. Hardware

The chosen framework requires a UWB transceiver chip for the drone, and for each beacon. Among UWB modules the Decawave DW1000 [17] was chosen for its low-cost, weight and power, but also because it is widely supported by existing firmware and an extensive body of research, thus reducing development risk. While the on-board sensor will be connected to the flight computer directly, anchors will require a microcontroller to operate.

10.3.4. Human Tracking

For the purpose of this work, the localization subsystem focused mainly on drone positioning. The following section will explain the state estimation method used for accurate drone localisation. Human tracking will be explored in future development and because of the more lenient requirement, RQ-WOS-01-SNS-01, the system will only use TDoA measurements and LM iterations for 2D positioning.

10.4. Sensor Fusion

Sensors produce noisy data that needs to be converted into useful state estimates that feed into the control subsystem. This section will present the remaining two sensors used for localisation followed by an introduction to the chosen data fusion algorithm.

10.4.1. Inertial Measurement Unit

The IMU provides the high-rate motion information (around 1 kHz) that drives the estimator. It combines a three-axis accelerometer and a three-axis gyroscope, together measuring the specific force and the angular rate of the drone. The IMU requires no external infrastructure and because it provides a higher rate than all other sensors, it serves as the natural source for propagating the state in the filter that will be presented later in Section 10.4.3.

Yaw Drift
One of the problems that arises from dead reckoning is drift. While roll and pitch can easily be corrected because the accelerometer measures the gravity direction, which is an absolute reference, thus bounding these rotations. Yaw, which by definition is the rotation about the gravity vector itself, accumulates drift. To solve this, some sensors also include a magnetometer, specifically 9-axis Degrees of Freedom (DoF) IMUs. This provides useful heading information that can recalibrate the yaw and prevents drift.

Hardware
The chosen IMU sensor for this work is the MPU 9250 [33]. It's a single chip 9-axis device that is low-cost, low-power and widely used in robotics research, thus ensuring low development risk.

10.4.2. Altitude Control
The last important component that has to be addressed before diving into the data fusion algorithm is the vertical ToF sensor. The UWB network provides a complete 3D positional fix, but the accuracy depends widely on the anchor constellation. The environment is a warehouse, that spans mostly in the horizontal direction, and with limited verticality. This means that the directions from the target to the beacons will lie closely to the horizontal plane, carrying little vertical information and inducing significant positioning errors.

Upward-facing mmWave radar
To mitigate this error, the drone will use one of its vertical mmWave radar as a ToF sensor that feeds into the state estimation to correct its vertical position. Due to the inherent nature of a warehouse, where the ground is cluttered with objects, the sensor will point upward, this also allows for better Line of Sight (LOS) as the bottom of the drone contains the grabber subsystem and legs. More details on the hardware and specifications of the mmWave module will be presented in Chapter 11.

10.4.3. Kalman Filter
The IMU provides high rate data that drifts, the TDoA and ToF measurements provide sparse absolute aiding, something needs to fuse them together for state estimation. Kalman filters are widely used for such applications in robotics and aerospace. The algorithms work by propagating the state of the drone at every IMU step while also keeping track of the uncertainty of its predictions. Once TDoA and ToF data is available, it nudges the state so it better fits both the predicted data and measurements. For example, when the filter is not entirely sure that the predictions are right and is feeling lost (has high uncertainty), it trusts the measurements more and the correction is larger. When the filter trusts it's predictions, the update is less substantial. While the IMU can provide up to 1 kHz data rate, the filter will sample at a quarter of that to reduce computation time. This does not impact performance as it is still less than the TDoA sample rate which drives the actual accuracy of the model.

Extended Kalman Filter
The standard Kalman filter [22] assumes that both the motion and measurement models are linear. The models in this system, however, are not linear, For example the TDoA measurement is a Euclidean distance $\|\mathbf{p}-\mathbf{a}\|$, and the ToF altitude involves a $1/\cos\theta$ tilt compensation. None of these can be written as a constant matrix acting on the state. The Extended Kalman Filter (EKF) [47] accommodates this by linearising the models about the current state estimate at each step.

Coupling
In this work the UWB measurements are fused in a loosely-coupled manner. With this method the calculated positions from Equation (10.1) are fed into the EKF. Based on batch size, after m measurements, for that many pairs, the position is estimated, then fed into the filter. A tightly-coupled configuration would feed the raw difference from Section 10.3.2 directly into the filter. While this can hypothetically provide better performance because the updates happen every TDoA measurement, not after m, it is more computationally complex, and requires a more intricate filter design. The loosely-coupled option also benefits from modularity, as the TDoA and LM frameworks can be built, verified and validated separately from the filter itself.

10.5. Verification and Validation
The driving requirement for the localisation subsystem is RQ-ATM-04-SNS-04 which bounds the positional error to 10 cm. To verify the sensor fusion of Section 10.4 and the TDoA logic of Section 10.3, the estimators were run against real flight data rather than simulated measurements. For this purpose the publicly available *UTIL* dataset [61] was used: an ultra-wideband time-difference-of-arrival (UWB-TDoA) indoor localisation dataset collected on a quadrotor drone at the University of Toronto. The dataset provides:

- Ground-Truth Vicon (motion-capture) data with millimeter level accurate pose, sampled at 200 Hz
- TDoA measurements per anchor pair, sampled at 480 Hz
- Accelerometer and gyroscope data from the IMU, sampled at 1009 Hz
- Vertical ToF measurements, sampled at 24 Hz

The dataset also includes six different trajectories, two different anchor constellations and three No Line of Sight (NLOS) configurations.

Hardware
The dataset utilises similar hardware for the UWB tag. The drone has a Decawave DW1000 just like the one selected in Section 10.3 and the same transceiver for the beacons. The IMU sample rate is well within the range of MPU-9250 chosen in Section 10.4. The vertical ToF sensor however is a laser pointing downwards. The laser's accuracy is fortunately similar to that of a mmWave radar, specifically when evaluating ranging capabilities, which is the main role of this sensor. The mmWave sensor can even outperform the laser in high tilt conditions, because it has a radar cone, it can always point up while adjusting for tilt. For the purpose of Verification, it is assumed that the mmWave radar will not induce any unforseen errors in the model.

Magnetometer
The main difference between the dataset configuration and the one chosen for this project is the absence of magnetometer readings. The dataset does provide additional optical flow measurements that can correct the yaw drift but does imply the need for an additional downward-facing camera. The lack of orientation measurements does not impact the positioning accuracy of this model, it only impacts yaw. For the purpose of testing the data fusion algorithm, ground-truth heading data has been fed back into the model with noise applied to simulate magnetometer readings, sampled at the Ground-truth Vicon rate. This does not impact any positional performance metrics mentioned later.

Results
The localisation accuracy is quantified using the Root Mean Square Error (RMSE) between the estimated position and the millimetre-accurate motion-capture ground truth. For the purpose of comparison between different configurations and NLOS conditions, one trajectory was selected to ensure enough vertical and horizontal displacemet.

LOS (ToF vs no ToF)
The impact of the vertical ToF was quantified and analysed in detail. The following figures outline the clear difference in performance that the sensor provides.

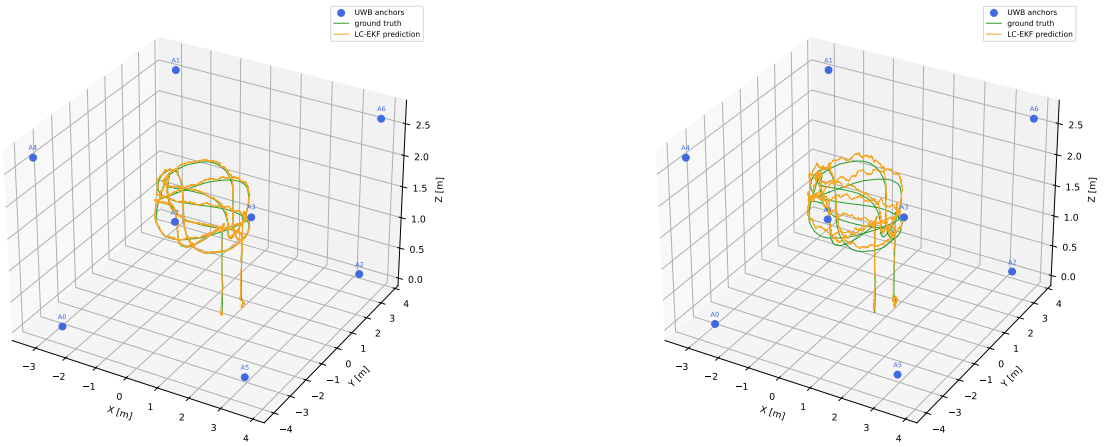


Figure 10.1: LOS trajectory with (left) and without (right) the ToF altitude aiding.

As seen in Table 10.2, for LOS with a ToF sensor, the error is around 4 cm which is well within the required accuracy. Without a ToF module, there would be a large error in the z (vertical) axis, around 9 cm which is highly undesirable.

Table 10.2: Position RMSE for const1 trial3, with and without ToF altitude aiding (LOS).

Configuration	x [m]	y [m]	z [m]	3D [m]
With ToF	0.022	0.023	0.022	0.039
Without ToF	0.023	0.025	0.094	0.100

LOS (Magnetometer vs no Magnetometer)

The impact of the magnetometer can not be quantified in the same manner as the ToF sensor because it does not influence the positional accuracy of the drone, however, the drift can be quantified. The following figures present a top-down view of the trajectory as well as the heading of the drone at the start and end of measurement:

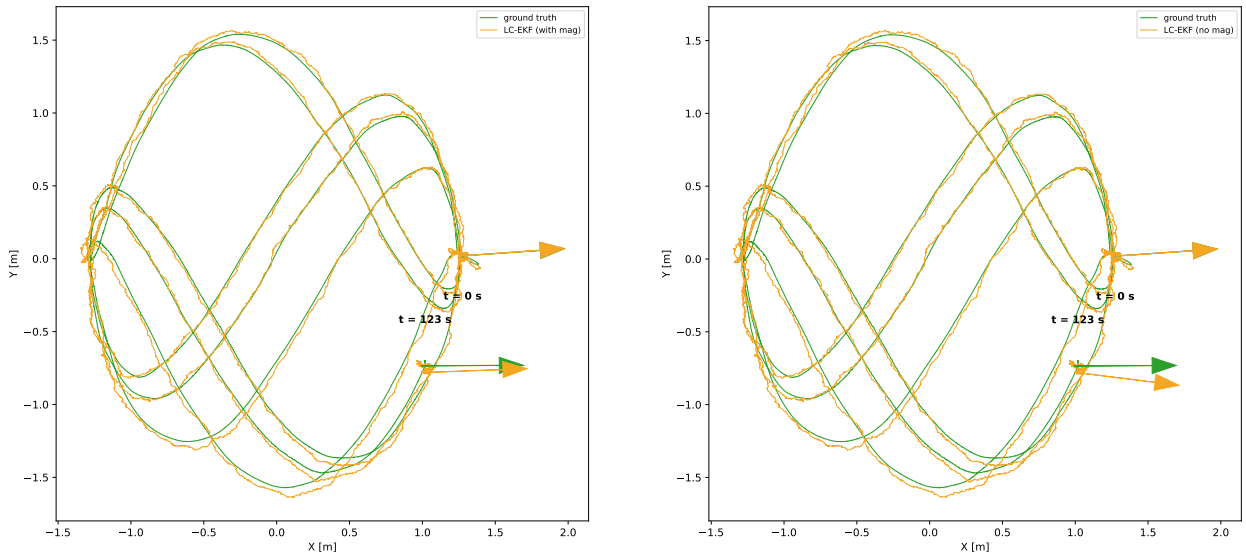


Figure 10.2: Top-down LOS trajectory with (left) and without (right) the Magnetometer Measurements.

As seen in Table 10.3, the heading displacement angle is 7.4 °, which is highly undesirable. While the magnetometer readings are generated from ground truth data, conclusions can be made about the EKF which correctly recalibrates the heading at every magnetometer tick.

Table 10.3: Heading deflection with and without the magnetometer (LOS).

Configuration	Start [deg]	Finish [deg]
With magnetometer	+0.2	+1.8
Without magnetometer	+0.0	−7.4

NLOS

No Line of Sight (NLOS) is a very important condition to take into consideration when designing a warehouse localisation system. The warehouse contains boxes, racks and other objects that can obstruct the UWB signals. The two analysed configurations are a simple metal box in front of one anchor, which is a realistic case that might be relevant to every day operations, and a highly unlikely worst case scenario with 4 anchors obstructed by boxes.

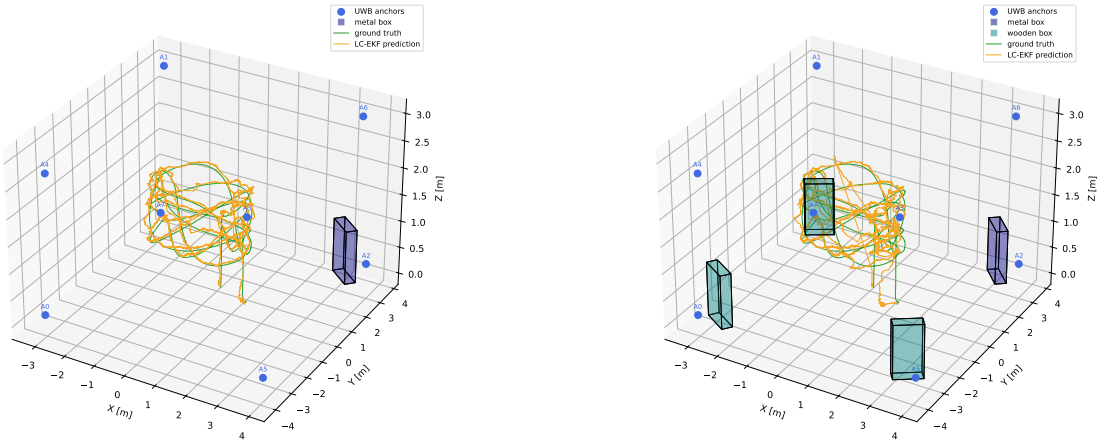


Figure 10.3: NLOS trajectory with one box (left) and four boxes (right).

With one obstruction, the model performs within the allowed error range with 9.2 cm RMSE as can be seen in Table 10.4. With four boxes the error rises to 31.4 cm which is three times the required margin.

Table 10.4: Position RMSE under NLOS, with a single metal obstacle, and all four floor-level anchors shielded.

Configuration	x [m]	y [m]	z [m]	3D [m]
Trial 2	0.062	0.049	0.048	0.092
Trial 3	0.144	0.179	0.214	0.314

Compliance

The results of the verification for the localisation subsystem can be seen in Table 10.5. Out of the eleven requirements, four were deemed verifiable within the scope of this project. The UWB hardware chosen in Section 10.3 and IMU chip chosen in Section 10.4 are modular and can operate in cold conditions, thus satisfying RQ-ENV-01-SNS-01 and RW-GWH-01-SNS-01. The data fusion model from Section 10.4 samples at a higher rate than the one required in RQ-ATM-02-SNS-01. RQ-ATM-04-SNS-04 is partially compliant due to the poor performance in heavy NLOS conditions. This will drive the UWB beacon detailed positioning in future development to ensure no such conditions can happen. Due to the positional accuracy of 3 cm with ideal LOS conditions, future development will have to incorporate mmWave accurate ranging for pickup, drop-off and ERS docking to comply with requirements RQ-ATM-04-SNS-01, 02 and 03. RQ-WOS-01-SNS-01 will have to be tested with a TDoA only approach for the tags the workers will use.

Table 10.5: Compliance matrix for the sensor subsystem requirements for localisation. Status: C = compliant, PC = partially compliant, NC = non-compliant, O = open/TBD.

Req ID	Method	Status	Evidence / Remarks
RQ-ENV-01-SNS-01	I	C	Modular sensor hardware - Reusable, supports disassembly requirement.
RQ-WOS-01-SNS-01	T*	C	Test stripped TDoA performance.
RQ-GWH-01-SNS-01	I	C	$\geq$ -40 ° [17], [33] - Component temperature range.
RQ-GWH-02-SNS-01	I	C	$\geq$ -40 ° [17], [33] - Component temperature range.
RQ-ATM-02-SEN-01	I	C	250 Hz - EKF state-output rate.
RQ-ATM-04-SNS-01	T*	O	UWB TDoA cannot reach 1 cm.
RQ-ATM-04-SNS-02	T*	O	As SNS-01 (drop-off phase).
RQ-ATM-04-SNS-03	T*	O	As SNS-01 (ERS phase).
RQ-ATM-04-SNS-04	T	PC	0.04 m (LOS), 0.31 m (NLOS). Exceeds 10 cm under heavy NLOS.
RQ-MFC-01-SNS-01	I*	O	Confirm supplier lead time $\leq$ 4 weeks.
RQ-MFC-02-SNS-01	D*	O	Integration-time target not yet defined.
RQ-DCM-01-SNS-01	D*	O	Disassembly-time target not yet defined.

Validation

While the verification section ensures that the subsystem partially meets its accuracy requirements against the dataset, validation will ensure that the system meets its operational purpose. Because the dataset utilizes real sensor data, this partially validates the model used for data fusion and the chosen sensor layout. A complete validation will have to evaluate a full operational cycle from pickup to drop-off and recharging. After developing the close-range mmWave localisation and establishing a robust UWB constellation, the system will be validated in an environment representative of an actual warehouse, with realistic NLOS conditions. In terms of hardware, as mentioned before, the setup used for the dataset does not have a nine axes IMU or a mmWave vertical ToF sensor but uses a laser instead, thus the actual sensor layout will have to be validated. Finally the environmental and operational requirements like RQ-MFC-02-SNS-01 and human localization that could not be verified within this project will have to be validated in later development phases.

10.6. Limitations and Recommendations

This section will touch on the limitations of the chosen localisation method as well as the future recommendations for the following design stages. After outlining the localisation method and performing the preliminary verification for the subsystem, a few conclusions can be made about the chosen configuration.

First, the TDoA algorithm works only when the anchors are completely synchronized. The induced error in the case of a drifting clock on one of the beacons will have to be tested and can prove to be problematic when scaling to a large warehouse.

Second, while the state estimation method used in Section 10.4 is tailored towards non-linear systems, it only uses a first order approximation thus it could struggle with aggressive dynamic maneuvers. One more important caveat is the interaction between the filter and the control subsystem, specifically how does the sample rate influence the performance of the controller. Full software integration and testing with other subsystems is important to evaluate the system's sensitivity to sensor output. For the purpose of this work, only the simple Kalman Filter and Extended Kalman Filter have been analysed, due to their relatively easy implementation. Several other models exist that could boost the localisation performance such as the Error State Kalman Filter or Unscented Kalman Filter.

Third, the human tracking method has not been tested or explored in this work. The 1 m positioning accuracy might seem as an easy target but might prove to be a limiting factor when using TDoA alone. The TDoA algorithm without IMU and a larger batch size might perform well in 2D but limited conclusions can be made at this time.

Fourth, while the verification procedure did provide useful localisation performance metrics, some caveats have to be acknowledged. The obvious hardware miss-match has to be mentioned, even though mmWave sensors produce similar ToF measurements like the laser used in the dataset. The usage of ground truth data to make up for the missing heading data also has to be taken into account. The trajectories from the dataset do not include almost any roll or pitch, thus there are limited conclusions to be made about how well the model can estimate these state variables. The UWB beacon constellations also heavily influence the results. While UWB positioning is reserved for future development and will most likely provide a denser network than in the tests, no guarantee can be made on the performance of such constellations. The TDoA algorithm will have to be tweaked to support more than eight anchors. Because the drone can listen to signals from all beacons, it will have to filter out anchors that are far away and might induce errors. The NLOS tests provide some estimate to the accuracy loss in case of beacon obstructions but are limited to the characteristics of the boxes used in the dataset and might not accurately represent real warehouse conditions. After building the drone prototype, it is important that the whole system is tested in a warehouse environment with a realistic UWB beacon network.

Lastly, the positional accuracy of the system is limited to 3 cm which does not comply with the close range 1 cm requirement. Future development will have to design, implement and test a close range mmWave localisation system to satisfy this requirement.

11

Collision Avoidance

Where the localization subsystem takes care of the necessary positional information on both drones and humans for path-planning, if an unregistered obstacle enters the calculated path, without a safeguard to avoid collisions the system is unable to ensure safe and efficient operations. The collision avoidance subsystem acts as the eyes of the drone, allowing it to dynamically avoid direct contact with detected obstacles during the entirety of the drones operations.

This chapter will present the overview of the collision avoidance subsystem. Section 11.1 explains the relevant requirements directly affecting design choices made. Section 11.2 briefly explains considered options prior to preliminary design. Section 11.3 presents the fundamental concepts of the mmWave radar and collision cone approach that make up the backbones of the collision avoidance subsystem. The subsequent sections, Section 11.4, Section 11.5, Section 11.6 presents the performance analysis of the designed system, the requirements it fulfils as well as a critical review of the limitations of the design.

11.1. Requirements

For the collision avoidance subsystem to ensure safe and efficient operations, it must be able to conduct a multitude of different functions, namely; detect obstacle size, detect obstacle distance relative to drone, and determine the next sequence of actions that guarantee no collision. To evaluate the extent to which these functions were fulfilled, requirements were formulated as in Table 4.1. Those that directly acted as critical requirements for this subsystem are listed in.

Table 11.1: Verification plan for sensor requirements

Requirement ID	Requirement	M	Method description
RQ-ENV-01-SNS-01	The sensor subsystem shall use reusable hardware.	I	Check properties of sensor hardware.
RQ-GOV-01-SNS-01	The sensor subsystem shall comply with NL communication regulations.	A	Compare subsystem to relevant regulations.
RQ-GWH-01-SNS-01	The sensor subsystem's hardware shall be functional in Cold-storage conditions.	T	Test sensor subsystem functioning in cold storage zone.
RQ-GWH-02-SNS-01	The sensor subsystem's hardware shall be functional in Ambient conditions.	T	Test sensor subsystem functioning in ambient storage zone.
RQ-GWH-01-SNS-01	The sensor subsystem's hardware shall be functional in Ambient conditions.	T	Test sensor subsystem functioning in ambient storage zone.
RQ-RLB-02-RTG-01	The routing subsystem shall plan collision avoidance paths maintaining a 30 cm distance between known obstacles and drone.	A	Simulate collision situation to analyse avoidance behaviours.

Continued on next page

Table 11.1 – Continued from previous page

Requirement ID	Requirement	M	Method description
RQ-RLB-02-CTR-01	The control subsystem shall safely handle unknown obstacle avoidance.	A	Simulate collision situation to analyse avoidance behaviours.
RQ-ATM-02-SEN-01	The sensor subsystem measure drone system status at a minimum frequency of 100 Hz.	I	Inspect the frequency of the sensor readout.
RQ-MFC-01-SNS-01	All Off-The-Shelf (OTS) sensor subsystem components shall have a maximum verified supplier lead time of 4 weeks.	I*	Check with manufactures and distributors whether sensor subsystems are available within time frame.
RQ-MFC-02-SNS-01	The sensor subsystem shall be designed such that a single trained technician can fully integrate onto drone in under 2 hours.	D	Trained technician shows that sensor subsystem can be fully integrated within time frame.
RQ-DCM-01-SNS-01	The sensor subsystem shall be integrated in such way that it can be disassembled by a single technician in under 30 minutes, using standard hand tools.	D	Trained technician shows that sensor subsystem can be disassembled with standard hand tools within the time frame.

11.2. Design Options

During the preliminary design phase, several obstacle detection architectures were evaluated. Unlike the localization subsystem, which was divided into on-board, anchor-based, and external tracking solutions, obstacle detection relies exclusively on on-board hardware.

The candidate technologies fell into three broad categories: active ranging sensors such as mmWave radar and LiDAR, which directly measure 3D spatial information; depth cameras, which reconstruct 3D geometry from stereo or structured-light imaging; and passive 2D approaches such as thermal or RGB cameras, which depend on computer vision algorithms to determine depth and classify obstacles. mmWave radar emerged as the most suitable option. It produces 3D point cloud data at a fraction of the cost and power consumption of LiDAR and depth cameras, and uniquely provides direct radial velocity measurements, a key advantage for dynamic obstacle avoidance. While LiDAR and depth cameras offer denser point clouds that better resolve obstacle geometry, this density is unnecessary for this task of obstacle avoidance. The system only needs to register the presence of an obstacle, not characterize its shape exactly. The low unit cost of mmWave sensors also enables deployment of multiple units, expanding the overall field of view without a significant mass or power penalty.

In the final design, nine mmWave radars are installed. Six on the underside of the arms and three on the upper surface of the body (see Section 15.1). Their orientations are arranged to achieve 360° near-full coverage of the drone's surroundings. Taking inspiration from the same principle used in surgical light-heads, where multiple angled sources eliminate shadows, this configuration ensures that no single arm occludes a sensor's view. Critically, it also guarantees persistent downward awareness, which is most valuable during item pick-up when the drone operates in closest proximity to obstacles and when the mmWave sensors double as the primary means of item detection.

11.3. Technical Background

11.3.1. Point cloud processing

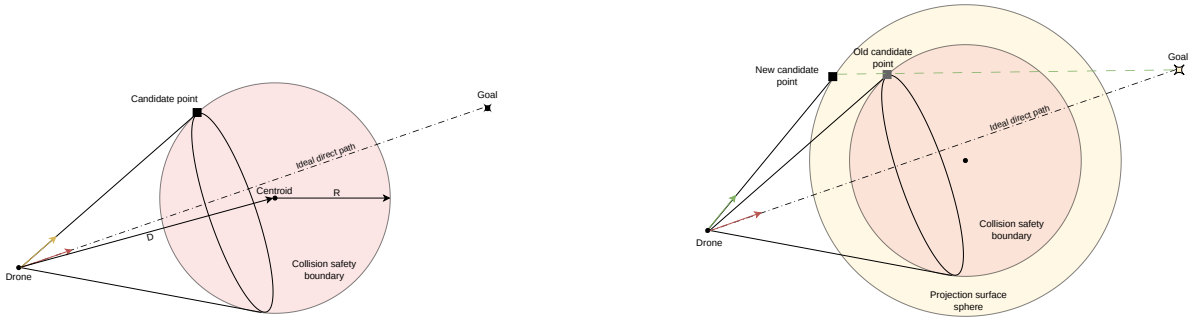
For mmWave radar, common OTS components are made to directly output 2D or 3D point cloud data directly, with Doppler velocity information for each point. In DRIVE-FC, outputs from all nine mmWave radar sensors are fused to construct a point cloud of the surrounding environment in all directions.

Obstacle identification is subsequently performed by clustering this point cloud using Density Based Spatial Clustering of Applications with Noise (DBSCAN). DBSCAN was selected above all for its versatility and noise handling. Firstly it requires no predetermined number of clusters, enabling simultaneous detection and allowing for separation of multiple obstacles without prior assumptions of the surrounding environment. Furthermore, it explicitly models noise, which directly improves detection precision given the susceptibility of mmWave radar to noise.

Each resulting cluster is then approximated as a spherical obstacle, following the common procedure for a

collision cone logic seen further in subsection 11.3.2, with the centroids of the clusters used as estimated positions and the sphere's radius equal to the cluster radius. This replaces the true obstacle geometry with a conservative volumetric representation, refereed to as the Collision Safety Boundary in the subsequent sections. The Collision safety boundary radius is defines as the cluster radius combined with a fixed clearance of 0.9 m, derived from the sum of the clearance posed by RQ-RLB-02-RTG-01 and the estimated drone radius of 0.6 m.

11.3.2. Collision cone



(a) Standard collision cone logic

(b) DRIVE-FC's collision cone logic

Figure 11.1: Comparison of Collision cone logics

Collision avoidance logics exist in many different variations, ranging from Control Barrier Functions to Vector Field Histograms, of which one of the simplest yet effective approaches is the collision cone logic. In two dimentions, the collision cone logic follows that given a detected obstacle, a circle is drawn that fully encircles it. The two tangent lines from the drone's current position to this circle define a triangular region in heading space within which a collision is imminent. The two points where the tangents intersect the circle are then called "aiming points" from which the system may choose to head towards one or another, depending on obstacle relative velocity, subjects kinematic constraints, and other relevant factors.

This translates to 3D quite trivially, with the circle becoming sphere, forming a cone of tangential lines, leading to a set of "aiming points" existing coplanar along a circle. A visualization of the 3D collision cone may also be seen in Figure 11.1a. For multiple obstacles, the combined set of possible candidate points becomes the subset of all aiming points, composed of points not overlapping with any collision cone formed, resulting in a set of potential aiming points like seen in Figure 11.2.

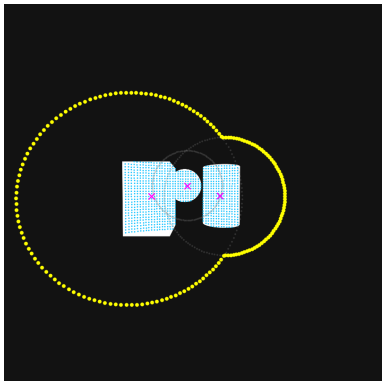


Figure 11.2: Visualization of multi-obstacle aiming point generation

The collision avoidance logic, unique to DRIVE-FC, builds on this fundamental implementation of the collision cone by using a secondary sphere to project the initial candidate aiming point along the line connecting the aiming point and goal. The secondary sphere is defined as a sphere concentric with the collision safety boundary, the radius for which is defined as:

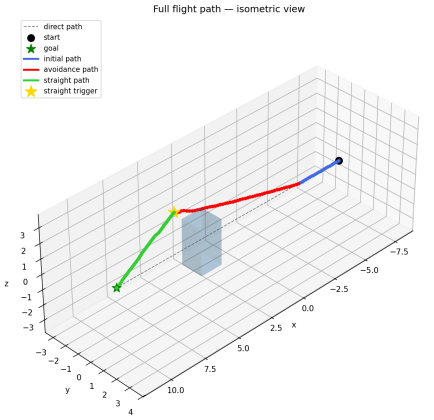
$$R_2 = R + kD \tag{11.1}$$

where k is an arbitrary scaling factor scaling the size of the sphere relative to distance between the obstacle and drone. By doing so, the logic aims to guarantee some level of separation between the obstacle and the collision safety boundary, reducing the spikes in changes in heading, a common phenomenon when in close proximity with the obstacle, near the surface of the collision safety boundary. In later simulations, value of k

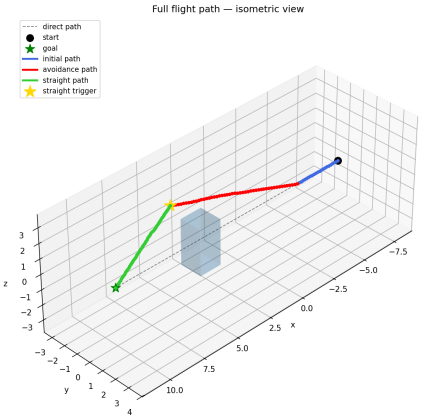
was fixed at a constant of k=0.2 as a result of a tuning campaign. Since oscillation corrections are of larger concern fro DRIVE-FC due to the centre of gravity having a larger moment arm, a mitigation in changes in heading was considered favourable. The visualization is also visible in Figure 11.1b.

11.4. Analysis

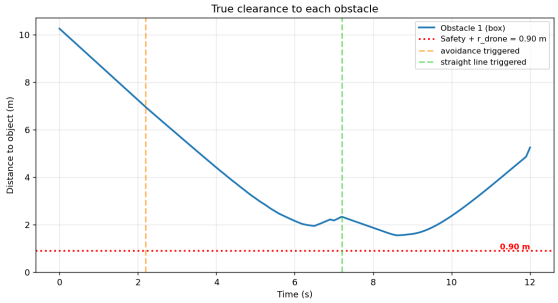
In order to assess the viability of the proposed collision avoidance logic, it was tested in two scenarios: single obstacle case and multi-obstacle case. As reference it was tested alongside the standard collision cone technique, following the method used by Jongho Park [37]. In simulation, key tracked performance metrics include, distance to obstacle, change in heading, and deviation from original direct path. Here distance to obstacle refers to the closest distance from the drone to any part of the obstacle, and change in heading refers to the change in heading between subsequent collision avoidance waypoints, which in simulation is set to match the frequency specified in Chapter 13.



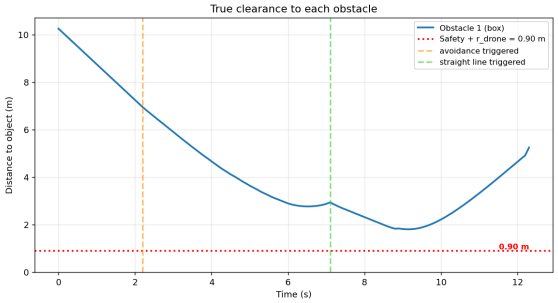
(a) Standard collision path



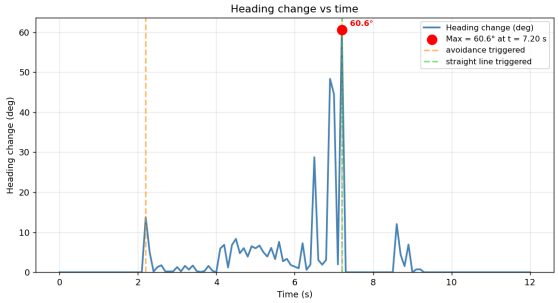
(b) DRIVE-FC's collision avoidance path



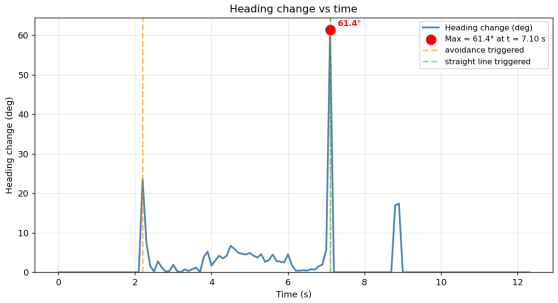
(c) Clearance from obstacles for standard case



(d) Clearance from obstacles for DRIVE-FC



(e) Changes in heading over time for standard approach



(f) Changes in heading over time for DRIVE-FC

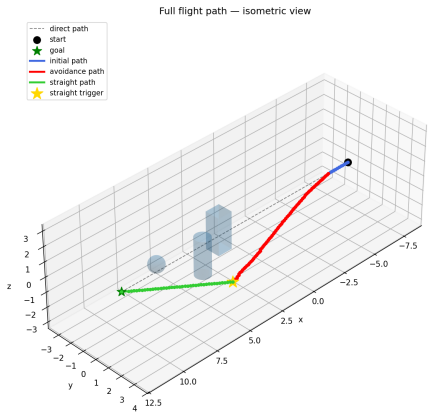
Figure 11.3: Comparison of Single obstacle avoidance performance

For the single obstacle avoidance case, at first glance performance seems rather indifferent, both having followed similar paths, showing similar clearance profiles, and both maximal changes in heading being around the 61°mark. However taking a close look into the performance metrics in Table 11.2 it is apparent that DRIVE-FC's collision avoidance results in a lower average heading change of 3.05°, almost 2.4°lower than the standard algorithm. This is further supported by the qualitative analysis of Figure 11.3e and Figure 11.3f,

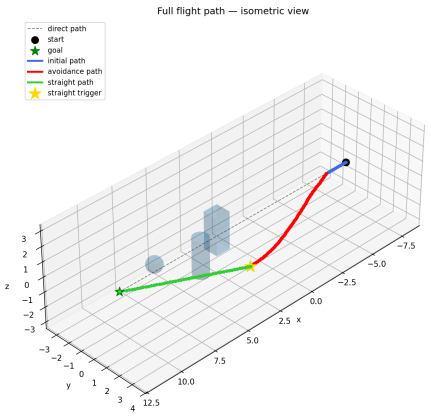
where for the standard algorithm has multiple peaks above or approaching 10°, as opposite to the proposed algorithm for DRIVE-FC which has no peaks above 10°aside from the two peaks at the decision point between avoidance and straight line motion. Both these observation support the more stable nature of DRIVE-FC's algorithm, with less severe jitter in heading. It is important to note that this seems to come at the expense of increased deviation, as both closest-distance-to-obstacle and maximal-deviation-from-path are significantly higher for DRIVE-FC. This can be attributed to the nature of the algorithm placing aiming points at positions further from the obstacles, which in turn also causes a larger deviation from the path. Although not inherently negative, for DRIVE-FC the increased deviation from path may increase the risk of drone to drone collision, thus requires proper attention.

Mode	Closest distance to any obstacle [m]	Average heading change in avoidance phase [deg]	Maximal deviation from original direct path [m]
Standard	1.554	5.43	2.886
DRIVE-FC	1.808	3.05	3.453

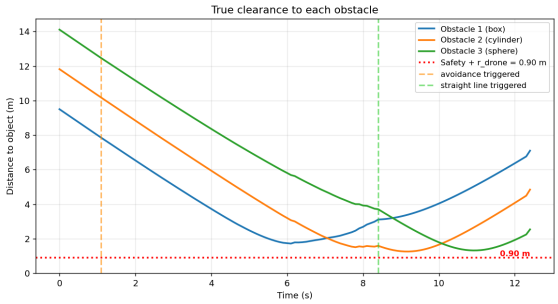
Table 11.2: Performance metrics for single obstacle avoidance



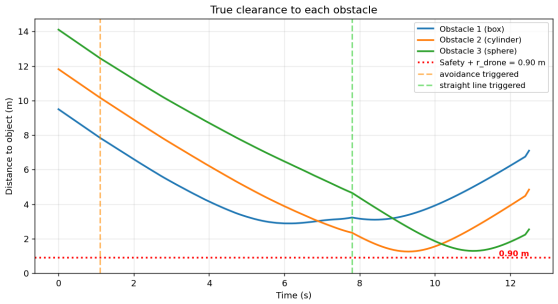
(a) Standard collision path



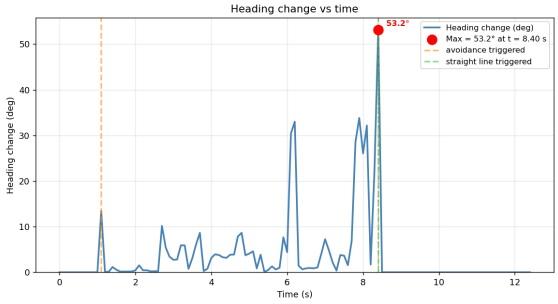
(b) DRIVE-FC's collision avoidance path



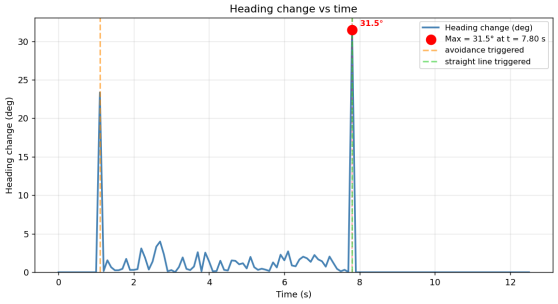
(c) Clearance from obstacles for standard case



(d) Clearance from obstacles for DRIVE-FC



(e) Changes in heading over time for standard approach



(f) Changes in heading over time for DRIVE-FC

Figure 11.4: Comparison of multi-obstacle avoidance performance

In the same manner as the single obstacle case, the performance seen from the multiple obstacle case also supports the more stable nature of DRIVE-FC's collision avoidance algorithm. Much like in Figure 11.3f, in Figure 11.4f it is again visible that peaks while in avoidance manoeuvrer does not exceed more than 10°, unlike for the standard approach as seen in Figure 11.4e. This effect is also visible when comparing Figure 11.4c and Figure 11.4d, where around t=8 seconds right before the straight line motion trigger, the standard approach enters jittery motion, as opposed to DRIVE-FC's motion which suggests a rather smooth motion before straight line motion trigger. This is most likely due to the common issue with the standard approach mentioned previously, as the drone lies near surface of the Collision Safety Boundary. This claim is also further supported by the lower average heading change as seen in Table 11.3. Furthermore, it is worth noting that in the multi-obstacle case, the maximal change in heading for DRIVE-FC is lower than the standard approach at 31.5°as opposed to 53.2°. Though direct causation relation is difficult to prove, this may be due to the larger deviation form the original path(seen from Table 11.3) where DRIVE-FC shows almost 40 cm more in deflection. This combined with the earlier trigger of the straight line motion to goal as observed from Figure 11.4b and Figure 11.4d suggests that the larger deflection earlier on allowed for line-of-sight to the goal to be achieved earlier, allowing for a longer approach requiring less change in heading. On a different note, one interesting difference between the single obstacle case and multi-obstacle case is that for the multi-obstacle case the closest-distance-to-obstacles are indistinguishable from one another, both around the 1.260 meter mark. From Figure 11.4c and Figure 11.4d, it is apparent that both reach the point with reference to the second obstacle (cylinder), possibly making a matter of obstacle geometry and not an inherent nature of the collision avoidance algorithms.

Table 11.3: Performance metrics for multi-obstacle avoidance

Mode	Closest distance to any obstacle [m]	Average heading change in avoidance phase [°]	Maximal deviation from original direct path [m]
Standard	1.260	5.45	2.885
DRIVE-FC	1.258	1.44	3.277

11.5. Verification and Validation

Compliance

The results for the verification of the collision avoidance subsystem can be seen in Table 11.4. Out of the 10 requirements, many of which overlap with the localization subsystem, 4 were deemed verifiable within the scope of this project. The mmWave hardware selected in Section 11.2 has documented operating temperatures between -40°to 85°satisfying RQ-GWH-01-SNS-01 and RQ-GWH-02-SNS-01. The mmWave sensors are also capable of operating at frequencies far beyond what is required by RQ-ATM-02-SEN-01 at 24GHz operating frequency. For RQ-RLB-02-RTG-01, simulations show clearance is assured under both single and multi-obstacle conditions. However due to the lack of kinematic constraints it has been registered as partially compliant.

Table 11.4: Compliance matrix for requirements for collision avoidance. Status: C = compliant, PC = partially compliant, NC = non-compliant, O = open/TBD.

Req ID	Method	Status	Evidence / Remarks
RQ-ENV-01-SNS-01	I	C	Modular sensor hardware - Reusable, supports disassembly requirement.
RQ-GOV-01-SNS-01	A	PC	mmWave sensor within NL regulations - Requires certification for use.
RQ-GWH-01-SNS-01	I	C	$\geq$ -40 °- Verify component temperature range.
RQ-GWH-02-SNS-01	I	C	$\geq$ 85 °- Verify component temperature range.
RQ-RLB-02-RTG-01	A	PC	Simulations verify clearance under multi-obstacle conditions - See Figure 11.4d; kinematic constraints not implemented.
RQ-ATM-02-SEN-01	I	C	250 Hz - Hardware limits far beyond requirement.
RQ-RLB-02-CTR-01	A*	O	Integration test with control systems remains incomplete.
RQ-MFC-01-SNS-01	I*	O	Confirm supplier lead time $\leq$ 4 weeks.
RQ-MFC-02-SNS-01	D*	O	Integration-time target not yet defined (TBD).
RQ-DCM-01-SNS-01	D*	O	Disassembly-time target not yet defined (TBD).

Validation

The previously conducted analysis of the collision avoidance subsystem demonstrates the performance of the subsystem given the artificial environment with idealized sensor outputs due to the lack of publicly available databases. Performance under real life conditions, with drone kinematics implementation would be necessary for a full validation of the system. Since only idealized sensor inputs were used, algorithm performance with

real sensor inputs in an environment representative of the actual warehouse would also have to be tested for full validation. This is to ensure the system works with factors such as sensor noise and environmental effects taken into account. Lastly, operational requirements such as RQ-MFC-01-SNS-01, RQ-MFC-02-SNS-01, and RQ-DCM-01-SNS-01 must also be validated down the line to ensure operational success of the project.

11.6. Limitations and Recommendations

Though in Section 11.4, the collision avoidance algorithm was verified to be capable of providing the necessary alternative path, it is not to say it is without its limitations. One critical limitation is the lack of kinematic constraints when planning the alternative path. The addition of kinematic constraints would not only add realism to the simulation, but by defining kinematic constraints, the planned path may be smoothened allowing to remove some of the slightly jittery behaviour observed in the single obstacle avoidance case. In a similar sense another limitation is the lack of an integrated test with the control subsystem. Without it the generated alternative path cannot be evaluated for applicability in companion with the designed control system for DRIVE-FC, thus is at risk of lacking realism.

On another note, a noteworthy limitation of the collision avoidance subsystem is that it it purely reactive and not predictive, leading to slower reaction times. With reactive collision avoidance systems, the manoeuvrer only acts based on the current state of the environment, thus if the environment contains a fast moving obstacle(such as a fellow drone) it may struggle with providing sufficient clearance depending on the relative velocities between the two objects. For future improvements, systems such as Optimal Reciprocal Collision Avoidance (ORCA) may be preferred which makes better use of the velocity information of the obstacles supplied by the mmWave radar.

Lastly, though not a limitation, one point of potential improvement is the definition of the secondary sphere used for aiming point projection. Currently the scaling factor k is defined as a constant, regardless of the situation, which in implementation manifests itself in the higher deviations from path as well as excessive clearance from obstacles. By changing the static variable to a dynamic variable dependent on factors such as relative velocity between obstacle and drone, or distance to obstacle (making the R_2 a non-linear relation to D), the system may see increases in performance overall. Even if the value remains constant, the value of k may be turned under warehouse-like environments for a more optimal value suited for DRIVE-FC's operations.

12

Communication

This chapter will present the communication subsystem. Section 12.1 will oultline the functions and requirements that dictate the design of the subsystem. in Section 12.2 the considered communication technologies are presented, followed by Section 12.3 where the network architecture and data budget are shown. Section 12.4 outlines the verification and validation procedures with the the compliance matrix. Finally, Section 12.5 focuses on the limitations of the subsystem design while providing relevant recommendations.

12.1. Requirements

The communication subsystem needs to reliably establish data exchange between drones, central computer, warehouse inventory and the ERS system. The following requirements drive the design of the subsystem:

Table 12.1: Communication subsystem requirements and verification plan

Requirement ID	Requirement	M	Method description
RQ-ENV-01-COM-01	The communication subsystem shall use reusable hardware.	I	Check properties of communication hardware.
RQ-GOV-01-COM-01	The communication subsystem shall comply with NL communication regulations.	I*	Check communication system certification.
RQ-GWH-01-COM-01	The communication subsystem's hardware shall be functional in Cold-storage conditions.	I	Inspect hardware operation temperature range.
RQ-GWH-02-COM-01	The communication subsystem's hardware shall be functional in Ambient conditions.	I	Inspect hardware operation temperature range.
RQ-RLB-04-COM-01	The communication subsystem shall communicate collision to the routing subsystem.	I*	Check that communication subsystem communicates collision.
RQ-ATM-02-COM-01	The communication subsystem shall have a minimal bandwidth of 1 Mbps between all communicating modules.	I	Check the bandwidth between all communicating modules.
RQ-ATM-02-COM-02	The communication subsystem shall guarantee a minimal inter-drone communication frequency of 1 Hz.	I*	Check the minimal frequency that is used between drones.
RQ-ATM-02-COM-03	The communication subsystem shall communicate with a expected data loss rate of no more than 10^{-6} at 10 m range.	T*	Measure data loss rate between two communicating bodies by the ratio of bits between sending and receiving.

Continued on next page

Table 12.1 – Continued from previous page

Requirement ID	Requirement	M	Method description
RQ-MFC-01-COM-01	All off-the-shelf communication subsystem components shall have a maximum verified supplier lead time of 4 weeks.	I*	Check with manufactures and distributors whether communication subsystem components are available within time frame.

12.2. Design Options

As part of the design process, several communication technologies have been evaluated, ranging from widely used standards like IEEE 802.11 Wifi, IEEE 802.15.4 UWB to more exotic solutions like a physical cable. The driving criteria for this trade-off were data rate and influence on other subsystems. WiFi proved to be a promising option, showing great data rate and minimal infrastructure needed. Cables offer unlimited data transfer, but were ultimately discarded as such a system would induce a lot of complexity for the routing subsystem while also requiring additional infrastructure. The deciding factor ended up being multifunctionality and UWB was chosen as the the main communication method. The ranging capabilities of the technology used in Chapter 10 outweigh the slight data-rate boost that other methods might offer.

12.3. Design Specifications

For the purpose of this project, the communication subsystem design was largely limited to selecting the method in preliminary design. This section will cover the communication network architecture and the available data budget from the UWB links.

12.3.1. Network Architecture

The communication network is organised in three tiers. At the lowest tier, the drones exchange data exclusively over the UWB link, receiving their assigned waypoints and planned paths and sending back their status. The fixed anchors, in addition to providing ranging signals for localization, they act as a middle point between drones and the central computer. The central computer acts as the hub of the network, it processes drone status, runs the routing solver and interfaces with the warehouse systems like order management or inventory management. A key efficiency of this architecture is that the localization and the communication data channel share the same physical mechanism as mentioned in Chapter 10.

12.3.2. Data Budget

The data handled by the system spans two distinct domains: high-rate streams that remain internal to each drone, and the comparatively low-rate traffic carried over the communication network. The on-board sensor data is processed locally on the flight computer and place no load on the communication subsystem. The communication network itself only carries the routing-related information (waypoints, drone status) over the UWB link. The relevant data rates, are summarised in Table 12.2:

Table 12.2: Communication data budget by link

Link	Data carried	Rate
Over-the-air (UWB)		
Anchors → drones	Waypoints, planned path	6.8 Mbps
Drones → anchors	Status, position	(shared)
Anchors ↔ central comp.	Aggregated traffic	6.8 Mbps
Wired		
Anchors ↔ anchors	Clock synchronisation	∼10 kbps
Central comp. ↔ order mgmt.	Order status	2 Mbps
Central comp. ↔ inventory mgmt.	Inventory state	5 kbps
Central comp. ↔ ERS	ERS availability	5 kbps

12.4. Verification and Validation

Verification of the communication subsystem combines inspection, analysis and testing for requirement compliance. Due to the fact that the hardware used for the communication subsystem is also used for the localization subsystem, requirements RQ-ENV-01-COM-01, RQ-GWH-01-COM-01 and RQ-GWH-02-COM-01

are already compliant. RQ-ATM-02-COM-01 is compliant because of the high bandwidth of the UWB link, at 6.8 Mbps compared to the required 1 Mbps. For RQ-GOV-01-COM-01, the DW1000 module is not certified in any country by the regulatory body governing radio emissions, thus it will have to be certified in future development.

Compliance

The compliance matrix for the communication subsystem is presented below:

Table 12.3: Compliance matrix for the communication subsystem requirements. Status: C = compliant, PC = partially compliant, NC = non-compliant, O = open/TBD.

Req ID	Method	Status	Evidence / Result, Remarks / Action
RQ-ENV-01-COM-01	I	C	Modular UWB hardware, Reusable
RQ-GOV-01-COM-01	I*	O	–, After DW1000 certification
RQ-GWH-01-COM-01	I	C	$\geq$ -40 ° [17], [33] - Verify component temperature range.
RQ-GWH-02-COM-01	I	C	$\geq$ -40 ° [17], [33] - Verify component temperature range.
RQ-RLB-04-COM-01	I*	O	Confirm collision message reaches routing.
RQ-ATM-02-COM-01	I	C	$\geq$ 6.8 Mbps [17]. Req. bandwidth is 1 Mbps
RQ-ATM-02-COM-02	I*	O	Check UWB/TDoA update rate in future testing.
RQ-ATM-02-COM-03	T*	O	Measure data-loss rate at range / under NLOS.
RQ-MFC-01-COM-01	I*	O	Confirm supplier lead time $\leq$ 4 weeks.

Validation

A complete validation can only be performed once a prototype and a representative warehouse environment are available. It must demonstrate the communication subsystem operating in a full mission cycle. The effect of NLOS conditions will have to be tested and validated. Finally, the validation must also confirm that the achieved update rates and latency are sufficient in practice for the control and routing subsystems.

12.5. Limitations and Recommendations

The communication subsystem was largely developed through analysis and several limitations follow from this. First, no dedicated testing was performed, the data-carrying function relies only on the bandwidth analysis done in Section 12.3. While Chapter 10 performed tests with real measurements for the UWB technology, the tests only considered the ranging capabilities of UWB. An extensive test will have to be performed to measure the data-rates and data-loss rates at different ranges and NLOS conditions. Second, the analysis considers link capacity but does not quantify the size of the routing packets that would be sent in operations with a full fleet. An estimate for the routing data size will have to be made. Like for other subsystems, an integrated test in warehouse conditions will have to be performed to properly validate the system.

13

Routing

The DRIVE-FC system is required to operate inside an e-grocery warehouse. This means that a large number of drones will be required to simultaneously navigate a confined space. This section discusses how the pathfinding problem is handled by the routing subsystem.

13.1. Requirements

The routing subsystem requirements are formulated in Table 13.1. The subsystem's primary function can be summarized as: the subsystem must take in orders and drone status, then compute and output instructions for the drones. It should also output order completion status to the warehouse main order scheduler. To ensure efficiency, performance and safety, the subsystem must perform these functions whilst fulfilling the requirements.

Table 13.1: Verification plan for routing requirements

Requirement ID	Requirement	M	Method description
RQ-QLT-01-RTG-01	The routing subsystem shall ensure drop-off height of no more than **30** cm.	D*	Through simulation it can be shown that the drone does not exceed the maximum drop-off height during drop-off.
RQ-WOS-01-RTG-01	The routing subsystem shall ensure a minimum 30 cm separation between drones and human workers.	T*	Test that drone keeps distance from human workers during motion.
RQ-RLB-02-RTG-01	The routing subsystem shall plan collision avoidance paths maintaining a 30 cm distance between known obstacles and drone.	A	Simulate collision situation to analyse avoidance behaviour.
RQ-RLB-03-RTG-01	The routing subsystem shall have emergency landing protocol in-case of single motor failure.	I*	Check presence of emergency landing protocol in case of single motor failure.
RQ-EFF-01	The system shall achieve a daily volume of at least 10,000 successful item transfers per 24 hours.	A	Analyse through usage of simulations whether the drone system is able to achieve a throughput of 10,000 items.
RQ-EFF-01-RTG-01	The routing subsystem shall ensure a maximal average time per item of **2** minutes.	A	Analyse through usage of simulations whether the drone system is able to achieve an average time per item of 2 minutes.

Continued on next page

Table 13.1 – Continued from previous page

Requirement ID	Requirement	M	Method description
RQ-ATM-01-RTG-01	The routing subsystem shall ensure a minimum of 30 cm separation between drones and static infrastructure.	D	Simulate whether drone keeps distance from static infrastructure.
RQ-ATM-01-RTG-02	The routing subsystem shall schedule ERS routing when drone energy is below 10%.	I	Check that drone automatically goes to ERS when energy is below 10%.
RQ-ATM-01-RTG-03	The routing subsystem shall schedule maintenance path.	D	Show that routing subsystem schedules maintenance.

13.2. Design Options

For the routing subsystem, several design options have been considered. Many are have been deemed as infeasible, mostly due to computational time constraints and collision avoidance concerns. The most compelling options have been integrated into the strawman concepts; these options are: Central S2M2 with Large Neighbourhood Search (LNS) and Distributed A* with dynamic collision avoidance.

13.2.1. Central S2M2

S2M2 [12] is a planning algorithm, which is able to solve multi-agent pathfinding problems; meaning that it can compute paths for drones to reach their destinations without colliding with each other or warehouse structures. Additionally, S2M2 can take into account kinematic constraints of the drone, including limited turning angles and limited accelerations. While S2M2 offers many compelling features, the computational time compared to A* is higher and the algorithm is advanced and difficult to implement. In order to help alleviate some of the computational complexity, LNS [25] will be added to the S2M2 algorithm, this makes computation faster, because it smartly decides which agents are going to collide and only routes those agents in a relevant neighbourhood. S2M2 computes the paths for all drones at the same time, combined with the computational effort, a central computer is required to efficiently run this calculation. The central computer then communicates the instructions to the drones.

13.2.2. Distributed A*

A* [53] is a well-known routing algorithm used to quickly find (sub-)optimal paths for single agents moving through graphs. A* is known for its low computational time, while still giving an (near) optimal solution. In addition, A* is relatively simple and already has many tested and verified implementations. This algorithm does require more set-up than S2M2, specifically the warehouse needs to be meshed, as the A* algorithm cannot inherently work with a continuous space. This meshing provides additional opportunities for optimization, however, it comes at a cost of more complexity in the algorithm's implementation. Furthermore, changes to the warehouse layout would require a re-meshing. In addition, A* does not handle any collision avoidance; this is where the dynamic collision avoidance is added. The dynamic avoidance [11] is handled by a vector field based algorithm that takes into account obstacles and objective waypoints to calculated where the drone has to move to avoid collisions. This A* approach gives the possibility of quick and optimal paths in relatively empty warehouses. However, it struggles in areas with more drone traffic, as the dynamic avoidance algorithm is less robust than competitors. One of the advantages of this approach is that the computations are quite quick and simple, meaning that the algorithm can be ran on the drone (distributed) and a central computer is not required.

13.2.3. Final Choice

The final chosen algorithm is Central S2M2. While the algorithm requires more computing power, it allows a higher number of drones to operate in the warehouse due to its safer route planning. This improves the system's ability to achieve high throughputs.

13.3. Detailed Design

13.3.1. Operating Principles

The drone routing subsystem's goal is to provide relevant commands and paths to individual drones that enables them to efficiently route around the warehouse and fulfil incoming orders. The subsystem consists of a central computer that uses a scheduler, Warehouse object database and Multi-Agent Pathfinding Algorithm to generate paths that it communicates to individual drones.

The central computer maintains a database of objects that represent the static layout of the warehouse within which the drones can be routed. Drone positions are updated when individual drones communicate their current relative positions. At other times, their positions are approximated using the routing algorithm's estimates.

The scheduler receives constant orders which it prioritizes and assigns to drop-off points. Afterwards, individual products within an order are assigned to drones.

When the scheduler decides to assign a task to a specific drone, its route is first simulated within the Multi-Agent Pathfinding (MAPF) algorithm. The pathfinding algorithm uses this simulation to create a set of waypoint that the drone should follow to reach its destination. This may include adjustments to the paths of other drones if their paths interfere. These paths are then sent to the individual drones. The pathfinding algorithm continues to model expected drone positions for the purposes of future drones' routing, updating them with real data whenever the drones communicate status updates.

13.3.2. Scheduler

The scheduler is the primary manager of warehouse tasks: it receives orders, positional data and drone energy data with which it determines which actions should be taken by drones to consistently fulfil orders, it handles most of the data transferred between systems within the routing subsystem.

The primary function of the scheduler is assigning products to drones for delivery. Whilst a first-in-first-out ordering of fulfilment has been considered, simulation has shown that this would result in drones spending more than half of their flight time waiting for their turn to perform drop-off. This would result in lowered throughput, less energy efficient deliveries and increased risk of collisions. As such, the scheduler was redesigned to assign products in a rotating order between drop-offs prioritized by distance from existing deliveries. Every drop-off sequentially assigns one of its requested products for delivery, the specific product selected is the one which is farthest from all currently assigned deliveries. Orders that are skipped over for delivery once are prioritized on the next pass of the cycle. In this way, drones are more spread out across the warehouse and drop-off stations whilst still ensuring a minimal order fulfilment time.

This system is illustrated in Figure 13.1. The diagram demonstrates how the products are prioritized over 4 simultaneously active orders over one cycle. Reading from drop-off 1 to drop-off 4: in drop-off 1, potatoes is chosen because it is farther than milk or eggs. In drop-off 2, neither products can be delivered because they are too close to existing deliveries (within drone collision diameter), so it is skipped over but will be prioritized next cycle. In drop-offs 3 and 4, pizza and onions are selected respectively because they are farthest within their orders. When all orders are checked, the loop repeats.

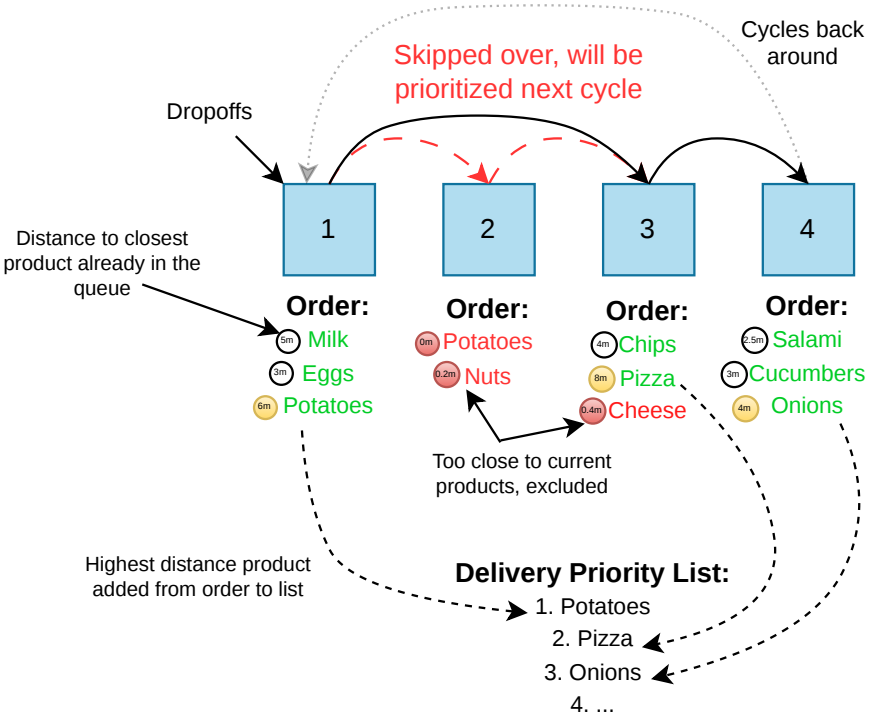


Figure 13.1: Product priority order demonstration

The size of the drone battery is one of the main limiting factors in drone flight time. As such, to ensure maximum average flight time, drones must be routinely recharged at preset ERS stations. At a base level, drones are assigned to recharge once their battery is drained below 25%, by simulation testing, it was determined that this was sufficient to perform a full delivery and return for recharge. However, during long-term testing within the simulation, it was determined that if the drone number below 25% exceeds the number of ERS, it is possible that the drones still waiting for their turn may run out of energy before batteries are swapped. As such, a 'battery balancer' was implemented: this algorithm reserves recharging time slots for 'routine' recharging and uses the remaining free ERS time to spread out drone energy equally by calling drones for recharge early. It attempts to ensure that at any given time, there are no more drones in need of a recharge than the number of ERS stations. This is illustrated in Figure 13.2.

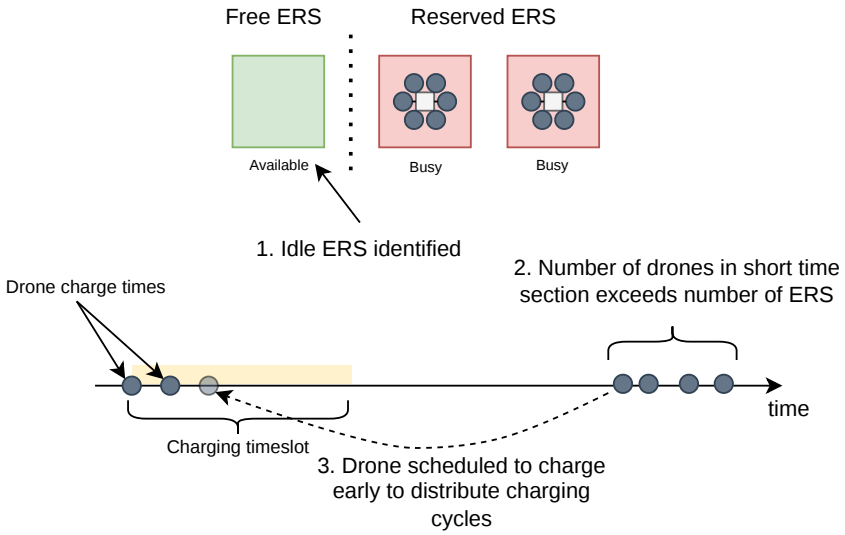


Figure 13.2: Recharge Cycle Balancer demonstration

13.3.3. Pathfinding

Drone paths are given a scaling penalty for all turns, a higher scaling penalty for all turn above 100° from the previous direction and a harsh fixed penalty for all turns above 150°. Additionally, a penalty is applied for jittery paths based on the vector difference between paths segments, this also penalizes rapid acceleration or deceleration, implementing a weak kinematic constraint. All of these factors can be tuned using weights to change their magnitude of impact, these were tuned manually by trial and error until an acceptable path smoothness was achieved. In this case, a path that the control system could follow consistently. This is displayed in Figure 13.3. Finally, gaining altitude and standing still are both penalized to reduce energy waste. This disincentivizes inefficient manoeuvrers and aids with path stability (the consistency of an undisturbed path segments between recalculations).

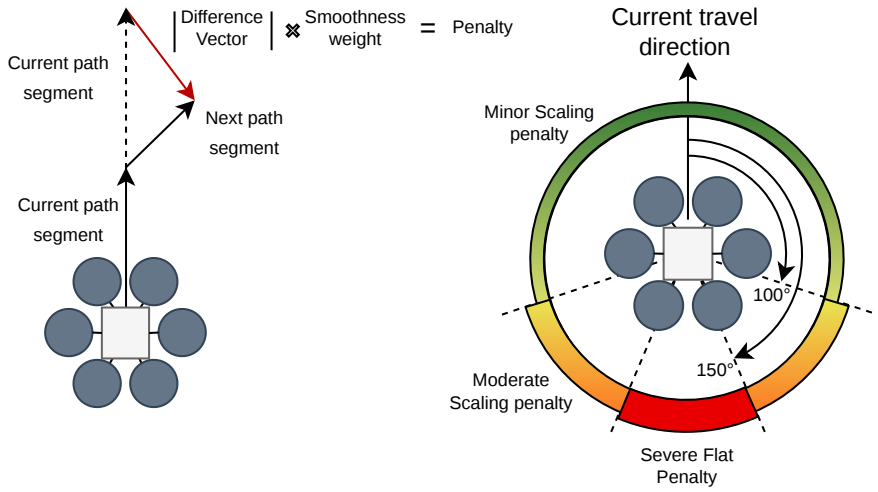


Figure 13.3: Drone path smoothing methods

13.4. Logistical Simulation

One of the most important requirements is the total throughput of the system. In order to estimate the value of the throughput, a high-level logistical simulation of the warehouse and drone system has been made. A secondary objective of the simulation was exporting the generated drone paths, which have been utilized in the design of the control subsystem.

13.4.1. Model Simplifications

Given the limited time and computational power of this pre-prototype design stage, certain assumptions and simplifications must be made to reduce complexity. These simplifications inevitably result in model inaccuracies which must be critically examined and recorded. These simplifications are listed in Table 13.2. Assumed values are listed in Table 13.3.

Table 13.2: Verification plan for sensor requirements

ID	Simplification	Effect	Reasoning
S1	**Drone velocity is to be constant and equal to the expected average velocity: 3 m/s**	This introduces a positioning error during segments of flight where the drone must accelerate or decelerate. Error increases with duration of inaccurate flight speed	Significantly improve computational performance of pathfinding. Error mitigated by the fact that short path segments where the drone cannot accelerate or decelerate in time will also have a lower positional error due to their short duration

Continued on next page

Table 13.2 – Continued from previous page

ID	Simplification	Effect	Reasoning
S2	**Paths are followed perfectly**	The drone control system may fail to perfectly follow the path resulting in a drone failing to avoid an obstacle as the pathfinding algorithm expected	This significantly improves pathfinding algorithm performance and the error is extremely difficult to eliminate completely. The likely effects of this simplification can be mitigated. The control system is designed to operate within a margin of error that is added to the drone size in the pathfinding algorithm. Additionally, in the event of a likely collision, the drone's sensors will attempt to course correct or stop the drone.
S3	**No communication delay**	The communication delay can cause a drone to lag behind expected positions/statuses. This can result in mistimed movement or actions.	The communication delay is miniscule on the time scales necessary for a significant error to occur. Additionally, it can be mitigated by adding an expected delay or exact time of execution
S4	**No sensor noise**	Sensor noise may result in drones failing to detect obstacles, detect obstacles that do not exist or introduce error to positioning data	Out of scope for logistical simulation, error in positioning data is expected and included in simulated drone size margins.
S5	**Pick-up, drop-off and battery swaps are modelled by hovering at a location for a specified amount of time**	Exact position of drone not correctly represented, especially noticeable during shelf access.	This is a conservative simplification that keeps the access area of shelves clear of other drones. Simulating actual pickup/recharging movements is out of scope for this logistical simulation. Pickup/drop-off time is set as a conservative estimate. These values are summarized in Table 13.3
S6	**Drone power consumption scales linearly with weight and altitude changes**	The simulation likely incorrectly estimates battery life and drain rate of drones during specific actions such as acceleration, ascension and payload transport.	Given the lack of accurate power draw data, it is extremely difficult to accurately estimate manoeuvrer energy drain. Given these limitations, it is reasonable to assume a linear increase in power draw with altitude and weight increase
S7	**Item orders follow a uniform distribution over the product categories**	Drones attempt to access all parts of the aisles equally over a long enough time period. Does not take into account potential product placement optimizations	A conservative estimate given that product order frequency would be a known variable under real conditions. Meaning that product placement could be optimized by placing frequently ordered items closer to drop-offs in parallel aisles to reduce congestion in specific aisles.
S8	**Drones are modelled as spheres bounding the entire drone**	Drone shape conservatively simplified, a minor amount of sub optimality is added to all heavily congested paths where drones would've otherwise been able to pass one another if modelled more accurately	This simplification significantly reduces computational load for collision detection. Additionally, this assumption is conservative and adds an extra margin of error by making drone orientation irrelevant to collision detection.

Continued on next page

Table 13.2 – Continued from previous page

ID	Simplification	Effect	Reasoning
S9	**Payload mass is assumed to be equally distributed between 0.5 and 1.0 kg and random**	Payload mass is likely to be inaccurate resulting in minor inaccuracy in power drain. Variance of drone power drain may be under or over-estimated	Given the aforementioned inaccuracy in power drain estimation and lack of accurate product data, the difference made by incorrectly estimating payload mass is negligible. Regardless, to be conservative, the upper range of possible value is assumed.
S10	**The system runs 24/7 without maintenance interruptions**	The throughput values do not completely represent long-term real world values.	Estimating maintenance cycle length and frequency is out of scope for this simulation, this simulation is meant to provide a sustained throughput value which can be used a performance baseline. This baseline can then be easily scaled up or down based on operating time or drone count

All assumed/estimated numerical values described in Table 13.2 are summarized in Table 13.3.

Table 13.3: Assumed parameter values or ranges of values

Parameter	Assumed Value or Range	unit
Payload mass	0.5 - 1.0	kg
Drone velocity	3	m/s
Drone nominal power drain	427	W
Battery replacement time	100	s
Payload drop-off time	6	s
Payload collection time	20	s
Assumed spherical collision radius	0.75	m

13.4.2. Logic

The simulation is made up of a system of interacting objects in continuous space, a scheduler which controls drone actions and a routing algorithm for traversing the space. All random variables of the simulation are kept constant using a randomizer seed to keep results consistent and comparable.

Simulated objects

The simulated objects include drones, ERS, No-Fly zones, shelves, drop-off points and products.

For the purposes of collision detection, drones are assumed to be spherical objects with a diameter equalling their maximum width. This simplification was made due to the relative ease with which collision of spherical objects can be determined. Using more abstract shapes would necessitate either slowing down the whole simulation or reducing the frequency of collision checks. So long as the drone's profile does not extend beyond this sphere, this shape can be considered conservative. The collision shape of shelves is bloated by 0.75 m is all directions such that any point along the bloated perimeter is 0.75 m away from the closest part of the shelf. This adds a margin of error for drone motion due to kinematic constraints.

Products spawn on any shelf object with equal probability, this can be considered a conservative estimate since in a real warehouse, the frequency of specific items being ordered is known; this allows for a more favourable placement of items such as placing frequently ordered items close to the drop-off points.

All actions performed by drones such as performing a product collection, product drop-off and recharging takes fixed time determined beforehand as a conservative estimate. The pickup time is assumed to be 20 s, the drop-off is assumed to be 6 s, and the battery replacement time is assumed to be 100 s.

The simulation additionally tracks all drone's energy and current power draw which is estimated from the baseline hover power and linearly scaled up depending on movement and carried weight. Recharging cycles are tracked and controlled by the scheduler.

Routing algorithm
The routing algorithm uses the object positions stored within the warehouse simulation alongside the start and goal positions provided by the scheduler to compute paths for individual drones. These paths attempt to minimize the net travel time for all drones.

The routing algorithm of this simulation uses a heavily modified version of the S2M2 implementation created by the authors of the S2M2 paper [12].

The planned LNS optimization would allow the algorithm to discern between existing drone paths that must change with new drone paths and those that do not. Due to the LNS method not being fully implemented, whenever a new drone requires a path, all drone paths are re-calculated. This significantly increases computational load and requires additional simplifications to compensate. For example: all drones share time-steps, this is primarily a limitation of computational power and could be changed to increase path quality in the future. Furthermore, to improve the ability of the control system to follow the path, the number of nodes in all paths is minimized.

13.4.3. Verification and Validation of Routing

To verify the compliance with the requirements outlined in Table 13.1, a logistical simulation performance analysis must be performed. These requirements can be grouped into the following analysis categories:

1. **Sustained Throughput**: The payload delivery performance of the system over a sustained time period.
2. **Sustained Charge Cycles**: The ability of the drones to maintain sufficient charge level over a sustained time period
3. **Separation and Avoidance**: The ability of the drone to correctly position themselves and avoid obstacles.
4. **Maintenance and Emergency Protocols**: The ability of the routing algorithm to appropriately react to emergency and maintenance situations to ensure the safety of the workers, drones and products.

The simulation was run for 74 minutes at a time-step of 0.15 s. By experimentation, this was deemed to be sufficient to stabilize start-up variables and serve as an accurate representation of baseline simulation performance.

Sustained Throughput
For the category of Sustained Throughput, The following requirements are relevant:

1. **RQ-EFF-01**: The system shall achieve a daily volume of at least 10,000 successful item transfers per 24 hours.
2. **RQ-EFF-01-RTG-01**: The routing subsystem shall ensure a maximal average time per item of 2 minutes.

The sustained system throughput can be primarily analysed by examining the number of products delivered per day and the average time per product delivery.

However, even if disturbances such as unexpected maintenance cycles, crashes or charge cycle rebalances are neglected, the throughput still regularly fluctuates as new orders are added. Furthermore, due to the random placement of products, over a short time period, this can have a statistically significant impact . As such, the 'sustained' throughput can be determined by examining the cumulative average of deliveries per day. As the total number of deliveries and timescale increases, any individual disturbances and randomness is gradually cancelled out until the sustained throughput tends towards a single value. This is visualized in Figure 13.4 on the right. The left graph of Figure 13.4 displays the daily throughput averaged over 60 s to demonstrate the variance in throughput over runtime.

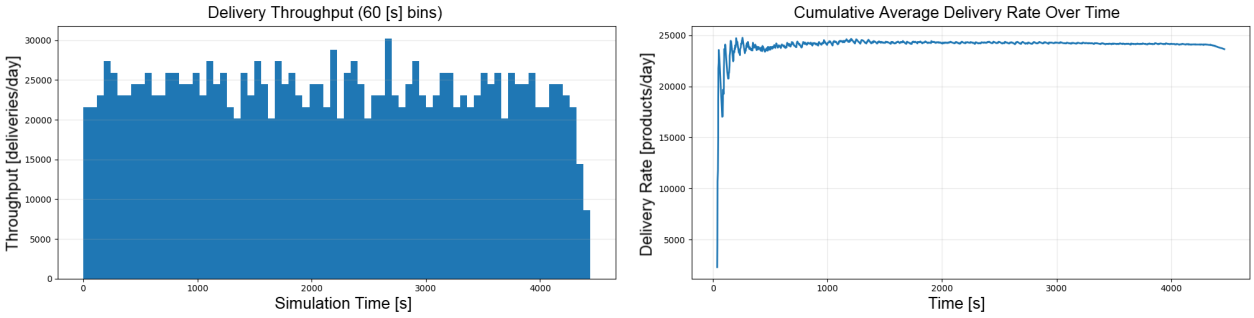


Figure 13.4: Changes in average daily throughput over runtime, collected and averaged with 60 s bins (left) and cumulative average throughput (right) of drone logistical simulation run for 74 minutes

This results in a sustained throughput of 24,003 products delivered per day and average product delivery time of 47.1 seconds, with a maximum time of 68.7 seconds. Note that the cumulative average value declines at the end of the graph primarily due to the simulation shutdown procedures and has no effect during operation. Both the system and subsystem requirements are met with a more than 100% margin.

Sustained Charge Cycles

For the category of Sustained Charge Cycles during routing, the following requirement is relevant:

- **RQ-ATM-01-RTG-02**: The routing subsystem shall schedule ERS routing when drone energy is below 10%.

The ability of the system to maintain a sustained drone energy can be analysed by running at least one full 'charge cycle' of the system. A 'charge cycle' is defined as a time period over which all drones are recharged at least once. This demonstrates that the charge cycle balancer has successfully distributed drone recharge timings to fit all 15 drones without overloading the ERS subsystem.

As a proof of concept, the simulation was run for 74 minutes during which at least three full charge cycle have passed. The simulation was run in the final design concept of the warehouse described in Chapter 5, except that only 2 ERS stations were used for 15 drones. This is a conservative estimate that assumes that one ERS station in each section is undergoing maintenance. Figure 13.5 displays the battery levels over the course of this experiment at expected nominal drain scaled linearly by weight and altitude gain with a constant conservative battery swap time of 100 s. Different colours represent different drones. The segment where battery level is completely flat represents battery replacement time. There is a level of variation between drones due to different carried weights, altitude gains and delays between receiving a recharge order and beginning the battery swap.

As can be seen on Figure 13.5, the initial battery level of all drones was roughly equal, after about 10% (the power loss threshold for charge cycle balancing) of power dropped, the charge cycle balancer began to send two drones at a time to break up the cluster. Once all drones have been charged at least once around 1200 s, at least one slot is generally kept reserved whilst the second is used by the balancer to adjust the system back towards equilibrium.

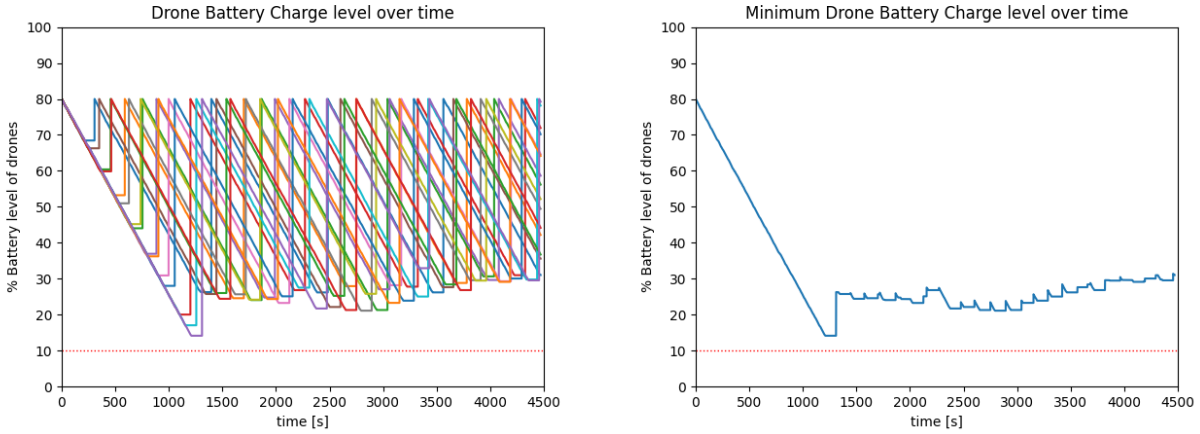


Figure 13.5: All Simulated drone battery levels (left) minimum battery level only (right) over 74 minute runtime. 15 drones, 2 ERS stations

As is clearly visible, over a sustained time scale, the power level never drops below 10%, fulfilling the requirement. The lowest recorded battery level was 14.2 %, this occurred during the initial pre-balance charge cycle. Furthermore, for the purposes of this test, the drones were only recharged to 80% of their maximum capacity, significantly extending battery lifespan.

Separation and Avoidance

For the category of Separation and Avoidance between drones and obstacles during routing, the following requirements are relevant:

1. **RQ-QLT-01-RTG-01**: The routing subsystem shall ensure drop-off height of no more than 30 cm.
2. **RQ-WOS-01-RTG-01**: The routing subsystem shall ensure a minimum 30 cm separation between drones and human workers.
3. **RQ-RLB-02-RTG-01**: The routing subsystem shall plan collision avoidance paths maintaining a 30 cm distance between known obstacles and drones.
4. **RQ-ATM-01-RTG-01**: The routing subsystem shall ensure a minimum of 30 cm separation between drones and static infrastructure.

To evaluate the fulfilment of these requirements, a collision detection system was created and used over the course of the 74 minute simulation. Additionally, 30 cm bloating was applied to both radii of the drone collision sphere and obstacles. The system was setup such that a drone to drone collision was reported whilst a drone to obstacle or drone to No-Fly-Zone collision resulted in a simulation shutdown and collision report.

Over the course of the simulation, 31 drone to drone collisions were detected by the conservative collision detection system. It should be noted that this collision detection system is bloated by a total of 30 cm and is completely spherical, massively overestimating actual drone volume. This is due to the limitations of the iteration between the pathfinding algorithm and goal scheduler where impossible paths are still attempted, the limitation mitigation recommendations are described in more detail within Section 13.4.4. Finally, it should be noted that the dynamic obstacle evasion system was not implemented in this simulation. Zero obstacle and No-Fly-Zone violations were reported. Drone positions were tracked and compiled, discretized and converted into a movement heat-map displayed in Figure 13.6. To reduce drop-off and ERS skewing the movement data, the heat-map only counts drones entering cells rather than time spent within.

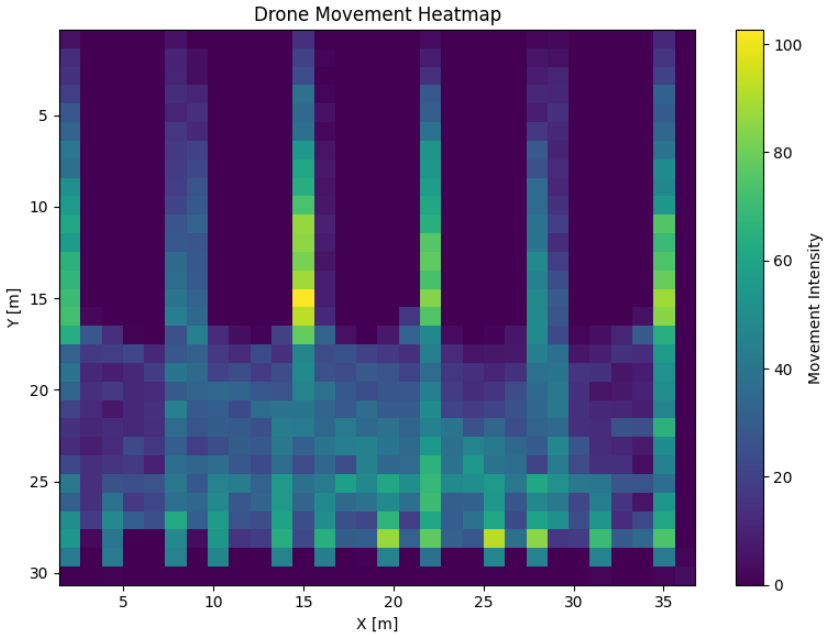


Figure 13.6: Drone movement (counting only number of times drones enter the cell) heat-map over 75 minute runtime. Intensity is normalized and ranges from 0 to 100

The algorithm successfully avoided all static obstacles and reacted successfully to an unscheduled addition of a No-Fly-Zone in the middle of the section, simulating a worker accessing the flight space for maintenance. Within 10 seconds, all drones had vacated the new No-Fly-Zone and all new paths successfully avoided it.

No drones ended their drop-off flight higher than 30 cm above the drop-off station.

Maintenance and Emergency Protocols
For this category, the following requirements are relevant:

1. **RQ-RLB-03-RTG-01**: The routing subsystem shall have emergency landing protocol in-case of single motor failure.
2. **RQ-ATM-01-RTG-03**: The routing subsystem shall schedule maintenance path.

A procedural motor failure generator was deemed out of scope for this simulation due to time constraints, however, manual testing shows that applying a No-Fly-Zone during runtime would allow a drone to successfully land and be avoided by other drones. New paths can be recalculated fast enough to avoid the descending drone. Since the propulsion system is designed with single motor failure in mind, the failing drone can maintain altitude until its flight-space is clear before descending.

On the other hand, a scheduled maintenance path is functionally no different from a regular product delivery path and was thus tested by applying a custom path that is triggered during runtime. This is effectively no different from a regular recharge order and was executed correctly.

Compliance Matrix

Table 13.4: Compliance matrix for the routing subsystem requirements. Status: C = compliant, PC = partially compliant, NC = non-compliant, O = open/TBD.

Requirement ID	Method	Status	Evidence and Remarks
RQ-QLT-01-RTG-01	D	C	Testing in simulation allows precise control of drone drop-off position. The drones consistently stop a distance of 10 cm above the drop-off point, below the 30 cm maximum.
RQ-WOS-01-RTG-01	T	O	Cannot be tested without prototype
RQ-RLB-02-RTG-01	A	PC	The routing system consistently paths around static obstacles but occasionally fails with drones. This is a conscious design decision due to the inability to switch targets based on impossibility of route
RQ-RLB-03-RTG-01	I	C	Emergency landing protocol present but only triggered manually
RQ-EFF-01-RTG-01	A	C	Simulation recorded an average of 47.1 s delivery time
RQ-ATM-01-RTG-01	D	C	Drones successfully avoid static obstacles
RQ-ATM-01-RTG-02	I	C	Drones consistently recharge before reaching sub 10% battery level
RQ-ATM-01-RTG-03	D	PC	The system is able to schedule and route drone towards a maintenance area but due to the lack of reliable maintenance cycle data, this is not done automatically yet.

Code Verification

The simulation and routing code were both extensively tested using a combination of unit tests, extreme value testing and integration testing.

1. Unit tests covered all implemented functions individually with expected typical input values checking for expected and/or manually precomputed outputs

2. Extreme value testing covered individual routing subsystems with extremes of expected input values testing for errors or values outside expected output ranges (ie: timesteps below 0.001 were amplified in pathfinding resulting in division by zero)

3. Integration testing was performed on all routing subsystems during integration into the full simulation

13.4.4. Simulation Limitations

The development of this simulation has revealed a variety of limitations both within the implementation of this simulation as well as within the system itself. Whilst most revealed limitations have already been examined and addressed, some still remain. Most of these can and should be assessed, and/or rectified before any physical deployments or prototypes of the system are attempted.

Due to computational limitations, the simulation was designed to run in variable or fixed frame rate: when consistency is required, the simulation can be run at a fixed frame-rate. This results in overall slower time-flow but ensures all time-steps are equal. On the other hand, for faster simulation, the time-steps can be multiplied or frame-rate set to vary, changing frames as soon as processing is finished. This would not be acceptable in a real system with physical drones since the central computer would be expected to provide consistent instructions on demand. This could be accomplished with a combination of: using a more performant programming language such as C, C++ or Rust, using a more computationally powerful central computer and/or utilizing parallel processing. It should be noted that in a real application, communication and action lag would result in additional delays, making consistent processing even more important.

Another significant limitation of the simulation is the inability to communicate the impossibility/impracticality of a path from the routing algorithm back to the scheduler. Once a goal is assigned to a drone and sent to the routing algorithm, it cannot be changed based on the results. The routing algorithm will still attempt to route a drone to an impossible to reach product, sometimes resulting in inefficient paths or collisions. This is a structural limitation created due to the need to make the scheduler flexible enough to fit any of the routing algorithms/structures tested in the design trade-off. Given additional development time, this could be resolved during real life implementation.

Due to development time constraints, the current simulation routing does not use the LNS method to optimize processing of paths. Any drone path that uses the routing algorithm requires all other drone paths to be recalculated. This results in a significantly higher computational load than was expected for the routing algorithm, necessitating other simplifications to compensate. For example: collision detection is sampled only at specific global time steps and struggles to detect drones that start moving and stop in its path. These limitations could be resolved by fully implementing the LNS method and increasing computational resources of the central computer. This would free up processing time to use a more costly collision sampling method

and give each drone its own dedicated sampling time-steps rather than using global time-steps.

13.5. Limitations and Recommendations

Currently, the routing algorithm structure has been fully designed. However, the implementation still lacks features like LNS and full kinematic constraints. These will have to be implemented for the final, real-world system.

While the current algorithm design satisfies the requirements, the system can be improved. Firstly, the scheduler could use a smarter algorithm, which also considers distance of drones to drop-off and pick-up locations. In addition, the current scheduler does not take into account which drones are going to be free in the future. One recommendation could be implementing features from the continuous-time, life-long, prioritized planning solver outlined by Combrink [15].

14

Control

This chapter will present the control system of the hexacopter drone. Firstly, in Section 14.1, the functions and requirements of this subsystem are described. Subsequently, the design options are discussed in Section 14.2, after which the design choice is described and explained in detail in Section 14.3. Then, the verification and validation procedures are presented in Section 14.4. Finally, limitations and recommendations are discussed in Section 14.5.

14.1. Requirements

The function of the control subsystem is to ensure the drone maintains stable and controlled flight during all operations. The system receives information on the trajectories by routing, and on states from the sensors. It is of essence that the drones are able to process this information to accurately follow the trajectories. To ensure complete and proper functioning, the system must comply with the requirements as shown in Table 14.1.

Table 14.1: Verification plan for control requirements

Requirement ID	Requirement	M	Method description
RQ-GWH-03-CTR-01	The control subsystem shall be able to accurately follow path within 0.15 m lateral deviation when carrying payload.	A	Analyse through means of a simulation.
RQ-GWH-03-CTR-02	The control subsystem shall be able to accurately follow path within 0.15 m vertical deviation when carrying payload.	A	Analyse through means of a simulation.
RQ-RLB-03-CTR-01	The control subsystem shall maintain stability in case of single motor failure.	A	Analyse flight dynamics of single motor failure.
RQ-RLB-04-CTR-01	The control subsystem shall enact forced shutdown upon collision detection.	D*	Demonstrate that control subsystem shutsdown upon collision.
RQ-ATM-04-CTR-01	The drone shall follow given path to destination within 0.15 m margin in altitude.	A	Simulate and show that the drone flies at the predetermined height.
RQ-ATM-04-CTR-02	The drone shall follow given path to destination within 0.15 m margin laterally.	A	Simulate and show that the drone flies at the predetermined position.

14.2. Design Options

Several control strategies were considered for the drone flight control system and evaluated in the subsystem trade-off. Following an initial screening, a number of unsuitable approaches were discarded, leaving primarily linear control methods. Due to the relatively simple flight trajectories, limited development time, and available expertise within the project team, the scope was restricted to almost exclusively linear control approaches.

Although multiple control methods were analysed, only the two methods incorporated into the final design concepts are discussed further. PID control achieved the highest score in the control subsystem trade-off analysis. Linear Model Predictive Control (Linear MPC) was considered for a high-performance concept due to its superior path-tracking capabilities. The characteristics of both methods are discussed in the following subsections.

14.2.1. PID

PID control is widely used across many drone flight control applications. Strong characteristics that are relevant for this mission include its simplicity. It is easy to implement, well established and can avoid obstacles by using, for example, a zero-velocity command. Furthermore, PID's scored moderately in the trade-off for disturbance robustness and path tracking accuracy. Controlling this drone with 6 degrees-of-freedom with a singular PID is not possible. Multiple cascaded loops are needed with gain scheduling to adjust the flight based on the payload condition. Path tracking scored moderately because it is a reactive method and, as such, does not predict errors which may result in overshooting at corners.

14.2.2. Linear MPC

Linear MPC is excellent for path tracking accuracy and obstacle avoidance. This method looks ahead to future waypoints and solves the optimisation problem in real-time. Constraints, such as obstacle avoidance requirements, can be incorporated directly into this optimisation problem. In addition, Linear MPC scored highly for disturbance robustness, as variations in payload are taken into account in the model. Consequently, it achieved the highest performance potential of the considered control methods. However, the method was deemed hard to implement as is requires the derivation of Lyapunov stability functions [10]. For this reason, Linear MPC was considered less practical to implement within the scope of the project despite its superior control performance.

14.3. Cascaded PID

As a result of the trade-off, PID control method was chosen for detailed design. In this section the layout of the constructed PID system will be elaborated upon. Firstly, the drone dynamics are explained, after which the control architecture is described. A cascaded PID architecture was chosen for this design, for which the tuning strategy as well as the resulting gains are presented in the last subsection.

14.3.1. Drone Dynamics

The drone is modelled as a rigid body hexacopter with six degrees-of-freedom, using a state vector of 13 elements: $[x, y, z, v_x, v_y, v_z, p, q, r, q_0, q_1, q_2, q_3]$. Here $[x, y, z]$ are the position coordinates, $[v_x, v_y, v_z]$ are the velocities in the respective axes, $[p, q, r]$ are roll, pitch, yaw rates, and $[q_0, q_1, q_2, q_3]$ are quaternions used to represent the 3D orientation of the hexacopter. Quaternions are used instead of Euler angles to avoid gimbal lock singularities [6].

A body-fixed frame is used as the coordinate system of the hexacopter, x_b pointing towards one of the propellers, z_b pointing down to Earth from the centre of gravity (c.g.) and y_b pointing perpendicular to both x_b and z_b, completing the drone coordinate frame. On the other hand, a North-East-Down frame is used for the warehouse, with x-axis pointing to North, y-axis pointing East and z-axis pointing down.

As the drone will carry payload up to 1 kg, any addition of payload is accounted for in the dynamics of the drone by updating the drone mass and the moment of inertias (I_{xx}, I_{yy}, I_{zz}). When updating the inertias, the parallel axis theorem is used [30]. This is done assuming the payload to act as a point mass 30 cm below the drone's c.g. which is approximated using the tentacle length. I_{zz} is assumed to remain unchanged, as the payload is attached along the z_b-axis.

The full rigid body dynamics of the drone are inherently non-linear and coupled, governed by the translational dynamics (Equation (14.1)) and rotational dynamics (Equation (14.2)).

$$\mathbf{a}_b = \frac{\mathbf{F}_{\text{ext}}}{m} - \boldsymbol{\omega}_b \times \mathbf{v}_b \tag{14.1}$$

$$\dot{\boldsymbol{\omega}}_b = \mathbf{I}^{-1} \left(\boldsymbol{\tau}_b - \boldsymbol{\omega}_b \times (\mathbf{I}\boldsymbol{\omega}_b) \right) \tag{14.2}$$

where $\mathbf{a}_b$ is the translational acceleration, $\mathbf{F}_{\text{ext}}$ is the external force vector which includes gravity and motor thrusts, m is the total drone mass, $\boldsymbol{\omega}_b$ is the angular velocity vector, $\mathbf{v}_b$ is the translational velocity vector,

$\dot{\boldsymbol{\omega}}_b$ is the angular acceleration vector, $\mathbf{I}$ is the inertia matrix, and $\boldsymbol{\tau}_B$ is the torque vector. Here, $\boldsymbol{\omega}_b \times \mathbf{v}_b$ represents the Coriolis acceleration term and $\boldsymbol{\omega}_b \times (\mathbf{I}\boldsymbol{\omega}_b)$ represents the non-linear gyroscopic cross-coupling torque vector.

14.3.2. Control architecture

For the control of the drone, the hexacopter dynamics explained previously are broken down into a series of linear and decoupled systems. This is achieved by the cascaded PID architecture shown in Figure 14.1. It depicts the layers of PID's that are used to fully control the drone movements. For lateral movement, this is visualised as one branch, while in reality this branch is created twice. One for movement in the x direction and one for movement in the y direction.

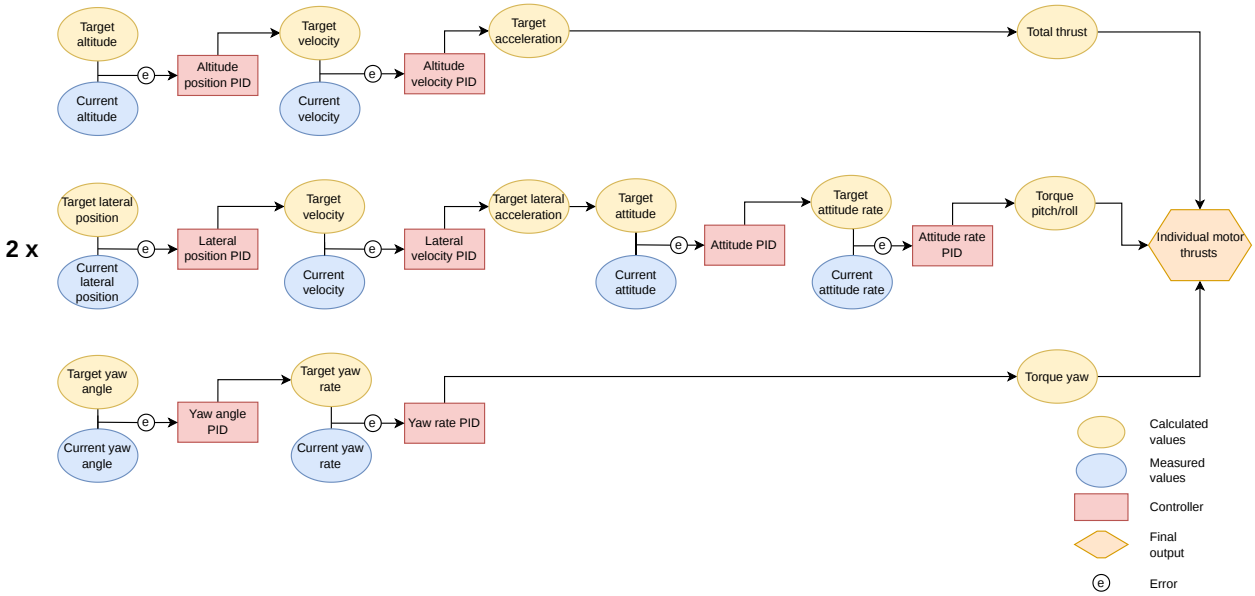


Figure 14.1: Cascaded PID architecture

This control architecture uses a fully cascaded PID loop structure to manage the flight dynamics of the hexacopter across four channels: altitude (z), lateral position (x and y), and yaw heading (ψ). Throughout the architecture, tracking errors (e) are computed by comparing calculated target values (represented by yellow ovals in Figure 14.1) with current values measured by sensors (represented by blue ovals) to sequentially feed the downstream PID controllers (red blocks).

The altitude channel cascades from position control to velocity control to calculate the desired vertical acceleration (a_z). From this, the total net thrust vector (F_{thrust}) is computed by:

$$F_{\text{thrust}} = m(a_z - g) \tag{14.3}$$

where g is the gravitational constant, m is the drone mass.

To control translational movement, the lateral control loop is duplicated to handle the x (longitudinal) and y (lateral) trajectories independently. These loops cascade through position and velocity to generate a target lateral acceleration, which is then mapped into target attitude angled. This is then passed into the inner attitude rate loops to output the corresponding pitch (τ_θ) and roll (τ_ϕ) control torques.

Conversely, the yaw loop operates independently of translation, receiving target heading inputs from the routing system when payload pick-up/drop-off sequences or ERS require an orientation change. This channel cascades sequentially through an angle loop to an angular rate loop, tracking similarly to the pitch and roll inner loops. Note that, because yaw tracking is handled independently, the term "attitude" in Figure 14.1 refers strictly to the pitch and roll channels. Ultimately, the calculated pitch, roll, and yaw torques are combined with the total thrust vector to determine the individual motor commands required for flight.

Furthermore, as illustrated in Figure 14.1, the control architecture positions the outermost loops on the left side, progressively moving toward the innermost loops on the right. In this configuration, the inner loops take the outputs of the outer loops as their reference targets. To ensure system stability and prevent control signal saturation, the inner rate loops are tuned to operate at a significantly higher execution frequency than the outer loops. The sequential tuning of the PID controllers, detailed in Section 14.3.3, is conducted systematically to execute this.

To find the target pitch angle (θ_{target}), and target roll angle (ϕ_{target}) from the target lateral accelerations (a_x and a_y), the non-linear drone dynamics are linearised near a steady-state hover, meaning the total thrust is approximately equal to gravity, leading to the following equations:

$$\theta_{\text{target}} = \arcsin\left(-\frac{a_x \cos\psi + a_y \sin\psi}{g}\right) \tag{14.4}$$

$$\phi_{\text{target}} = \arcsin\left(\frac{-a_x \sin\psi + a_y \cos\psi}{g}\right) \tag{14.5}$$

where ψ is the current yaw heading. As mentioned earlier, the yaw heading is determined by the routing, and during cruise it is set to zero. This simplifies Equation (14.4) to $\theta_{\text{target}} = \arcsin(\frac{-a_x}{g})$ and Equation (14.5) to $\phi_{\text{target}} = \arcsin(\frac{a_y}{g})$.

Flight Modes

During operation it is desired the drone can adapt its behaviour based on the momentary mission at hand. Therefore, several flight modes are created:

- Cruise - Empty
- Cruise - Payload
- Hover
- Take-off & Landing
- Pick-up item & Drop-off item
- Engage with ERS
- One-Engine-Inoperative (OEI)

These modes can be categorised into three main categories for control purposes:

- **Cruise**: Includes all cruise modes with or without payload.
- **Stable**: Includes all modes where stability is prioritised over speed (hover, take-off, landing, pick-up, drop-off, ERS).
- **OEI**: A separate category is created for OEI condition to ensure that drone can safely fly back to be maintained.

Maximum allowed angles, velocities and accelerations may be adjusted based on the wanted drone characteristics. If this is not sufficient, further tuning is necessary. The maximum values for each category can be found in Table 14.2. The values have been determined to ensure safe operations across all flight modes.

Table 14.2: Maximum limits and attitude constraints across flight mode categories

Parameter	Cruise	Stable	OEI
Maximum horizontal velocity [m/s]	3.0	1.5	1.0
Maximum vertical velocity [m/s]	3.0	1.5	1.0
Maximum vertical acceleration [m/s^2]	2.0	1.0	1.0
Maximum tilt angle [°]	30.0	5.0	15.0
Maximum attitude torque [Nm]	2.0	2.0	1.0

Waypoint Interpolation

The trajectories generated by the routing subsystem typically consist of only a limited number of waypoints, each associated with a desired arrival time. Directly using these waypoints as targets for a PID controller would result in discontinuous position commands. Such step changes cause an instantaneous increase in the control error, often leading to aggressive control actions and therefore, overshoot.

To mitigate this behaviour, the waypoint sequence is interpolated into a continuous reference trajectory in the lateral and vertical plane. The interpolation respects the prescribed maximum velocity while ensuring smooth transitions between waypoints. The required acceleration and deceleration profiles are determined based on the maximum velocity, distance of each trajectory segment and the available time between consecutive waypoints. As a result, the controller receives continuously varying position references, improving tracking performance.

14.3.3. Tuning

Having established the drone dynamics and received the trajectories from the routing subsystem, the controller gains were tuned. Due to the complexity of the cascaded control architecture, this process was performed in multiple stages, which are discussed in this subsection. Note that at the stage of tuning, the empty drone mass had been estimated to be 4.3 kg. Thus all tuning has been done with this assumption.

Linear Tuning

First, the non-linear dynamic model was linearised around the hover condition, allowing the equations of motion to be simplified considerably. At this operating point, the drone is assumed to have zero translational and angular velocity, zero roll and pitch angle, and thrust equal to the drone weight. Using the resulting linearised model, together with the drone mass, moments of inertia and desired control bandwidths, preliminary proportional gains were determined for the control loops.

The resulting linearised dynamics were used to derive simplified transfer functions for the individual control loops. Preliminary proportional gains were then selected by specifying desired closed-loop bandwidths and matching the resulting closed-loop transfer functions to the standard first-order form:

$$TF(s) = \frac{\omega_c}{s + \omega_c},$$

where ω_c denotes the desired bandwidth. For the rotational rate loops, the plant is given by:

$$G_\omega(s) = \frac{1}{Is},$$

which yields a closed-loop bandwidth of approximately:

$$\omega_{\text{rate}} \approx \frac{K_\omega}{I}.$$

Consequently, the preliminary rate-loop gains were selected as $K_\omega = I\omega_{\text{rate}}$.

Assuming the inner rate loops are significantly faster than the outer loops, the attitude, altitude and position controllers can each be approximated as first-order systems. An analogous derivation was applied to the remaining control loops, thus resulting in three forms for the preliminary proportional gains:

$$K_{\text{pos/ang}} = \omega, \qquad K_{\text{vel}} = m\omega, \qquad K_{\text{rate}} = I\omega. \tag{14.6}$$

Here, ω denotes the desired closed-loop bandwidth, m the vehicle mass, and I the moment of inertia about the controlled axis. Bandwidth ratios were subsequently used to enforce a separation in dynamics between the cascaded loops, ensuring that inner loops respond faster than outer loops. These preliminary gains provided a stable initial estimate for the subsequent optimisation process.

Hooke-Jeeves Optimisation

The preliminary gains were then refined using the Hooke–Jeeves optimisation method. This derivative-free pattern search algorithm explores the design space by evaluating the effect of parameter perturbations while retaining information on favourable search directions [1]. The method relies on both a suitable initial gain set and a well-defined cost function to efficiently converge towards a satisfactory solution.

Determining the cost function was an iterative process, which consisted of experimenting with different cost functions and analysing the resulting gains using the simulation. The cost function was then modified to improve the performance of the drone where it was deemed necessary. The used cost function is as follows:

$$\text{Cost} = 1 \cdot \text{Tracking Penalty} + 2 \cdot \text{Overshoot Penalty} + 2 \cdot \text{Boundary Penalty} + 0.05 \cdot \text{Jitter Penalty}.$$

Here, the tracking penalty penalises for deviations from the trajectory on average. The overshoot penalty describes the maximum deviation from the trajectory. This is penalised heavily (weight of 2), as overshooting the trajectory in a warehouse environment would cause safety issues. Furthermore, it was observed that for some paths with sharp turns near the edges of the warehouse, the drone struggled to stay within the physical limits of the warehouse. Thus an additional parameter was introduced to penalise for going outside the physical bounds of the warehouse. Finally, the jitter penalty penalises for large changes in motor thrusts to avoid overheating of the motors. This has a relatively low weight due to the scale of the term being larger than the rest. Weights were chosen based on importance to flight behaviour and size of the term such that the cost function is not dominated by one term.

In control system optimisation, there are many local minima to be found. Gains tuned on a single trajectory may provide excellent performance for that specific route while performing poorly on others. To obtain a gain set with good overall performance, the optimisation was therefore evaluated on a dataset of five random trajectories generated by the routing subsystem. This number of trajectories was chosen to balance the improvement in gain results against the increase in computational time, given the project's time constraint. The tuning was performed in a cascaded manner, meaning the inner loops were tuned first, working their way to the outer loops. The result of the tuning for an empty drone can be seen in Table 14.3. Here, it can be noted that most integral gains are zero. This could be caused by the fact that the proportional and derivative terms are enough for stable behaviour, as integral terms only come at play if there are persistent biases present. In addition, it is suspected that the cascaded controller structure already compensates for steady-state errors through the outer loops.

Table 14.3: PID Gains of drone with no payload

Control path	Loop	Controller output	P	I	D
Altitude	Altitude velocity PID	Target vertical acceleration	16.2	0	0.0175
	Altitude position PID	Target vertical velocity	1.94	0	0.116
X lateral motion	Pitch rate PID	Pitch torque	0.572	0.0188	0
	Pitch angle PID	Target pitch rate	5.06	0.0188	0
	X velocity PID	Target pitch angle	5.64	0	0
	X position PID	Target X velocity	2.68	0	0.00375
Y lateral motion	Roll rate PID	Roll torque	1.13	0	0
	Roll angle PID	Target roll rate	5.63	0	0.253
	Y velocity PID	Target roll angle	3.20	0	0
	Y position PID	Target Y velocity	2.05	0	0.00750
Yaw	Yaw rate PID	Yaw torque	0.443	0	0
	Yaw angle PID	Target yaw rate	1.50	0	0

The drone will pick-up items up to 1 kg during the operations which change the dynamics of the drone. These gains were tested for trajectories with payload, however the tracking performance was not sufficient. Thus the gains have been tuned for the drone carrying 1 kg of payload and can be seen in Table 14.4.

Table 14.4: PID Gains of drone with 1 kg payload

Control path	Loop	Controller output	P	I	D
Altitude	Altitude velocity PID	Target vertical acceleration	17.6	0	0.0238
	Altitude position PID	Target vertical velocity	1.84	0	0.0915
X lateral motion	Pitch rate PID	Pitch torque	2.68	0.228	0
	Pitch angle PID	Target pitch rate	6.58	0.0375	0
	X velocity PID	Target pitch angle	6.98	0.00125	0
	X position PID	Target X velocity	3.24	0	0.00375
Y lateral motion	Roll rate PID	Roll torque	4.06	0.209	0
	Roll angle PID	Target roll rate	6.26	0	0.247
	Y velocity PID	Target roll angle	3.20	0	0
	Y position PID	Target Y velocity	2.11	0	0.0100
Yaw	Yaw rate PID	Yaw torque	0.443	0	0
	Yaw angle PID	Target yaw rate	1.50	0	0

As the items to be picked up have varying masses, an interpolated gain scheduling has been implemented. The controller linearly interpolates between the no payload and 1 kg of payload gains and assigns the correct gains depending on the mass of the payload being picked up. This can be done as the payload to be picked up is known prior during scheduling of the drones, and is an input for the control subsystem.

Furthermore OEI condition was considered as per RQ-RLB-03-CTR-01. The controller have been retuned for a flight assuming one motor is not functional. The results of the tuning can be found in Table 14.5.

Table 14.5: PID Gains of drone under OEI conditions

Control path	Loop	Controller output	P	I	D
Altitude	Altitude velocity PID	Target vertical acceleration	16.0	0.0400	0.563
	Altitude position PID	Target vertical velocity	0.688	0	0.169
X lateral motion	Pitch rate PID	Pitch torque	6.08	0.0560	0.0550
	Pitch angle PID	Target pitch rate	7.83	0.0300	0
	X velocity PID	Target pitch angle	2.16	0.0200	0.119
	X position PID	Target X velocity	0.0810	0	0.0830
Y lateral motion	Roll rate PID	Roll torque	8.17	0.0360	0.00490
	Roll angle PID	Target roll rate	19.8	0.0170	0.00250
	Y velocity PID	Target roll angle	2.78	0.0110	0.337
	Y position PID	Target Y velocity	0.956	0	0.338
Yaw	Yaw rate PID	Yaw torque	0.443	0	0
	Yaw angle PID	Target yaw rate	1.50	0	0

14.4. Verification and Validation

In order to verify the requirements, the performance of the tuned gains were analysed through plotting the trajectory tracking performance of the control subsystem. Using the gains defined in Table 14.3, the tracking performance for empty cruise condition is plotted for a path generated by routing, as can be seen in Figure 14.2.

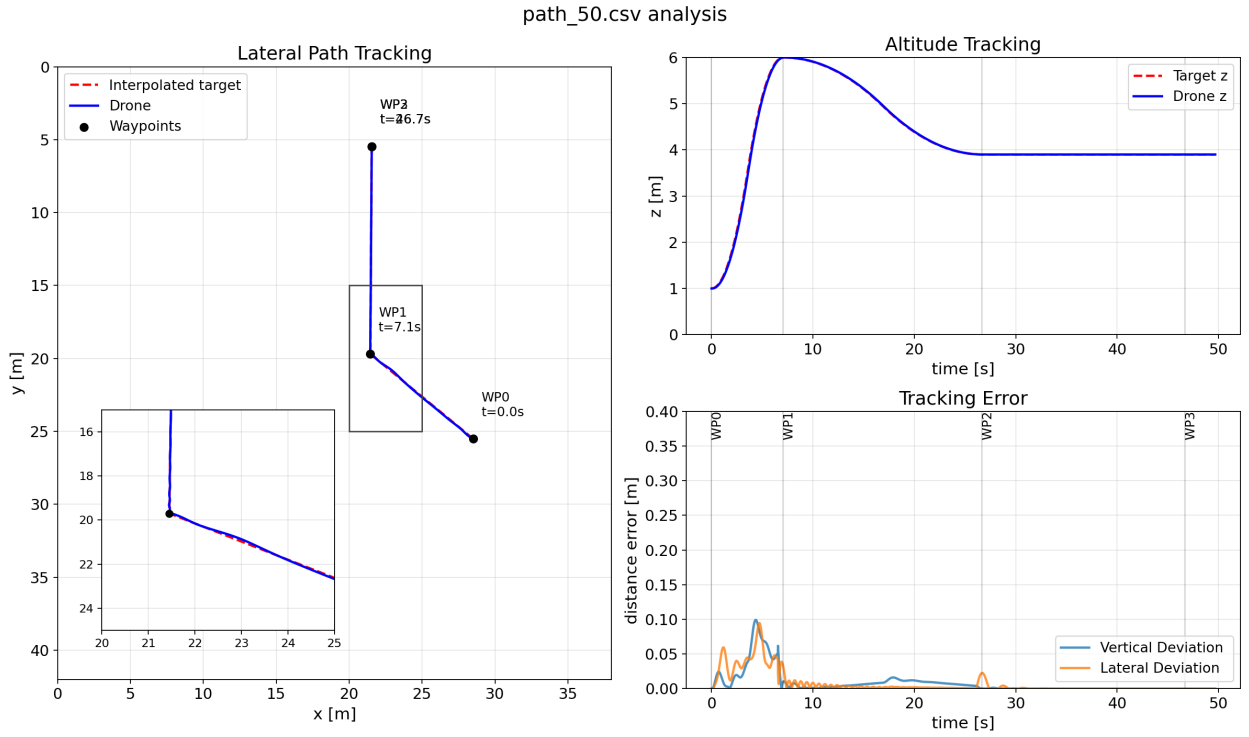


Figure 14.2: Path tracking performance for empty drone

This plot shows shows the lateral path tracking on the left, the altitude tracking at top right, and the tracking error at bottom right plot for a general path generated by routing. The lateral and altitude tracking plots show the full dimensions of the warehouse, with the shelves extending in the y-direction. In the tracking plots, the waypoints and the target path between these paths are shown by red dotted lines. The drone trajectory is shown by the blue lines.

The tracking error plot on the bottom right shows the deviation in both the lateral (orange line) and vertical (light blue line) trajectories. It plots the shortest perpendicular distance from the drone to the interpolated target against time. To comply with the requirements: RQ-GWH-03-CTR-01, RQ-GWH-03-CTR-02, RQ-ATM-04-CTR-01, RQ-ATM-04-CTR-02; the errors should be under 0.15 m.

After analysing these plots for a variety of paths, it is concluded that the control subsystem can follow the trajectories with an accuracy of 0.15 m both in lateral and vertical directions for similar paths to the one shown in Figure 14.2. The same plots have been made for flight with payload as well, where the plots show similar tracking performance with an accuracy of 0.15 m.

However, this accuracy is not maintained throughout all the paths that have been tested. For paths with waypoints which have very short distance between them, such as when the drone is queuing for drop-off, the drone tends to overshoot as it struggles to decelerate to a stop in such a short distance. An example of such path tracking performance can be seen in Figure 14.3.

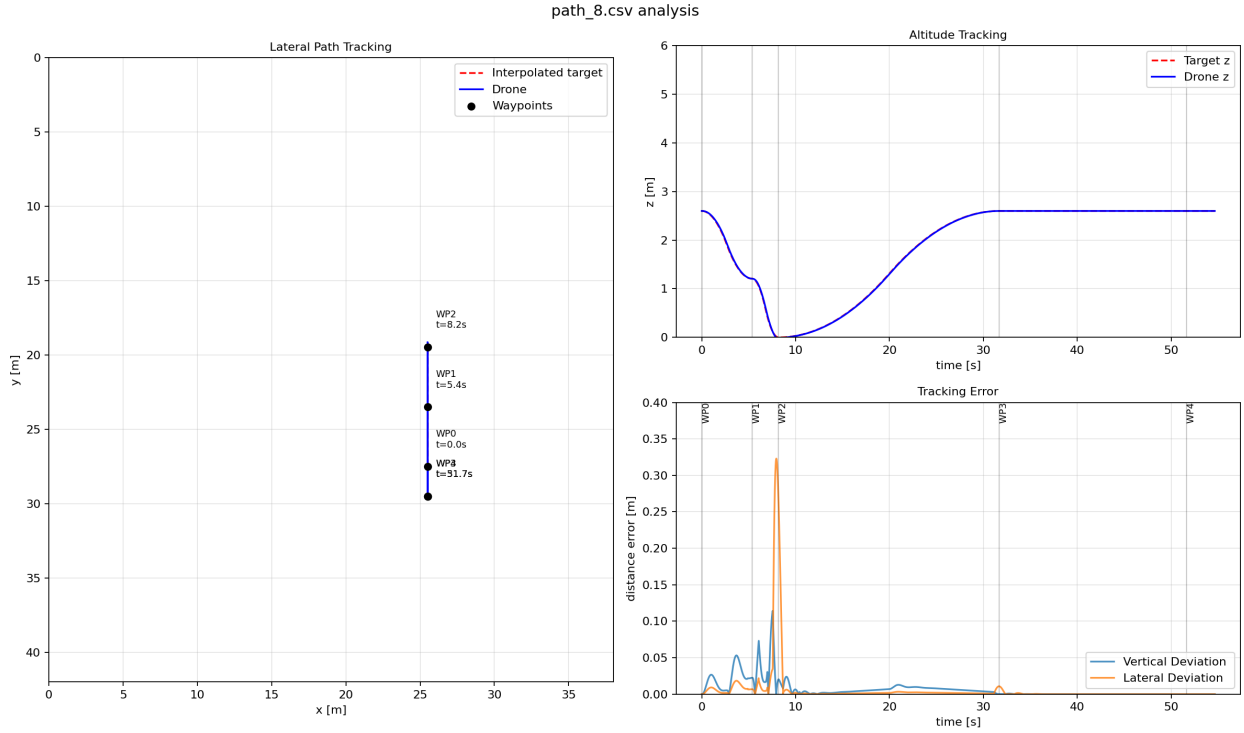


Figure 14.3: Path tracking performace for drone with 1 kg payload

Figure 14.3 shows the path tracking performance for a route generated by routing for a drone carrying 1 kg of payload. Here it can be seen that the lateral deviation is greater than 0.15 m, due to overshoot at waypoint 2. Similar performance was observed for paths with short distances between the waypoints, where the tracking accuracy is not within 0.15 m.

OEI condition

Motor failures are expected to be a significant cause of failures, due to the very high number of flight hours and the associated motor wear. As per RQ-RLB-03-CTR-01, a stable flight is desired in OEI condition. The gains in Table 14.5 have been tested for the paths generated by routing. This revealed that even with the retuned gains the drone cannot follow majority of the paths laterally, although it maintains an acceptable altitude control. This is due to a fundamental limitation in the drone's control authority when an engine fails. With one of its six motors disabled, the drone must continuously fight an unbalanced rolling moment just to stay level. This requires the remaining motors to operate at a high differential thrust, consuming a large portion of their available power. When the path requires lateral manoeuver, the motors have little remaining capacity to create the additional torque needed to pitch or roll into a turn. As a result, the drone cannot achieve the required lateral acceleration to follow the path's curvature, causing it to drift off the path. The altitude controller, however, remains effective because it primarily relies on the total thrust of the remaining motors, which is still sufficient to counteract gravity. This demonstrates that while the drone can survive and maintain altitude in an OEI condition, its ability to perform complex lateral maneouvers is severely compromised. Thus to mitigate OEI condition, the controller must trigger an immediate override upon detecting an engine failure. The system will abandon the active mission path and requests a "safe landing" contingency trajectory from routing. After landing, this area can be made a No-Fly-Zone and the drone can be handled by a worker to be taken to maintenance. It must be noted, however, that the closed-loop bidirectional data interface linking the controller overrides back to the dynamic path-routing engine remains a conceptual architecture. While the individual control gains have been verified in isolation, the real-time software inter-process communication framework connecting these two subsystems has not yet been implemented.

Compliance

For the control subsystem, six requirements were identified of which five were deemed within the scope of this project. The compliance matrix for the requirements can be seen in Table 14.6.

Table 14.6: Compliance matrix for the control subsystem requirements for localization. Status: C = compliant, PC = partially compliant, NC = non-compliant, O = open/TBD.

Req ID	Method	Status	Evidence and Remarks
RQ-GWH-03-CTR-01	A	C	Tracking error within 0.15 m
RQ-GWH-03-CTR-02	A	C	Tracking error within 0.15 m
RQ-RLB-03-CTR-01	A	PC	OEI procedure for landing explained, should implement conceptual architecture to software
RQ-RLB-04-CTR-01	D*	O	Demonstrate that control subsystem shuts down upon collision.
RQ-ATM-04-CTR-01	A	C	Tracking error within 0.15 m
RQ-ATM-04-CTR-02	A	C	Tracking error within 0.15 m

14.5. Limitations and Recommendations

The flight control architecture developed and evaluated within this chapter represents a foundational baseline for the drone fleet. However, as demonstrated during the verification and validation phases (Section 14.4), there are operational boundaries and structural limitations persist. This section details these constraints and outlines actionable recommendations for future system iterations.

Controller Tuning and Optimisation Methodologies

The tracking accuracy of the drone currently exhibits performance variations across different path geometries, indicating that the cascaded PID gains have not yet reached a global optimum. To achieve better performance, several modifications to the tuning methodology are recommended:

- **Cost function refinement:** The resulting gain sets are highly sensitive to the multi-objective cost function definition. Minor variations in penalty scaling constants yield distinct results, thus a sensitivity analysis of the cost function weights is recommended.
- **Expansion of the optimization dataset:** The optimisation algorithm currently evaluates performance across a set of five paths. Training the optimiser on a significantly larger and more diverse path library will guarantee more robust, generalized gains that adapt to various trajectory classes.
- **Alternative optimization methods:** While the Hooke-Jeeves direct search algorithm was selected for its ease of implementation, it is inherently prone to being stuck in a local minima. Future iterations should explore more advanced global optimisation techniques, such as a Genetic Algorithm framework [32].
- **Tuning for final drone mass:** As all design is an iterative process, the gains must be tuned for the final drone mass instead of the initial mass used.

Path Tracking Accuracy

The drone's path tracking accuracy remains highly dependent on spatial separation between waypoints. While the current control system guides the drone accurately along long segments toward warehouse aisles and drop-off points, it exhibits transient overshooting for paths with short waypoint intervals.

To maintain the required accuracy across all paths, it is recommended to categorise the paths based on their geometric characteristics (e.g., segment lengths and turn radius) and isolate tuned gain sets for each class. Integrating a dynamic Gain Scheduling block would then allow the controller to switch the gains based on the path being followed.

Furthermore, to maintain acceptable tracking margins with the baseline controller, the maximum operational velocity was bounded at 3.0 m/s. To increase warehouse throughput and efficiency, a future iteration would be to optimise for faster cruise speeds. The current tuning parameters cannot support a major increase in cruise velocity without comprising the tracking accuracy, as highlighted in the sensitivity analysis (Section 15.5).

Subsystem Integration and Full Simulation

Due to project timeline constraints, several complex components required for full deployment remain unexecuted. The item pick-up/drop-off, take-off/landing, ERS procedures must be modelled in higher detail and paths must be generated for these actions. This will require a more comprehensive tuning of the yaw PID loop, which was only evaluated using a linearized representation of the multi-body dynamics.

Additionally, the baseline simulation operates under an idealised situation where the data received by the sensors is perfect. In physical implementation, there is stochastic noise and drift; consequently, the controller

should be tested to with the modelled sensor noise distributions.

Finally, the simulation currently handles pre-computed path files as a passive input. Real-world execution demands a closed-loop, bidirectional data link between the flight controller and the routing subsystem. This interface is necessary to implement the safety overrides needed for the OEI procedures explained in Section 14.4. Ultimately, a complete software integration combining the path planning, localization, collision avoidance, and control subsystems within a multi-agent environment should be completed to validate the full autonomous fleet design.

15

Final Design

This chapter covers the final integrated design of the DRIVE-FC drone. Section 15.1 goes over the final configuration of the drone and presents a 3D render. Section 15.2 covers the system architecture, looking at the relations and dependencies between hardware, software, electronics and data handling. Lastly, Section 15.3 evaluates whether the designed subsystems fit within the allocated resource budget.

15.1. Configuration Layout

The results of the design process presented in the previous sections is the DRIVE-FC drone. The final design features a hexacopter configuration with vertically unevenly spaced, asymmetric, looped propellers. The flight computer is mounted between the top two panels of the drone body. This is the heart of the system and contains the drone's flight and grabbing controls as well as the IMU and UWB sensors. mmWave sensors are placed on each of the six drone arms and on top of the body. The swappable battery is positioned directly beneath the flight computer. At the bottom of the drone structure, the grabber system is installed. This consists of an arm equipped with a tentacle-like grabbing mechanism. The fully integrated drone is shown in Figure 15.1.

Figure 15.1: DRIVE-FC drone layout

15.2. System Architecture

System architecture refers to the interaction between all components. This section will present an overview of the system architecture divided into hardware, software, data handling, and electrical architectures. Note that all part specific values such as data rate, power and voltage have been estimated based on references mentioned for each subsystem component in previous chapters.

Hardware architecture

Hardware architecture primarily focuses on the connections between key physical components, to distinguish power bus, data bus, and physical connections from one another. As seen in Figure 15.2, the battery is supplying power to the flight computer, sensors, and actuators either directly or through the Electronic Speed Controller (ESC). Similarly, the flight computer acts as the core for data processing and communication, taking in all sensor inputs and instructing actuators. It is worth noting that for sake of simplicity, the voltage converter seen included in Figure 15.5 has been omitted here.

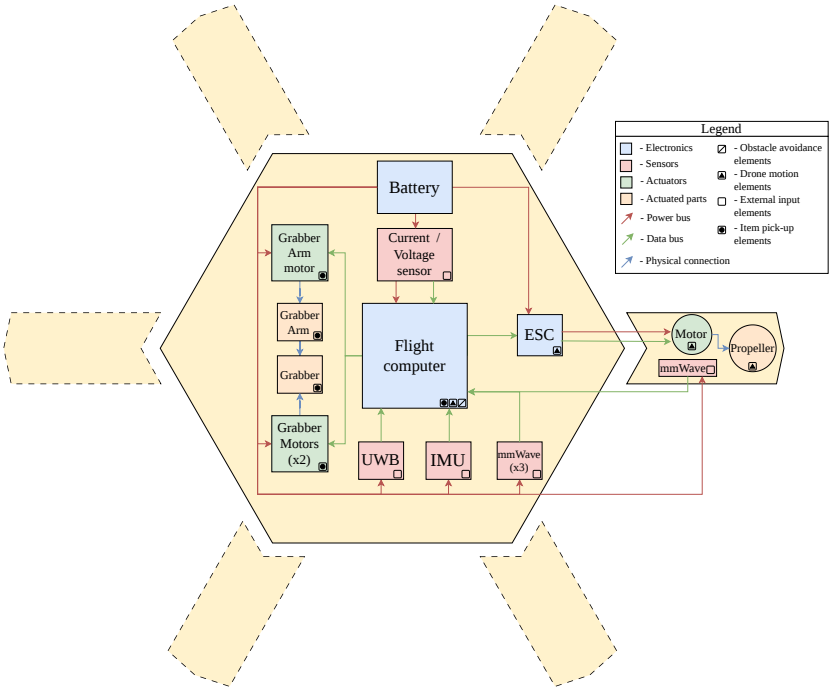


Figure 15.2: Hardware architecture diagram

Software architecture

While hardware architecture emphasises connections between physical components, software architecture outlines the connections between different computational processes. This ranges from data retrieval and processing, to action planning and actuation. Figure 15.3 shows the architecture in the form of a block diagram. In the diagram, one noticeable feature is the bypass from global path-planning to waypoints, omitting local path-planning as a whole. As mentioned earlier in Chapter 11, this stems from the fact that local path-planning only becomes necessary once an alternative path is deemed necessary by the obstacle avoidance algorithm. For regular operations, the bypass may be used instead.

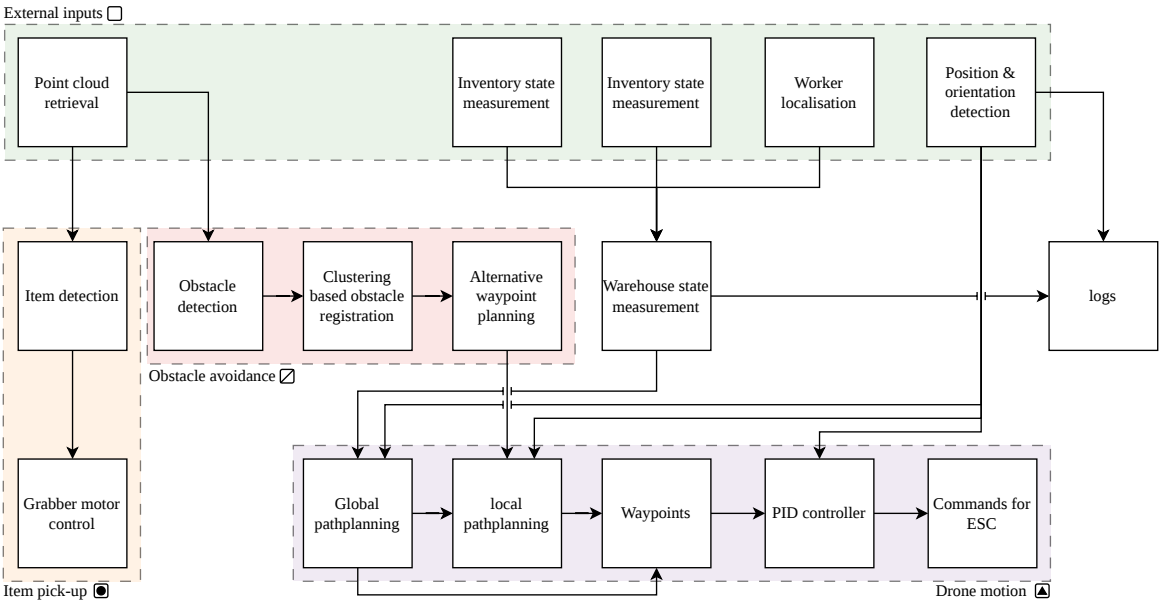


Figure 15.3: Software architecture diagram

Data handling

In the hardware architecture, the data bus was briefly mentioned, only as a type of connection between physical components. The data handling architecture builds on that by refining the resolution, detailing what is communicated as well as the estimated data rate. This can be seen in Figure 15.4. The diagram captures both data handling between internal drone components and key environmental elements surrounding the system. The diagram shows that the UWB module is the only interface between the drone and the external environment as mentioned in Chapter 12. Although this simplifies the communication network, it does pose a potential bottleneck as a failure in the UWB module causes complete loss of communication to the drone. This is partially mitigated by assuring communication over multiple UWB anchors, further mentioned in Chapter 10 and Chapter 19.

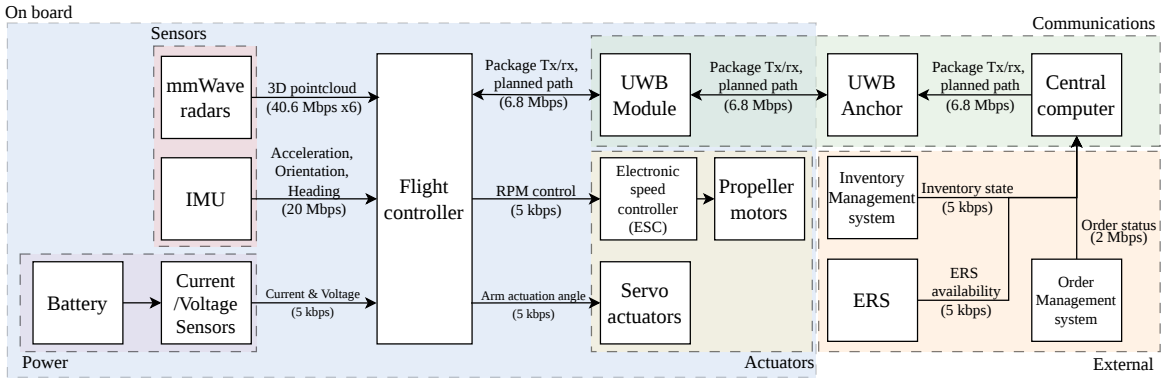


Figure 15.4: Data Handling Diagram

Electrical architecture

The electrical diagram, like the data handling diagram, elaborates on the hardware architecture except now for the power bus, instead of the data bus. Figure 15.5 shows the connections using varying line types, where each type represents a different level in voltage, provided by the DC/DC switching controllers. Though summarized as a singular block, each line with a specific voltage is regulated by a DC/DC switching controller of its own. The reference mmWave radar runs on 3.6V, in contrast to the industry standard of 3.3V. Fortunately, all other sensor components are also rated for this voltage. As for the motors, voltage and power values are derived from the necessary performance as defined by flight profiles and maximal weight to be picked up.

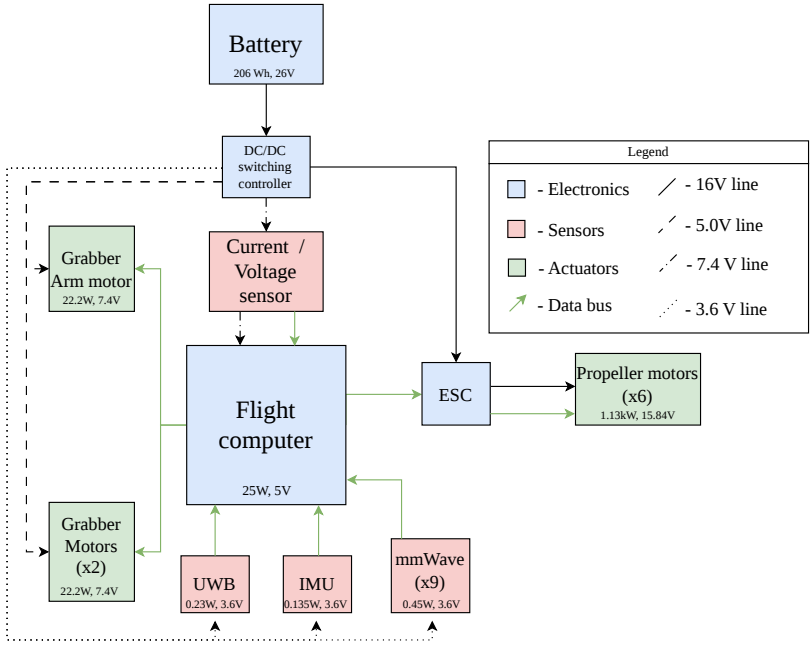


Figure 15.5: Electrical block diagram

15.3. Technical Resource Budgets

Budgets are set up to ensure the design stays feasible and that each subsystem is aware of the limits in which must be designed. This section discusses how the each technical resource budget (mass, power and data) has been allocated and used. In this section, only the subsystems directly relating to the drone body are considered, as these are most limiting.

15.3.1. Mass Budget

As mass has a significant influence on drone performance, a strict mass budget is required. The mass budget is presented in Table 15.1. The allocated total mass was determined based on a payload mass fraction of 0.22 [20]. A mass margin of 20% was included in the allocation. The mass budgets for the body, battery, and propulsion subsystems were estimated by working backwards from the total mass and distributing the available mass based on engineering judgement. From the table, it can be observed that the final drone mass is 1.315 kg lower than initially allocated and 2.915 kg lower than the allocated mass including margin.

Table 15.1: Mass Budget

Subsystem	Allocated budget [kg]	Budget used [kg]	Comments
Body	2.40	2.00	
Battery	2.00	1.82	
Sensors	0.00	Negligible	One sensor is about 10 grams.
Control Computer	0.00	0.045	
Propulsion	2.00	1.14	0.63 kg for motor, 6 x 0.085 kg for propellers.
Grabber	0.80	0.40	
Payload	1.00	1.00	Up to 1.00 kg of payload.
Total	8.20	6.41	
Total incl. Margin (20%)	9.80	–	

15.3.2. Power Budget

It is essential that there is sufficient power for each of the subsystems to work as expected. This requires that, in a worse case scenario, the battery can deliver enough power for each of the subsystems to operate in their max power configuration. The battery has been configured to have 8 cells in series and 3 in parallel, with each cell delivering 3.3 V and a maximal continuous discharge of 52 A. Therefore, the battery is able to deliver a maximum discharge power of 4118.4 W, which is more than is required.

The second consideration for the power budget lies in that the drone needs to be able to fly for a minimum

amount of time. This time has been set at 4 minutes at a worst case scenario, such that the drone is able to safely manoeuvre and return to the ERS. Assuming a depth of discharge of 60% (80% → 20%) the battery has a capacity of 123.6 Wh. With the minimum fly time and depth of discharge constraints the maximum possible power usage in a worst-case scenario is 1854 W.

Table 15.2: Power Budget

Subsystem	Allocated budget [W]	Budget used [W]	Comments
Propulsion	1200	1130	The allocated budget for propulsion stems from an estimate that a drone uses 100 W per kg of force [26].
Sensors	15	4.185	Based on the low power nature of sensors 15 was deemed enough [4].
Communication	5	0.23	UWB is known for its extreme low power usage.
Control	10	25	Flight computers are highly optimized systems, designed to minimize power consumption during usage.
Grabber	100	66.6	The grabber subsystem is a novel solution and there is little literature on power consumption, so a large part of the remaining power budget was allocated.
Total	1455	1226.015	
Total incl. Margin (20%)	1741	-	

From Table 15.2, it can be seen that each of the subsystems remains in budget with the exception of the control subsystem. As the other subsystems are well within their budgets the control subsystem was granted a larger power budget, as such the choice has been made to use an off-the-shelf Raspberry Pi 5 instead of a highly optimized customized flight computer, to reduce costs, project complexity and development time. The Raspberry Pi also allows for on-board obstacle detection and collision avoidance, which is not the case for simple flight computers. This change was discussed with and approved by the system engineer. The overall power used falls well within the budget, allowing for additional power usage for further post-DSE detailed design if necessary.

15.4. Performance Analysis

With the completion of detailed design, the most critical performance parameters must be summarized and analysed as a whole. Parameters are selected based on their relevance to the subsystem and system requirements. These parameters are organized into table 15.3

Table 15.3: Subsystem performance summary

Part	Parameter	Value	Units
Drone body subsystem			
Drone weight	Drone weight	5.41	[kg]
Propulsion subsystem			
Propeller (MID 118)	Propeller radius (r_{prop})	18.75	[mm]
Propeller (MID 118)	Propeller pitch (p_{prop})	7.0	[in]
Propeller (MID 118)	Propeller separation angle (γ)	30.0	[deg]
Propeller (MID 118)	Maximum chord (c_{max})	38	[mm]
Propeller (MID 118)	Design efficiency (η_{design})	0.62	[-]
Propeller (MID 118)	Average efficiency (η_{avg})	0.17	[-]

Continued on next page

Table 15.3 – continued from previous page

Part	Parameter	Value	Units
Propeller (MID 118)	Vertical hub offset	30.4	[mm]
Propeller (MID 118)	Total mass	85	[g]
Propeller (MID 118)	BPF (hover)	53	[Hz]
Propeller (MID 118)	BPF (design speed)	67	[Hz]
Powertrain	$(T/W)_{max}$	≥ 2.0	[-]
Grabber subsystem			
Tentacle grabbing system	Number of tentacles	6	[-]
Tentacle grabbing system	Grabber system mass	0.4	[kg]
Power subsystem			
Battery	Battery capacity	206	[Wh]
Battery	Battery mass	1.82	[kg]
Routing subsystem			
Simulated drone fleet (15 drones 2 ERS)	Throughput	24003	[products/day]
Simulated drone fleet (15 drones 2 ERS)	Minimal recorded battery level	14.78%	[-]
Simulated drone fleet (15 drones 2 ERS)	Average product delivery time	38.7	[s]
Control subsystem			
Cascaded PID	Cruise speed	3	[m/s]
Cascaded PID	Tracking accuracy	0.15	[m]
Localisation subsystem			
UWB localisation (LOS)	Positional accuracy (3D RMSE)	0.04	[m]
UWB localisation (NLOS, 1 obstruction)	Positional accuracy (3D RMSE)	0.09	[m]
UWBlocalisation (NLOS, worst case)	Positional accuracy (3D RMSE)	0.31	[m]
Warehouse subsystem			
Warehouse picking area	Warehouse height	6	[m]
Warehouse picking area	Picking zone width	42.1	[m]
Warehouse picking area	Aisle length	16.8	[m]
Warehouse picking area	Aisle width	2.5	[m]
Warehouse picking area	Pallet width	0.12	[m]
Warehouse picking area	Pallet depth	0.08	[m]
Warehouse picking area	Shelf height	1.9	[m]
Warehouse picking area	Number of shelves	3	[-]
Warehouse picking area	Number of SKU's in picking area	500	[-]

15.5. Sensitivity Analysis

With the design of the system complete, some analysis needs to be done to determine its robustness. Throughout the design process, many assumptions and estimates have been used, which may not be accurate. Furthermore, subsystems have taken performance figures from other subsystems which may not be accurate. To determine how robust the system is against changes in parameters, sensitivity analyses are performed which are explained in the following sections.

The full system will be analysed in its response to changes in system or subsystem parameters. The parameters analysed mostly correspond to the technical resources that were budgeted. These parameters are varied to cover an appropriate range and the effects on the throughput are analysed relative to the current throughput of 24,003 products/day as a percentage increase or decrease. Thus the current parameters are indicated by 0% in the sensitivity analysis tables.

Available Energy

The available energy on the drone is 206 Wh. This energy is primarily used by the propulsion subsystem, although some energy is also used for the flight computer, sensors, communications, and the grabber. Changes in available energy will result in more frequent battery swaps being needed, which increases ERS demand and increases drone downtime.

Since the ERS capacity balancing algorithm used in the routing subsystem allocates drones to swap batteries, an increase in capacity will not change the throughput at the system at all, with drones being sent to swap batteries well before their battery is depleted. Decreasing the battery capacity does slightly decrease throughput, but the main effect is more ERS being necessary.

Mass

Changing the drone mass will have an affect on the power required as shown in Figure 15.6.

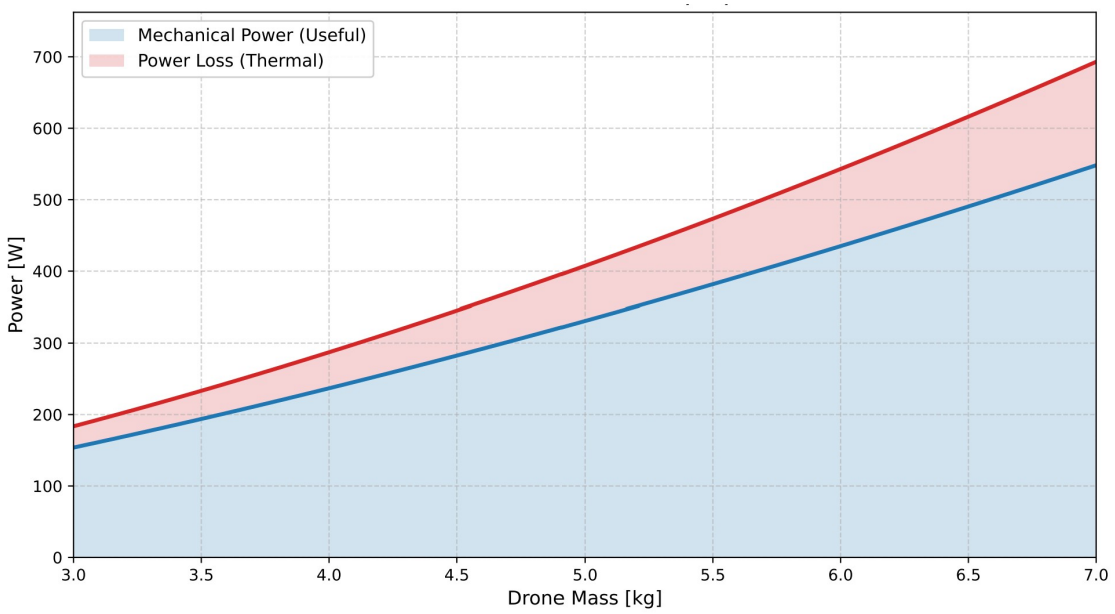


Figure 15.6: Relation between drone mass and power

For a heavier drone, the propulsion system will consume more power, as larger thrust is required to counterbalance gravity. Thus, changing the drone mass has the same effect as changing the available energy explained previously, but inversely, with higher mass leading to lower throughput.

Number of Drones and ERS

The current design has a total number of 15 drones operating in the warehouse to deliver the required throughput within the budget. The number of drones have been varied in the logistical simulation and the relative throughput of each has been calculated as shown in Table 15.4.

Table 15.4: Number of drones and the relative throughput

Number of Drones	7	8	10	11	12	13	14	15	16	17	18	19	20
Relative Throughput [%]	-58	-51	-40	-31	-22	-15	-7	0	7	12	16	26	36

Since the requirement is 10,000 items per day, this could still be met while having eight drones in total. However, since the budget allows for it, 15 drones is chosen to increase throughput massively, which makes more sense financially.

Furthermore the effect of changing the number of ERS have been analysed. The simulations using different numbers of ERS show that there are no significant changes in system performance due to this change. There is a minimum number of ERS below which there is not sufficient capacity to keep all drones powered, which is one ERS per side of the warehouse (ambient and cold). Additionally, a higher number of ERS allows for more frequent swaps. This does increase downtime for drones, but will increase the lifetime of batteries, which may

end up financially beneficial. Due to a lack of quantification of the effect of depth of discharge on battery degradation, no decision could be made based on this effect.

Warehouse Size

The warehouse size will affect the average length of paths, which in turn affects each drone's throughput. To quantify the effect of warehouse size on throughput, the depth and width will be considered separately. The sensitivity analysis of varying the depth of the warehouse can be found in Table 15.5. The same analysis for varying the width of the warehouse by changing the number of aisles can be seen in Table 15.6.

Table 15.5: Warehouse depth and the relative throughput

Warehouse Depth [m]	20	25	31	35	40
Relative Throughput [%]	6	6	0	-8	-10

Table 15.6: Warehouse width and the relative throughput

Number of aisles	8	10	12	14	16
Relative Throughput [%]	6	10	0	-6	-6

It can be seen that a smaller warehouse in both depth and width increases throughput as the drone would fly shorter paths. However as explained in Chapter 5, the current parameters are chosen to fit 500 pallets. If the variety of the items is constrained, then a smaller warehouse layout can increase throughput further.

Pick-up/Drop-off Time

The pick-up procedure has been estimated to take 20 seconds in the current logistical simulation. This value was chosen a realistic estimate considering the full pick-up procedure: flying into the shelf, aligning with the item, engaging the grabber arm and the grabber, picking up the item, and finally flying back into the aisle. As this is an estimated value, a sensitivity analysis is performed to see the effects on the throughput, as shown in Table 15.7.

Table 15.7: Pick-up times and the relative throughput

Pick-up Time [s]	5	10	15	18	20	22	25	30	50
Relative Throughput [%]	37	33	10	8	0	-10	-12	-20	-41

Similar to the pick-up time estimation, the drop-off procedure has been estimated to take 6 seconds in the current logistical simulation. This includes approaching the drop-off point, engaging the grabber to drop-off the item and flying out of the drop-off point. As this is just added time in the simulation, it would have similar results as varying the pick-up time, hence a sensitivity analysis is omitted.

It can be concluded that having shorter pick-up/drop-off times increase the throughput as expected. To make better estimations on these times, a full pick-up/drop-off procedure must be simulated and timed. Depending on the limiting characteristics of the procedures, the system could be improved in these areas to reduce the pick-up/drop-off times.

Battery Swap Time

Similarly to pick-up/drop-off times, the battery swap time has been estimated to take 100 seconds. This estimate can be improved by simulating the battery swap procedure. The sensitivity analysis of varying the battery swap time can be seen in Table 15.8.

Table 15.8: Battery swap times and the relative throughput

Battery Swap Time [s]	10	100	200	400
Relative Throughput [%]	7	0	-11	-15

Here the battery swap time has been varied across a large variety of times and the effect it has on throughput it minimal compared to the effect of pick-up/drop-off times. This is due to drones only having to swap

batteries every 17 minutes, whereas pick-up/drop-off procedures taking place with every order. Hence small improvements in the battery swap would have little to no effect on the throughput.

Cruise Speed

Cruise speed is another parameter that affects the throughput. In the current simulation it is set to 3 m/s to be able to achieve the required path tracking accuracy. The effect of changing the cruise speed on the throughput can be seen in Table 15.9.

Table 15.9: Cruise speed and the relative throughput

Cruise speed [m/s]	1	2	3	5	8
Relative Throughput [%]	-49	-18	0	20	35

This shows that higher throughput can be achieved by increasing cruise speed. However the current controller cannot track the paths with the required accuracy at higher cruise speeds. A plot of maximum deviation against the cruise speed is shown in Figure 15.7 for the same path as Figure 14.2. Throughout this analysis, the controller gains and maximum limits are kept the same as detailed in Chapter 14.

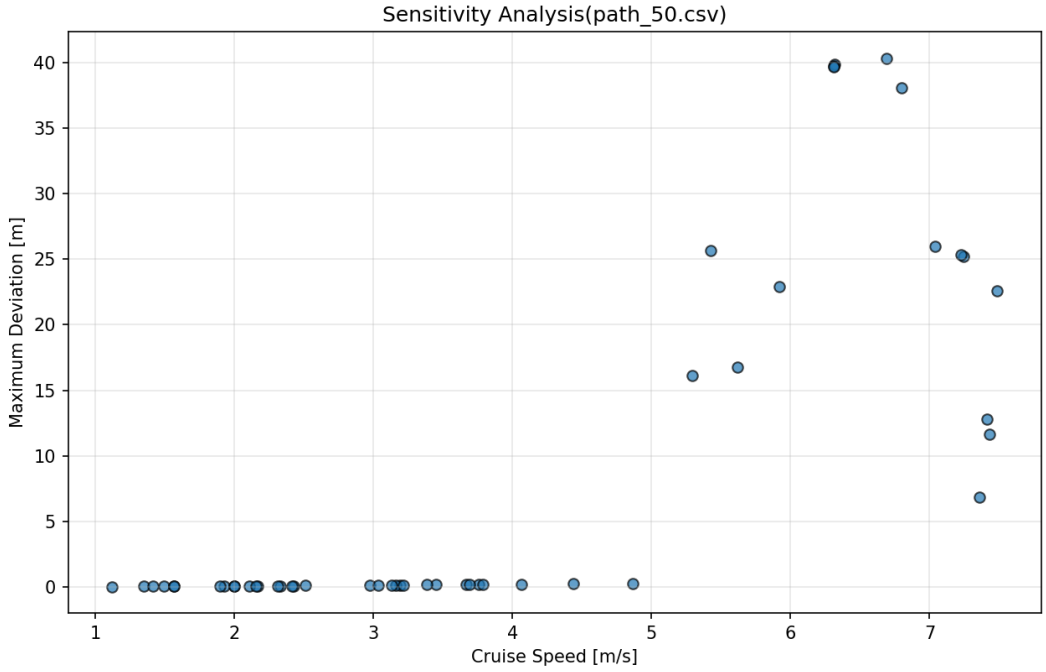


Figure 15.7: Sensitivity Analysis for changing cruise speed

Figure 15.7 shows that the maximum deviation increases slightly up until a cruise speed of 5 m/s. The maximum deviation observed until this point is approximately 0.3 m. Going above this cruise speed, the maximum deviation increases greatly. This behaviour is driven by the maximum tilt being limited to 30°, thus the drone cannot slow down in time and and flies past the waypoint, leading to a large deviation. Note that this is plotted for a path with a sharp turn to demonstrate the worst performance.

Additionally, it should be noted that the controller gains were not dynamically retuned to compensate for the varying speeds in this specific study. While a retuned gain framework would theoretically optimise tracking metrics, initialising a multi-variable tuning optimization was omitted due to project timing constraints. The current fixed-gain sensitivity analysis demonstrates the performance of the current control architecture for varying only the cruise speeds.

16

Cost Analysis

The previous chapters established DRIVE-FC's technical performance across multiple performance metrics such as throughput, positioning accuracy, and system reliability. However, the technical performance alone does not guarantee the successful deployment of the system. The commercial outlook of the project must be taken into account for a full feasibility analysis. With RQ-BDT-01 and RQ-BDT-02 being critical requirements for this project, the project must be thoroughly analysed to stay within these margins. This chapter evaluates the economic viability of the DRIVE-FC system, first through a cost-per-unit breakdown in Section 16.1, followed by a full cash flow analysis in Section 16.2, and finally a sensitivity analysis of the commercial outlook in Section 16.3.

16.1. Cost per Unit

As bounded by RQ-BDT-02, the cost per unit (cost per drone) for DRIVE-FC is a crucial measurement of the projects financial performance. For the drone, many components contribute to the total cost of a singular drone, spanning flight-critical hardware such as the flight controller, ESCs, rotors, and motors, to sensing and navigation subsystems including the UWB modules, mmWave radar, and IMU, through to structural elements such as the frame and propellers, and electronics including onboard flight computer and circuitry. Each component, while for some parts costs may be low in isolation, must be accounted for in full, as cumulative costs snowball rapidly into a non-negligible sum. In Figure 16.1 costs are broken down by piece, with each part's price based on either prices of off-the-shelf component references, or estimated manufacturing and material costs.

Drone unit price(excl. battery)			
Component	Price	Quantity	Total price
SEN0395 (mmWave)	€30.26	9	€272.34
DWM1000 (UWB)	€15.97	1	€15.97
MPU6000 (IMU)	€17.95	1	€17.95
Electronic speed controllers (ESCs)	€23.99	1	€23.99
Tentacle motor	€43.44	3	€130.32
Raspberry pi 5	€94.60	1	€94.60
Tentacles	€3.50	1	€3.50
Aluminium rod	€5.32	1	€5.32
Bolts	€5.00	1	€5.00
Cables	€0.54	3	€1.62
Arm	€20.00	1	€20.00
Aluminium sheets	€5.31	1	€5.31
Power converter	€23.50	4	€94.00
Drone motors	€83.71	6	€502.26
Assembly	600		€600.00
			-
			-
			-
			-
			-
			-
Total			€1,792.18

Figure 16.1: Breakdown of unit cost of drone

It is important to note that battery costs have been excluded from this cost-per-unit analysis as batteries undergo periodic replacements, and thus were considered part of CAPEX and Operational Expenditure (OPEX). Likewise, manufacturing labour costs have also been included into CAPEX, labelled as part of total manufacturing cost, as it scales weakly with production volume at low quantities. Assembly labour cost, by contrast, has been included into cost-per-unit as it scales directly with the quantity of drones to be produced.

This brings the total per unit cost to €1,792.18, far below the limit of €10,000 from RQ-BDT-02, with approximately 82% headroom. The substantial margin allows to absorb any price fluctuations or minor design oversight that may arise. As visible from Figure 16.1, the main cost drivers are the motors for the propellers and motors, which on their own totals to around €630.58. This is expected, as these elements are mission critical and a lower cost option severely impacts the performance of the system as a whole.

16.2. Cash Flow Analysis

Having established the per-unit cost of the drone, the following section extends the analysis to the full financial picture of the DRIVE-FC project, and the commercial outlook it presents when in operation. In doing so a cash flow analysis was conducted for which the list of inputs and assumptions can be seen in Chapter A. Prior to discussing the full extent of the cash flow analysis, key assumptions and estimations must be highlighted. Most importantly, with the majority of DRIVE-FC's economic value coming from saved labour costs, average wage growth rate was considered a crucial factor. Based on the Dutch wage growth rate during the period of 1991-2026[34], a fixed rate of 4.3% has been selected for the analysis. Additionally another key factor is the Weighted Average Cost of Capital (WACC), for which 8% has been assumed based on various industry averages presented by KPMG[21]. The life span of the project was estimated to be around 10 years, based on the Life-Cycle Assessment (LCA), with a ramp factor of 100% across all years for simplicity of calculations. Finally for CAPEX, a contingency margin of 20% has been added to the estimated CAPEX value, to account for possible unaccounted for costs or deviations in price, which brings the total CAPEX to €1,169,360.45, just under the restriction set by RQ-BDT-01. Details of the cash flow analysis can be found in Chapter A. From the cash flow analysis, key financial performance metrics were extracted, summarized in Table 16.1.

Net Present Value [$]	Discounted payback years	Savings [%]
€11,885,989.37	2	27.01%

Table 16.1: Financial performance metrics

Three key performance metrics have been selected for the financial analysis, namely those seen in Table 16.1: Net Present Value (NPV), discounted payback years, and savings relative to manual labour. NPV is the sum of all projected cash flows over the systems lifespan, discounted to the value at present day, representing the financial gain of the investment in today's context. The discounted payback period is the number of years (rounded up) required for the cumulative discounted cash flows to recover the initial investment. Lastly, the savings relative to manual labour is calculated from the ratio between the present value of estimated manual labour costs with no drones, and the net present value of the project.

Based on the key metrics, it is evident that even after considering WACC, the discounted payback period is in just 2 years, with a NPV of near 12 million euros considering the 10 year project lifespan, resulting in a 27.01% saving in operational costs total. This significant cost saving can be attributed to the high system throughput of nearly 25,000 items a day, allowing to automate the tasks of 27 full time employees with a fleet of 17 drones (including the extra drones in case one goes into maintenance). All of these metrics support the claim that not only is DRIVE-FC a financially feasible project, it presents significant financial benefits for the client, given the high output that can be achieved.

16.3. Sensitivity Analysis

The financial analysis of DRIVE-FC presents a fairly positive outlook on the economic impact of DRIVE-FC in action. However, to assess the validity and volatility of the analysis conducted, a sensitivity analysis was conducted. For the sensitivity analysis 3 major variables were considered, namely CAPEX, OPEX, and system throughput. The full extent of the sensitivity analysis can be seen in Chapter A.

The sensitivity analysis revealed that a change in CAPEX results in relatively marginal changes to the discounted payback period, only increasing to 3 years if the CAPEX exceeds €1,450,000. On the contrary, the sensitivity analysis revealed that OPEX is a highly sensitive variable, highlighted as a dominant source

of financial risk. One critical value that was revealed through the sensitivity analysis is that if the OPEX exceeds €5,500,000 the project no longer is profitable. Moreover, if the OPEX were to increase by more than €250,000 from the current estimate, the payback period immediately crosses over to 3 years, with 3.81% reduction in savings in the 10 year span. Finally, throughput, perhaps counter-intuitively, turned out to be not as sensitive of a variable as one would imagine. Even at an annual drone system throughput of 10,750.000 the payback period remains 2 years with the saving increasing by roughly 4%, despite the near 25% increase in throughput.

17

Manufacturing Assembly and Integration

This chapter will present the manufacturing, assembly and integration procedures to be followed to prepare the system up for the operations stage. Each custom part will be assigned a manufacturing method or process as well as a material. OTS parts will either mention the material or the exact model.

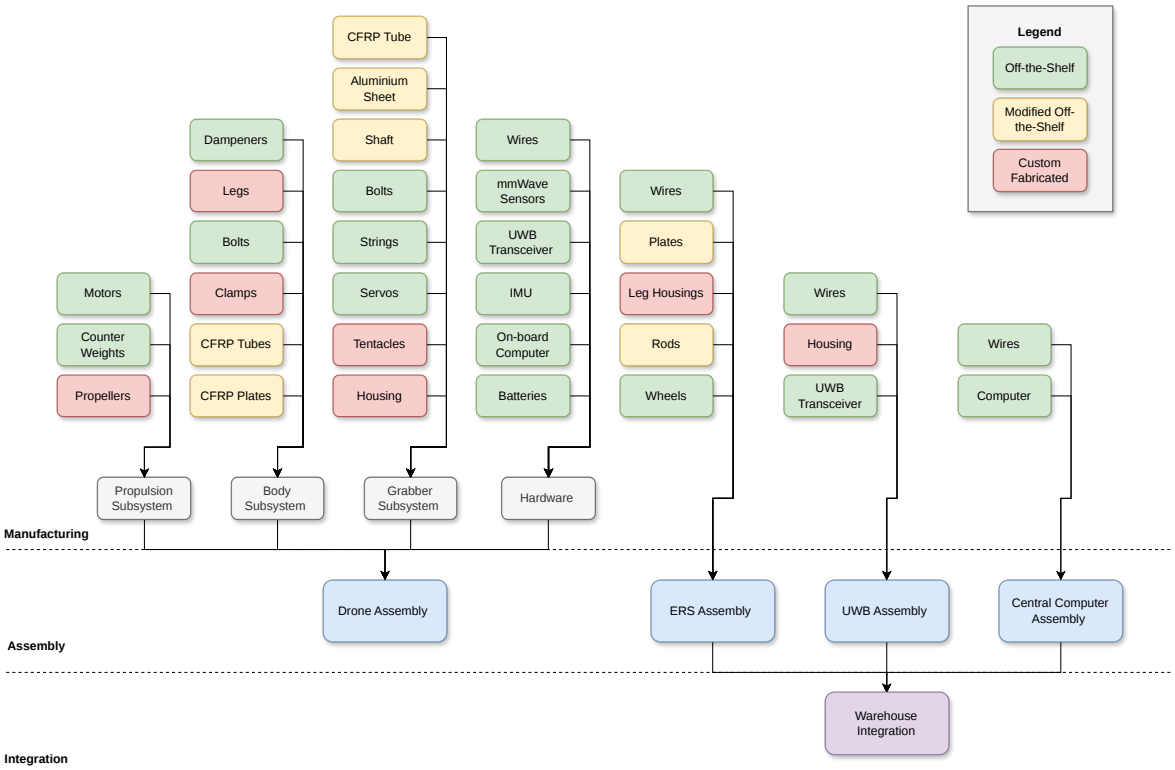


Figure 17.1: Manufacturing Assembly and Integration Plan

17.1. Manufacturing

The following table will outline most of the parts used in the drone assembly, ERS, UWB beacons and the central computer.

Table 17.1: Manufacturing and procurement plan per component

Part	Category	Method / Process	Material/Model
Propulsion subsystem			
Motors	OTS	Procured	Neumotors 3810/17/279
Propellers	Custom	3D-printed / Injection moulded	Nylon CF
Counter weights	OTS	Procured	Stainless steel
Body subsystem			
Plates	Mod. OTS	Sheet stock, cut, drilled to shape	CFRP
Tubes	Mod. OTS	Tube stock, cut to length	CFRP
Legs	Custom	CNC-machined	Aluminium
Dampeners	OTS	Procured	Rubber
Bolts	OTS	Procured	M6×110mm, stainless steel
Clamps	Custom	3D-printed	PP
Grabber subsystem			
Tentacles	Custom	3D-printed	TPU
Grabber housing	Custom	3D-printed	PP
Servos	OTS	Procured	goBILDA 2000 Series Torque Servo
Strings	OTS	Procured, cut to length	UHMWPE
Bolts	OTS	Procured	M4X10mm, stainless steel
Shaft	Mod. OTS	Rod stock, cut to length	Aluminium
Sheet	Mod. OTS	Sheet stock, cut, bent and drilled to shape	Aluminium
Tube	Mod. OTS	Tube stock, cut to length	CFRP
Hardware			
UWB transceiver	OTS	Procured	Decawave DW-1000
IMU	OTS	Procured	MPU-9250
mmWave sensors	OTS	Procured	SEN0395
On-board comp.	OTS	Procured	Raspberry pi 5 4gb
Batteries	OTS	Procured	Lithium iron phosphate ANR26650m1B
Wiring	OTS	Procured	-
ERS			
Plates	Mod. OTS / Custom	Sheet stock cut to size / load-bearing plates CNC-machined	Aluminium
Leg housings	Custom		
Rods	OTS	Procured	Aluminium
Wheels	OTS	Procured	Aluminium
Wiring	OTS	Procured	-
UWB anchors			
Anchor housing	Custom		
UWB transceiver	OTS	Procured	Decawave DW-1000
Wiring	OTS	Procured	-
Central computer			
Computer	OTS	Procured	PCSpecialist 4U rack workstation

Continued on next page

Table 17.1 – continued from previous page

Part	Category	Method / Process	Material/Model
Wiring	OTS	Procured	-

As the table shows, most parts are off-the-shelf with a small set of modified off-the-shelf components and a limited number of custom fabricated parts. This bias towards standard parts directly supports the cost, lead time and maintainability targets (RQ-BDT, RQ-MFC-01, RQ-MNT) because off-the-shelf hardware keeps unit costs down, lead times short and spares readily available. For custom works, it is bounded to simple 3D-printing or CNC machining leaving the propellers as the only part that requires significant manufacturing development due to their unusual shape. For procurement it must be ensured that all off-the-shelf parts remain within the four-week lead-time limit of RQ-MFC-01. The manufactured and procured parts feed into the four assembly blocks of Figure 17.1, discussed in the following section.

17.2. Assembly

Assembly is organised into the four parallel streams shown in Figure 17.1: the drone, the ERS, the UWB anchors and the central computer, which are each built and checked independently before coming together at warehouse integration.

Drone Assembly

The drone is assembled per subsystem starting from propulsion. The propellers are fit on to the motors which are clamped on the tube arms. Dampener rings are installed on the tubes. The grabber is assembled by fitting the tentacles, strings and servos in the housing, then attaching it to the arm rod. The main body is built by stacking the CFRP plates with the bolts and installing the legs. The sensors, on-board computer and power hardware are installed and wired according to Figure 15.5. Finally the arms are attached and the motors are wired in to complete the assembly. Throughout, the design is modular and fastener-based, allowing a single technician to assemble the drone with simple hand tools (RQ-MFC-02). This also allows for disassembly and maintenance (RQ-DCM-01). The use of similar interchangeable plates and tubes simplifies assembly and spare handling.

Warehouse Systems Assembly

The ERS is assembled by integrating the battery-swap mechanism and charging electronics into its frame made from aluminium rods and plates. The units are then stacked vertically to minimize footprint. The UWB beacons are assembled by installing the transceiver into the housing while wiring everything. The central computer requires minimal assembly as it is procured as a fully configured workspace unit. These assemblies flow directly into the integration part in the next section.

17.3. Integration

The warehouse integration step consists of installing the ERS Assembly, UWB beacons and Central Computer in the warehouse. The ERS will be positioned at the corner of the warehouse and wired to the power source. After establishing a UWB constellation for the warehouse, the beacons will be installed in their respective locations, either on the walls, floor, racks or ceiling and will be wired to a power source and into the warehouse communication network to connect to the central computer for data and timing synchronisation.

18

Sustainable Development

In this chapter, the sustainability of the DRIVE-FC project is analysed and evaluated. Section 18.1 presents a background on the sustainable development goals, which form the guidelines for DRIVE-FC's sustainable development strategies moving forward. Section 18.2 covers the life cycle analysis performed on the project and goes over some initial conclusions. Finally, Section 18.3 utilises the United Nations (UN)'s sustainable development goals and results from the life cycle analysis to form a three part sustainable development strategy that has been used throughout the design of DRIVE-FC and will continue to be used moving forward.

18.1. Sustainability Overview

As the design team, the client and the market as a whole is committed to a sustainable future, it is important that it is considered in all aspects of the project. The UN Brundtland Commission: "meeting the needs of the present without compromising the ability of future generations to meet their own needs." [40]. This means that outside of the common and narrowly defined environmental sustainability, social and economic sustainability are also considered. Three especially relevant sustainable development goals have been identified for this project, these are: "Decent Work and Economic Growth", "Industry, Innovation and Infrastructure" and "Responsible Consumption and Production". The associated icons are shown in Figure 18.1.

(a) SDG 8 **(b)** SDG 9 **(c)** SDG 12

Figure 18.1: Relevant Sustainable Development Goals [50]

18.2. Life Cycle Analysis

To properly understand and evaluate the environmental impact of DRIVE-FC operations, a life cycle analysis is performed. This analysis looks at the Carbondioxide (CO_2) emissions of all phases of the project, from manufacturing to decommission. Data is gathered from databases containing CO_2 emissions per kg for all raw materials used in productions and estimating the kWh the drone system will use over the course of the 10 year project lifetime, and converting this into kg CO_2 emissions using the electricity conversion factor for electricity usage in the Netherlands. Since the project is in a very preliminary phase and not much is known about the expected long term status of DRIVE-FC, it is estimated, based on similar drone swarm operations, that 83% of battery production CO_2 emissions is emitted when decomissioning the drones. For the drone materials and

ERS materials, a rough estimate of 70% and 40% usage is taken. Furthermore, the extra materials that are necessary as a result of maintenance, repair and overhaul are already taken into account in this estimation. The result of the life cycle analysis is presented in a sankey diagram shown in Figure 18.2 to illustrate where the CO_2 originate from throughout the project lifetime.

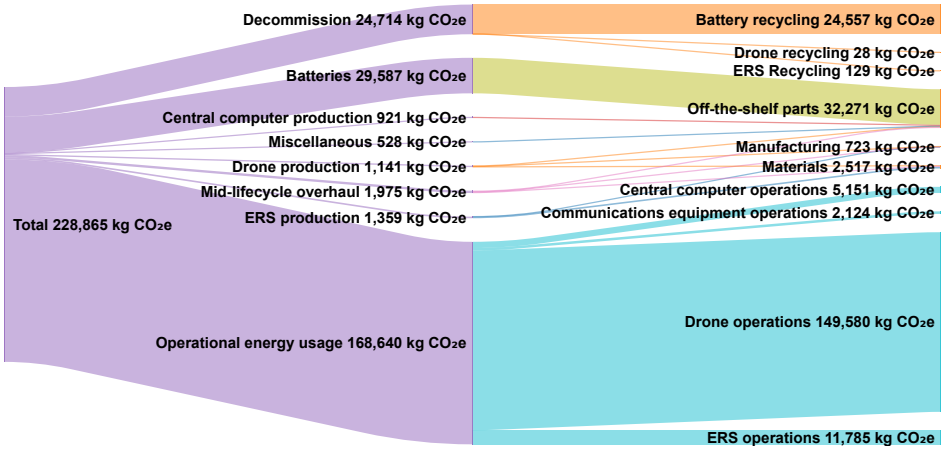


Figure 18.2: Summary of life cycle analysis results [28]

From the estimated CO_2 calculations, it is clear that the environmental impact of the system is quite significant. To provide further context, the CO_2 emissions for the human equivalent system is calculated and these numbers are compared in Table 18.1.

Table 18.1: System comparison CO_2 emissions

	kg CO_2e	**Difference to base**
Human picking system (base)	201117	-
DRIVE-FC	228160	+27043

Table 18.1 shows that, in its current design, DRIVE-FC is less sustainable than when using an equivalent human picking system and therefore will need further iterations in the future to improve its sustainability. The bulk of DRIVE-FC's carbon footprint derives from operational energy usage, and most of what remains to battery manufacturing and decomission. Reducing drone energy consumption should be the main target in improving the design between the prototyping and commissioning stages. This will not only reduce the drone energy usage, which alone represents around 65% of total equivalent emissions, but will also decrease the number of replacement batteries needed. This is significant since battery replacements represent an additional 24% of the total equivalent emissions.

18.3. Sustainable Development Strategy

In accordance with identified relevant sustainable development goals presented in Section 18.1 and the result from the life cycle analysis shown in Section 18.2, a three part strategy has been developed, focused on environmental, social and economic sustainability.

18.3.1. Environmental Sustainability

A three pronged approach is taken to develop an environmentally sustainable product, these are: minimizing the environmental impact during manufacturing, minimizing the environmental impact during operation, and finally, minimizing environmental impact during decommissioning, satisfying the twelfth sustainable development goal (Figure 18.1c). To ensure that the environmental impact is minimized each possible design will be compared on their sustainability to decide what the final design will be. Furthermore, aspects of environmental sustainability are considered in the design of the physical components of the system. To address sustainability, at least 70% of the mass of the system consists of recycled/reusable materials. The system will also be fully electrically powered, allowing for sustainable energy sources to be used. The life cycle analysis has highlighted that the operational usage is the largest contributor to CO_2 emissions, and currently likely the reason that it is unsustainable compared to human based picking solutions. So future design steps must focus on reducing drone energy usage, for example by further optimising flight time, or better coordination

with peak hours in the warehouse. This could potentially decrease the number of batteries needed. In the long term, the possibility to develop rechargeable batteries that fit within budget requirements can also be investigated.

Outside of the product itself the development process will also be conducted sustainably. This includes choosing to minimize the environmental impact during simulations, for example, avoiding high performance computing when possible, and using power hungry manufacturing machines effectively. Local material sources will also be considered first when identifying suppliers to minimise any transport costs and emissions. Furthermore limiting printing and choosing for a digital first approach to minimize paper usage.

18.3.2. Social Sustainability

There will be careful attention to the social impacts of the system looking at all of the stakeholders, even if the stakeholders have little influence on the project outcomes. The proposed system increases the automation in fulfilment centres. This raises some social issues such as worker displacement, especially as the proposed system will most negatively impact lower skilled and lower educated workers. A plan will be made how to minimize this impact which will help satisfy the eighth sustainable development goal (Figure 18.1b). A common example of how this could be minimized is through retraining displaced workers to maintain the system. Furthermore, the system will be designed with the wellbeing of the workers in mind. The warehouse and drone routing design ensures that human-drone interactions are kept to a minimum and as such that there are no potential injury causing collisions. In addition, a very strict noise limit has been set, to minimise disruption for workers.

18.3.3. Economic Sustainability

To ensure a long and prosperous use of the DRIVE-FC system it is key to ensure that the system is economically sustainable. This includes the ability of the system to generate a consistent profit for the client while building resilient industry thus satisfying the eighth and ninth sustainable development goal (Figure 18.1a, Figure 18.1b). For this reason it is essential that the system is reliable and that clients do not get overrun with maintenance or reparation costs. During the design phase this has been taken into account and will continue to be taken into account in the next steps of the project.

19

RAMS Analysis

The Reliability, Availability, Maintainability and Safety (RAMS) analysis consists of analysing the operational life of the system. If any of these categories perform poorly, it indicates the system is not ready for clients to use. In this section, each of these are carefully and critically considered and evaluated. The definition of each category used is based on the Airbus definition of RAMS [3]:

- **Reliability** — The system's ability to perform its intended function without failure. For analysis, the probability of failure is mainly considered.
- **Availability** — The system's ability to be operational and accessible when needed. For analysis both the uptime and downtime are considered, trying to maximise the former and minimise the latter.
- **Maintainability** — The system's ease of maintenance or repairs when necessary. For analysis the accessibility of components, modularity and ease of troubleshooting are considered.
- **Safety** — The system's ability to prevent hazards, accidents, and risks during the entire lifecycle. For analysis it consists of identifying possible hazards and how well the system does at reducing the risks.

19.1. Reliability

Reliability was integrated into the design process from the beginning to guarantee a high-quality system for the client. This focus is directed by requirement **RQ-RLB** (see Table 4.1). Additionally, a technical risk analysis was performed to identify and mitigate operational risks to levels As Low as Reasonably Practicable (ALARP).

The primary threats to system reliability stem from either individual drone component failures or dependencies on supporting infrastructure, such as the ERS, communication beacons, or the central management computer. To mitigate these risks, structural and functional redundancies have been systematically incorporated:

- Propulsion Redundancy: A hexacopter configuration was selected over a traditional quad-copter to avoid full drone failure in a motor or propeller failure.
- Sensor Redundancy: Overlapping mmWave sensors ensure the drone maintains dynamic obstacle avoidance capabilities even if a single sensor fails.
- Infrastructure Redundancy: The deployment of 132 UWB beacons ensures robust localization coverage, maintaining normal operations even during localized beacon dropouts.
- Operational Redundancy: The ERS features multiple docking stations, ensuring continuous battery-swapping capabilities if a single dock malfunctions.

Despite these redundancies, verification of the physical system reliability requires prototype manufacturing and long-term testing. Consequently, while the architecture is designed for high reliability, physical validation remains open at this stage of development.

The primary bottleneck in overall system reliability is the multi-agent routing algorithm. In its current stage of development, the framework exhibits an average collision rate of one conflict every 2.4 minutes when 15

drones are simultaneously active as explained in Section 13.4.3. Within the simulation environment, a collision is defined as a spatial breach where two or more UAVs violate each other's designated clearance radii.

19.2. Availability

The availability of the system is defined by how much uptime the drone has compared to its downtime. Two availability states are considered: **1)** a partial system availability where the system works in a degraded state, and **2)** a situation where the system is completely unavailable. The former may occur when a drone drops an item that needs to be cleaned up, while the latter occurs when communication between the drones and the central computer completely fails.

To map out how different events can cause (partial) system shutdowns, a simplified fault tree analysis is performed for both the partial and full system unavailability situations. These are shown in Figure 19.1 and Figure 19.2.

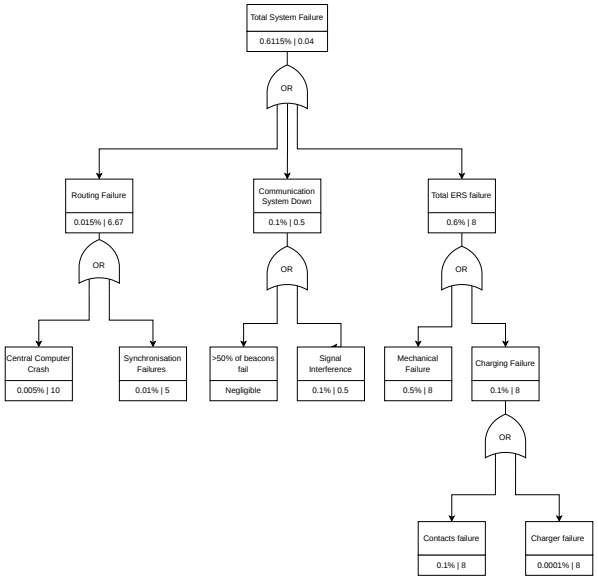


Figure 19.1: Fault tree analysis for a complete system failure

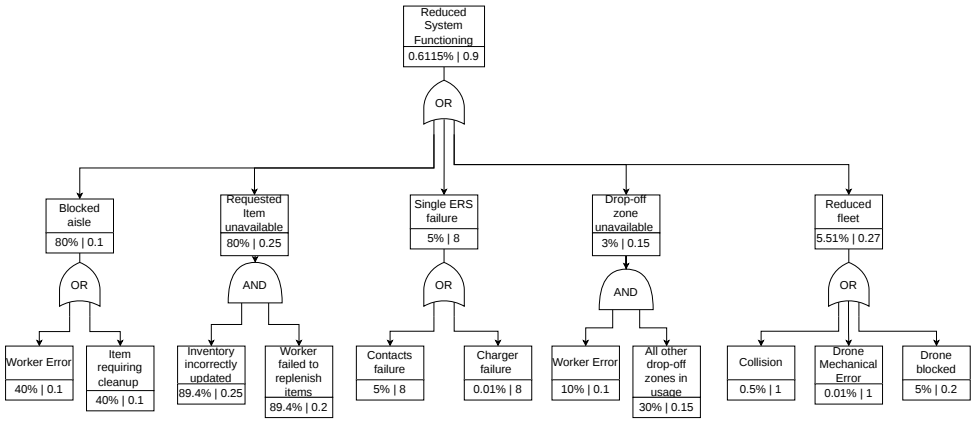


Figure 19.2: Fault tree analysis for a partial system failure

Below each event in the figure that may cause a shutdown of the system, the probability of it occurring in a day is given, along with the mean time to resolve the issue in hours, separated by a pipe. The probabilities and the weighted mean time are calculated to find the total downtime for both the partial and complete shutdown scenarios. For the 'AND' gates, it is assumed that the resolution time is equal to the longest time, meaning that the resolution of the issues occur simultaneously.

From the fault tree analysis, it is found that there is an expected daily downtime of 0.04 hours for total system failures, as shown in Figure 19.1. Based on the warehouse being operational for 20 hours a day, this results in

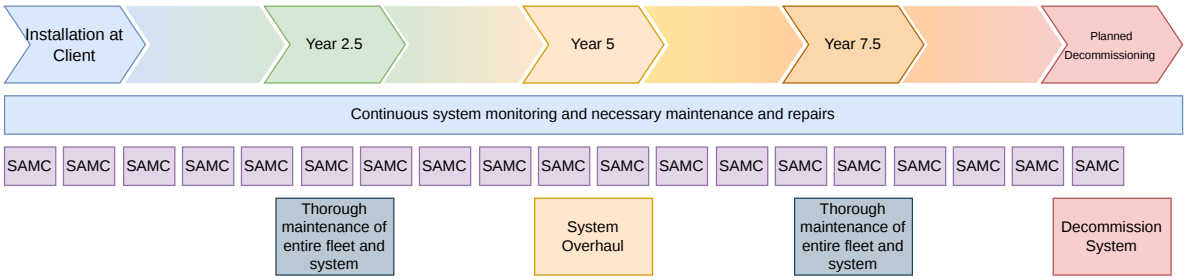


Figure 19.3: Maintenance schedule for project duration

an uptime of **99.8%** in some state. However, although the system is operational, it may be operating in a degraded state. As shown, the expected daily downtime due to a partial system failure is 0.9 hours. This translates to the system operating in its optimal state **95.5%** of the time.

19.3. Maintainability

Maintainability is closely linked to the availability, as maintenance and repair must be simple and fast enough for the drones to quickly be operational after any issues that could occur. This also closely fits in with the system requirement of **RQ-MNT** (system shall be maintainable).

To improve maintainability the design has opted for off-the-shelf components whenever possible, and when custom components are necessary these are reusable. For example, the structural plates in the body are all the same, allowing for the plates to be interchangeable. Although this does come with worse weight and structural optimization, this will reduce the amount of custom made components that are needed to be stored for maintenance and most components being very well accessible.

The drone is also fully modular allowing it to be quickly taken apart and put back together. As of the current design, no parts are bonded together, allowing for quick changing of components with simple hand tools.

The ease of troubleshooting was not necessarily considered during the design of the drone. However, most components can be visually inspected for any damage. For electrical issues, it is possible to find problems with the battery health using health modelling, and motors can be visually inspected. Finally, sensors, although not directly diagnosable, will cause the drone to act unpredictably, so any issues will quickly become evident.

To ensure maintenance is performed sufficiently regularly, a maintenance schedule has been set up. The maintenance schedule is shown in Figure 19.3. As shown, through the entire planned project duration, the system health is monitored and repair and maintenance is performed when necessary. An example would be that if data shows that a battery is performing worse than expected, it is removed from the fleet and examined and repaired or replaced if necessary. A Semi-annual Maintenance Check (SAMC) is performed, as expected, twice a year to ensure no glaring issues are found which could pose a risk to safety. This is a shallow check, mainly consisting of inspection of the drones and supporting systems. After 2.5 years and 7.5 years of operation, in-depth maintenance is performed on the entire system, including measuring motor performance, component cleaning, and testing electrical connections. At year five, a full overhaul of the system is performed in which many parts of the drones are replaced, as many have reached their life lifespan. Finally, after ten years, the system is planned to be decommissioned and removed from the warehouse.

19.4. Safety

Since the requirement **RQ-WOS** (system shall ensure worker well-being) is a system requirement, safety of the system has been carefully considered throughout the entire design process. Four separate redundant systems are used to ensure that accidents are avoided. To ensure no risky situations occur in the first place, the warehouse is split into two zones: worker zones and drone zones. In case a worker does have to go into the drone zone, the drones can be set to completely avoid that area, once again physically separating humans from the drones. In case a human enters the drone zone and the safety zone around the human does not get activated the drones are equipped with sensors which detect unexpected obstacles, and can manoeuvre around them without any human inputs whatsoever. Finally, in case a drone is dangerously out of control a ”drone out of control plan” is used, as described in the contingency of technical risk **TR-OPS-08**. Through the redundancy in the human avoidance system and the final plan to deal with an unsafe environment, many things have to go wrong before having any issues with safety.

The safety outside of normal operations is also analysed. The materials are proven and can be assumed to be safe. The main consideration is the battery, for which handling and disposal needs to be done correctly. As the batteries are used intensively they are the most likely component to malfunction. For this reason care must be taken when using the batteries, and a designated battery disposal box needs to be present between decommissioning the batteries and disposing of them. Finally carbon fibre may be sharp if it breaks, and can cause very fine dust, in case of a collision broken carbon fibre needs to be carefully disposed of.

20

Project Outlook

This chapter presents the future outlook of the DRIVE-FC by defining all the steps to be taken post-DSE. This includes all the steps until the start of operations and the actions to be taken during operations, until decommissioning.

20.1. Project Development Logic

The main concept of the drone is complete, however, more detailed design is required to produce a complete product. In each of the subsystems further design is necessary, as shown in Figure 20.1. Outside of designing, further verification and validation needs to be performed to ensure compliance with the project requirements, using simulations, prototyping and testing the drone, as described in Section 4.1. Any discrepancies found during the verification and validation process both in the simulations and prototype phases will be iteratively corrected in the finalising design phase. Once the prototype is fully validated, the Manufacturing, Assembly and Integration (MAI) process starts, to produce the system at full scale, ready for the client. Then the system starts to operate, and then finally, when necessary, the system is decommissioned in a safe and sustainable way. During the finalisation of the project, after the DSE, close contact will be kept with the client for milestone reviews.

20.2. Project Development Schedule

Figure 20.1 presents the list of tasks that form the next steps of the DRIVE-FC project. The next page presents these tasks in a timeline for the coming year. The Gantt chart is structured on a week by week basis, and expected duration of tasks is rounded off to the nearest working week (five days) as a first estimate. A key assumption made in the estimation of this timeline is that the effort and manpower remains the same as it has been in the past ten weeks. Hence a similar level of progress is expected over the coming year. Below is a list of key deadlines for the project development.

- **September 2026**: Design finalised. Start of detailed simulation phase
- **October 2026**: Start of (prototype) manufacturing phase.
- **Decemeber 2026**: Start of prototype testing phase
- **February 2027**: Testing completed. If acceptance test is met, start of product manufacturing phase. If the acceptance test is not met, return to appropriate phase, depending on the cause of the failure to pass the test.
- **1st week of March 2027**: Perform final integration test between drones and warehouse.
- **March 2027**: Start DRIVE-FC operations.

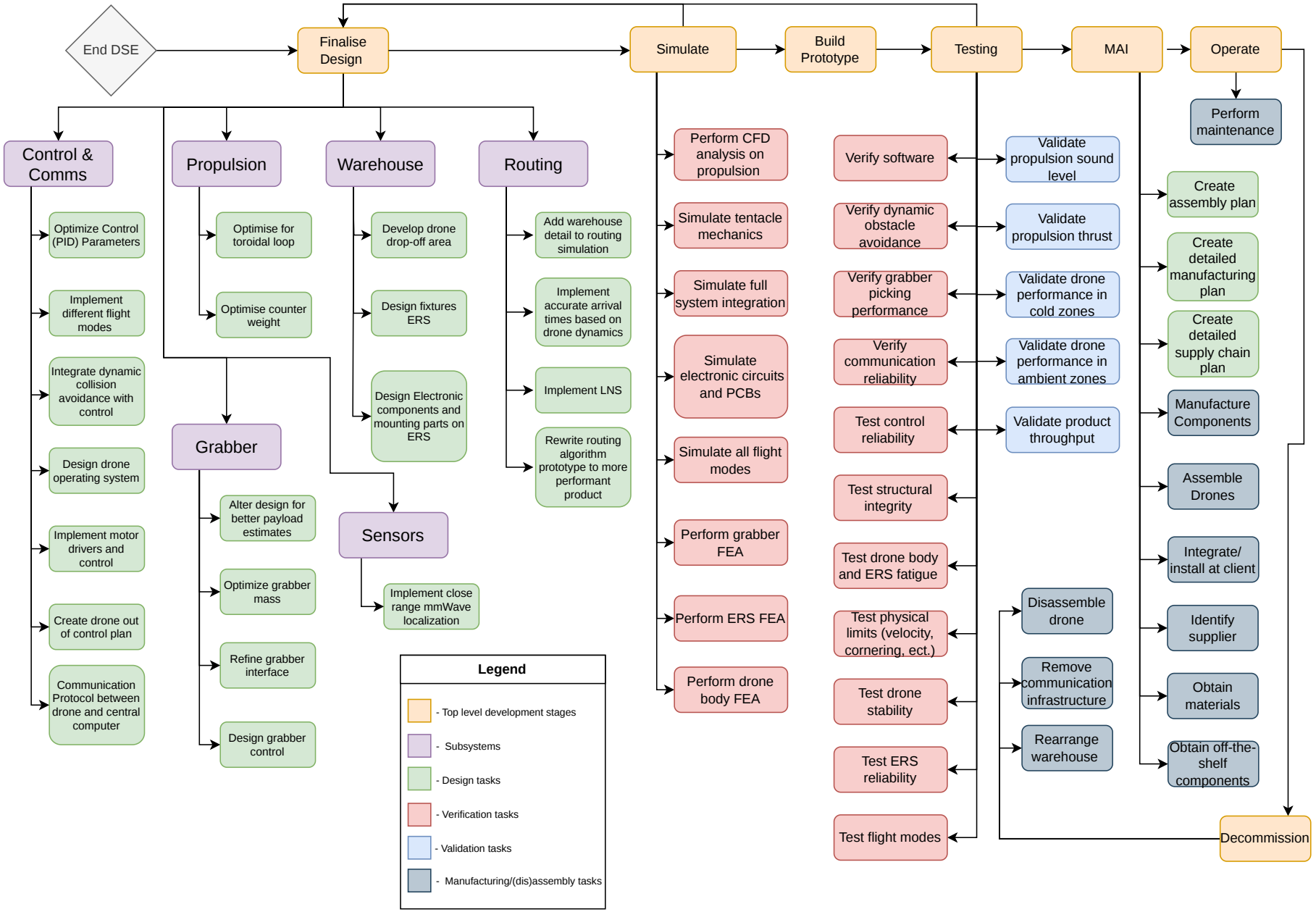
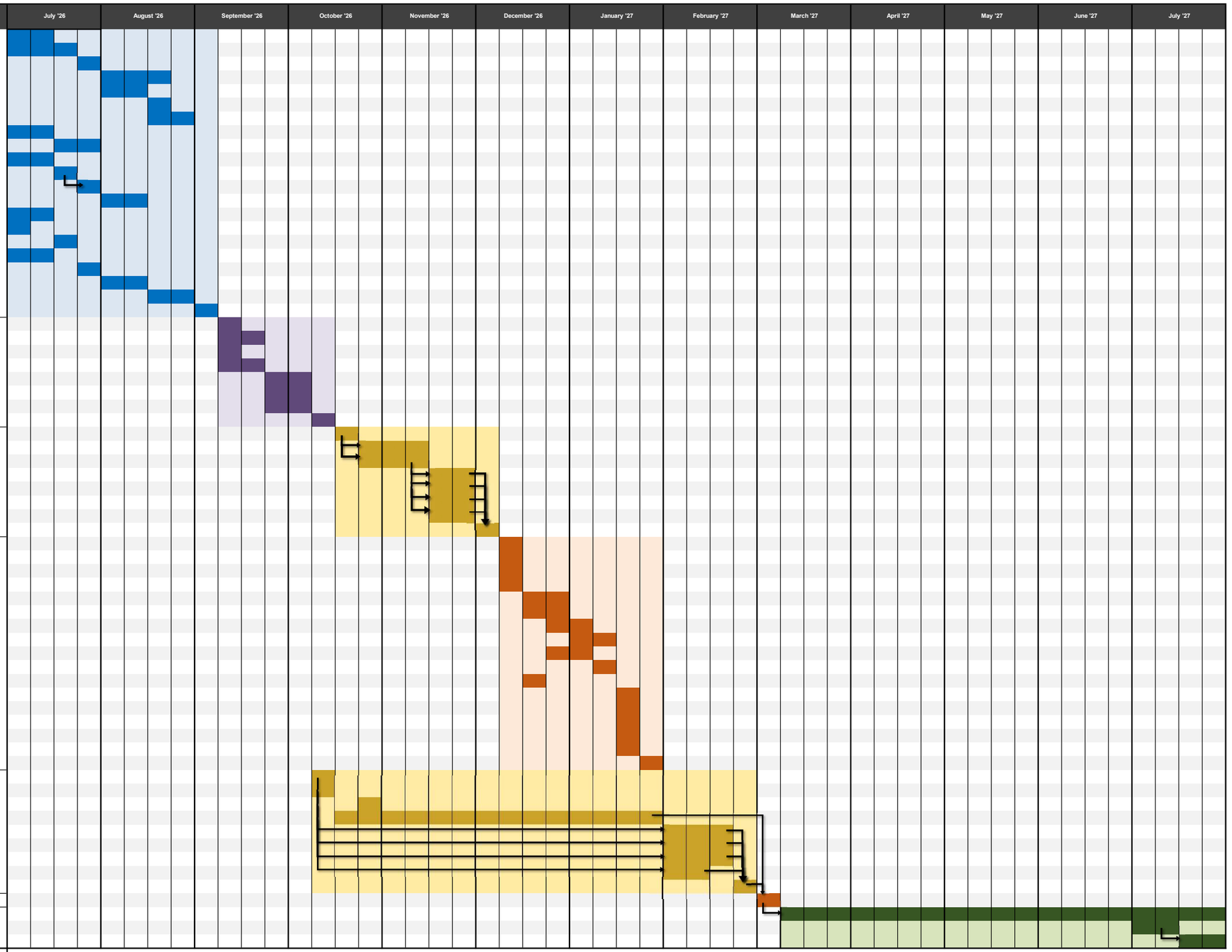


Figure 20.1: Future Project Development Logic

Project Phase	Task ID	Task	Expected duration (working days)
DES: Design	2	Optimize Control Parameters	10
	3	Implement different flight modes	15
	4	Integrate dynamic collison avoidances with control	5
	5	Design drone operating system	15
	6	Implement motor drivers and control	10
	7	Create drone out of control plan	5
	8	Communication protocal between drone and central computer	10
	9	Optimise toroidal loop	10
	10	Optimise counter weight	10
	11	Alter grabber design for better payload estimates	10
	12	Optimize grabber mass	5
	13	Refine grabber mass	5
	14	Design grabber control	10
	15	Develop drone drop-off area	10
	16	Design fixtures ERS	5
	17	Design elecctronic components and mounting parts on ERS	5
	18	Implement close range mmWave localization	10
	19	Add warehouse detail to routing simulation	5
	20	Implement accurate arrival tames based on drone dynamics	10
	21	Implement LNS	10
	22	Rewrite routing algorithm prototype to more perfromant product	5
SIM: Simulation	23	Perform CFD analysis on propulsion	5
	24	Simulate tentacle mechanics	10
	26	Simulate electronic circuits and PCB's	5
	27	Simulate all flight modes	10
	28	Perform grabber FEA	10
	29	Perform ERS FEA	10
	30	Perform drone body FEA	10
	31	Perform full system integration simulation acceptance test	2
MIA: Manufacturing, Integration & Assembly	32	Identify supplier	5
	33	Obtain materials	15
	34	Obtain off-the-shelf components	15
	35	Manufacture prototype airframe	15
	36	Manufacture prototype grabbing tool	10
	37	Manufacture prototype propulsion system	10
	38	Manufacture prototype electronics/hardware	10
	39	Integrate all manufactured prototype subsystem	5
TES: Testing	40	Verify software	5
	41	Verify dynamic obstacle avoidance	5
	42	Verify grabber picking performance	5
	43	Verify communication reliability	5
	44	Test control reliability	10
	45	Test structural integrity	10
	46	Test drone body and ERS fatigue	10
	47	Test physical limits	10
	48	Test drone stability	10
	49	Test ERS reliability	5
	50	Test flight modes	5
	51	Validate propulsion sound level	5
	52	Validate propulsion thrust	5
	53	Validate drone performace in cold zones	5
	54	Validate drone performance in ambient zones	5
	55	Validate drone throughput	5
	56	Perform final system acceptance tests	5
MIA: Manufacturing, Integration & Assembly	57	Create assembly plan	5
	58	Ceate detailed manufacturing plan	5
	59	Create detailed supply chain plan	5
	60	Make adjustments to warehouse	70
	61	Manufacture final airframe	15
	62	Manufacture final grabber	15
	63	Manufacture final propulsion system	15
	64	Manufacture final electronics/hardware	10
	65	Integrate all manufactured subsystems	5
TES: Testing	66	Test integration between drone system and warehouse	5
OPS: Operations	68	Start drone operations	2600
	69	Perform scheduled maintenance	10
	70	Repair broken/malfunctioning parts	10



Legend

Design phase	Task active
Simulation phase	Task active
Testing phase	Task active
Manufacturing phase	Task active
Operations phase	Task active

Once the drone starts operating, RQ-RLB-01 states that the system will run continuously for ten years. To maintain this operation, it is estimated that approximately once every half a year, a general maintenance is scheduled to inspect the drones. This will be done in a staggered format to ensure the system can keep operating while this occurs. In general maintenance is expected to be not very invasive, only being a system level inspection to see if there are no areas for concern. At three points will the maintenance be more extensive than normal.

- **September 2029**: This is the 2.5 year mark for the operation so the maintenance inspection will go over all individual subsystems extensively.
- **March 2032**: At the halfway point of the operation, several drone parts will be overhauled as a part of preventive maintenance.
- **September 2034**: Same inspection as at the 2.5 year mark.

After ten years of operation, which according to the current planning will be March 2037, the operation will be shut down. Figure 20.2 shows a timeline of how the decommission phase is expected to proceed. A similar time scale to the project development Gantt chart is used.

Project Phase	Task ID	Task	Expected duration (working days)
OPS: Operations	66	Start drone operations	2600
DEC: Decommission	71	Deactivate drones	5
	72	Remove sensitive data	15
	73	Recycle recyclable components	10
	74	Dispose hazardous materials safely	10
	75	Archive documentation	15

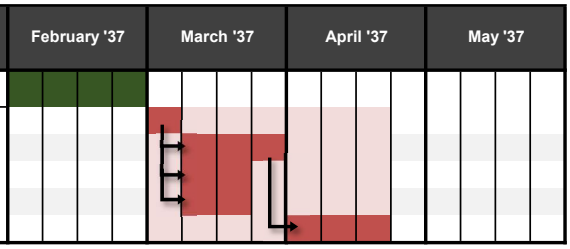


Figure 20.2: End of life Gantt chart

21

Conclusion

At the start of the project, the design team was tasked with designing an autonomous integrated drone-based intra-warehouse transport system for an e-grocery fulfilment centre. The projects goal was to have a system capable of reaching a daily throughput of 10,000 items that weigh less than 1 kg per day while also be safe to operate in a warehouse by meeting noise constraints and having collision avoidance protocols. Over the course of ten weeks, DRIVE-FC was developed in four project phases: initial project planning, followed by conceptual, preliminary and finally detailed design phases.

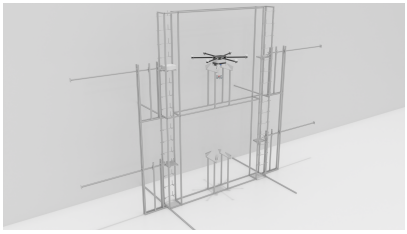
In the planning phase of the project, the first steps of the project were worked out fully and 11 subsystems were identified: Propulsion, Control, Localization, Collision Avoidance, Grabber, Communication & Data Handling, Drone Body, Power, Routing and Warehouse. Following this, a market analysis was performed to identify the key stakeholder requirements for the project and each subsystem researched all possible solutions to meet these requirements during the conceptual phase. Each subsystem then performed trade-offs based on criteria stemming from the subsystem requirements to converge to a final chosen concept per subsystem.

In the preliminary design phase, key drone parameters were sized. The number of grabber arms was weighed up against the grabber system weight and the design working space was determined. The battery was sized to meet the power requirements following from propulsion system requirements and the propellers themselves were sized to find the optimal point between drone mass and noise level. From this an initial weight for the drone body could also be estimated and the minimal flight time was also established by minimising production cost. The final preliminary design yielded an operational envelope of feasible drone designs.

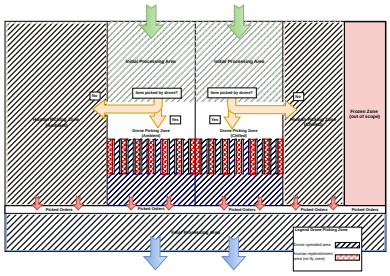
Combining the requirements and the results from the operational envelope, the final design was established after further analysis performed in the detailed design phase. The result is DRIVE-FC, a system of 15 drones operating in its own distinctive area within the warehouse. Drone and human picking areas are separated using physical barriers while stock replenishment in the drone picking area is kept segregated from humans by implementing no fly zones. The drones find the optimal paths using an S2M2 routing algorithm with a large neighbourhood search structure.

The drones themselves are configured in a hexacopter configuration using asymmetric looped propellers. The drone grabber is a tentacle-based system with six tentacles. The total drone diameter is 1.2 meters and the grabber can be extended to 29 centimetres in length. The total weight of the drone is 5.41 kg and the drone is able to carry a payload weight of up to 1 kg. The drones are fitted with nine mmWave sensors (one on each of the drone arms and three on top of the drone body) for collision avoidance and contain a fused IMU-UWB-Vertical time of flight sensor to localize the drone's position in the warehouse, which contains multiple UWB beacons mounted on the ceiling and shelves. The drone is controlled using a cascaded PID controller positioned in the top of the body. In between the flight controller and the grabber system, the drone is powered by swappable batteries that are replaceable using a custom designed ERS. The warehouse has been sized to fit the determined number of items to be carried by drones while minimizing operating area. In this area, two ERS's, each containing 30 charging batteries, are used to keep the 15 drone system operational. In addition to the drone picking area, a concept has been developed for determining which items get picked by a drone or a human. The final concept for the drone, ERS and warehouse are shown below in Figure 21.1

(a) Final Drone design



(b) Final ERS design



(c) Final warehouse design

Figure 21.1: Final designs

The chosen final system has been analysed to ensure it complies with the requirements. By plotting accuracy of tracking paths set by routing, the PID controllers is able to follow certain paths within the required 15 centimetre margin. However, some paths still leave some accuracy to be desired. The sensor's positional accuracy has been verified using a real time data set and gives a 4 centimetre accuracy, which is better than the predetermined requirement. The grabber system has been analysed based on its geometry and loading to ensure it meets the 1 kg pickup requirement. Finally, the high-level logistical simulation ran the chosen configuration and resulted in a final throughput of 24,003 items picked a day.

This result provides a robust foundation for attaining the project objective. Nevertheless, several aspects remain to be addressed. The biggest strength of the current design is its throughput. The target throughput at the start of the project was 10,000 items a day, and the simulation has estimated that DRIVE-FC's throughput is 24,003 items a day, a 240% overachievement. This number can be put into perspective as the simulation does not consider the pre- and post-processing stages and also assumes a very high reliability, both during battery swaps, but also drone reliability in general. Both of these aspects are areas that can be further developed in future iterations of this project. Nonetheless, taking these limitations into account, it can be expected that DRIVE-FC will meet the daily throughput requirement. Another strength of this project is its financial feasibility. The project meets the €1,200,000 budget requirement based upon the initial cost breakdown and also estimates that it will run a profit after just two years. DRIVE-FC is expected to run for ten years; that results in eight years of profit. Once more, these are initial estimates and subject to change once details of the design is more developed. Lastly, the current concept warehouse configuration is designed such that the remainder of the warehouse operations is undisturbed and that the safety (both noise and risk of collisions) of the workers is kept in mind.

The conceptuality of the design is one of the projects biggest drawbacks. Most subsystems have only been sized and verified on a theoretical level. Several verification methods require physical testing. For example, friction-testing the grabber system and drone subsystems, like the propeller and body, require further simulation methods such as a CFD or FEM analysis, for which there was neither enough time nor resources, during the ten week duration of the DRIVE-FC project. In addition, some subsystems still have to be tested when integrated with each other. The most important subsystem for integration testing is collision avoidance. Currently, collision avoidance and control have been yet to be tested when integrated, and the routing algorithm currently also does not communicate route feasibility to the scheduler. It is recommended that these are the first steps that should be made if the project continues. Another key drawback of the system is its sustainability footprint. Based upon an initial life cycle analysis, it can be concluded that DRIVE-FC is not environmentally sustainable when compared to an equivalent human based picking system. The analysis also illustrates that the operational energy usage and battery manufacturing are the largest contributors to the CO_2 footprint. As such, steps must be taken to ensure this usage is kept to a minimum, either by optimising the routes flown by the drone to minimise battery usage or looking into more long-term, energy efficient charging solutions. To summarise these recommendations, the key next steps have been listed below.

- **Integrate collision avoidance with pathfinding, communication and control subsystems** and perform integration tests to assess its performance.
- **Perform CFD and FEM analysis** on the drone's propellers and body to further quantify its performance and optimise parameters where necessary.
- **Develop the warehouse system integration concepts** to strengthen the business case and provide the throughput simulation with more accurate estimates

- **Perform environmental testing** in the warehouse to evaluate the drone's performance in real world conditions.
- **Perform a proper drone reliability analysis** to more accurately estimate the operational throughput
- **Investigate more energy efficient drone charging methods** to ensure the project becomes more sustainable than a human equivalent operation.

Although it remains to be seen if DRIVE-FC, in its current state, functions in real world testing. The project demonstrates that designing an autonomous intra-warehouse drone-based picking system is possible from a technical, operational, safety and financial point of view. It provides a base for solutions alternative to conveyor belts and AGV's in the continuously growing automated warehouse operation market. DRIVE-FC has provided the design team with an extensive learning experience where creativity, imagination and a systems engineering project approach have been central all throughout its 10 week duration.

Throughout the project, various AI tools such as, Grok, Claude, Gemini & ChatGPT have been used to aid in source finding, formatting, code support and image generation. This statement serves as an acknowledgement of this use.

References

[1] *(1) (PDF) Automatic PID tuning, a heuristic optimization approach*. URL: https://www.researchgate.net/publication/268268955_Automatic_PID_tuning_a_heuristic_optimization_approach.

[2] Inc. ACP Composites. *Mechanical Properties of Carbon Fiber Composite Materials, Fiber / Epoxy resin (120°C Cure)*. URL: https://acpcomposites.com/wp-content/uploads/2023/12/Mechanical-Properties-of-Carbon-Fiber-Composite-Materials.pdf (visited on 06/17/2026).

[3] Airbus Protect. *Safety 101: Understanding operational reliability terminology*. July 2023. URL: https://www.protect.airbus.com/blog/safety-101-understanding-operational-reliability-terminology/.

[4] Nusin Akram et al. "Energy Consumption Modeling and Flight Time Analysis of Micro Drones". In: *IEEE Access* 13 (2025), pp. 109854–109866. ISSN: 21693536. DOI: 10.1109/ACCESS.2025.3581944.

[5] *Amazon's tiny robot drives do the heavy lifting*.

[6] V. Artale, C.L.R. Milazzo, and A. Ricciardello. "Mathematical modeling of hexacopter". In: *Applied Mathematical Sciences* 7 (July 2013), pp. 4805–4811. ISSN: 13147552. DOI: 10.12988/ams.2013.37385. URL: http://www.m-hikari.com/ams/ams-2013/ams-97-100-2013/37385.html.

[7] Chingiz Arystanbekov, Vladimir Golubev, and Basman Elhadidi. "Aeroacoustic and Aerodynamic Assessment of Propellers with Uneven Blade Spacing". In: *30th AIAA/CEAS Aeroacoustics Conference, 2024*. American Institute of Aeronautics and Astronautics Inc, AIAA, 2024. ISBN: 9781624107207. DOI: 10.2514/6.2024-3221.

[8] Francesco Bandinelli et al. "Recycling a carbon fiber-reinforced polyamide through 3D printing: A mechanical and physicochemical analysis". In: *Composites Part B: Engineering* 294 (Apr. 2025). ISSN: 13598368. DOI: 10.1016/j.compositesb.2025.112147.

[9] Barnes W. McCormick. "AerodynamicAeronauticsFlightMechanics". In: (1979).

[10] Hamid Bentarzi et al. "Backstepping Control of Drone". In: *Engineering Proceedings 2022, Vol. 14, Page 4* 14.1 (Jan. 2022), p. 4. ISSN: 2673-4591. DOI: 10.3390/ENGPROC2022014004. URL: https://www.mdpi.com/2673-4591/14/1/4/htm%20https://www.mdpi.com/2673-4591/14/1/4.

[11] Martin Braquet and Efstathios Bakolas. "Vector Field-based Collision Avoidance for Moving Obstacles with Time-Varying Elliptical Shape". In: *IFAC-PapersOnLine* 55.37 (2022), pp. 587–592. ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2022.11.246. URL: https://www.sciencedirect.com/science/article/pii/S2405896322028890.

[12] Jingkai Chen et al. "Scalable and Safe Multi-Agent Motion Planning with Nonlinear Dynamics and Bounded Disturbances". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 35.13 (May 2021), pp. 11237–11245. DOI: 10.1609/aaai.v35i13.17340. URL: https://ojs.aaai.org/index.php/AAAI/article/view/17340.

[13] Mingtai Chen, Tianming Liu, and Tiegang Fang. "Experimental and Numerical Analysis of a Toroidal Propeller". In: *Journal of Aerospace Engineering* 38.6 (Nov. 2025). ISSN: 0893-1321. DOI: 10.1061/jaeeez.aseng-6423.

[14] Stephen Cimpoeru et al. *The Science of Armour Materials*. Oct. 2016, p. 229. ISBN: 978-0-08-101002-0.

[15] Alvin Combrink, Sabino Francesco Roselli, and Martin Fabian. *Prioritized Planning for Continuous-time Lifelong Multi-agent Pathfinding*. 2025. arXiv: 2503.13175 [eess.SY]. URL: https://arxiv.org/abs/2503.13175.

[16] *Corvus Robotics*. URL: https://www.corvus-robotics.com/.

[17] *DECAWAVE DWM1000 Transceiver Module Positioning*. URL: https://electropeak.com/transceiver-module-positioning-with-an-accuracy-of-10-cm-dwm1000-product-decawave?srsltid=AfmBOopthc7RkjYh_EF4RPlc-HF2vPAgLpqiJ1_IbgcPv7z8HxyRvRdQuy0.

[18] Md Fazle Rabbi, Rajesh Nandi, and Mohammad Mashud. "Induce Drag Reduction of an Airplane Wing". In: *American Journal of Engineering Research (AJER)* 4.6 (2015), pp. 219–223. ISSN: 2320-0936. URL: www.ajer.org.

[19] Ricardo Fernandez-Aldama et al. “High-fidelity simulations of airfoil vortex-induced vibrations: From 2D to blade-like aspect ratios”. In: *Journal of Physics: Conference Series*. Vol. 2767. 2. Institute of Physics, 2024. DOI: 10.1088/1742-6596/2767/2/022054.

[20] Miguel Figliozzi. “Multicopter drone mass distribution impacts on viability, performance, and sustainability”. In: *Transportation Research Part D: Transport and Environment* 121 (Aug. 2023), p. 103830. ISSN: 13619209. DOI: 10.1016/j.trd.2023.103830.

[21] *Industry-specific cost of capital parameters*. URL: https://atlas.kpmg.com/de/en/deal-advisory-services/cost-of-capital-and-multiples/cost-of-capital.

[22] R E Kalman. “A New Approach to Linear Filtering and Prediction Problems”. In: *Journal of Basic Engineering* 82.1 (June 1960), pp. 35–45. ISSN: 0021-9223. DOI: 10.1115/1.3662552. URL: https://doi.org/10.1115/1.3662552.

[23] Kim Tae. *UC Irvine Electronic Theses and Dissertations Title Reduction of Tonal Propeller Noise by Means of Uneven Blade Spacing*. Tech. rep. University of California, Irvine, 2016. URL: https://escholarship.org/uc/item/9q75v9t9.

[24] Joshua Laber, Ravindra Thamma, and E Daniel Kirby. *The Impact of Warehouse Automation in Amazon's Success*. Tech. rep. 2020. URL: www.ijiset.com.

[25] Jiaoyang Li et al. “Anytime Multi-Agent Path Finding via Large Neighborhood Search”. In: *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*. AAMAS '21. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2021, pp. 1581–1583. ISBN: 9781450383073.

[26] Ligpower. “ligpower_u7_v2_0_kv490_industrial_uav_motor_for_10_datasheet”. In: (). URL: https://cdn.robotshop.com/media/T/Tmo/RB-Tmo-287/pdf/ligpower_u7_v2_0_kv490_industrial_uav_motor_for_10_datasheet.pdf.

[27] LithiumWerks. *ANR26650m1B Power Cell*. URL: https://lithiumwerks.com/products/lithium-ion-26650-cells/ (visited on 06/17/2026).

[28] Mats Macke and Joost Couwenberg. *Drive-FC LCA*. URL: https://github.com/MATSMACKE/drive-fc-lca (visited on 06/17/2026).

[29] Donald W Marquardt. *An Algorithm for Least-Squares Estimation of Nonlinear Parameters*. Tech. rep. 2. 1963, pp. 431–441.

[30] T. H. G.. Megson. *Introduction to aircraft structural analysis*. Butterworth-Heinemann, 2018. ISBN: 9780081020760.

[31] Metric Multistandard. *Minimum Ultimate Tensile Loads Proof Loads*. URL: https://www.metricmcc.com/images/steel_bolts_tensile_proof_loads.pdf (visited on 06/15/2026).

[32] Andri Mirzal, Shinichiro Yoshii, and Masashi Furukawa. *PID Parameters Optimization by Using Genetic Algorithm*. Tech. rep. Sapporo, Japan: Graduate School of Information Science and Technology Hokkaido University.

[33] *MPU-9250 Product Specification Revision 1.0 MPU-9250 Product Specification*. Tech. rep. 2014.

[34] *Netherlands Hourly Wage Growth YoY*. URL: https://tradingeconomics.com/netherlands/wage-growth.

[35] *Online Grocery Market Size & Share | Industry Report, 2033*. URL: https://www.grandviewresearch.com/industry-analysis/online-grocery-market.

[36] *OSRS: Ocado Storage & Retrieval System*. URL: https://ocadointelligentautomation.com/systems/osrs-ocado-storage-retrieval-system.

[37] Jongho Park and Youdan Kim. “Collision avoidance for quadrotor using stereo vision depth maps”. In: *IEEE Transactions on Aerospace and Electronic Systems* 51.4 (Oct. 2015), pp. 3226–3241. ISSN: 00189251. DOI: 10.1109/TAES.2015.140222.

[38] Cai Qi et al. “Advances in degradation mechanism and sustainable recycling of LiFePO4-type lithium-ion batteries”. In: *Energy Storage Materials* 71 (2024), p. 103623. ISSN: 2405-8297. DOI: https://doi.org/10.1016/j.ensm.2024.103623. URL: https://www.sciencedirect.com/science/article/pii/S2405829724004495.

[39] G Renukadevi and Javin Tony. “The Evolution of Robotics: Current Trends and Future Trajectories”. In: (2025). ISSN: 2349-9249. URL: www.tijer.org.

[40] *Report of the World Commission on Environment and Development: Our Common Future*. Tech. rep. United Nations Brundtland Commsion, 1987.

[41] SciPy Docs. *Trust-Region Constrained Algorithm*. URL: https://docs.scipy.org/doc/scipy/tutorial/optimize.html#constrained-minimization (visited on 06/15/2026).

[42] ServoCity. *2000 Series 5-Turn, Dual Mode Servo*. URL: https://www.servocity.com/2000-series-5-turn-dual-mode-servo-25-2-torque/ (visited on 05/20/2026).

[43] SmallRig. *SmallRig 870 15mm Carbon Fiber Rod*. URL: https://www.kamera-express.nl/smallrig-870-15mm-carbon-fiber-rod-20cm-8inch-2pcs (visited on 06/16/2026).

[44] SNS Insider. *Warehouse Automation Market Size to Exceed $71.25 Billion by 2033, at 15.93% CAGR | Research by SNS Insider*. Dec. 2025.

[45] Felix Starke, Magdalena Worm, and Jean Pujol. *The future of grocery shopping*. 2022. URL: https://www.strategyand.pwc.com/de/en/industries/consumer-markets/future-of-grocery-shopping.html.

[46] *The Amazon Robotics Family: Kiva, Pegasus, Xanthus, and more... - AllAboutLean.com*. URL: https://www.allaboutlean.com/amazon-robotics-family/.

[47] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *PROBABILISTIC ROBOTICS*. Tech. rep. 1999.

[48] Mitsubishi Forklift trucks. *AXiA ES SBP10-16N3(I)(R)(S) Series*. URL: https://mitforklift.com/en/gu/mitsubishi-axia-es-sbp16n3-sbp12n2c-pedestrian-stackers/specifications (visited on 06/17/2026).

[49] U.S Department of Transportation. *Clarus Concept of Operations*. Tech. rep. U.S Department of Transportation, Oct. 2005. URL: https://web.archive.org/web/20090705102900/http://www.itsdocs.fhwa.dot.gov/jpodocs/repts_te/14158.htm#_Toc113261370.

[50] United Nations. **1515900* Transforming our world: the 2030 Agenda for Sustainable Development Transforming our world: the 2030 Agenda for Sustainable Development*. English. Tech. rep. New York: United Nations, 2015, pp. 0–35. URL: https://digitallibrary.un.org/record/803352?ln=en&v=pdf#files.

[51] Dima Usov et al. *Aerodynamics and Acoustics of Asymmetric Propellers with Uneven Blade Spacing*. Tech. rep.

[52] *Verity | Warehouse Intelligence Platform - Autonomous Inventory & Intelligent AI Orchestration*. URL: https://www.verity.net/.

[53] Yingrui Wang et al. "Improved A* Algorithm and Dynamic Obstacle Avoidance Method for Path Planning and Self- Positioning of Autonomous Guided Vehicles". In: *2024 International Conference on Intelligent Robotics and Automatic Control (IRAC)*. 2024, pp. 607–612. DOI: 10.1109/IRAC63143.2024.10871658.

[54] Zhanchi Wang, Nikolaos M. Freris, and Xi Wei. "SpiRobs: Logarithmic spiral-shaped robots for versatile grasping across scales". In: *Device* 3.4 (2025), p. 100646. ISSN: 2666-9986. DOI: https://doi.org/10.1016/j.device.2024.100646. URL: https://www.sciencedirect.com/science/article/pii/S2666998624006033.

[55] *Warehouse Automation Market - Industry Size & Growth 2025 - 2031*. Tech. rep. Mordor Intelligence. URL: https://www.mordorintelligence.com/industry-reports/warehouse-automation-market.

[56] Wei Wei et al. "Analysis and Evaluation of Aerodynamic Noise Characteristics of Toroidal Propeller". In: *Drones* 8.12 (Dec. 2024). ISSN: 2504446X. DOI: 10.3390/drones8120753.

[57] *What is Brief History of AutoStore Company? – businessmodelcanvastemplate.com*. URL: https://businessmodelcanvastemplate.com/blogs/brief-history/autostore-brief-history.

[58] *World's Fastest AS/RS | 4x Space & 99.8% Uptime | AutoStore*. URL: https://www.autostoresystem.com/.

[59] Chengjian Xu et al. "Future greenhouse gas emissions of automotive lithium-ion battery cell production". In: *Resources, Conservation and Recycling* 187 (2022), p. 106606. ISSN: 0921-3449. DOI: https://doi.org/10.1016/j.resconrec.2022.106606. URL: https://www.sciencedirect.com/science/article/pii/S0921344922004402.

[60] Shufeng Zhai and Wanyue Huxiao. "Study on Noise Reduction of Fans with Uneven Blade Spacing". In: *IOP Conference Series: Earth and Environmental Science*. Vol. 508. 1. Institute of Physics Publishing, June 2020. DOI: 10.1088/1755-1315/508/1/012162.

[61] Wenda Zhao et al. *UTIL: An Ultra-wideband Time-difference-of-arrival Indoor Localization Dataset*. 2024. URL: https://arxiv.org/abs/2203.14471.

A

Cost Breakdown

Inputs & Assumptions		
Overall params		
Project lifespan	10	years
Starting year	2026	
Discounnt rate (WACC)	8%	
Fully-loaded wage multiplier	1.00x	
Total annual warehouse throughput	44,000,000	/year
Throughput		
Time per item	1.50	min/item
Per drone daily throughput	1,846	picks/drone/day
Daily throughput	23,998	picks/day
Annual throughput	8,759,270	picks/year
Per drone annual throughput	515,251	picks/drone/year
Avg employee throughput	332,640	picks/year
Market CAGR	15.93%	/year
Number of drones inactive	4	drones
Labor		
Average gross annual wage	€ 35,728.00	EUR/FTE/year
Fully-loaded annual cost per FTE	€ 35,728.00	EUR/FTE/year
FTE count for human only	133	FTEs
Remanining FTE with drone system	106	FTEs
CAPEX (one time costs)		
Number of drones	17	drones
Per-drone unit cost(excl. battery)	€ 1,792.18	EUR/drone
Drone fleet cost	€ 32,669.58	EUR
Installation	€ 148,200.00	EUR
Communication infrastructure	€ 3,162.06	EUR
ERS	€ 31,155.40	EUR
Central computer	€ 5,280.00	EUR
Manufacturing costs	€ 50,000.00	EUR
Development labour costs	€ 704,000.00	EUR
Subtotal CAPEX(pre-contingency)	€ 974,467.04	EUR
Contingency	20%	% of total
Total CAPEX	€ 1,169,360.45	EUR
OPEX (annual, full operations)		
Battery unit cost	€ 129.56	EUR/battery
Battery replacements per year	204	/year
Annual battery costs	€ 26,430.24	EUR/year
Maintainance & repairs	€ 70,000.00	EUR/year
Drone attiritions	1%	% of fleet
Attrition replacement cost	€ 1,000.00	EUR/year
Electricity for charging	€ 5,000.00	EUR/year
Motor replacements	€ 6,776.00	EUR/year
Insurance	€ 10,000.00	EUR/year
Human labour costs	€ 3,787,168.00	EUR/year
Total annual OPEX	€ 3,905,384.24	EUR/year
End of life		
Ramp factor - Year 1	100%	% of full efficiency
Ramp factor - Year 2	100%	% of full efficiency
Ramp factor - Year 3+	100%	% of full efficiency
Drone fleet life	5	years
Salvageable costs	2%	% of CAPEX

Figure A.1: List of main inputs and assumptions for cash flow analysis

Year	0	1	2	3	4
Calendar year	2026	2027	2028	2029	2030
In operation(1/0)	0	1	1	1	1
Ramp factor	0%	100%	100%	100%	100%
Throughput & Labor					
Annual throughput	0	8759270	8759270	8759270	8759270
Minimum employee count	-	106	106	106	106
Annual minimul labor cost	-	€4,751,824.00	€5,131,969.92	€5,542,527.51	€5,985,929.71
Drone-scenario labour cost	-	€3,787,168.00	€4,090,141.44	€4,417,352.76	€4,770,740.98
Gross labour cost saving	-	€964,656.00	€1,041,828.48	€1,125,174.76	€1,215,188.74
Costs					
System OPEX	-	€3,905,384.24	€4,208,357.68	€4,535,569.00	€4,888,957.22
Net operating saving	-	€846,439.76	€923,612.24	€1,006,958.52	€1,096,972.50
CAPEX	€1,169,360.45	-	-	-	-
Salvage value	-	-	-	-	-
Net Cash Flow	(€1,169,360.45)	€846,439.76	€923,612.24	€1,006,958.52	€1,096,972.50
Discount factor	1.000	0.926	0.857	0.794	0.735
Discounted Cash Flow	(€1,169,360.45)	€783,740.52	€791,848.63	€799,356.14	€806,307.53
Cumulative DCF	(€1,169,360.45)	(€385,619.93)	€406,228.70	€1,205,584.84	€2,011,892.37
Payback reached (y/n)	no	no	yes	yes	yes

5	6	7	8	9	10
2031	2032	2033	2034	2035	2036
1	1	1	1	1	1
100%	100%	100%	100%	100%	100%
8759270	8759270	8759270	8759270	8759270	8759270
106	106	106	106	106	106
€6,464,804.09	€6,981,988.42	€7,540,547.49	€8,143,791.29	€8,795,294.60	€9,498,918.16
€5,152,400.25	€5,564,592.27	€6,009,759.66	€6,490,540.43	€7,009,783.66	€7,570,566.36
€1,312,403.84	€1,417,396.15	€1,530,787.84	€1,653,250.86	€1,785,510.93	€1,928,351.81
€5,270,616.49	€5,682,808.51	€6,127,975.90	€6,608,756.67	€7,127,999.90	€7,688,782.60
€1,194,187.60	€1,299,179.91	€1,412,571.60	€1,535,034.62	€1,667,294.69	€1,810,135.57
€30,467.06	-	-	-	-	€30,467.06
-	-	-	-	-	€23,387.21
€1,163,720.54	€1,299,179.91	€1,412,571.60	€1,535,034.62	€1,667,294.69	€1,803,055.72
0.681	0.630	0.583	0.540	0.500	0.463
€792,008.64	€818,703.72	€824,221.96	€829,331.44	€834,062.45	€835,163.67
€2,803,901.01	€3,622,604.73	€4,446,826.69	€5,276,158.13	€6,110,220.58	€6,945,384.25
yes	yes	yes	yes	yes	yes

Figure A.2: Full Cash flow table of DRIVE-FC

Sensitivity analysis					
Changed Variable: CAPEX	NPV	Payback period	Discounted payback period	Net present labour savings	Savings (%)
	€ 11,885,989.37	0.8964	2.00	€ 13,872,605.12	27.01%
€ 950,000.00	€ 11,915,077.83	0.8739	2.00	€ 13,872,605.12	27.08%
€ 1,000,000.00	€ 11,855,633.66	0.9199	2.00	€ 13,872,605.12	26.95%
€ 1,050,000.00	€ 11,796,189.49	0.9659	2.00	€ 13,872,605.12	26.81%
€ 1,100,000.00	€ 11,736,745.32	1.0119	2.00	€ 13,872,605.12	26.68%
€ 1,150,000.00	€ 11,677,301.16	1.0579	2.00	€ 13,872,605.12	26.54%
€ 1,200,000.00	€ 11,617,856.99	1.1039	2.00	€ 13,872,605.12	26.41%
€ 1,250,000.00	€ 11,558,412.82	1.1499	2.00	€ 13,872,605.12	26.27%
€ 1,300,000.00	€ 11,498,968.65	1.1959	2.00	€ 13,872,605.12	26.13%
€ 1,350,000.00	€ 11,439,524.48	1.2419	2.00	€ 13,872,605.12	26.00%
€ 1,400,000.00	€ 11,380,080.32	1.2879	2.00	€ 13,872,605.12	25.86%
€ 1,450,000.00	€ 11,320,636.15	1.3339	3.00	€ 13,872,605.12	25.73%
Changed Variable: OPEX	NPV	Payback period	Discounted payback period	Net present labour savings	Savings (%)
	€ 11,885,989.37	0.8964	2.00	€ 13,872,605.12	27.01%
€ 3,500,000.00	€ 14,606,150.62	0.683905073	2.00	€ 13,872,605.12	33.20%
€ 3,750,000.00	€ 12,928,630.27	0.801025856	2.00	€ 13,872,605.12	29.38%
€ 4,000,000.00	€ 11,251,109.92	0.966550514	2.00	€ 13,872,605.12	25.57%
€ 4,250,000.00	€ 9,573,589.57	1.218301328	3.00	€ 13,872,605.12	21.76%
€ 4,500,000.00	€ 7,896,069.22	1.647384258	3.00	€ 13,872,605.12	17.95%
€ 4,750,000.00	€ 6,218,548.87	2.543035532	5.00	€ 13,872,605.12	14.13%
€ 5,000,000.00	€ 4,541,028.52	5.572931873	6	€ 13,872,605.12	10.32%
€ 5,250,000.00	€ 2,863,508.17	never	8	€ 13,872,605.12	6.51%
€ 5,500,000.00	€ 1,185,987.82	never	9	€ 13,872,605.12	2.70%
€ 5,750,000.00	€ 491,532.53	never	never	€ 13,872,605.12	-1.12%
€ 6,000,000.00	€ 2,169,052.88	never	never	€ 13,872,605.12	-4.93%
Changed Variable: throughput	NPV	Payback period	Discounted payback period	Net present labour savings	Savings (%)
	€ 11,885,989.37	0.8964	2.00	€ 13,872,605.12	27.01%
8,250,000	€ 11,317,578.71	0.953248512	2.00	€ 13,304,194.46	25.7%
8,500,000	€ 11,601,784.04	0.923973663	2.00	€ 13,588,399.79	26.4%
8,750,000	€ 11,885,989.37	0.896443337	2.00	€ 13,872,605.12	27.0%
9,000,000	€ 11,885,989.37	0.896443337	2.00	€ 13,872,605.12	27.0%
9,250,000	€ 12,170,194.70	0.870506107	2.00	€ 14,156,810.45	27.7%
9,500,000	€ 12,454,400.04	0.846027581	2.00	€ 14,441,015.79	28.3%
9,750,000	€ 12,738,605.37	0.822888069	2.00	€ 14,725,221.12	29.0%
10,000,000	€ 12,738,605.37	0.822888069	2.00	€ 14,725,221.12	29.0%
10,250,000	€ 13,022,810.70	0.800980625	2.00	€ 15,009,426.45	29.6%
10,500,000	€ 13,307,016.03	0.7802094	2.00	€ 15,293,631.78	30.2%
10,750,000	€ 13,591,221.37	0.760488234	2.00	€ 15,577,837.12	30.9%

Figure A.3: Sensitivity analysis of Cost-benefit estimations