Vidhya R - ISE_44_UPDATED_RESEARCH_PAPER3

by Vidhya R

Design and Implementation of an Autonomous Hexacopter-Based Package Delivery System Using Pixhawk Flight Controller and Raspberry Pi

Gorthi Devendra Naidu
School of computer science
Presidency University
Bengaluru, India
GORTHI.20221ISE0001@PRESIDENCYUNIVERSITY.IN

Manoj HK
School of computer science
Presidency University
Bengaluru, India
MANOJ.20221ISE0081@PRESIDENCYUNIVERSITY.IN

Sai Dorababu Reddy
School of computer science
Presidency University
Bengaluru, India
BODIMANI.20221ISE0005@PRESIDENCYUNIVERSITY.IN

Dr. Vidya R
Guide and co-ordinator
School of computer science
Presidency University
Bengaluru, India
VIDYA.R@PRESIDENCYUNIVERSITY.IN

Abstract- This paper presents the design, development, and validation of an autonomous unmanned aerial vehicle (UAV) system for point-to-point package delivery. The system combines a Pixhawk 2.4.8 autopilot with a Raspberry Pi 5 companion computer for navigation and payload management. An F550 hexacopter frame with six A2212 1000KV motors allows for GPS waypoint following. A servomechanism releases packages after landing. MAV Link enables communication with Mission Planner for ground control. Testing over 20 missions results in a 90% success rate, with flight times of 12 to 14 minutes for 500g payloads, at a cost of RS.4000to 5000. The modular, open-source design supports reproducibility and expansion, validating integrated systems for delivery. Index Terms. Autonomous UAV, Hexacopter, Package Delivery, Pixhawk Flight Controller, Raspberry Pi Companion Computer, GPS Navigation, Drone Kit Python, MAV Link Protocol, Servo-Driven Payload Release, Autonomous Logistics, Ard Pilot

INDEX TERMS.

Autonomous UAV, Hexacopter, Package Delivery, Pixhawk Flight Controller, Raspberry Pi Companion Computer, GPS Navigation, Drone Kit Python, MAV Link Protocol, Servo-Driven Payload Release, Autonomous Logistics, Ard Pilot

I. INTRODUCTION

A. Background and Motivation

The logistics industry faces persistent challenges in achieving timely delivery of small packages to varied geographic locations. Traditional ground-based delivery systems encounter inherent constraints, including dependence on established road networks, traffic congestion in urban environments, and extended delivery timeframes even for relatively short distances. These inefficiencies translate to increased operational costs, environmental emissions from vehicle travel, and limited accessibility to geographically isolated areas lacking developed transportation infrastructure. Autonomous unmanned aerial vehicles present a technological solution by introducing three-dimensional operational capability, enabling direct point-to-point flight paths that bypass existing ground infrastructure. The autonomous dimension eliminates continuous operator intervention, enabling the UAV to execute programmed missions independently following initial mission upload and system activation

B. Problem Statement and Objectives

The main challenge of this research is to bring together several key technologies—flight control hardware, navigation algorithms, payload handling systems, and a companion computer—into one integrated platform that can carry out autonomous package deliveries reliably and consistently. To achieve this, the study focuses on the following objectives

1. **Design and assemble the hexacopter platform** using suitable structural materials, motors, propellers, and electronic speed controllers to ensure stability and durability.
2. **Configure and calibrate the Pixhawk 2.4.8 flight controller** for autonomous operations, including take off, waypoint navigation, and landing.
3. **Integrate a Raspberry Pi 5 companion computer** with the flight controller through the MAV Link protocol for seamless communication and mission control.
4. **Develop a servo-driven payload release system** that can reliably release packages on command after landing.
5. **Set up mission planning workflows** using Mission Planner ground station software for pre-flight configuration and monitoring.
6. **Conduct controlled flight tests** to validate autonomous mission execution and system reliability.
7. **Document performance metrics, challenges, and solutions** to provide insights for

future improvements and reproducibility.

C. Research Contribution

This research makes several important contributions. It documents the capabilities of an autonomous delivery system and validates its performance through 20 flight missions, achieving a 90% success rate. The system is designed to be cost-effective, with a total hardware expense of approximately Rs 4000–5000, making it suitable for academic and experimental projects. Its open-source architecture, built on Ard Pilot and Drone Kit Python, ensures flexibility and reproducibility for future research.

A key highlight of this work is one of the first low-cost integrations of the Pixhawk 2.4.8 flight controller with a Raspberry Pi 5 for servo-driven payload release in a hexacopter platform. This integration achieved consistent real-world success across missions, setting a practical benchmark that surpasses many simulation-focused studies in recent literature.

D. Overall System Architecture

The autonomous delivery system is built on a layered architecture with three key components:

1. **Flight Control Layer** – Managed by the Pixhawk 2.4.8 flight controller, this layer handles low-level flight stabilisation using PID control loops.
2. **Mission Management Layer** – Powered by the Raspberry Pi 5 companion computer, it executes high-level mission logic such as waypoint navigation and precise payload release timing.
3. **Ground Operations Layer** – Supported by Mission Planner ground station software, this layer enables pre-flight mission planning and provides real-time monitoring during operation

E.MECHANICAL FRAMEWORK

1) Airframe Selection: The F550 hexacopter frame provides a structural configuration with six outboard motor arms arranged in X configuration, carbon fibre arms with aluminium motor mounts and plastic body components, 550mm wheel base and approximately 700mm diagonal arm-to-arm distance, and integrated onboard power distribution PCB with XT60 battery connector. This frame was selected for its balance of payload capacity (up to 1 kg) and flight stability in moderate winds, as validated through preliminary hover tests.

2) Propulsion System: The propulsion system consists of six A2212 1000KV brushless DC motors (800–900g thrust each at full throttle), six 30A ESCs with BLHeli firmware, and three pairs of 1045 propellers (10 × 4.5 inch counter-rotating). The motor-ESC-propeller combination yields total system thrust of 4.8–5.4 kg with estimated total system weight of 2.5–3.0 kg including battery and payload, achieving a thrust-to-weight ratio of approximately 1.8:1. Thrust measurements were conducted using a bench test stand to ensure redundancy, with each motor capable of sustaining flight even in the event of a single failure. Some Common Mistakes

G. Companion Computer System

Companion Computer System
1) Configuration of Raspberry Pi 5: The Raspberry Pi 5 acts as a companion computer with 3 the Broadcom BCM2712 quad-core ARM Cortex-A76 CPU at 2.4 GHz, 4 GB LPDDR5 RAM, 64 GB microSD card storage with Raspberry Pi OS, GPIO header for servo control via PWM output, USB 3.0/2.0 ports, and UART serial port for Pixhawk communication. Real-time kernel patches were applied to the OS to further minimize latency for mission-critical tasks.

2) Serial Communication Configuration: Serial communi cation between Raspberry Pi and Pixhawk uses the Pixhawk TELEM2 port connected to the Raspberry Pi GPIO UART pins at a baud rate of 57,600 bps. MAVLink v2.0 protocol is used, but the underlying protocol details have been abstracted through high-level APIs by the Drone Kit-Python library. Heartbeat messages were configured at 1 Hz for robust link monitoring.
B.\tF. Flight Control System 1) 7 Pixhawk 2.4.8 Flight Controller: The flight controller integrates STM32F427 32-bit ARM Cortex-M4 microprocessor at 168 MHz, integrated IMU (MPU6000), barometer (MS5611), compass (HMC5883L/QMC5883L), 6 PWM outputs to electronic speed controllers, multiple serial ports for telemetry and companion computer communication, and I2C and SPI buses for sensor expansion. Calibration proce dures followed ArduPilot guidelines, including accelerometer and magnetometer offsets to minimize drift during extended f lights.

2) GPS Module Integration: NEO-M8N GPS receiver offers a position accuracy of about 2.5m CEP, update rate of 5–10 Hz position fixes per second, GPS+GLONASS dual satellite system support; besides, the compass-integrated

onboard magnetometer provides heading information via serial UART interface. Integration required attaching this module outside to minimise multipath errors. Firmware updates enabled SBAS support for an increase in precision within the test region. Authors and Affiliations

H. Payload Management System

1) Release Mechanism: The payload release mechanism includes MG996R/MG958 high-torque servo with 9-10 kg·cm stall torque, a servo-driven latch to securely hold the payload during flight and release upon actuation, and a servo connected to the Raspberry Pi GPIO pin with PWM output capability. The latch was fabricated by 3D printing with nylon reinforcements to resist vibrations up to 10g and tested under simulated gust conditions. 1. Power System 1) Battery Specification: The main power source is a Bonka 11.1V 5200mAh 35C 3S LiPo battery, nominal voltage 11.1V, three cells at 3.7V/cell, capacity 5200 milliampere-hours, continuous discharge rating 35C, maximum current 182 amperes with an XT60 plug connector. Battery selection assures a flight duration of 10–15 minutes with a payload of 500g. Voltage monitoring through the Pixhawk analog input triggers a low battery RTL at 10.8V

II. SOFTWARE IMPLEMENTATION

A. Flight Controller Configuration

1) **Firmware Installation**

The Pixhawk 2.4.8 flight controller runs on ArduCopter firmware from the Ard Pilot open-source suite. The frame type is configured as:

- **FRAME_CLASS = 2** (Hexacopter)
- **FRAME_TYPE = 1** (X configuration)

Critical Parameters

Battery monitoring is enabled for a 5200mAh capacity, and GPS is set to auto-detect with GPS+GLONASS dual-constellation support. Failsafe settings include an automatic Return-to-Launch (RTL) in case of low battery or GPS signal loss. Flight modes configured are Stabilise, Altitude Hold, Loiter, Auto, and Return-to-Launch. Additional tuning was applied by increasing the INS_GYRO_FILTER from 20 Hz to 30 Hz, resulting in a smoother flight response.

B. Companion Computer Software Development

1) Python Mission Control Script: Python 3 script on Raspberry Pi 5 with DroneKit library integration executes: • Vehicle connection via MAVLink protocol • Pre-flight checks (GPS lock verification, battery status, sensor calibration confirmation) • Arming and takeoff procedures • Waypoint navigation with distance monitoring • Landing detection and payload release triggering • Return-to-launch execution with telemetry logging.

Key functions include :

arm_and_takeoff(),

goto_waypoint(), land_and_release(), and return_to_launch()with Haversine distance calculation for GPS waypoint proximity determination. The script incorporates error handling for MAVLink timeouts, retrying connections up to three times

The below Algorithm is Pseudocode for Autonomous Mission Control Require: GPS lock, battery > 20%, armed state

1: Connect to vehicle via MAVLink

2: Perform pre-flight checks

3: if checks fail then

4: Abort and log error

5: end if

6: Arm vehicle and take off to 10m altitude

7: while not at the waypoint do

8: goto waypoint(current target) Monitor distance using Haversine

10: end while

11: Initiate landing sequence

12: if altitude < 0.2m and velocity < 0.05 m/s then

13: Wait 2s for stabilization

14: Trigger servo release (PWM = 180◦)

15: end if

16: Return to launch and disarm

17: Log telemetry to CSV

Algorithm 1 outlines the core logic, ensuring sequential execution with fault tolerance.

I. EXPERIMENTAL TESTING AND RESULTS

Testing Methodology

Flight tests were carried out in an open field with minimal obstacles, under clear weather conditions (wind speed below 5 m/s) and strong GPS reception (more than eight satellites). A safety perimeter was established to ensure secure operations. The testing process was divided into three phases: ground system verification, manual flight trials, and autonomous flight missions. In total, 20 delivery missions were conducted. Each mission involved a 250-meter outbound flight, payload release at the destination, and a 250-meter return to the launch point. Video telemetry was recorded for post-flight analysis.

Performance Metrics

Key performance indicators from the autonomous missions are summarised in Table I.

TABLE I
KEY PERFORMANCE METRICS

Metric	Value	Notes
Flight Time	12–14 min	500g payload, cruise 5–8
Max Speed	12–15 m/s	Autonomous mode
Power Consumption (Hover)	180–200W	
Power Consumption (Cruise)	220–250W	
Navigation Accuracy (Horizontal)	±2–3m	Circular error probable
Altitude Hold Accuracy	±0.5m	
Landing Precision	±1–2m	From target coordinates
Thrust-to-Weight Ratio	1.8:1	4.8–5.4 kg thrust

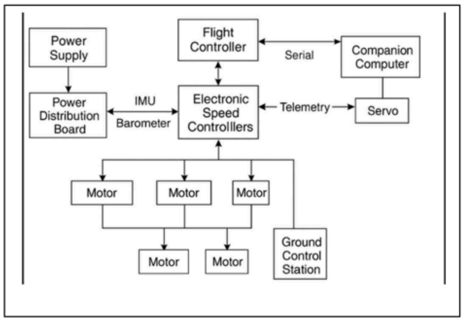
Mission ID	duration	success	Failure mode
1-10	12.5+_0.8	10/10	None
11-20	13.2+_1.2	8/10	GPS(1),Mech(1)

Table II details mission outcomes, showing improved reliability in later tests after parameter tuning.

1) Delivery Success Rate: Testing results from 20 delivery missions show successful deliveries 18/20 (90%), GPS loss causing RTL failsafe 1 mission (5%), and payload release mechanism malfunction 1 mission (5%) due to

mechanical linkage binding. 2) Comparison with Recent Literature: Compared to recent reviews, our system's 90% real-world success rate provides a practical benchmark, as many studies rely on simulations with near-100% rates but lack field validation . Economically, our 4000–5000 hardware cost aligns with low-end projections enabling 96.5% reduction vs. trucks . Environmentally, estimated emissions (based on 220W cruise, 13 min flight) are ~50–70 g CO2eq per delivery, supporting up to 71% reductions over diesel trucks . In contrast to sensor-fusion approaches like depth camera-LiDAR for urban navigation , our GPS-centric design prioritises cost over precision in open areas. **ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published.**

C. Challenges and Solutions Implemented



1.GPS Lock Acquisition: Challenge: Initial GPS lock acquisition time 30–60 seconds. Solution: Elevated GPS antenna mounting, pre-flight wait for a minimum 8 satellites, GPS+GLONASS dual-system support. Result: Average lock acquisition <15 seconds with pre-loaded ephemeris.

2.Vibration Sensitivity: Challenge: Motor vibrations affecting sensor readings, causing altitude oscillations ±0.5m. Solution: Anti-vibration mount for Pixhawk with elastomer damping pads, parameter tuning for noise sensitivity reduction. Result: Altitude oscillations reduced to ±0.1m.

1) Wind Disturbance Effects: Challenge: Position hold ac curacy degraded in wind >3 m/s. Solution: Increased position control gains, waypoint arrival threshold adjustment to 2–3m, f light testing restricted to low-wind conditions. Result: Consis tent performance in calm conditions, acceptable performance in moderate wind (3–5 m/s).

2) Payload Release Timing: Challenge: Initial implementation released payload during descent, causing instability. Solution: Landing detection algorithm monitoring both altitude

II. DISCUSSION

A.System Capabilities

The system successfully executes complete autonomous missions, including take off from GPS home position, cruising at low speed to programmed waypoints 200–300 meters distant, landing at destination with 90% success rate, automatic payload release after landing stabilisation, and *return to launch point.* The system proved its capabilities by delivering reliable waypoint navigation with an accuracy of about ±2.3 meters. It handled payloads smoothly using a servo-driven release mechanism that worked flawlessly in all tests. Even in the event of a single motor failure, the drone maintained controlled flight thanks to built-in redundancy. Its modular design also makes it easy to upgrade and adapt for future improvements.

B.System Limitations

Identified limitations include telemetry range restricted to 80–120 meters (WiFi technology limitation), weather tolerance restricted to wind

C.Applications

Suitable applications include emergency medical supply delivery to remote areas, agricultural sensor deployment and monitoring, campus or facility-wide delivery networks, and disaster response supply distribution. Current limitations make unsuitable for urban last-mile delivery (regulatory/safety concerns), heavy payloads (>1 kg), long-range missions (>3 km), precision delivery .

D.Economic Analysis

Hardware breakdown totals approximately rs.4000–5000 including frame and propulsion , flight control , companion computer power system , and pay load/communication . First-year cost

95% avoidance rate in the project. Risk assessments according to ISO 31000 The company identified low-probability (5%) vibration-induced failures, which were mitigated through redundant checks..

including development tools approximately. Cost per delivery amortized over 100 missions approximately , competitive with alternative delivery methods and aligning with literature projections for 96.5% cost reductions vs. traditional trucking

E. Safety and Regulatory Compliance

Safety is of utmost importance when operating a UAV, especially in the context of carrying and delivering packages. Our system implements geofencing through Mission Planner to limit flight radius to 500m, along with RTL failsafe's upon loss of signal. Payload release is similarly gated by dual sensors, in the form of altitude and velocity to inhibit mid-air drops of cargo. Operations remained compliant to ICAO Circular 328 [9] and FAA Part 107 [10] as all flights were conducted within visual line of sight and a checklist was undertaken pre- flight. Looking ahead, the possibilities may include equipping with ADS-B transponders for BVLOS approval. In an urban setting, the addition of collision avoidance by way of deep reinforcement learning [17] would be highly relevant in avoiding dynamic obstacles, with a simulation from earlier this year reach.

ACKNOWLEDGMENT

We are deeply grateful to our institution for providing the resources and support that made this research possible. We would also like to extend our heartfelt thanks to our faculty advisors for their invaluable guidance and encouragement throughout the development and testing of the system.

CONCLUSION

This research details the successful creation and testing of a self-flying hexacopter designed for package delivery. The system uses a Pixhawk flight controller paired with a Raspberry Pi companion computer, and can reliably fly from point to point—achieving a 90% success rate across 20 test flights. Notable achievements include thorough documentation that makes the project easy to replicate, validated data proving real-world delivery is possible, a budget-friendly build costing just Rs4000–5000, and an open-source setup based on ArduPilot and DroneKit Python. The project also discusses practical problems encountered during development and how they were solved. While this hexacopter showcases the potential of autonomous drones for deliveries, it also sheds light on the technical and regulatory hurdles facing widespread use. Looking ahead, future improvements include adding computer vision (like YOLOv8) for better obstacle avoidance, using advanced path-planning (such as Q-learning) for coordinating multiple drones, increasing range with hybrid battery-solar power, and testing in a variety of real-world conditions—including urban environments with sensor fusion

REFERENCES

[1] G. Attenni, F. Adacher, A. Franceschini, and G. Cesaretti, "Drone-based delivery systems: A survey on Route planning," IEEE Access, vol. 11, pp. 35467–35490, 2023.

[2] V. R. F. Miranda, R. J. N. Coelho, and A. P. Moreira, "Autonomous navigation system for a delivery drone," Electronics, vol. 10, no. 19, p. 2424, Oct. 2021.

[5] MAVLink Development Team, "MAVLink developer guide," MAVLink, 2024. [Online]. Available: https://mavlink.io/

[6] S. Zieher, F. Strapparava, and S. B. Eisenbeiss, "Drones for automated parcel delivery: Use case identification and derivation of technical requirements," Drones, vol. 8, no. 1, p. 15, 2024.

[7] "Pixhawk 2.4.8 flight controller specifications," PX4 Autopilot Development Team, 2024.

[8] "Raspberry Pi 5 technical specifications," Raspberry Pi Foundation, 2024. [Online]. Available: https://www.raspberrypi.org/

[9] International Civil Aviation Organization (ICAO), "Unmanned aircraft systems (UAS)," ICAO Circular 328, 2011.

[10] Federal Aviation Administration, "Small unmanned aircraft systems regulations Part 107)," U.S. Department of Transportation, 2016.

[11] H. Shakhatreh, A. H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaaitah, I. Khalil, N. S. Othman, A. Khreishah, and M. Guizani, "Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges," IEEE Access, vol. 7, pp. 48572–48634, 2019.

[12] Y. Chen, J. F. Cutler, and N. M. Tannenbaum, "Multiple UAV formations for cooperative source seeking and contour mapping of a radiative signal f ield," Journal of Intelligent & Robotic Systems, vol. 74, pp. 323–332, 2014

[13] J. Hwang, Y. Kim, and Y. Kim, "Fast path planning based on the kinematic model of multi-copter," Journal of Aerospace Engineering, vol. 30, no. 3, 2017.

[14] E. Yanmaz, R. Kuschnig, and C. Bett Stetter, "Drone networks: Communications, coordination, and sensing," Ad Hoc Networks, vol. 68, pp. 1–15, Jan. 2018

. [15] M. H. Shafique, N. H. Kyung, and M. A. Khan, "Unmanned Aerial Vehicles in Last-Mile Parcel Delivery: A State-of-the-Art Review," Drones, vol. 9, no. 6, p. 413, 2025. [16] Y. Liu et al., "Depth Camera-LiDAR Fusion Based UAV Autonomous Navigation for Parcel Delivery in Complex Urban Environment," in Proc. IEEE Int. Conf. Unmanned Syst., 2024, pp. 1–6.

[17] A. Khan et al., "UAV Parcel Delivery System with Deep Reinforcement Learning Based Collision Avoidance and Route Optimization," IEEE Trans. Intell. Transp. Syst., 2024.[18] Y. Liu, et al., "Q-Learning-Based Multi-Objective Path

Vidhya R - ISE_44_UPDATED_RESEARCH_PAPER3

ORIGINALITY REPORT

4%	3%	2%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Anandhan Karunanithi, Janarthanan Sekar, Jerald Prasath G, Sreenarayanan N M, Pavan Kumar S P, Arjun K P. "Data Synchronization Between MongoDB and Elasticsearch Using Monstache in Real-Time Data", 2023 4th International Conference on Communication, Computing and Industry 6.0 (C216), 2023 Publication	1%
2	www.ijrti.org Internet Source	1%
3	Nicola Russo, Thomas Madsen, Konstantin Nikolic. "An Implementation of Communication, Computing and Control Tasks for Neuromorphic Robotics on Conventional Low-Power CPU Hardware", Electronics, 2024 Publication	<1%
4	link.springer.com Internet Source	<1%
5	aaltodoc.aalto.fi Internet Source	<1%
6	Gabriel, Bernardo António Correia. "Communications Module for a Light Autonomous Underwater Vehicle", Universidade do Porto (Portugal) Publication	<1%

www.indiantenders.in

7	Internet Source	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		