

TRƯỜNG ĐẠI HỌC BÁCH KHOA - ĐẠI HỌC ĐÀ NẴNG
KHOA ĐIỆN TỬ VIỄN THÔNG



BÁO CÁO LẬP TRÌNH MẠNG

ĐỀ TÀI
MÔ PHỎNG DRONE TÌM KIẾM VÀ TRÁNH
VẬT CẢN TRONG PX4

GVHD: TS Nguyễn Văn Hiếu

Họ và tên: Nguyễn Thanh Tùng - 106200283 - 20KTMT2

Nguyễn Nam Phương - 106220266 - 22KTMT2

Đà Nẵng, ngày 13 tháng 12 năm

1. Giới thiệu chung:	2
1.1 Bối cảnh và động cơ đề tài:	2
1.2 Mục tiêu đề tài:	3
1.3 Phạm vi và giới hạn:	3
2. Cơ sở lý thuyết:	3
2.1 Tổng quan PX4 Autopilot:	3
2.2 Giao thức MAVLink và MAVSDK:	4
2.3 Mô phỏng với Gazebo:	4
2.4 Điều khiển Offboard:	4
3. Kiến trúc hệ thống:	4
3.1 Các thành phần hệ thống:	4
3.2 Kiến trúc tổng thể:	4
4. Thiết kế thuật toán:	5
4.1 Thuật toán lawnmower:	5
4.2 Phát hiện vật cản:	5
4.3 Thuật toán tránh vật cản:	5
4.4 Ghi log dữ liệu:	6
5. Thực nghiệm và đánh giá:	6
5.1 Kịch bản mô phỏng:	6
5.2 Các metrics đánh giá:	8
5.3 Phân tích kết quả:	8
7. Kết luận:	9
8. Hướng phát triển:	9

Thành viên	Đóng góp
Nguyễn Thanh Tùng	60%
Nguyễn Nam Phương	40%

Bảng 1: bảng mức độ đóng góp

1. Giới thiệu chung:

1.1 Bối cảnh và động cơ đề tài:

Trong những năm gần đây, thiết bị bay không người lái (UAV - Unmanned Aerial Vehicle), thường gọi là drone, ngày càng được ứng dụng rộng rãi trong nhiều lĩnh vực như quan sát, giám sát môi trường, nông nghiệp thông minh và đặc biệt là tìm kiếm cứu nạn trong các khu vực thiên tai.

Trong các tình huống khẩn cấp như động đất, lũ hoặc sạt lở đất, việc tiếp cận hiện trường bằng con người gặp nhiều nguy hiểm và hạn chế. Drone có khả năng tiếp cận nhanh, quan sát từ trên cao và thu thập dữ liệu hiệu quả. Tuy nhiên, để drone hoạt động an toàn và hiệu quả, cần có các thuật toán điều khiển tự động cũng như khả năng tránh vật cản trong môi trường phức tạp.

Trước khi triển khai trên drone thật, việc mô phỏng đóng vai trò quan trọng nhằm kiểm tra thuật toán, đánh giá độ ổn định và giảm thiểu rủi ro. Do đó, đề tài này tập trung xây dựng một hệ thống mô phỏng drone sử dụng PX4 SITL kết hợp Gazebo, cho phép thực hiện nhiệm vụ tìm kiếm theo quỹ đạo định sẵn và tránh vật cản một cách tự động.

1.2 Mục tiêu đề tài:

Mục tiêu của đề tài bao gồm:

- Xây dựng môi trường mô phỏng drone sử dụng PX4 SITL và Gazebo.
- Thiết kế thuật toán bay tự động theo mẫu tìm kiếm lawnmower.
- Triển khai thuật toán tránh vật cản ở mức nhiệm vụ (mission).
- Thu thập và phân tích dữ liệu bay để đánh giá hiệu quả hệ thống.

1.3 Phạm vi và giới hạn:

Trong phạm vi đề tài, hệ thống chỉ xét các vật cản tĩnh được khai báo trước trong môi trường mô phỏng. Việc phát hiện vật cản dựa trên khoảng cách hình học, không sử dụng dữ liệu cảm biến hình ảnh hoặc LiDAR. Hệ thống được triển khai và đánh giá trong môi trường mô phỏng, chưa áp dụng cho drone thật.

2. Cơ sở lý thuyết:

2.1 Tổng quan PX4 Autopilot:

PX4 là một nền tảng autopilot mã nguồn mở phổ biến cho UAV, hỗ trợ nhiều loại phương tiện bay khác nhau. PX4 cung cấp các khối điều khiển mức thấp như attitude control, position control và các cơ chế an toàn.

PX4 SITL (Software In The Loop) cho phép chạy firmware PX4 trực tiếp trên máy tính, kết hợp với môi trường mô phỏng để kiểm tra thuật toán mà không cần phần cứng thật.

2.2 Giao thức MAVLink và MAVSDK:

MAVLink là giao thức truyền thông tầng ứng dụng được thiết kế cho các hệ thống UAV. MAVLink hoạt động trên nền UDP hoặc TCP, cho phép truyền lệnh điều khiển và dữ liệu telemetry theo thời gian thực.

MAVSDK là thư viện cấp cao giúp lập trình viên điều khiển drone thông qua MAVLink bằng nhiều ngôn ngữ, trong đó có Python. Trong đề tài này, MAVSDK Python được sử dụng để gửi lệnh điều khiển offboard tới PX4.

2.3 Mô phỏng với Gazebo:

Gazebo là một trình mô phỏng vật lý mã nguồn mở, cho phép mô phỏng môi trường, vật cản và tương tác vật lý. Gazebo kết nối với PX4 thông qua các plugin, cung cấp dữ liệu cảm biến mô phỏng cho autopilot.

2.4 Điều khiển Offboard:

Offboard mode là chế độ cho phép hệ thống bên ngoài gửi setpoint điều khiển trực tiếp cho PX4. Để duy trì chế độ này, setpoint phải được gửi liên tục với tần số đủ cao, nếu không PX4 sẽ kích hoạt failsafe.

3. Kiến trúc hệ thống:

3.1 Các thành phần hệ thống:

- PX4 SITL Autopilot: điều khiển bay, cung cấp interface MAVlink
- Gazebo: mô phỏng vật lý và môi trường.
- MAVSDK Python: điều khiển drone ở tầng mission/offboard.

3.2 Kiến trúc tổng thể:

Hệ thống được xây dựng theo kiến trúc phân tầng. PX4 đảm nhiệm điều khiển mức thấp, trong khi logic nhiệm vụ và tránh vật cản được triển khai ở tầng ứng dụng thông qua MAVSDK Python. Các thành phần giao tiếp với nhau bằng giao thức MAVLink trên nền UDP.



Hình 1: kiến trúc hệ thống

3.3 Kiến trúc phần mềm:

Phần mềm Python được tổ chức thành các module:

- Core: Kết nối và điều khiển cơ bản
- Planner: Sinh quỹ đạo và tránh vật cản
- Mission: Điều phối nhiệm vụ
- Analysis: phân tích dữ liệu log

4. Thiết kế thuật toán:

4.1 Thuật toán lawnmower:

Thuật toán lawnmower tạo ra các waypoint song song nhằm bao phủ toàn bộ khu vực tìm kiếm. Khoảng cách giữa các dải được xác định bởi tham số strip width.

4.2 Phát hiện vật cản:

Khoảng cách giữa drone và vật cản được tính bằng công thức Euclid trong hệ tọa độ NED. Khi khoảng cách nhỏ hơn tổng bán kính vật cản và safety margin, thuật toán tránh được kích hoạt.

4.3 Thuật toán tránh vật cản:

Thuật toán tránh vật cản sử dụng phương pháp Reactive Obstacle Avoidance bằng Waypoint Shifting. Khi phát hiện nguy cơ va chạm, hệ thống tạo một waypoint trung gian lệch sang một hướng an toàn, sau đó quay trở lại quỹ đạo ban đầu.

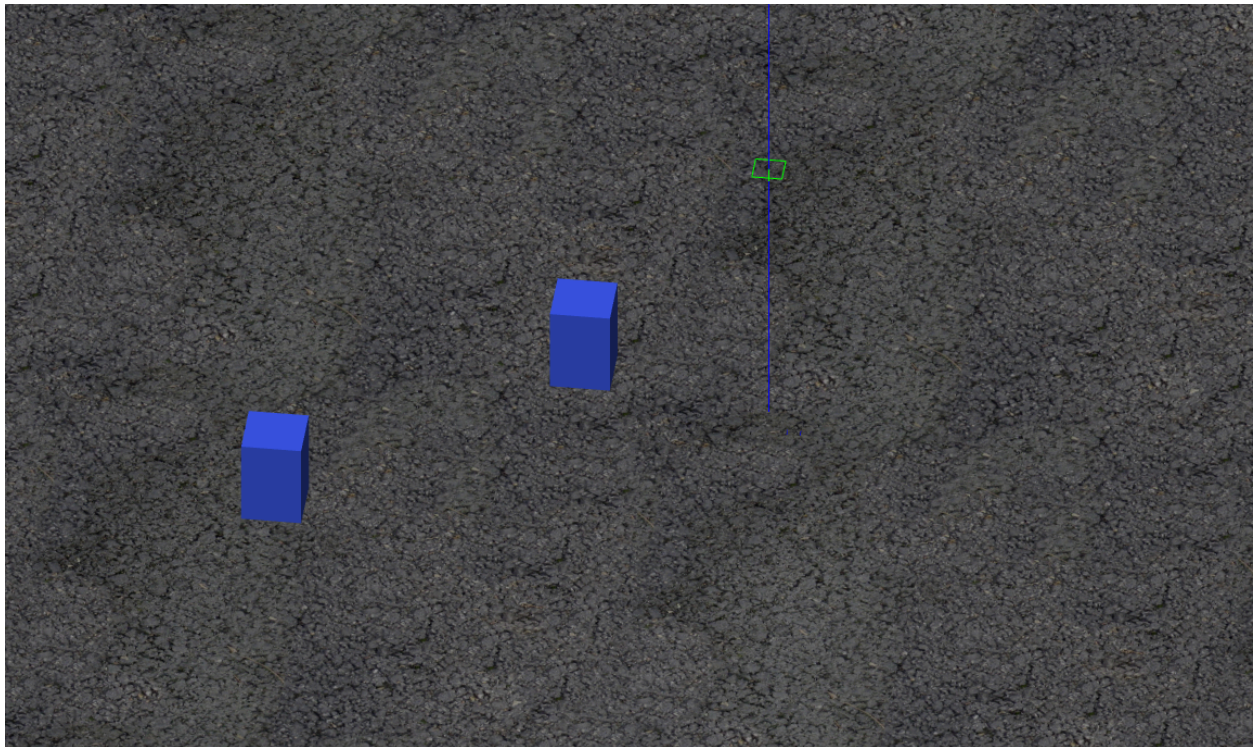
4.4 Ghi log dữ liệu:

Trong quá trình bay, hệ thống ghi lại vị trí drone, thời gian, waypoint và các sự kiện quan trọng vào file CSV. File log được sử dụng để phân tích và đánh giá hiệu quả hệ thống.

5. Thực nghiệm và đánh giá:

5.1 Kịch bản mô phỏng:

Drone thực hiện nhiệm vụ tìm kiếm trong khu vực 20m x 10m, với hai vật cản tĩnh được đặt trong môi trường Gazebo.



Hình 2: Mô phỏng vật cản

```

[control] offboard started
[mission] starting mission with 10 waypoints
[mission] waypoint 1/10 -> (0.00,0.00,-3.00)
[mission] heading to waypoint (0.00,0.00)
[control] goto target (0.00,0.00), dist=0.61 m, time~0.30s, steps=6
[control] goto done
[mission] waypoint 2/10 -> (0.00,10.00,-3.00)
[mission] heading to waypoint (0.00,10.00)
[control] goto target (0.00,10.00), dist=10.36 m, time~5.18s, steps=103
[control] goto done
[mission] waypoint 3/10 -> (5.00,10.00,-3.00)
[mission] heading to waypoint (5.00,10.00)
[control] goto target (5.00,10.00), dist=6.81 m, time~3.40s, steps=68
[control] goto done
[mission] waypoint 4/10 -> (5.00,0.00,-3.00)
[mission] obstacle detected near current pos -> compute avoid point
[mission] goto avoid point (4.15,5.07)
[control] goto target (4.15,5.07), dist=5.07 m, time~3.38s, steps=67
[control] goto done
[mission] heading to waypoint (5.00,0.00)
[control] goto target (5.00,0.00), dist=7.53 m, time~3.77s, steps=75
[control] goto done
[mission] waypoint 5/10 -> (10.00,0.00,-3.00)
[mission] heading to waypoint (10.00,0.00)
[control] goto target (10.00,0.00), dist=5.35 m, time~2.67s, steps=53
[control] goto done
[mission] waypoint 6/10 -> (10.00,10.00,-3.00)
[mission] heading to waypoint (10.00,10.00)
[control] goto target (10.00,10.00), dist=10.10 m, time~5.05s, steps=101
[control] goto done
[mission] waypoint 7/10 -> (15.00,10.00,-3.00)
[mission] heading to waypoint (15.00,10.00)
[control] goto target (15.00,10.00), dist=5.45 m, time~2.72s, steps=54
[control] goto done
[mission] waypoint 8/10 -> (15.00,0.00,-3.00)
[mission] heading to waypoint (15.00,0.00)
[control] goto target (15.00,0.00), dist=9.45 m, time~4.73s, steps=94
[control] goto done
[mission] waypoint 9/10 -> (20.00,0.00,-3.00)
[mission] heading to waypoint (20.00,0.00)
[control] goto target (20.00,0.00), dist=5.30 m, time~2.65s, steps=52
[control] goto done
[mission] waypoint 10/10 -> (20.00,10.00,-3.00)
[mission] heading to waypoint (20.00,10.00)
[control] goto target (20.00,10.00), dist=9.90 m, time~4.95s, steps=99
[control] goto done
[mission] mission finished. avoidance_count = 1
[control] goto target (0.00,0.00), dist=21.78 m, time~10.89s, steps=217
[control] goto done
[control] stopping offboard
[control] landing...

```

Hình 3: log trong quá trình chạy


```

results > mission_log.csv
1 timestamp,north_m,east_m,down_m,event,wp_index,wp_x,wp_y
2 1764255258.9511366,-0.6995624899864197,-0.447716623544693,-0.8703457117080688,pre_waypoint_check,1,0,0
3 1764255259.3840275,-0.6927077770233154,-0.4259328544139862,-1.3698358535766602,reached_waypoint,1,0,0
4 1764255259.9029906,-0.6658682823181152,-0.34986063838005066,-2.0247700214385986,pre_waypoint_check,2,0,10
5 1764255265.2337708,0.23107436299324036,8.199470520019531,-3.1128029823303223,reached_waypoint,2,0,10
6 1764255265.7652311,-0.526481032371521,9.481764793395996,-3.135467052459717,pre_waypoint_check,3,5,10
7 1764255268.6265142,2.692983627319336,10.23218822479248,-3.089263677597046,reached_waypoint,3,5,10
8 1764255269.1490462,3.7734830379486084,10.140068054199219,-3.0963449478149414,pre_waypoint_check,4,5,0
9 1764255272.6979764,4.153652191162109,6.730997562408447,-3.120363712310791,avoided_at_start,4,5,0
10 1764255276.21341,4.721930980682373,2.277956485748291,-3.096872329711914,reached_waypoint,4,5,0
11 1764255276.743729,4.870144844055176,1.2489099502563477,-3.0845437049865723,pre_waypoint_check,5,10,0
12 1764255280.9247541,8.904666900634766,0.3346354365348816,-3.0888400077819824,reached_waypoint,5,10,0
13 1764255281.4569952,9.761548042297363,0.16796322166919708,-3.0803873538970947,pre_waypoint_check,6,10,10
14 1764255286.549248,9.902524948120117,7.973160743713379,-3.156179666519165,reached_waypoint,6,10,10
15 1764255287.0761218,9.942639350891113,8.937688827514648,-3.1763296127319336,pre_waypoint_check,7,15,10
16 1764255289.7367516,12.723775863647461,9.628597259521484,-3.1682910919189453,reached_waypoint,7,15,10
17 1764255290.2793593,13.788260459899902,9.710736274719238,-3.1570820808410645,pre_waypoint_check,8,15,0
18 1764255295.2886145,14.704812049865723,1.990159511566162,-3.021145820617676,reached_waypoint,8,15,0
19 1764255295.8256116,14.850236892700195,1.0063985586166382,-3.0091755390167236,pre_waypoint_check,9,20,0
20 1764255298.52597,17.686283111572266,0.32082897424697876,-2.9929404258728027,reached_waypoint,9,20,0
21 1764255299.0518396,18.71702003479004,0.262441486120224,-2.988818407058716,pre_waypoint_check,10,20,10
22 1764255304.1240964,19.719083786010742,8.033391952514648,-3.0836029052734375,reached_waypoint,10,20,10
23

```

Hình : Hình ảnh file log

5.2 Các metrics đánh giá:

Thông số	Giá trị
Tổng quãng đường bay	38.4m
Tổng thời gian	52.1s
Số lần tránh vật cản	1
Khoảng cách tối thiểu	3.4m
Độ mượt đường bay	19.6

Bảng 2: Giá trị các thông số

5.3 Phân tích kết quả:

Kết quả cho thấy drone hoàn thành nhiệm vụ an toàn, tránh vật cản thành công và duy trì quỹ đạo ổn định. Độ mượt đường bay ở mức chấp nhận được cho hệ thống tránh vật cản phản ứng.

7. Kết luận:

- Drone thực hiện nhiệm vụ lawnmower thành công.
- Thuật toán Reactive Obstacle Avoidance - Waypoint Shifting hoạt động ổn định.
- Mission an toàn: không va chạm, khoảng cách tối thiểu $>$ bán kính obstacle.
- Hệ thống cho phép tái tạo kịch bản nhiều và ghi lại log để phân tích.

8. Hướng phát triển:

- Thêm camera/YOLO để phát hiện người trong khu vực tìm kiếm.
- Dùng global planner (A^* , RTT^*) để tránh vật cản tối ưu.
- Dùng LiDAR thay cho mô phỏng obstacle tĩnh.
- Tích hợp đa drone trong cùng một mission.