

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP
NGÀNH CÔNG NGHỆ THÔNG TIN

**ĐỀ TÀI : LẬP KẾ HOẠCH LỘ TRÌNH BAO
PHỦ CHO THIẾT BỊ BAY KHÔNG NGƯỜI LÁI**

Giảng Viên Hướng Dẫn : TS. Nguyễn Mạnh Cường

Sinh viên thực hiện : Lã Huy Hoàng

Mã sinh viên : 2020602560

Lớp : CNTT - K15

HÀ NỘI – 2024

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP
NGÀNH CÔNG NGHỆ THÔNG TIN

**ĐỀ TÀI : LẬP KẾ HOẠCH LỘ TRÌNH BAO
PHỦ CHO THIẾT BỊ BAY KHÔNG NGƯỜI LÁI**

Giảng Viên Hướng Dẫn : TS. Nguyễn Mạnh Cường

Sinh viên thực hiện : Lã Huy Hoàng

Mã sinh viên : 2020602560

Lớp : CNTT - K15

HÀ NỘI – 2024

Mục lục

LỜI CẢM ƠN	2
LỜI NÓI ĐẦU	1
NỘI DUNG	4
CHƯƠNG 1: KHẢO SÁT VÀ PHÁT BIỂU BÀI TOÁN	4
1.1 Khảo sát các loại Drone trên thị trường hiện tại	4
1.1.1 Khái niệm về drone, drone là gì?	4
1.1.2 Cấu tạo của drone	4
1.1.3 Vật liệu chế tạo drone	7
1.1.4 Điều khiển và điều hướng drone	7
1.1.5 Nguyên lý hoạt động và cơ chế vận hành	8
1.1.6 Ứng dụng của drone sản xuất và đời sống	10
1.2 Một số ứng dụng trí tuệ nhân tạo trong Drone	14
1.2.1. Giới thiệu:	14
1.2.2. Một số ứng dụng trí tuệ nhân tạo trong Drone:	15
1.2.3. Ưu điểm:	16
1.2.4. Hạn chế:	16
1.2.5. Kết luận:	17
1.3 Phát biểu bài toán	17
CHƯƠNG 2: MỘT SỐ PHƯƠNG PHÁP GIẢI QUYẾT BÀI TOÁN	19
2.1 Một số nghiên cứu giải quyết bài toán tiêu biểu	19
2.2 Phương pháp phân rã	20
2.1.1 Phân rã tế bào chính xác (Exact cellular decomposition):	21
2.1.2 Phân rã hình thang (Trapezoidal Decomposition):	22
2.1.3 Phân rã Boustrophedon(Boustrophedon Decomposition):	24
2.1.4 Phân rã dựa trên Morse (Morse-Based Decomposition):	25
2.1.5 Phương pháp dựa trên lưới(Grid-Based Methods):	27
2.1.6 Phạm vi phủ sóng dựa trên mạng nơ-ron trên bản đồ lưới (Neural Network-Based Coverage on Grid Maps):	29
2.3 Phương pháp tối ưu hóa toán học	30
2.4 Mô hình quy hoạch đường đi phủ sóng của UAV tiết kiệm năng lượng	32
CHƯƠNG 3 : THỰC NGHIỆM	36
3.1 Quy trình thực nghiệm	36
3.2 Dữ liệu đầu vào	36

3.3 Mô hình hóa và giải bài toán tối ưu	44
3.2.1 Xây dựng mô hình HCPP	44
3.2.2 Giải mô hình HCPP	45
3.4 Tổng hợp kết quả	48
CHƯƠNG 4 : MÔ PHỎNG BẰNG MICROSOFT AIRSIM	51
4.1. Giới thiệu về các công cụ mô phỏng	51
4.2 Giới thiệu về Microsoft Airsim	51
4.3 Giới thiệu về Px4-autopilot	58
4.4 Giới thiệu về QgroundControl	64
4.5 Phương thức kết nối	66
4.6 Kết quả chạy mô phỏng	67
KẾT LUẬN	71
Tài Liệu Tham Khảo	72

DANH MỤC HÌNH ẢNH

Hình 1.1 Động cơ của Drone

Hình 1.2 Bảng mạch và các bộ vi xử lý trung tâm của Drone

Hình 1.3 Cánh quạt của Drone

Hình 1.4 Khung và giá đỡ của Drone

Hình 1.5 Pin của Drone

Hình 1.6 Bộ điều khiển Drone

Hình 1.7 Hình ảnh Drone cánh cố định

Hình 1.8 Hình ảnh Drone cánh quạt

Hình 1.9 Hình ảnh Drone phục vụ quân sự an ninh

Hình 1.10 Hình ảnh Drone phục vụ hoạt động dự báo thời tiết, khí tượng

Hình 1.11 Hình ảnh Drone phục vụ nhiếp ảnh, làm phim

Hình 1.12 Hình ảnh Drone phục vụ khảo sát địa hình và thiết lập bản đồ

Hình 1.13 Hình ảnh Drone phục vụ sản xuất nông nghiệp

Hình 1.14 Hình ảnh Drone phục vụ giao nhận, logistics

Hình 1.15 Hình ảnh Drone phục vụ tìm kiếm cứu hộ cứu nạn

Hình 2.1 Hình ảnh phương pháp phân rã tế bào chính xác

Hình 2.2 Hình ảnh phương pháp phân rã hình thang

Hình 2.3 Hình ảnh phương pháp phân rã Boustrophedon

Hình 2.4 Hình ảnh phương pháp phân rã dựa trên Morse

Hình 2.5 Hình ảnh phương pháp phân rã dựa trên lưới

Hình 2.6 Hình ảnh phương pháp phạm vi phủ sóng dựa trên mạng nơ-ron trên bản đồ lưới

Hình 3.1 Hình ảnh quy trình thực nghiệm

Hình 3.2 Hình ảnh khu vực bị phân rã và che phủ

Hình 3.3 Hình ảnh biểu diễn khu vực bao phủ và tất cả các điểm nằm trong

Hình 3.4 Hình ảnh biểu diễn các điểm nằm trong khu vực tuần tra

Hình 3.5 Hình ảnh biểu diễn hướng di chuyển từ một điểm

Hình 3.6 Hình ảnh phần mềm CPLEX

Hình 3.7 Hình ảnh đầu vào của bài toán: Input-1

Hình 3.8 Hình ảnh đầu ra sau khi giải bài toán: Output-1

Hình 3.9 Hình ảnh đầu vào của bài toán : Input-2

Hình 3.10 Hình ảnh đầu ra sau khi giải bài toán : Output-2

Hình 4.1 Hình ảnh mô phỏng AirSim

Hình 4.2 Hình ảnh mô phỏng AirSim : môi trường

Hình 4.3 Hình ảnh mô phỏng AirSim

Hình 4.4 Hình ảnh mô phỏng AirSim

Hình 4.5 Hình ảnh mô phỏng AirSim : API

Hình 4.6 Hình ảnh ứng dụng Epic Games Launcher

Hình 4.7 Hình ảnh ứng dụng Epic Games Launcher : Cài đặt Unreal Engine

Hình 4.8 Hình ảnh ứng dụng Visual Code 2022 : Installation Workloads

Hình 4.9 Hình ảnh ứng dụng Visual Code 2022 : Installation Individual components

Hình 4.10 Hình ảnh cài đặt AirSim trên Developer Command Prompt for VS 2022

Hình 4.11 Hình ảnh phần mềm PX4-autopilot

Hình 4.12 Hình ảnh cài đặt Ubuntu

Hình 4.13 Hình ảnh phần mềm QGroundControl

Hình 4.14 Hình ảnh phương thức kết nối giữa các ứng dụng

LỜI CẢM ƠN

Trải qua hành trình đầy ý nghĩa của khóa học, giờ đây khi bản báo cáo tốt nghiệp đã được hoàn thành, em xin gửi lời tri ân sâu sắc đến những người đã đồng hành và tiếp thêm sức mạnh cho em trên con đường này.

Đầu tiên, em xin bày tỏ lòng biết ơn chân thành đến Ban Giám hiệu, cùng toàn thể quý thầy cô giáo Trường Đại học Công Nghiệp Hà Nội, đặc biệt là Khoa Công Nghệ Thông Tin. Nhờ sự dùu dắt và truyền đạt kiến thức của quý thầy cô, em đã có nền tảng vững chắc để tiếp cận và nghiên cứu đề tài tốt nghiệp.

Em xin gửi lời cảm ơn đặc biệt đến Tiến sĩ Nguyễn Mạnh Cường - người thầy hướng dẫn tận tâm, người đã dành cho em những lời khuyên quý báu, những góp ý cẩn thận và sự động viên nhiệt tình trong suốt quá trình thực hiện đề tài. Nhờ sự hướng dẫn của thầy, em đã có thể hoàn thành bài báo cáo một cách tốt nhất.

Hành trình chinh phục tri thức không thể trọn vẹn nếu thiếu đi sự đồng hành của gia đình và bạn bè. Em xin gửi lời cảm ơn chân thành đến bố mẹ, những người đã luôn yêu thương, động viên và tạo điều kiện cho em học tập. Em cũng xin cảm ơn những người bạn thân thiết đã luôn bên cạnh, chia sẻ và hỗ trợ em trong suốt quá trình học tập.

Cuối cùng, em xin gửi lời cảm ơn đến tất cả những ai đã góp phần giúp đỡ em hoàn thành bài báo cáo tốt nghiệp này. Em xin chúc quý thầy cô sức khỏe dồi dào, hạnh phúc và thành công trong sự nghiệp. Em cũng chúc cho gia đình và bạn bè của em luôn bình an và hạnh phúc.

Em xin chân thành cảm ơn !

Sinh viên: Lã Huy Hoàng

LỜI NÓI ĐẦU

Kể từ khi ra đời, thiết bị bay không người lái (Drone) đã chứng minh tiềm năng to lớn trong nhiều lĩnh vực như nông nghiệp, giao hàng, giám sát và an ninh, khảo sát và lập bản đồ, quay phim và chụp ảnh, tìm kiếm cứu nạn. Tuy nhiên, việc ứng dụng Drone còn nhiều hạn chế do hệ thống pháp lý chưa hoàn thiện, cơ sở hạ tầng thiếu hụt, nguồn nhân lực chưa đáp ứng và kinh nghiệm sử dụng chưa đầy đủ.

Sự phát triển vượt bậc của trí tuệ nhân tạo (AI) trong những năm gần đây mở ra tiềm năng to lớn cho việc ứng dụng Drone. AI có thể giúp Drone tự động hóa các nhiệm vụ phức tạp, nâng cao hiệu quả hoạt động và mở rộng phạm vi ứng dụng.

Nội dung quyển báo cáo này sẽ đề xuất nghiên cứu mô hình bài toán lập kế hoạch lộ trình bao phủ cho Drone dưới dạng quy hoạch toán học, đồng thời phân tích các phương pháp giải quyết mô hình và thực hiện thực nghiệm trên môi trường mô phỏng. Mục tiêu của đề tài là thúc đẩy việc sử dụng Drone trong cả lĩnh vực quân sự và dân sự, mang lại lợi ích cho cộng đồng và quốc gia.

Đề tài này có ý nghĩa khoa học và thực tiễn quan trọng. Về mặt khoa học, đề tài này sẽ cung cấp một cái nhìn tổng quan về tiềm năng ứng dụng của Drone trong các lĩnh vực dân sự và quân sự, đồng thời đề xuất một kế hoạch lộ trình cụ thể để triển khai Drone một cách hiệu quả. Về mặt thực tiễn, đề tài này sẽ góp phần thúc đẩy việc ứng dụng Drone, mang lại lợi ích cho cộng đồng.

Với những tiềm năng to lớn và ý nghĩa khoa học, thực tiễn quan trọng, việc nghiên cứu ứng dụng Drone hứa hẹn sẽ mang lại nhiều lợi ích cho cộng đồng và góp phần thúc đẩy sự phát triển của xã hội.

Nội dung quyển báo cáo đề án tốt nghiệp sẽ bao gồm các chương như sau:

Chương 1: Khảo sát và phát biểu bài toán

Trong chương 1, em tiến hành nghiên cứu các báo cáo, bài viết khoa học, sách, v.v. về Drone. Sau đó em có tìm hiểu về các loại Drone, tính năng, ưu điểm, nhược điểm của từng loại. Nghiên cứu về các ứng dụng tiềm năng của Drone. Phân tích các rào cản và thách thức trong việc triển khai Drone. Tham khảo các kế hoạch lộ trình phát triển Drone của các chuyên gia trên thế giới.

Sau phần khảo sát sẽ đến phần trình bày về bài toán được đặt ra là đưa ra được lộ trình bao phủ cho máy bay không người lái.

Chương 2: Mô hình bài toán

Sau khi đã phát biểu và xác định rõ ràng được yêu cầu bài toán, chương 2 sẽ trình bày các kỹ thuật giải quyết bài toán hiện có cùng các ưu và nhược điểm của chúng, cũng như các nghiên cứu nổi bật đã đạt được thành công nhất định từ những kỹ thuật đó. Tiếp đến sẽ đề xuất giải pháp của mình cho bài toán được đặt ra.

Chương 3: Kết quả thực nghiệm

Chương 3 sẽ tập trung trình bày về quá trình thực nghiệm cũng như các kết quả đạt được với kỹ thuật giải quyết bài toán được đề xuất ở chương 2.

Tại đây cũng tiến hành so sánh kết quả thực nghiệm thu được từ phương pháp em đề xuất giải quyết bài với một số phương pháp phổ biến hiện nay và đưa ra nhận xét.

Chương 4: Mô phỏng bằng microsoft Airsim

Cuối cùng là chương 4 sẽ giới thiệu các phần mềm dùng để mô phỏng quá trình chạy thực tế của Drone thông qua điều khiển tự động theo đường đi đã được tính toán.

Phần kết luận:

Cuối cùng, phần kết luận sẽ tóm tắt các kết quả đạt được, đề xuất hướng phát triển và mở rộng để tài nghiên cứu trong tương lai.

Qua quá trình thực hiện đề tài, em đã có được cái nhìn tổng quan về bài toán lập kế hoạch đường đi bao phủ (CPP) nói chung và mô hình MILP (Mixed Integer Linear Programming) nói riêng. Em đã trải nghiệm thực tế quá trình xây dựng bài toán, xử lý dữ liệu đầu vào và thiết lập các ràng buộc cho bài toán để tối ưu hóa hiệu suất.

Đặc biệt, em hy vọng báo cáo này không chỉ là kết quả nghiên cứu cá nhân mà còn là tài liệu hữu ích cho những ai quan tâm đến lĩnh vực học máy và trí tuệ nhân tạo. Bằng cách trình bày chi tiết quy trình nghiên cứu, các vấn đề gặp phải và giải pháp đã thử nghiệm, em mong muốn đề tài này sẽ cung cấp nguồn thông tin đầy đủ và hữu ích cho cộng đồng nghiên cứu.

Nghiên cứu này góp phần mang lại cái nhìn sâu sắc hơn về khả năng ứng dụng của mô hình MILP trong giải bài toán CPP, đồng thời mở ra những thách thức và hướng phát triển tiềm năng cho tương lai. Hy vọng rằng báo cáo

này sẽ là nguồn tài nguyên hữu ích, khơi gợi sự quan tâm và nghiên cứu sâu hơn trong lĩnh vực này.

NỘI DUNG

CHƯƠNG 1: KHẢO SÁT VÀ PHÁT BIỂU BÀI TOÁN

1.1 Khảo sát các loại Drone trên thị trường hiện tại

1.1.1 Khái niệm về drone, drone là gì?

Drone là tên gọi chung chỉ các thiết bị bay được điều khiển từ xa hoặc tự động hóa. Các thiết bị bay không người lái (flycam, Drone) này không có người lái hay điều khiển trực tiếp bên trong mà thường được vận hành nhờ hệ thống, phần mềm được lập trình tự động hoặc có người vận hành, điều khiển từ xa, ở khu vực an toàn hơn.

1.1.2 Cấu tạo của drone

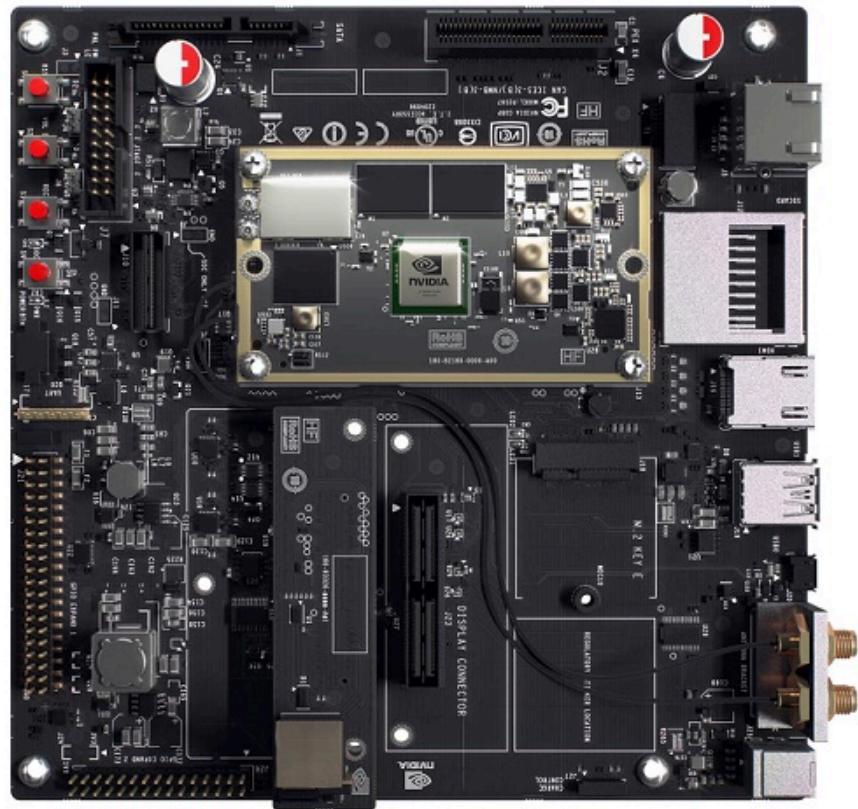
Drone thường được cấu tạo với các thành phần chính, bao gồm:

- Động cơ



Hình 1.1 Động cơ của Drone

- Bảng mạch và các bộ vi xử lý trung tâm



Hình 1.2 Bảng mạch và các bộ vi xử lý trung tâm của Drone

- Các chi tiết khác như cánh quạt (với flycam, Drone)



Hình 1.3 Cánh quạt của Drone

- Các bộ phận để gá thiết bị như khung, giá đỡ, gimbal



Hình 1.4 Khung và giá đỡ của Drone

- Nguồn cấp năng lượng hoạt động cho drone thường là pin công suất lớn.



Hình 1.5 Pin của Drone

1.1.3 Vật liệu chế tạo drone

Vật liệu chế tạo drone thường sử dụng các vật liệu đảm bảo tính chất nhẹ, độ bền cao, nhằm đảm bảo các đặc tính yêu cầu như:

- Tính thông dụng (dễ dàng chế tạo thay thế)
- Độ cơ động (vật liệu nhẹ giúp giảm trọng lượng của drone, đảm bảo thời lượng bay)
- Thiết thực (giá cả phù hợp)
- Hoạt động hiệu quả (thích nghi với nhiều điều kiện môi trường và thời tiết, khí hậu khác nhau).

Một số loại vật liệu chế tạo drone phổ biến hiện nay có thể kể tên như:

- Compozit (composite) dạng sợi thủy tinh (GPRP);
- Compozit (composite) dạng sợi carbon (CFRP);
- Compozit (composite) dạng sợi aramid (AFRP).
- Compozit (composite) sợi gốm liên tục (CFCC);

1.1.4 Điều khiển và điều hướng drone

Người điều khiển, vận hành có thể điều khiển drone ở khoảng cách xa và lập trình drone bay theo lộ trình dựa trên hệ thống định vị GPS hay xác lập bản đồ tự động từ điểm xuất phát (với những loại drone cao cấp có công nghệ tự động lập bản đồ trong quá trình bay tĩnh từ điểm xuất phát).

Bộ điều khiển drone:



Hình 1.6 Bộ điều khiển Drone

Các loại drone và flycam thường được trang bị những bộ điều khiển có tần số sóng 2.4GHz trở lên. Bộ điều khiển thường có ăng-ten (có thể gấp lại), có hai nút xoay 360 độ, các nút chức năng khác (với các dòng flycam và drone cao cấp).

Một số loại drone còn trang bị điều khiển sử dụng công nghệ đường truyền sóng riêng, cho khả năng bay xa và trần bay cao (như các flycam của DJI). Hoặc sử dụng các tín hiệu truyền sóng thông qua tín hiệu Wifi (chỉ sử dụng ở cự ly và phạm vi hoạt động ngắn, thường thấy ở các dòng flycam giá rẻ).

1.1.5 Nguyên lý hoạt động và cơ chế vận hành

Các loại drone thông thường được trang bị cánh quạt, sử dụng nguồn năng lượng cấp từ pin. Cá biệt, những dòng drone Drone (drone kích thước lớn) trang bị động cơ phản lực có thể bay với tốc độ lên tới 800km/h và phạm vi hoạt động lên tới 15km.

Một số loại drone có thể tự động hoạt động và bay theo lộ trình (được người dùng thiết lập) hoặc có thể điều khiển thủ công (trực tiếp do người điều khiển).

Hiện nay có những loại drone nào?

Hiện nay, những loại drone và flycam phổ thông có kích thước và trọng lượng vừa và nhỏ thường trang bị hai cơ chế cánh quạt chủ yếu, gồm:

Drone cánh cố định: loại drone này có phần cánh dài, thường cần một khoảng đường băng dài để lấy đà cất cánh (một số loại kích thước vừa có thể sử dụng máy, bệ phóng di động). Ưu điểm của drone loại này thường có thời gian bay lâu, trần bay và phạm vi hoạt động xa. Cho phép thực hiện các nhiệm vụ đặc biệt, như do thám, khảo sát, trinh sát.



Hình 1.7 Hình ảnh Drone cánh cố định

Drone cánh quạt: loại drone này sử dụng cơ chế cánh quạt tương tự như máy bay trực thăng. Chính vì vậy, chúng có thể cất cánh tại chỗ hoặc những khu vực cất cánh nhỏ. Thường những loại drone này có kích thước vừa và nhỏ nên được sử dụng phục vụ cho các hoạt động quay phim, chụp hình, khảo sát địa hình.



Hình 1.8 Hình ảnh Drone cánh quạt

1.1.6 Ứng dụng của drone sản xuất và đời sống

- Quân sự, an ninh



Hình 1.9 Hình ảnh Drone phục vụ quân sự an ninh

Ban đầu, những mẫu drone kích thước lớn (Drone) xuất hiện phổ biến trong các hoạt động quân sự với vai trò như một thiết bị trinh sát, do thám địa hình hoặc có thể tiêu diệt mục tiêu với độ chính xác cao (nếu được vũ trang).

- Các hoạt động dự báo thời tiết, khí tượng



Hình 1.10 Hình ảnh Drone phục vụ hoạt động dự báo thời tiết, khí tượng

Sau này, nhận thấy tiềm năng của drone, các lĩnh vực khác cũng lần lượt thử nghiệm và áp dụng drone. Chẳng hạn, trong hoạt động dự báo thời

tiết (thu thập các thông tin, hình ảnh từ độ cao), nhằm giúp chuyên gia phân tích hình thái thời tiết để đưa ra những dự báo.

- Drone phục vụ dân sự và doanh nghiệp

Nhờ hoạt động nghiên cứu và phát triển, các loại drone kích thước vừa và nhỏ được trang bị các loại camera khác nhau phục vụ các nhu cầu dân sự cũng từ đó lần lượt ra đời và phổ biến ngày nay trong nhiều lĩnh vực của cuộc sống, như:

- Nhiếp ảnh, làm phim (không ảnh)



Hình 1.11 Hình ảnh Drone phục vụ nhiếp ảnh, làm phim

Các loại drone phục vụ nhiếp ảnh trên không (không ảnh) được trang bị camera được gọi là flycam. Camera của drone có độ phân giải và chất lượng hình ảnh sắc nét cao (những loại drone cao cấp có thể được trang bị camera quay phim 4K), cho phép thực hiện quay phim và chụp hình từ phạm vi và trần bay ở tầm trung (dưới 1000m). Tùy theo giá thành và mục đích sử dụng, các loại flycam này còn được chia ra thành drone phục vụ nhiếp ảnh nghiệp dư hay nhiếp ảnh chuyên nghiệp.

- Khảo sát địa hình và thiết lập bản đồ



Hình 1.12 Hình ảnh Drone phục vụ khảo sát địa hình và thiết lập bản đồ

Những loại drone và flycam phục vụ hoạt động khảo sát, thiết lập bản đồ thực địa thường trang bị module LIDAR hay RTK hỗ trợ quá trình quét và chụp ảnh, thiết lập bản đồ địa hình. Đặc biệt là bản đồ 3D mô phỏng.

- Sản xuất nông nghiệp



Hình 1.13 Hình ảnh Drone phục vụ sản xuất nông nghiệp

Các loại drone phục vụ nông nghiệp hiện nay chủ yếu thực hiện các nhiệm vụ như: rải phân bón, phun thuốc trừ sâu, và gieo hạt giống. Cá biệt một số dòng drone cao cấp trang bị camera đa quang phổ giúp thu thập, phân tích dữ liệu biến đổi của mặt đất, cây trồng, khu vực canh tác. Có thể kể đến như Mavic 3M hay Phantom 4M.

- Công nghiệp

Hiện nay, một số loại drone phục vụ công tác kiểm tra, giám sát các hoạt động sản xuất, chẳng hạn như: Drone giám sát đường dây truyền tải điện, phát hiện những điểm bất thường trên đường dây truyền tải điện nhờ trang bị camera ảnh nhiệt, xử lý các vật thể lạ tiếp cận hay vướng vào đường dây truyền tải điện.

- Một số lĩnh vực khác như dầu khí, cũng sử dụng những ứng dụng của drone tương tự.
- Giao nhận, logistics



Hình 1.14 Hình ảnh Drone phục vụ giao nhận, logistics

Hiện nay, một số công ty và tập đoàn hoạt động trong lĩnh vực bán lẻ, logistics và vận tải ở khắp nơi trên thế giới đã và đang thử nghiệm, sử dụng drone như là một phương tiện giao hàng tự động trong đời sống, đặc biệt là trong và sau đại dịch Covid-19 diễn ra. Tầm quan trọng của drone trong việc duy trì chuỗi cung ứng ngày càng được nhìn nhận một cách nghiêm túc và xúc tiến hơn.

- Tìm kiếm cứu nạn



Hình 1.15 Hình ảnh Drone phục vụ tìm kiếm cứu hộ cứu nạn

Các loại drone phục vụ công tác tìm kiếm cứu nạn cũng có thể trang bị camera ảnh nhiệt, giúp tìm kiếm người gặp nạn (theo tín hiệu nhiệt độ phát ra từ cơ thể nạn nhân). Xác định nguồn gây ra các vụ cháy (dựa theo chênh lệch về vùng nhiệt độ) để có phương án dập tắt nguồn gây cháy, giảm thiệt hại ở mức tối đa.

1.2 Một số ứng dụng trí tuệ nhân tạo trong Drone

1.2.1. Giới thiệu:

Drone hay còn gọi là phương tiện bay không người lái, đang ngày càng phổ biến và được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau. Nhờ khả năng bay linh hoạt, thu thập dữ liệu chính xác và hiệu quả, drone mang đến nhiều tiềm năng to lớn cho các ngành công nghiệp và hoạt động của con người.

Trí tuệ nhân tạo (AI) đang ngày càng khẳng định vị trí của mình trong nhiều lĩnh vực, bao gồm cả lĩnh vực robot. Một trong những ứng dụng quan trọng của AI trong drone là giải quyết bài toán lập kế hoạch lộ trình bao phủ cho thiết bị bay không người lái (CPP). CPP là bài toán tìm kiếm đường đi tối ưu cho robot để đảm bảo nó có thể bao phủ toàn bộ khu vực làm việc. Bài toán này có nhiều ứng dụng thực tế như robot dọn dẹp nhà cửa, robot thu thập dữ liệu, robot kiểm tra an ninh, v.v.

1.2.2. Một số ứng dụng trí tuệ nhân tạo trong Drone:

Dưới đây là một số ứng dụng tiềm năng của drone:

- Nông nghiệp:

Giám sát và đánh giá mùa vụ: Drone có thể được sử dụng để chụp ảnh và video từ trên cao, giúp theo dõi tình trạng phát triển của cây trồng, phát hiện sâu bệnh và đưa ra biện pháp phòng trừ kịp thời.

Phun thuốc bảo vệ thực vật: Drone phun thuốc có thể giúp tiết kiệm lượng thuốc sử dụng, tăng hiệu quả phun thuốc và giảm thiểu nguy cơ ảnh hưởng đến sức khỏe con người và môi trường.

Bón phân và gieo hạt: Drone có thể được sử dụng để bón phân và gieo hạt một cách chính xác và hiệu quả, giúp tăng năng suất cây trồng.

- Giao hàng:

Giao hàng hóa: Drone có thể được sử dụng để giao hàng hóa một cách nhanh chóng và hiệu quả, đặc biệt là ở những khu vực khó tiếp cận.

Giao hàng y tế: Drone có thể được sử dụng để vận chuyển thuốc men và vật tư y tế đến những khu vực hẻo lánh hoặc trong trường hợp khẩn cấp.

- Giám sát và an ninh:

Giám sát an ninh: Drone có thể được sử dụng để giám sát an ninh khu vực, phát hiện hành vi phạm pháp và hỗ trợ lực lượng chức năng trong việc truy bắt tội phạm.

Giám sát giao thông: Drone có thể được sử dụng để giám sát giao thông, phát hiện tắc nghẽn và hỗ trợ điều tiết giao thông.

Giám sát môi trường: Drone có thể được sử dụng để giám sát môi trường, phát hiện ô nhiễm và theo dõi biến đổi khí hậu.

- Khảo sát và lập bản đồ:

Khảo sát địa hình: Drone có thể được sử dụng để khảo sát địa hình một cách nhanh chóng và chính xác, giúp tạo ra các bản đồ chi tiết phục vụ cho nhiều mục đích khác nhau.

Lập bản đồ 3D: Drone có thể được sử dụng để chụp ảnh và video từ nhiều góc độ khác nhau, giúp tạo ra các bản đồ 3D sống động và trực quan.

- Quay phim và chụp ảnh:

Chụp ảnh và quay phim từ trên cao: Drone có thể chụp ảnh và quay phim từ trên cao, giúp tạo ra những góc nhìn độc đáo và ấn tượng.

Quay phim quảng cáo: Drone có thể được sử dụng để quay phim quảng cáo cho các doanh nghiệp, giúp quảng bá sản phẩm và dịch vụ một cách hiệu quả.

- Tìm kiếm cứu nạn:

Tìm kiếm người mất tích: Drone có thể được sử dụng để tìm kiếm người mất tích trong khu vực rộng lớn, đặc biệt là ở những khu vực khó tiếp cận.

Cứu hộ thiên tai: Drone có thể được sử dụng để hỗ trợ cứu hộ trong trường hợp thiên tai, giúp cung cấp thông tin về tình hình thiệt hại và hỗ trợ cứu người.

- Ngoài những ứng dụng trên, drone còn có thể được ứng dụng trong nhiều lĩnh vực khác như:

Kiểm tra và bảo trì: Drone có thể được sử dụng để kiểm tra và bảo trì các công trình, đường dây điện, cầu cống,...

Nghiên cứu khoa học: Drone có thể được sử dụng để thu thập dữ liệu khoa học trong các lĩnh vực như khí tượng, hải dương học, địa chất,...

Giải trí: Drone có thể được sử dụng cho các mục đích giải trí như chơi game, đua drone,...

1.2.3. Ưu điểm:

- Tính linh hoạt: Drone có thể bay đến những khu vực khó tiếp cận bằng con người hoặc các phương tiện khác.
- Hiệu quả: Drone có thể thu thập dữ liệu nhanh chóng và chính xác hơn so với các phương pháp truyền thống.
- Tiết kiệm chi phí: Drone có thể giúp giảm chi phí cho các hoạt động như khảo sát, giám sát, giao hàng.
- An toàn: Drone có thể giảm thiểu nguy cơ cho con người trong các hoạt động nguy hiểm như kiểm tra đường dây điện hoặc cứu hộ thiên tai.
- Dữ liệu trực quan: Drone có thể cung cấp dữ liệu trực quan dưới dạng hình ảnh và video, giúp dễ dàng phân tích và đánh giá.

1.2.4. Hạn chế:

- Giá thành: Drone có thể khá đắt đỏ, đặc biệt là các loại drone chuyên dụng.

- Thời gian bay: Drone có thời gian bay hạn chế, thường chỉ từ 15 đến 30 phút mỗi lần sạc.
- Tầm hoạt động: Tầm hoạt động của drone bị giới hạn bởi phạm vi truyền sóng.
- Điều kiện thời tiết: Drone có thể bị ảnh hưởng bởi điều kiện thời tiết như gió mạnh, mưa lớn.
- Vấn đề an ninh: Việc sử dụng drone có thể tiềm ẩn nguy cơ an ninh như xâm phạm quyền riêng tư, tấn công mạng.
- Quy định pháp lý: Việc sử dụng drone cần tuân thủ các quy định pháp luật, có thể khác nhau ở mỗi quốc gia.

1.2.5. Kết luận:

Drone là một công nghệ mới với tiềm năng ứng dụng vô cùng rộng rãi. Với sự phát triển nhanh chóng của công nghệ, drone sẽ ngày càng được sử dụng nhiều hơn nữa trong các lĩnh vực khác nhau của đời sống.

1.3 Phát biểu bài toán

Ngày nay, Drone được ứng dụng rộng rãi trong các nhiệm vụ trinh sát, do thám, giám sát, cảnh báo. Tuy nhiên, việc tối ưu hóa hiệu quả hoạt động của Drone vẫn còn nhiều thách thức. Bài toán lập kế hoạch lộ trình bao phủ tối ưu cho Drone đóng vai trò quan trọng trong việc nâng cao hiệu quả, tiết kiệm thời gian, chi phí và đảm bảo an toàn cho các nhiệm vụ bay.

Bài toán lập kế hoạch lộ trình bao phủ tối ưu cho Drone là một vấn đề quan trọng trong lĩnh vực điều khiển tự động và ứng dụng của drone. Bài toán này liên quan đến việc lập kế hoạch cho drone di chuyển qua một khu vực cụ thể để thu thập dữ liệu hoặc thực hiện các nhiệm vụ như giám sát, kiểm tra, hoặc giao hàng. Mục tiêu là tối ưu hóa lộ trình di chuyển của drone để đảm bảo rằng nó bao phủ một phạm vi lớn nhất có thể hoặc thực hiện các nhiệm vụ cụ thể một cách hiệu quả nhất, trong khi đồng thời tuân thủ các ràng buộc như hạn chế thời gian hoặc năng lượng tiêu thụ.

Đối với bài toán lập kế hoạch lộ trình bao phủ tối ưu cho drone, các đầu vào chính bao gồm:

- Thông tin về khu vực cần bao phủ: Đây là thông tin về hình dạng, kích thước, và các ràng buộc về vị trí của khu vực cần drone bao phủ. Thông tin này có thể được cung cấp dưới dạng tọa độ GPS, bản đồ, hoặc mô tả chi tiết về địa hình.
- Vị trí và số lượng các điểm quan trọng cần bao phủ: Đây là thông tin về các điểm cần drone bao phủ, như vị trí của các đối tượng cần giám sát, kiểm tra, hoặc các khu vực quan trọng cần được giám sát.

- Thông tin về drone: Bao gồm các thông số về drone như tốc độ tối đa, thời gian hoạt động, sức chứa năng lượng, và các ràng buộc khác liên quan đến khả năng hoạt động của drone.
- Ràng buộc và yêu cầu: Bao gồm các ràng buộc như thời gian cần thiết để hoàn thành nhiệm vụ, yêu cầu về mức độ bao phủ của khu vực, và bất kỳ hạn chế nào về tài nguyên như năng lượng hoặc thời gian.

Nhiệm vụ của bài toán lập kế hoạch lộ trình bao phủ tối ưu cho drone là thiết kế một lộ trình di chuyển cho drone sao cho nó có thể bao phủ một khu vực cụ thể một cách hiệu quả nhất, đồng thời tuân thủ các ràng buộc và yêu cầu được đưa ra. Cụ thể, nhiệm vụ này bao gồm:

- Tối ưu hóa Bao phủ: Tạo ra một lộ trình di chuyển cho drone sao cho nó có thể bao phủ tất cả các điểm quan trọng hoặc khu vực được chỉ định một cách hiệu quả nhất. Mục tiêu là đảm bảo rằng không có phần nào của khu vực cần bao phủ bị bỏ sót và drone tiêu tốn ít năng lượng hoặc thời gian nhất có thể.
- Tối ưu hóa Ràng buộc: Đảm bảo rằng lộ trình được thiết kế tuân thủ các ràng buộc và yêu cầu được đặt ra, như hạn chế thời gian, tốc độ tối đa của drone, hoặc năng lượng tiêu thụ.
- Tối ưu hóa Hiệu suất: Tạo ra một lộ trình di chuyển cho drone sao cho nó hoàn thành nhiệm vụ một cách hiệu quả nhất, có thể làm được nhiều nhiệm vụ trong thời gian ngắn nhất hoặc tiêu tốn ít năng lượng nhất có thể.

Dựa trên các thông tin này, thuật toán sẽ tạo ra một lộ trình tối ưu cho drone, đảm bảo rằng nó có thể bao phủ một cách hiệu quả nhất các điểm quan trọng trong khu vực đã chỉ định và tuân thủ các ràng buộc và yêu cầu được đưa ra.

CHƯƠNG 2: MỘT SỐ PHƯƠNG PHÁP GIẢI QUYẾT BÀI TOÁN

Chương 2 này sẽ trình bày một số nghiên cứu giải quyết bài toán tiêu biểu. Tiếp đến là sẽ trình bày các phương pháp để giải quyết bài toán “Lập kế hoạch đường đi bao phủ cho Drone” hiện có trên thế giới như Phương pháp phân rã và Phương pháp tối ưu toán học. Cuối cùng sẽ trình bày mô hình bài toán sẽ được áp dụng.

2.1 Một số nghiên cứu giải quyết bài toán tiêu biểu

Việc giải quyết bài toán CPP truyền thống thường dựa trên các phương pháp toán học như lập kế hoạch di chuyển, tìm kiếm đường đi ngắn nhất, v.v. Tuy nhiên, những phương pháp này thường gặp khó khăn khi môi trường làm việc phức tạp, có nhiều chướng ngại vật hoặc thay đổi liên tục.

AI mang đến những giải pháp mới cho bài toán CPP. Nhờ khả năng học tập và thích ứng, các thuật toán AI có thể giúp robot di chuyển hiệu quả hơn trong môi trường phức tạp, đảm bảo bao phủ toàn bộ khu vực làm việc và tiết kiệm thời gian di chuyển.

Dưới đây là các nghiên cứu tiêu biểu đã nghiên cứu và giải quyết bài toán Lập kế hoạch lô trình bao phủ :

- "A Survey of Coverage Path Planning Algorithms for Unmanned Aerial Vehicles" (2016): Nghiên cứu này tổng hợp và phân tích các thuật toán lập kế hoạch lô trình bao phủ cho drone. Nó giới thiệu các phương pháp như phân chia ô, phân tích hình học và tối ưu hóa đồng thời. Kết quả của nghiên cứu là việc cung cấp một cái nhìn tổng quan về các thuật toán hiện có và ưu nhược điểm của chúng.
- "Coverage Path Planning for Unmanned Aerial Vehicles in Windy Environments" (2017): Nghiên cứu này tập trung vào việc giải quyết bài toán lập kế hoạch lô trình bao phủ cho drone trong môi trường có gió. Phương pháp được đề xuất sử dụng mô hình động học để ước lượng tốc độ và hướng gió, từ đó tối ưu hóa lô trình bao phủ. Kết quả cho thấy phương pháp này giúp cải thiện hiệu suất bao phủ trong môi trường gió.
- "Efficient Coverage Path Planning for Unmanned Aerial Vehicles using Voronoi Diagrams" (2018): Nghiên cứu này đề xuất sử dụng biểu đồ Voronoi để tạo lập lô trình bao phủ hiệu quả cho drone. Phương pháp này chia khu vực cần bao phủ thành các vùng con, sau đó tính toán đường đi dọc theo biên của các vùng này. Kết quả cho thấy phương pháp Voronoi giúp giảm đáng kể độ phức tạp tính toán và cải thiện hiệu quả bao phủ.

- "Optimal Coverage Path Planning for Unmanned Aerial Vehicles using Genetic Algorithm" (2019): Nghiên cứu này áp dụng thuật toán di truyền để tìm ra lộ trình bao phủ tối ưu cho drone. Phương pháp này sử dụng quần thể các lộ trình tiềm năng, sau đó tối ưu hóa chúng thông qua quá trình tiến hóa. Kết quả cho thấy phương pháp di truyền giúp tìm ra lộ trình tối ưu và cải thiện hiệu suất bao phủ.
- "Coverage Path Planning for Multiple Unmanned Aerial Vehicles in Dynamic Environments" (2020): Nghiên cứu này tập trung vào việc lập kế hoạch lộ trình bao phủ cho nhiều drone trong môi trường động. Phương pháp được đề xuất sử dụng kỹ thuật phân bổ và phối hợp tác động để tạo lập lộ trình cho mỗi drone sao cho chúng không xung đột và bao phủ đầy đủ khu vực. Kết quả cho thấy phương pháp này giúp tối ưu hóa việc phân chia nhiệm vụ và cải thiện hiệu suất bao phủ.

Các phần dưới đây sẽ trình bày 2 phương pháp chính là phương pháp phân rã và phương pháp tối ưu toán học.

2.2 Phương pháp phân rã

Phương pháp này tập trung vào việc phân chia bài toán thành các phần nhỏ hơn và giải quyết chúng một cách độc lập trước khi kết hợp kết quả lại với nhau. Trong bài toán này, khu vực cần bao phủ có thể được chia thành các phần nhỏ hơn, và sau đó mỗi phần được xử lý riêng biệt bằng cách sử dụng các thuật toán hoặc chiến lược cụ thể. Các kết quả từ các phần này sau đó được kết hợp để tạo ra một lộ trình tổng thể cho drone.

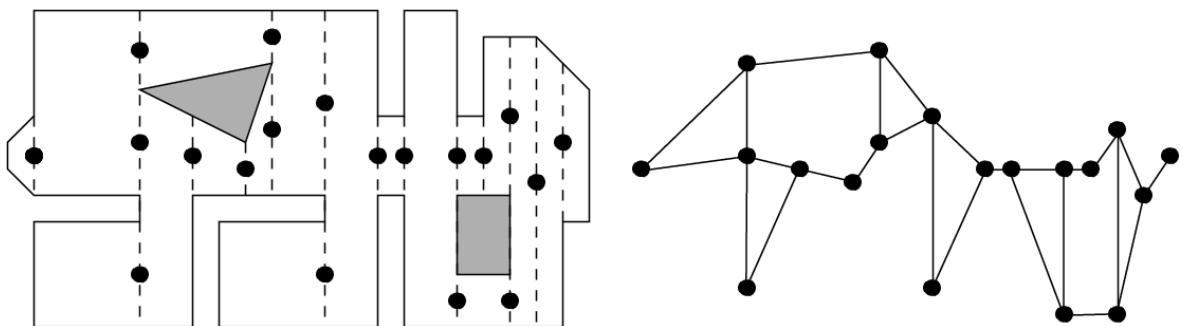
Đối với nhiệm vụ tuần tra, chúng ta có thể xem xét bài toán CPP trong môi trường 2D. Thông thường hình dạng của khu vực tuần tra có thể lồi hoặc không lồi. Đối với những vùng có hình dạng thông thường thì không cần phân rã. Các mẫu có quy hoạch đường dẫn đơn giản, chẳng hạn như boustrophedon hoặc hình vuông là đủ để bao phủ toàn bộ khu vực không phức tạp mà không bị chồng chéo.

Do khu vực tuần tra trong thực tế thường có hình dạng phức tạp nên hầu hết các thuật toán CPP đều phân tách khu vực quan tâm trong các ô. Điều này có thể cho phép các thuật toán thực thi tốt trên cả vùng lồi và không lồi.

Dưới đây là một số phương pháp phân rã phổ biến và hay được sử dụng

:

2.1.1 Phân rã tế bào chính xác (Exact cellular decomposition):



Hình 2.1 Hình ảnh phương pháp phân rã tế bào chính xác

Phương pháp Exact Cellular Decomposition là một kỹ thuật phân tích không gian để giải quyết bài toán lập kế hoạch đường đi bao phủ cho drone. Kỹ thuật này chia không gian 2D hoặc 3D thành các ô (cell) nhỏ, mỗi ô đại diện cho một phần của không gian. Sau đó, các ô này được sắp xếp vào các lớp (layers) dựa trên mức độ quan trọng ưu tiên.

Sau khi thực hiện phương pháp Exact Cellular Decomposition, quá trình giải quyết bài toán sẽ diễn ra như sau:

- Phân tích không gian: Đầu tiên, không gian cần được phân tích thành các ô nhỏ hơn, có thể có kích thước đồng nhất hoặc không đồng nhất, tùy thuộc vào yêu cầu cụ thể của bài toán.
- Xác định ưu tiên của các ô: Các ô có thể được xác định ưu tiên dựa trên các yếu tố như độ quan trọng của khu vực cần bao phủ, hoặc cần thiết đặt ưu tiên cho việc bao phủ các khu vực quan trọng hơn trước.
- Tạo lớp và sắp xếp các ô: Các ô được sắp xếp vào các lớp dựa trên ưu tiên đã xác định, sao cho các ô quan trọng hơn được đặt ở các lớp ưu tiên cao hơn.
- Thực hiện lập kế hoạch đường đi: Sau khi không gian đã được phân tích và các ô đã được sắp xếp vào các lớp, drone sẽ di chuyển qua các ô theo lộ trình đã được xác định trước. Quá trình di chuyển này có thể được tối ưu hóa để đảm bảo rằng drone bao phủ tất cả các ô quan trọng một cách hiệu quả nhất, tuân thủ các ràng buộc và yêu cầu của bài toán.

Bằng cách này, phương pháp Exact Cellular Decomposition cung cấp một cách tiếp cận hợp lý để giải quyết bài toán lập kế hoạch đường đi bao phủ cho drone bằng cách chia không gian thành các ô và quản lý chúng theo mức độ ưu tiên.

Có hai loại ECD chính:

- ECD dựa trên lưới (Grid-based ECD): Môi trường được chia thành các ô vuông hoặc hình chữ nhật có kích thước bằng nhau.

- ECD dựa trên Voronoi (Voronoi-based ECD): Môi trường được chia thành các tế bào dựa trên các điểm Voronoi.

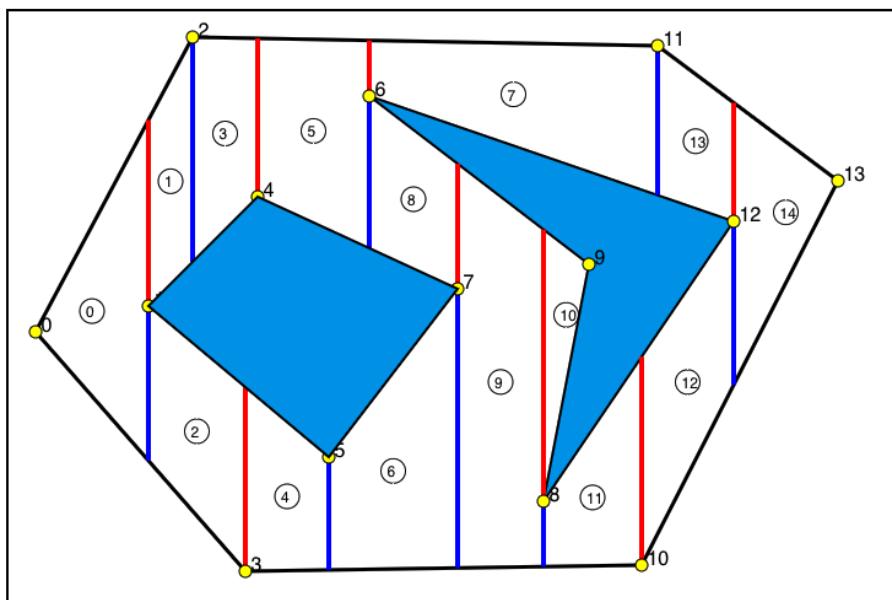
Ưu điểm của ECD:

- Giảm số lượng trạng thái cần xem xét: ECD giúp đơn giản hóa bài toán lập kế hoạch đường đi bằng cách chia nhỏ môi trường thành các tế bào nhỏ.
- Tăng hiệu quả: ECD có thể giúp drone di chuyển nhanh hơn và hiệu quả hơn bằng cách cho phép nó tập trung vào các tế bào cụ thể cần được phủ.
- Dễ dàng triển khai: ECD là một kỹ thuật tương đối đơn giản và dễ dàng triển khai.

Nhược điểm của ECD:

- Độ chính xác: Độ chính xác của ECD phụ thuộc vào kích thước của các tế bào. Các tế bào nhỏ hơn sẽ dẫn đến độ chính xác cao hơn, nhưng cũng sẽ làm tăng số lượng trạng thái cần xem xét.
- Tính toán: ECD có thể đòi hỏi nhiều tính toán, đặc biệt là đối với các môi trường lớn.

2.1.2 Phân rã hình thang (Trapezoidal Decomposition):



Hình 2.2 Hình ảnh phương pháp phân rã hình thang

Phương pháp Trapezoidal Decomposition là một kỹ thuật được sử dụng để phân tích và lập kế hoạch đường đi bao phủ cho drone trong không gian 2D. Đầu tiên, không gian được chia thành các hình thang nhỏ hơn, thường được hình thành bởi các đoạn thẳng song song hoặc gần song song với nhau.

Các điểm quan trọng trong không gian, như các điểm cần kiểm tra hoặc giám sát, được xác định và sử dụng để tạo ra lộ trình cho drone.

Quá trình giải quyết bài toán sẽ diễn ra như sau:

- Phân tích không gian thành các hình thang: Không gian cần được phân tích thành các hình thang nhỏ hơn. Điều này thường được thực hiện bằng cách sử dụng các thuật toán hình học như thuật toán Trapezoidal Decomposition.
- Xác định các điểm quan trọng: Các điểm quan trọng trong không gian được xác định, ví dụ như các điểm cần kiểm tra, giám sát, hoặc các vị trí quan trọng khác.
- Tạo lộ trình di chuyển cho drone: Dựa trên các hình thang đã được phân tích và các điểm quan trọng, một lộ trình di chuyển được tạo ra cho drone. Lộ trình này sẽ bao gồm các đoạn đường đi qua các hình thang để đảm bảo rằng drone di chuyển qua mỗi hình thang và bao phủ tất cả các điểm quan trọng trong không gian.
- Tối ưu hóa lộ trình: Cuối cùng, lộ trình di chuyển có thể được tối ưu hóa để đáp ứng các yêu cầu cụ thể của bài toán, như tối thiểu hóa thời gian hoặc năng lượng tiêu thụ. Các thuật toán tối ưu hóa có thể được áp dụng để điều chỉnh lộ trình để đạt được mục tiêu này.

Phương pháp Trapezoidal Decomposition cung cấp một cách tiếp cận hiệu quả để giải quyết bài toán lập kế hoạch đường đi bao phủ cho drone trong không gian 2D, bằng cách sử dụng các hình thang để phân rã không gian và tạo ra lộ trình di chuyển cho drone.

Ưu điểm của TD:

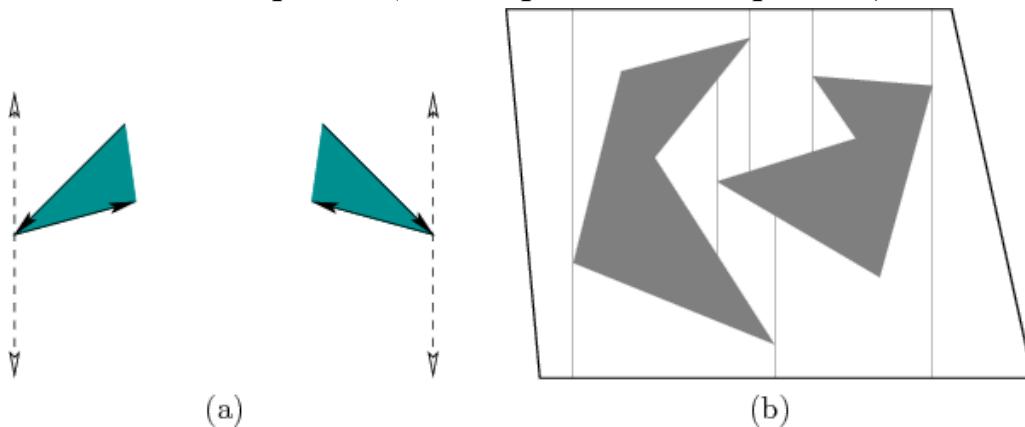
- Giảm số lượng trạng thái cần xem xét: TD giúp đơn giản hóa bài toán lập kế hoạch đường đi bằng cách chia nhỏ môi trường thành các hình thang nhỏ.
- Tăng hiệu quả: TD có thể giúp Drone di chuyển nhanh hơn và hiệu quả hơn bằng cách cho phép nó tập trung vào các khu vực cụ thể cần được phủ.
- Dễ dàng triển khai: TD là một kỹ thuật tương đối đơn giản và dễ dàng triển khai.

Nhược điểm của TD:

- Độ chính xác: Độ chính xác của TD phụ thuộc vào kích thước của các hình thang. Các hình thang nhỏ hơn sẽ dẫn đến độ chính xác cao hơn, nhưng cũng sẽ làm tăng số lượng trạng thái cần xem xét.

- Tính toán: TD có thể đòi hỏi nhiều tính toán, đặc biệt là đối với các môi trường lớn.

2.1.3 Phân rã Boustrophedon(Boustrophedon Decomposition):



Hình 2.3 Hình ảnh phương pháp phân rã Boustrophedon

Phương pháp Boustrophedon Decomposition là một kỹ thuật phân rã không gian được sử dụng để giải quyết bài toán lập kế hoạch đường đi bao phủ cho drone hoặc robot di chuyển trong không gian 2D. Tên gọi "boustrophedon" xuất phát từ tiếng Hy Lạp cổ điển, có nghĩa là "di chuyển như bò cày".

Dưới đây là phác thảo chi tiết về cách phương pháp Boustrophedon Decomposition hoạt động:

- Phân tích không gian thành các vùng hình chữ nhật: Không gian cần được phân tích thành các vùng hình chữ nhật nhỏ hơn, có thể có kích thước đồng nhất hoặc không đồng nhất, tùy thuộc vào đặc điểm của không gian và yêu cầu của bài toán.
- Xác định các vùng bắt đầu và kết thúc của quãng đường: Từ các vùng hình chữ nhật đã được phân tích, các vùng bắt đầu và kết thúc của quãng đường bao phủ được xác định. Các vùng này thường được chọn sao cho drone hoặc robot có thể di chuyển một cách hiệu quả từ vùng bắt đầu đến vùng kết thúc mà không cần phải quay đầu.
- Lập kế hoạch di chuyển boustrophedon: Drone hoặc robot di chuyển qua các vùng bắt đầu và kết thúc theo một cách lần lượt, giống như cách mà một con bò cày đi qua một cánh đồng. Trong mỗi vùng, chúng di chuyển theo đường zigzag, chuyển hướng 180 độ sau mỗi hàng hoặc cột.
- Đảm bảo bao phủ đầy đủ của không gian: Quá trình lập kế hoạch đường đi bao phủ boustrophedon đảm bảo rằng toàn bộ không gian

được bao phủ một cách đầy đủ và hiệu quả. Mỗi điểm trong không gian 2D được bao phủ ít nhất một lần bởi đường đi của drone hoặc robot.

Phương pháp Boustrophedon Decomposition cung cấp một cách tiếp cận đơn giản và hiệu quả để giải quyết bài toán lập kế hoạch đường đi bao phủ cho drone hoặc robot trong không gian 2D. Đặc biệt, nó phù hợp cho các ứng dụng như kiểm tra, giám sát, hoặc làm sạch tự động trong môi trường bờ mặt phẳng.

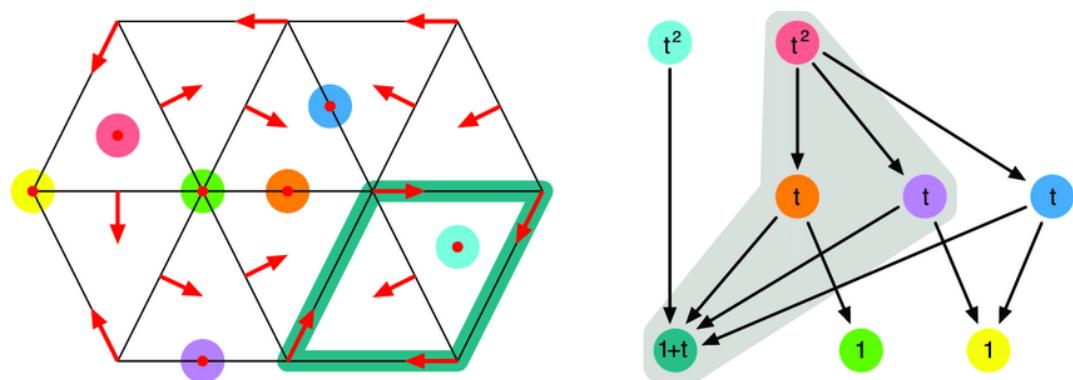
Ưu điểm của BD:

- Đơn giản: BD là một kỹ thuật đơn giản và dễ dàng triển khai.
- Hiệu quả: BD có thể giúp Drone Di chuyển hiệu quả bằng cách giảm thiểu số lần quay đầu.
- Độ chính xác: BD có thể đạt được độ chính xác cao nếu kích thước của các dải được chọn phù hợp.

Nhược điểm của BD:

- Có thể không phù hợp với tất cả các môi trường: BD có thể không hiệu quả đối với các môi trường có nhiều chướng ngại vật.
- Có thể tạo tách các đường đi không tối ưu: BD có thể tạo tách các đường đi dài hơn so với các kỹ thuật khác.

2.1.4 Phân rã dựa trên Morse (Morse-Based Decomposition):



Hình 2.4 Hình ảnh phương pháp phân rã dựa trên Morse

Phương pháp Morse-Based Decomposition là một phương pháp tiếp cận phức tạp được sử dụng để giải quyết bài toán lập kế hoạch đường đi bao phủ cho drone. Phương pháp này dựa trên lý thuyết đồ thị và lý thuyết độ cao Morse trong đại số đa điểm.

- Dưới đây là một phác thảo chi tiết về cách phương pháp Morse-Based Decomposition hoạt động:

- Xây dựng đồ thị trạng thái: Đầu tiên, không gian cần được biểu diễn dưới dạng một đồ thị trạng thái, trong đó các nút đại diện cho các trạng thái khác nhau của hệ thống, và các cạnh đại diện cho các quan hệ giữa các trạng thái.
- Xác định các điểm quan trọng: Các điểm quan trọng trong không gian, như các vị trí cần được bao phủ, được biểu diễn dưới dạng các nút đặc biệt trong đồ thị trạng thái.
- Phân rã đồ thị: Sử dụng lý thuyết độ cao Morse, đồ thị trạng thái được phân rã thành các vùng không gian con, được gọi là các lớp Morse, sao cho mỗi lớp chứa một số lượng nhỏ nhất các điểm quan trọng.
- Lập kế hoạch đường đi: Một lộ trình di chuyển được tạo ra bằng cách đi qua các lớp Morse theo một cách tiếp cận đặc biệt, đảm bảo rằng mỗi điểm quan trọng được bao phủ ít nhất một lần.
- Tối ưu hóa lộ trình: Cuối cùng, lộ trình di chuyển có thể được tối ưu hóa để giảm thiểu thời gian hoặc năng lượng tiêu thụ, hoặc để đáp ứng các yêu cầu cụ thể của bài toán.

Phương pháp Morse-Based Decomposition là một phương pháp phức tạp nhưng mạnh mẽ, cho phép giải quyết các bài toán lập kế hoạch đường đi bao phủ cho drone trong không gian 2D hoặc 3D một cách hiệu quả và linh hoạt. Điều này giúp tối ưu hóa việc sử dụng drone trong các ứng dụng thực tế như giám sát môi trường, tìm kiếm và cứu hộ, hoặc giao hàng tự động.

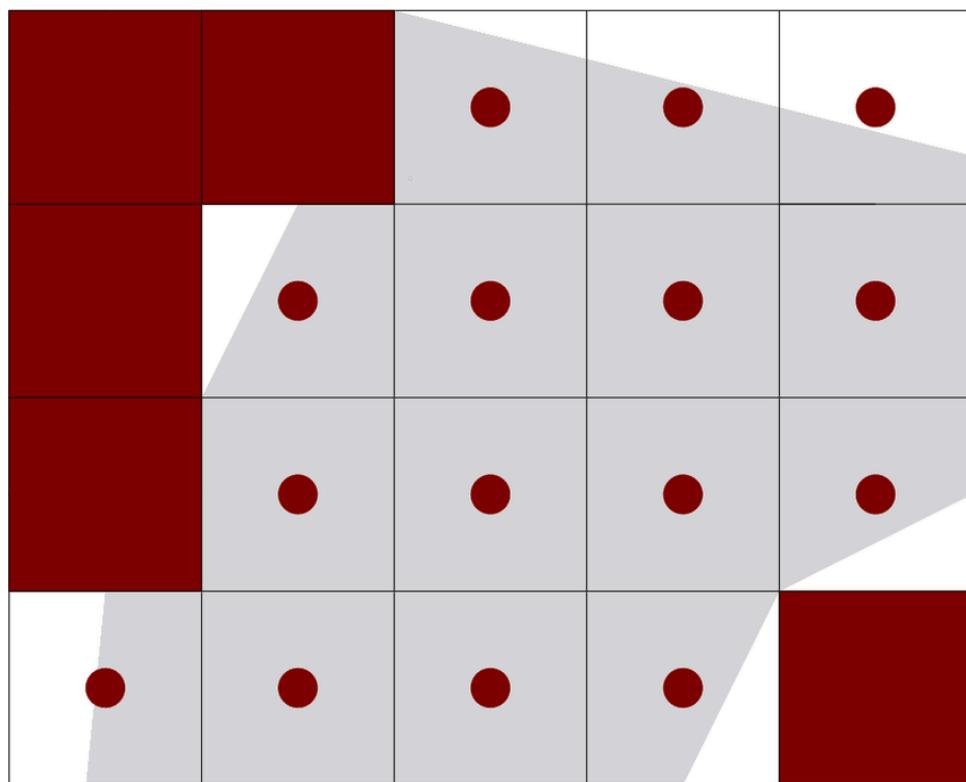
Ưu điểm của MBD:

- Có thể tạo ra các đường đi tối ưu: MBD có thể tạo ra các đường đi ngắn hơn so với các kỹ thuật khác.
- Có thể xử lý các môi trường phức tạp: MBD có thể hiệu quả đối với các môi trường có nhiều chướng ngại vật.
- Có thể được sử dụng cho các drone có nhiều hình dạng: MBD có thể được sử dụng cho các drone có hình dạng khác nhau, không chỉ drone hình tròn hoặc hình vuông.

Nhược điểm của MBD:

- Có thể đòi hỏi nhiều tính toán: MBD có thể đòi hỏi nhiều tính toán hơn so với các kỹ thuật khác.
- Có thể khó triển khai: MBD có thể khó triển khai hơn so với các kỹ thuật khác.

2.1.5 Phương pháp dựa trên lưới(Grid-Based Methods):



Hình 2.5 Hình ảnh phương pháp phân rã dựa trên lưới

Phương pháp Grid-Based Methods là một kỹ thuật thường được sử dụng để giải quyết bài toán lập kế hoạch đường đi bao phủ cho drone và các hệ thống tự động khác. Cách tiếp cận này chia không gian thành một lưới hình chữ nhật (grid), trong đó mỗi ô của lưới đại diện cho một phần nhỏ của không gian. Mỗi ô trong lưới có thể được xác định là "đã bao phủ" nếu nó đã được drone hoặc robot đi qua, hoặc "chưa bao phủ" nếu nó chưa được đi qua.

Dưới đây là cách phương pháp Grid-Based Methods hoạt động:

- Phân tích không gian thành lưới: Không gian cần được phân tích và chia thành các ô hình chữ nhật có kích thước đồng nhất. Kích thước của lưới có thể được điều chỉnh tùy thuộc vào độ chính xác và yêu cầu của bài toán.
- Xác định các điểm quan trọng: Các điểm quan trọng trong không gian, như các vị trí cần kiểm tra hoặc giám sát, được xác định và gắn với các ô tương ứng trong lưới.
- Thực hiện lập kế hoạch đường đi: Drone được hướng dẫn đi qua từng ô của lưới để bao phủ tất cả các điểm quan trọng. Có nhiều phương pháp để lập kế hoạch đường đi trong lưới, bao gồm thuật toán đi theo hình chữ Z, thuật toán quét hàng ngang, thuật toán quét hàng dọc, hoặc thuật toán A*.

- Tối ưu hóa lô trình: Sau khi lô trình được tạo ra, nó có thể được tối ưu hóa để giảm thiểu thời gian hoặc năng lượng tiêu thụ, hoặc đáp ứng các yêu cầu cụ thể của bài toán.

Phương pháp Grid-Based Methods là một cách tiếp cận đơn giản và linh hoạt để giải quyết bài toán lập kế hoạch đường đi bao phủ cho drone, và nó đã được áp dụng rộng rãi trong nhiều ứng dụng thực tế.

Có hai loại GBM chính:

- Phương pháp di chuyển ô (Cell-by-cell methods): drone di chuyển từ ô này sang ô khác theo một thứ tự nhất định, ví dụ như di chuyển theo hàng hoặc theo cột.
- Phương pháp di chuyển đường (Path-based methods): drone di chuyển từ ô này sang ô khác theo một đường đi được lập kế hoạch trước.

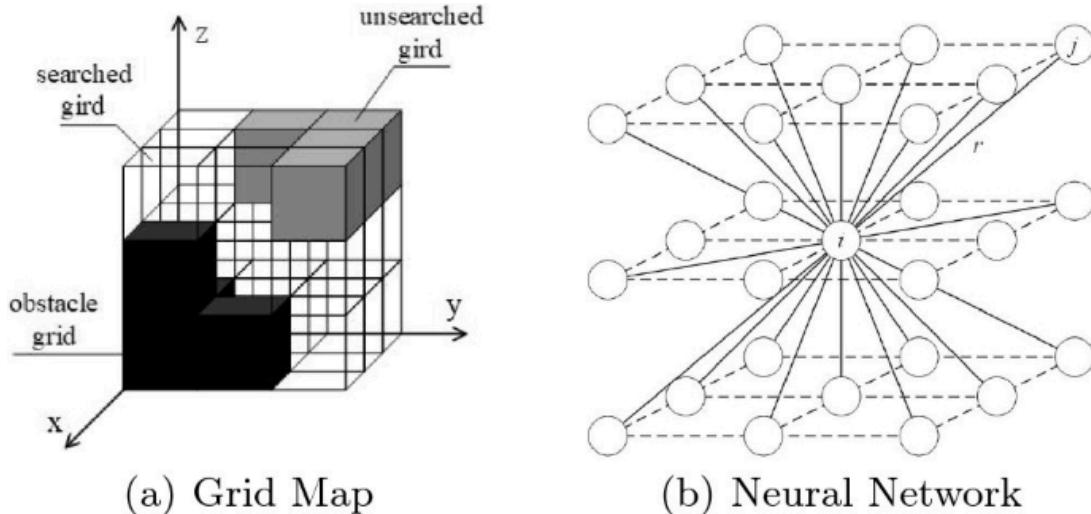
Ưu điểm của GBM:

- Đơn giản: GBM là một tập hợp các kỹ thuật đơn giản và dễ dàng triển khai.
- Hiệu quả: GBM có thể tìm kiếm các đường đi ngắn hơn so với các kỹ thuật khác.
- Có thể xử lý các môi trường phức tạp: GBM có thể hiệu quả đối với các môi trường có nhiều chướng ngại vật.

Nhược điểm của GBM:

- Độ chính xác: Độ chính xác của GBM phụ thuộc vào kích thước của các ô. Các ô nhỏ hơn sẽ dẫn đến độ chính xác cao hơn, nhưng cũng sẽ làm tăng số lượng trạng thái cần xem xét.
- Tính toán: GBM có thể đòi hỏi nhiều tính toán hơn so với các kỹ thuật khác, đặc biệt là đối với các môi trường lớn.

2.1.6 Phạm vi phủ sóng dựa trên mạng nơ-ron trên bản đồ lưới (Neural Network-Based Coverage on Grid Maps):



Hình 2.6 Hình ảnh phương pháp phạm vi phủ sóng dựa trên mạng nơ-ron trên bản đồ lưới

Phương pháp Neural Network-Based Coverage on Grid Maps là một phương pháp tiếp cận hiện đại trong việc giải quyết bài toán lập kế hoạch đường đi bao phủ cho drone hoặc robot di chuyển trong không gian 2D. Phương pháp này sử dụng mạng nơ-ron nhân tạo để học và dự đoán các đường đi hiệu quả trên các bản đồ lưới (grid maps).

Dưới đây là phác thảo chi tiết về cách phương pháp Neural Network-Based Coverage on Grid Maps hoạt động:

- Xây dựng mô hình mạng nơ-ron: Đầu tiên, một mô hình mạng nơ-ron được xây dựng để học cách lập kế hoạch đường đi bao phủ trên các bản đồ lưới. Mô hình này có thể sử dụng các kiến trúc mạng nơ-ron như mạng nơ-ron hồi tiếp (RNN), mạng nơ-ron tích chập (CNN), hoặc mạng nơ-ron hồi quy (LSTM) để học cách điều chỉnh đường đi dựa trên dữ liệu đầu vào.
- Thu thập dữ liệu huấn luyện: Dữ liệu huấn luyện được thu thập từ các mô phỏng hoặc thực tế về các trạng thái và hành động của drone hoặc robot trong quá trình di chuyển và bao phủ không gian. Dữ liệu này bao gồm thông tin về các vị trí hiện tại, các vị trí tiếp theo dự kiến, và các hành động cần thực hiện để di chuyển từ một vị trí đến vị trí khác.
- Huấn luyện mô hình mạng nơ-ron: Mô hình mạng nơ-ron được huấn luyện trên dữ liệu thu thập để học cách dự đoán các đường đi bao phủ hiệu quả. Trong quá trình này, mạng nơ-ron sẽ học cách điều chỉnh

đường đi dựa trên môi trường và các yếu tố khác như ràng buộc hoặc hạn chế.

- Áp dụng mô hình để lập kế hoạch đường đi: Khi mô hình đã được huấn luyện, nó có thể được sử dụng để dự đoán các đường đi bao phủ trên các bản đồ lưới mới. Drone hoặc robot sẽ tiếp tục di chuyển theo các đường đi được dự đoán, đảm bảo rằng không gian được bao phủ một cách hiệu quả và tự động.

Phương pháp Neural Network-Based Coverage on Grid Maps cung cấp một cách tiếp cận linh hoạt và mạnh mẽ để giải quyết bài toán lập kế hoạch đường đi bao phủ cho drone hoặc robot trong không gian 2D, đồng thời khai thác sức mạnh của học máy và mạng nơ-ron nhân tạo. Điều này giúp tối ưu hóa việc sử dụng drone trong nhiều ứng dụng thực tế như giám sát, kiểm tra, hoặc làm sạch tự động.

Ưu điểm của NN-Coverage:

- Hiệu quả: NN-Coverage có thể giúp drone di chuyển hiệu quả hơn bằng cách dự đoán các khu vực mà nó có thể phủ.
- Độ chính xác: NN-Coverage có thể dự đoán các khu vực mà drone có thể phủ với độ chính xác cao.
- Khả năng thích ứng: NN-Coverage có thể thích ứng với các môi trường mới bằng cách được huấn luyện trên dữ liệu mới.

Nhược điểm của NN-Coverage:

- Yêu cầu dữ liệu: NN-Coverage yêu cầu một lượng lớn dữ liệu để huấn luyện mạng nơ-ron.
- Tính toán: NN-Coverage có thể đòi hỏi nhiều tính toán hơn so với các kỹ thuật khác.

2.3 Phương pháp tối ưu hóa toán học

Hầu hết các phương pháp được giới thiệu ở trên đều phù hợp cho các bài toán CPP trong đó diện tích tuần tra nhỏ và chỉ sử dụng một drone. Ngoài ra, nếu sử dụng nhiều drone thì sự tương tác giữa các drone gần như không tồn tại. Điều này làm giảm hiệu quả của bài toán CPP và mục tiêu chính chỉ là phạm vi phủ sóng tối đa.

Các phương pháp dựa trên phân rã cũng có thể được mở rộng cho trường hợp triển khai nhiều UAV, chẳng hạn như: Multi-Drones Boustrophedon Decomposition, Multi-Drones Spanning Tree Coverage, Multi-Drones Neural Network-Based Coverage, Multi-Drones Graph-Based and Boundary

Coverage. Nhưng như đã đề cập ở trên, mục tiêu chính vẫn là độ bao phủ của đường đi. Đối với các khu vực tuần tra rộng lớn, phức tạp, chúng em quan tâm đến thuật toán CPP có thể cho phép triển khai nhiều drone. Hơn nữa, cần có sự phối hợp giữa các drone để đạt được nhiều mục tiêu như phạm vi bao phủ tối đa, thời gian tuần tra tối thiểu, mức tiêu thụ năng lượng tối thiểu, v.v. Điều này dẫn đến việc phải giải quyết các vấn đề tối ưu hóa, thay vì chỉ sử dụng các phương pháp phân rã. Bài toán bao phủ một khu vực có nhiều drone nằm ở việc tính toán các tuyến đường tối ưu nhằm giảm thiểu thời gian phủ sóng.

Phương pháp tối ưu hóa toán học là một trong những cách tiếp cận chính để giải quyết bài toán lập kế hoạch đường đi bao phủ cho drone. Các loại mô hình toán học thường được sử dụng trong phương pháp này bao gồm:

- Mô hình Tuyến tính (Linear Model): Trong mô hình tuyến tính, mục tiêu và các ràng buộc được biểu diễn dưới dạng các phương trình và bất phương trình tuyến tính. Các hàm mục tiêu và ràng buộc đều là các hàm tuyến tính của biến quyết định. Mô hình tuyến tính thường dễ giải quyết và có thể được sử dụng khi bài toán lập kế hoạch đơn giản và các ràng buộc tương đối đơn giản.
- Mô hình Bậc hai (Quadratic Model): Trong mô hình bậc hai, hàm mục tiêu và các ràng buộc có thể chứa các thành phần bậc hai của các biến quyết định. Mô hình này thường được sử dụng để mô tả các tương tác phức tạp hơn giữa các biến và ràng buộc trong bài toán.
- Mô hình Phi tuyến (Nonconvex Model): Trong mô hình phi tuyến, hàm mục tiêu hoặc các ràng buộc có thể là phi tuyến, tức là không phải là các hàm lồi hoặc không thể chứng minh được là lồi. Mô hình này thường gặp phải những thách thức lớn hơn trong quá trình giải quyết vì các phương pháp tối ưu hóa có thể không thể đảm bảo tìm ra giải pháp tối ưu toàn cục.
- Mô hình Tối ưu hóa lồi (Convex Optimization): Trong mô hình này, bài toán được biểu diễn dưới dạng một hàm mục tiêu (objective function) cần tối ưu trên một không gian lồi. Các hàm mục tiêu có thể là việc tối thiểu hóa thời gian hoặc năng lượng tiêu thụ của drone, hoặc tối đa hóa hiệu suất bao phủ. Các phương pháp tối ưu hóa lồi như phương pháp gradient descent, phương pháp Newton, hoặc các thuật toán tối ưu hóa lồi khác có thể được áp dụng để giải quyết bài toán này.
- Mô hình Tối ưu hóa Khoảng cách (Distance Optimization): Trong mô hình này, mục tiêu là tối thiểu hóa khoảng cách di chuyển của drone trong quá trình bao phủ không gian. Các thuật toán như thuật toán Dijkstra, thuật toán A*, hoặc các phương pháp tìm kiếm bài toán ngắn nhất khác có thể được sử dụng để giải quyết bài toán này.

- Mô hình Tối ưu hóa đồ thị (Graph Optimization): Trong mô hình này, không gian và các điểm quan trọng được biểu diễn dưới dạng đồ thị, với các đỉnh đại diện cho các vị trí trong không gian và các cạnh đại diện cho các kết nối giữa các vị trí. Bài toán sau đó trở thành việc tìm kiếm đường đi ngắn nhất hoặc lô trình tối ưu trên đồ thị này, sử dụng các thuật toán như thuật toán Dijkstra hoặc thuật toán Bellman-Ford.
- Mô hình Tối ưu hóa động (Dynamic Optimization): Trong mô hình này, các ràng buộc động như hạn chế tốc độ hoặc tốc độ di chuyển của drone được tính toán và áp dụng trong quá trình lập kế hoạch đường đi. Các phương pháp như lập kế hoạch động, lập kế hoạch động biến, hoặc lập kế hoạch động tuyến tính có thể được sử dụng để giải quyết bài toán này.

Mỗi loại mô hình toán học có ưu điểm và hạn chế riêng, và việc lựa chọn một loại mô hình phụ thuộc vào độ phức tạp của bài toán, tính chất của dữ liệu, và các yêu cầu cụ thể của ứng dụng. Trong bài toán lập kế hoạch đường đi bao phủ cho drone, sự lựa chọn giữa các loại mô hình này sẽ phụ thuộc vào yếu tố như độ phức tạp của không gian, yêu cầu về thời gian, và khả năng tối ưu hóa được áp dụng.

2.4 Mô hình quy hoạch đường đi phủ sóng của UAV tiết kiệm năng lượng

Trong việc nghiên cứu về mô hình CPP, em đã tham khảo mô hình được trình bày trong bài báo "Energy-Efficient UAVs Coverage Path Planning Approach" của các tác giả Gamil Ahmed, Tarek Sheltami, Ashraf Mahmoud và Ansar Yasar, được công bố vào tháng 3 năm 2022.

Máy bay không người lái (UAV), thường được gọi là máy bay không người lái, đã thu hút được sự chú ý đáng kể nhờ tính năng nhanh nhẹn, cơ động và linh hoạt. Chúng đóng một vai trò quan trọng trong các nhiệm vụ trinh sát, kiểm tra, tình báo và giám sát hiện đại. Lập kế hoạch đường dẫn phủ sóng (CPP), một trong những khía cạnh quan trọng quyết định chất lượng của hệ thống thông minh, tìm kiếm quỹ đạo tối ưu để bao phủ toàn bộ khu vực quan tâm (ROI). Tuy nhiên, thời gian bay của UAV bị hạn chế do hạn chế về pin và có thể không bao phủ toàn khu vực, đặc biệt là ở khu vực rộng lớn. Vì vậy, tiêu thụ năng lượng là một trong những vấn đề thách thức nhất cần được tối ưu hóa. Trong bài báo này, chúng em đề xuất một thuật toán quy hoạch đường bao phủ hiệu quả về năng lượng để giải quyết vấn đề CPP. Mục tiêu là tạo ra một đường phủ sóng không va chạm nhằm giảm thiểu mức tiêu thụ năng lượng tổng thể và đảm bảo bao phủ toàn bộ khu vực. Để làm được điều đó, đường bay được tối ưu hóa và số vòng quay được giảm xuống để giảm thiểu mức tiêu thụ năng lượng. Phương pháp được đề xuất trước tiên sẽ phân rã ROI thành một tập hợp các ô tùy thuộc vào dấu vết của camera UAV. Sau

đó, vấn đề quy hoạch đường đi bao phủ được hình thành, trong đó giải pháp chính xác được xác định bằng cách sử dụng bộ giải.

Với ROI, kho và UAV, mục đích là giảm thiểu mức tiêu thụ năng lượng và đồng thời tối đa hóa vùng phủ sóng. Đầu ra của thuật toán lập kế hoạch đường đi được đề xuất là đường bay không va chạm tối ưu với mức tiêu thụ năng lượng tối thiểu cho tất cả các máy bay không người lái với các yêu cầu sau: UAV có thể tiếp nhiên liệu tại một kho, tức là trạm gốc. Hơn nữa, các UAV cất cánh và hạ cánh tại kho. Để đạt được mục đích này, các giả định sau được đưa ra:

- Quỹ đạo bay của UAV được thiết kế tĩnh. Nói cách khác, đường đi của UAV đã được lên kế hoạch trước, tức là trước khi bắt đầu nhiệm vụ.
- Máy bay không người lái được trang bị một camera quay mặt tải xuống với khả năng bao phủ đã được biết trước.
- Máy bay không người lái bay ở độ cao cố định và do đó, bài toán đã được đơn giản hóa thành bài toán CPP 2D.

ROI có thể là bất kỳ hình dạng, hình vuông, hình tròn, đa giác nào, v.v. Mục tiêu là bao phủ hoàn toàn ROI với mức tiêu thụ năng lượng tối thiểu. Do đó hàm mục tiêu là một bài toán tối thiểu hóa. Hơn nữa, các biến quyết định được lấy cảm hứng từ công thức MILP cổ điển.

Mục tiêu là đảm bảo hoàn toàn ROI với mức tiêu thụ năng lượng tối thiểu. Do đó hàm mục tiêu là một bài toán tối thiểu hóa. Hơn nữa, các biến quyết định được lấy cảm hứng từ công thức MILP cổ điển.

Gọi N là một số vị trí hoặc điểm tham chiếu ($1\text{-kho}, 2, \dots, N$) trong ROI được yêu cầu phải được bảo vệ bởi N_d máy bay không người lái, D_{ij} là khoảng cách từ vị trí i đến vị trí j , E_u là năng lượng sẵn có của một máy bay không người lái u , E_{ij} là năng lượng tiêu thụ để bay từ vị trí i đến vị trí j , OC_{ij} là chi phí do chướng ngại vật địa hình và có thể được tính như sau:

$OC_{ij} = \inf$ nếu tại vị trí (i, j) có chướng ngại vật, $OC_{ij} = 0$ với trường hợp còn lại

Biến quyết định x_{uij} có thể được biểu diễn như sau:

$$x_{uij} = 1 \text{ nếu drone } u \text{ bay từ vị trí } i \text{ đến vị trí } j, \text{ ngược lại } x_{uij} = 0$$

Mục tiêu của những ràng buộc này là tạo ra một đường đi không va chạm để ROI được bao phủ hoàn toàn với mức tiêu thụ năng lượng tối thiểu, tức là vấn đề giảm thiểu. Vì vậy, hàm mục tiêu xem xét cả phạm vi phủ sóng và nhiệm vụ an toàn của máy bay không người lái, tức là tránh va chạm với các chướng ngại vật trên địa hình.

Với định nghĩa này, các công thức của MILP được liệt kê dưới đây

Hàm mục tiêu là một bài toán tối thiểu hóa và có thể được biểu diễn dưới dạng:

$$\min T = \sum_{i=1}^N \sum_{j=1}^N (E_{ij} * X_{uij} + OC_{ij} * X_{uij}), u \in [1, N_d] \quad (1)$$

Tuân theo các ràng buộc sau:

$$\sum_{j=1}^N X_{uij} = 1, \forall i = 2, \dots, N, \forall u = 1, \dots, N_d \quad (2)$$

$$\sum_{j=1}^N X_{u1j} = \sum_{j=2}^N X_{uj1} = u, \forall u = 1, \dots, N_d \quad (3)$$

$$\sum_{i=1}^N \sum_{j=1}^N E_{ij} * X_{uij} \leq E_u, \forall u = 1, \dots, N_d \quad (4)$$

$$0 < E_{ij} * X_{uij} + E_{j1} \leq E_u, \forall i, j = 1, \dots, N, \forall u = 1, \dots, N_d \quad (5)$$

$$\sum_{u=1}^{N_d} \sum_{j=1}^N X_{uij} = 1, \forall i = 1, \dots, N \quad (6)$$

$$X_{uij} \in [0, 1], \forall i, j = 1, \dots, N \quad (7)$$

- Hàm mục tiêu trong biểu thức (1) giảm thiểu tổng mức tiêu thụ năng lượng di chuyển trên tất cả điểm tham chiếu bên trong khu vực cần lập kế hoạch. Phần đầu tiên của hàm mục tiêu là tính năng lượng để chuyển động trên tất cả các điểm tham chiếu. Trong khi phần thứ hai sẽ thêm một hình phạt, chi phí,...nếu đường đi từ điểm tham chiếu i đến điểm tham chiếu j đang đi qua chướng ngại vật ngăn va chạm với chướng ngại vật địa hình.
- Ràng buộc trong phương trình (2) chỉ ra rằng mỗi vị trí được đi trước và đi trước bởi chính xác một vị trí khác ngoại trừ kho (trình tự duy nhất của đường dẫn UAV).
- Ràng buộc trong phương trình (3) chỉ ra rằng mỗi UAV khởi hành từ kho và quay lại kho,...vị trí ban đầu của máy bay không người lái hoặc trạm gốc.
- Ràng buộc trong phương trình (4) chỉ ra rằng năng lượng là cần thiết để hoàn thành chuyến đi bằng máy bay không người lái phải ít hơn năng lượng sẵn có của máy bay không người lái. Đối với những khu vực rộng lớn mà năng lượng sẵn có của máy bay không người lái không đủ

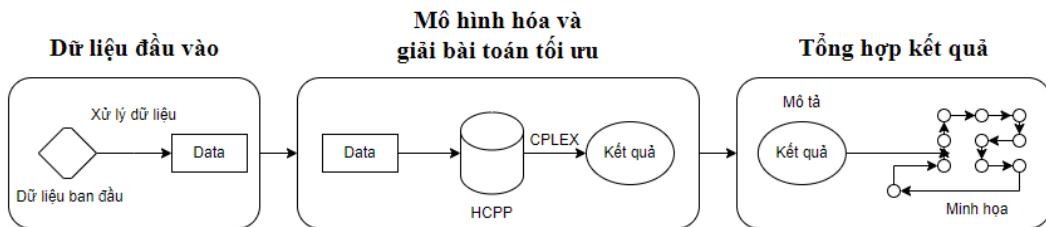
để bao phủ khu vực đó, máy bay không người lái sẽ quay trở lại trạm gốc để sạc lại và hoàn thành phạm vi phủ sóng của khu vực.

- Ràng buộc trong phương trình (5) giải thích rằng máy bay không người lái không thể bay từ điểm tham chiếu i đến điểm tham chiếu j trừ khi nó có đủ sức mạnh để di chuyển từ điểm tham chiếu i đến điểm tham chiếu j và từ điểm tham chiếu j đến kho là đủ. Ràng buộc này đảm bảo việc quay trở lại nơi lưu trữ một cách an toàn.
- Đối với nhiều UAV, ràng buộc trong phương trình (6) đảm bảo rằng chỉ có một UAV bay từ điểm tham chiếu i đến điểm tham chiếu j và chỉ một lần.
- Cuối cùng, ràng buộc trong phương trình (7) cho thấy $x_{u i j}$ là biến nhị phân.

Công thức chính xác cho CPP đã được đề xuất dựa trên MILP để tìm kiếm quỹ đạo tối ưu cho máy bay không người lái sao cho đường đi được tạo ra giảm thiểu mức tiêu thụ năng lượng và đáp ứng mọi ràng buộc trong quá trình thực hiện nhiệm vụ. Cách tiếp cận được đề xuất có thể phân chia ROI thành các tiểu vùng một cách hiệu quả. Chiến lược phân bổ các tiểu vùng được đề xuất đã góp phần giảm tiêu thụ năng lượng.

CHƯƠNG 3 : THỰC NGHIỆM

3.1 Quy trình thực nghiệm



Hình 3.1 Hình ảnh quy trình thực nghiệm

Quy trình thực nghiệm sẽ bao gồm 3 bước cơ bản lần lượt là :

Bước 1: Dữ liệu đầu vào : tại đây dữ liệu ban đầu sẽ được xử lý để sao cho phù hợp với mô hình

Bước 2: Mô hình hóa và giải bài toán tối ưu: Tại bước này sẽ thực hiện mô hình hóa bài toán bằng ngôn ngữ lập trình Python và tiến hành giải bài toán bằng công cụ CPLEX.

Bước 3: Tổng hợp kết quả : Sau khi đã giải bài toán tối ưu chúng ta sẽ thu được kết quả, kết quả này sẽ thu được dưới dạng các biến x, ở đây chúng ta sẽ cần xử lý, tổng hợp kết quả và đưa ra định dạng cần thiết để minh họa được kết quả thu được.

Sau đây chúng ta sẽ đi làm rõ cụ thể các bước để thực hiện quá trình thực nghiệm.

3.2 Dữ liệu đầu vào

Dữ liệu thực nghiệm của chúng ta sẽ có dạng Json với đầu vào là 1 file input *.plan, file input *.plan có dạng như sau:

```
{
  "fileType": "Plan",
  "geoFence": {
    "circles": [
    ],
    "polygons": [
    ],
    "version": 2
  },
  "groundStation": "QGroundControl",
  "mission": {
    "cruiseSpeed": 15,
    "firmwareType": 12,
    "globalPlanAltitudeMode": 1,
    "hoverSpeed": 5,
    "items": [
    ]
  }
}
```

```
{
    "AMSLAltAboveTerrain": null,
    "Altitude": 50,
    "AltitudeMode": 1,
    "autoContinue": true,
    "command": 22,
    "doJumpId": 1,
    "frame": 3,
    "params": [
        0,
        0,
        0,
        null,
        21.03484371,
        105.82542945,
        50
    ],
    "type": "SimpleItem"
},
{
    "AMSLAltAboveTerrain": null,
    "Altitude": 50,
    "AltitudeMode": 1,
    "autoContinue": true,
    "command": 16,
    "doJumpId": 2,
    "frame": 3,
    "params": [
        0,
        0,
        0,
        null,
        21.03479315,
        105.82719742,
        50
    ],
    "type": "SimpleItem"
},
{
    "AMSLAltAboveTerrain": null,
    "Altitude": 50,
    "AltitudeMode": 1,
    "autoContinue": true,
    "command": 16,
    "doJumpId": 3,
    "frame": 3,
    "params": [
        0,
        0,
        0,
        null,
        21.032397,
        105.82707599,
        50
    ],
    "type": "SimpleItem"
},
{
    "AMSLAltAboveTerrain": null,
    "Altitude": 50,
    "AltitudeMode": 1,
    "autoContinue": true,
    "command": 16,
    "doJumpId": 4,
    "frame": 3,
    "params": [
        0,
        0,
        0,
        null,
        21.03197605,
        105.82504651,
        50
    ],
    "type": "SimpleItem"
},
{
```

```

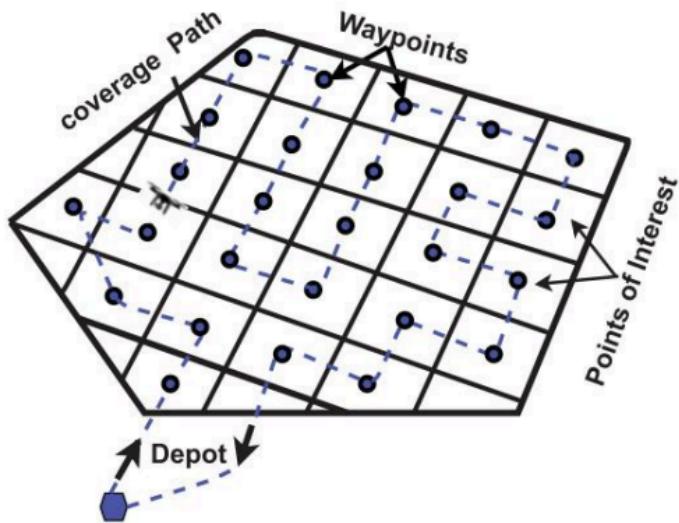
    "AMSLAltAboveTerrain": null,
    "Altitude": 50,
    "AltitudeMode": 1,
    "autoContinue": true,
    "command": 16,
    "doJumpId": 5,
    "frame": 3,
    "params": [
        0,
        0,
        0,
        null,
        21.03353842,
        105.82455215,
        50
    ],
    "type": "SimpleItem"
},
{
    "autoContinue": true,
    "command": 20,
    "doJumpId": 6,
    "frame": 2,
    "params": [
        0,
        0,
        0,
        0,
        0,
        0,
        0
    ],
    "type": "SimpleItem"
}
],
"plannedHomePosition": [
    21.03484371,
    105.82542945,
    19
],
"vehicleType": 2,
"version": 2
},
"rallyPoints": {
    "points": [
    ],
    "version": 2
},
"version": 1
}
}

```

Từ file plan này chúng ta sẽ thu được các tọa độ và thứ tự các tọa độ tương ứng của nó.

Sau khi thu được file plan input cho bài toán chúng ta sẽ đi tiền xử lý dữ liệu:

Để đạt được phạm vi bao phủ tốt hơn, khu vực sẽ được phân tách thành các ô. Kích thước ô được xác định bởi dấu chân camera của UAV và có thể được bao phủ hoàn toàn khi UAV đi qua trung tâm của nó

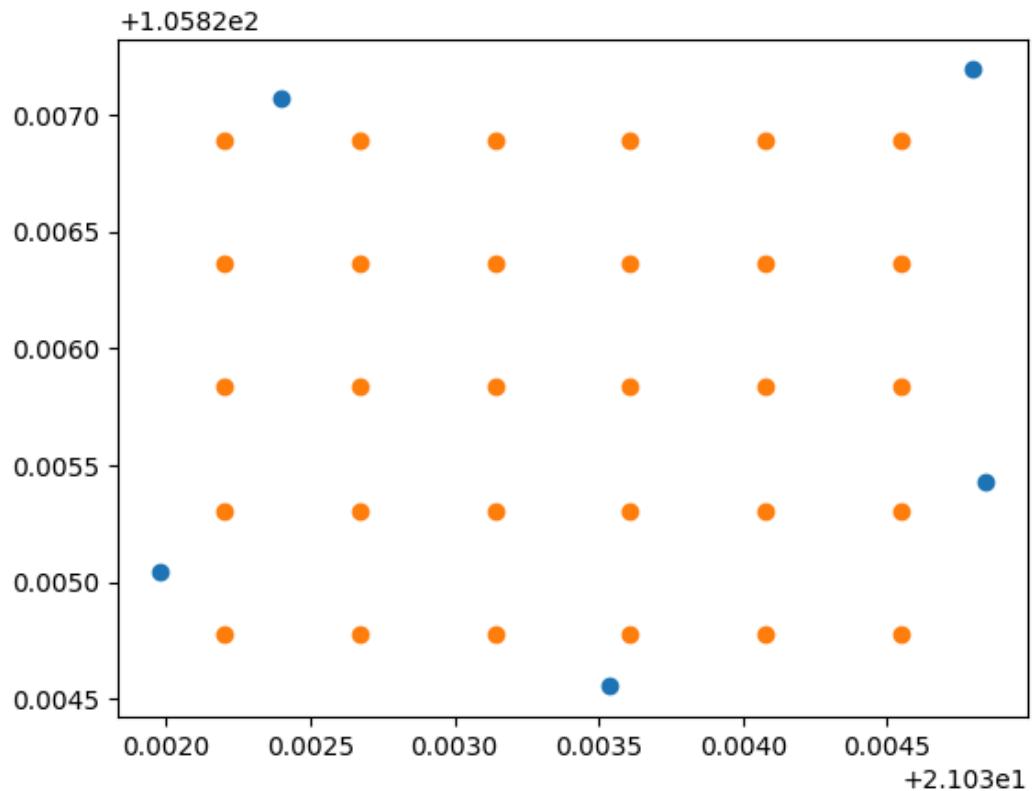


Hình 3.2: Hình ảnh khu vực bị phân rã và che phủ

Bước 1: Thiết lập khu vực bao phủ

Khu vực bao phủ: Là khu vực hình chữ nhật bao trùm toàn bộ các điểm dữ liệu đầu vào. Mục đích của việc tạo khu vực bao phủ là để đảm bảo rằng tất cả các điểm cần thiết đều được bao gồm trong quá trình xử lý dữ liệu sau này.

Dựa trên dữ liệu tọa độ các điểm thu thập được, ta tiến hành khởi tạo một khu vực hình chữ nhật bao trùm toàn bộ các điểm này. Sau đó, lọc ra các điểm nằm trong khu vực bao phủ để tạo thành tập dữ liệu mới.

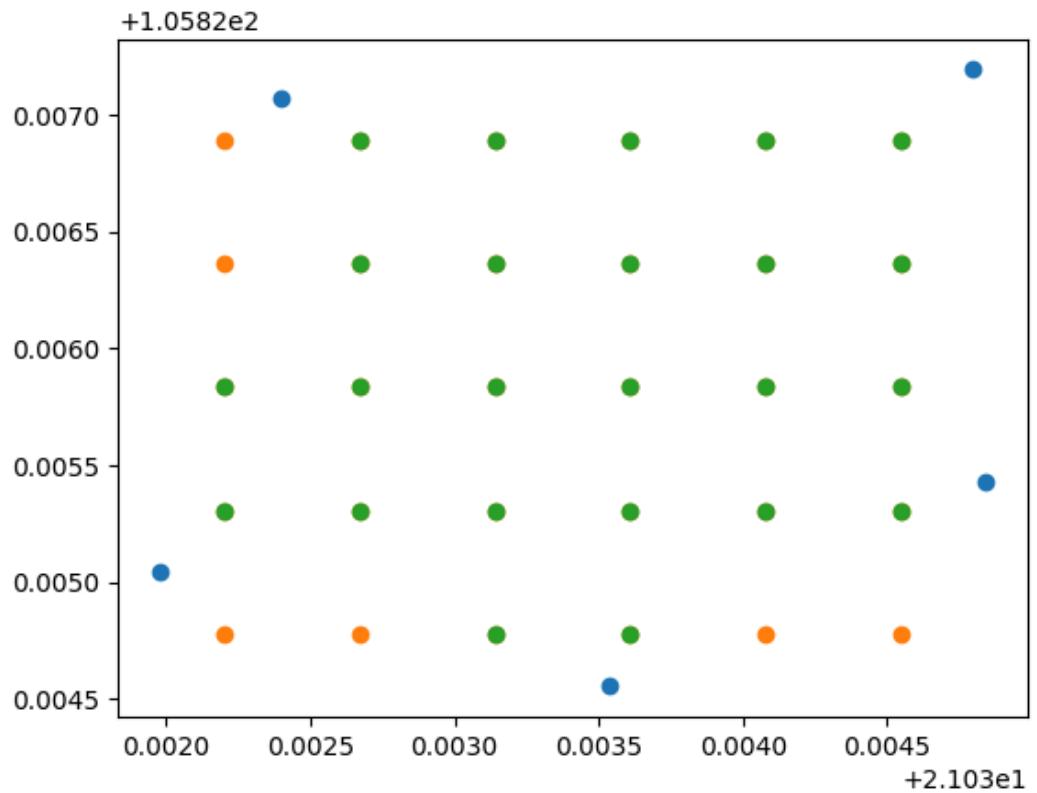


Hình 3.3: Hình ảnh biểu diễn khu vực bao phủ và tất cả các điểm nằm trong

Bước 2: Xác định điểm thuộc khu vực tuần tra

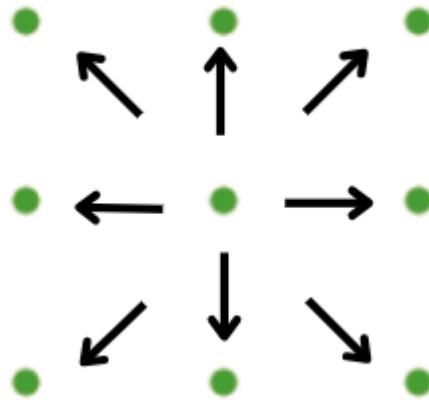
Khu vực tuần tra: Là khu vực cụ thể mà ta muốn drone di chuyển và thực hiện nhiệm vụ. Khu vực này có thể nhỏ hơn hoặc bằng khu vực bao phủ.

Tiếp theo, ta cần phân biệt các điểm nằm trong khu vực cần được bao phủ (khu vực tuần tra) với các điểm nằm ngoài khu vực này. Để thực hiện điều này, ta có thể sử dụng một thuộc tính đánh dấu (flag) cho các điểm thuộc khu vực tuần tra. Ví dụ, ta có thể gán giá trị true cho thuộc tính point.user đối với các điểm nằm trong khu vực tuần tra và false cho các điểm còn lại.



Hình 3.4 Hình ảnh biểu diễn các điểm nằm trong khu vực tuần tra

Sau khi thu thập dữ liệu về các điểm nằm trong khu vực tuần tra, ta tiến hành xác định các điểm lân cận đối với mỗi điểm. Mỗi điểm có thể di chuyển theo 8 hướng: trên, dưới, trái, phải, trên trái, trên phải, dưới trái, dưới phải. Tiếp theo, ta tính toán lượng năng lượng tiêu thụ khi di chuyển từ một điểm đến một điểm lân cận. Năng lượng tiêu thụ được xác định dựa trên khoảng cách giữa hai điểm.



Hình 3.5 Hình ảnh biểu diễn hướng di chuyển từ một điểm

3.3 Mô hình hóa và giải bài toán tối ưu

3.2.1 Xây dựng mô hình HCPP

Hàm mục tiêu: Biểu diễn hàm mục tiêu cần tối ưu hóa (tối thiểu hóa hoặc tối đa hóa) bằng hàm số tuyến tính của các biến quyết định.

Chúng ta có hàm mục tiêu ban đầu là :

$$\min T = \sum_{i=1}^N \sum_{j=1}^N (E_{ij} * X_{uij} + OC_{ij} * X_{uij}), u \in [1, N_d]$$

hàm mục tiêu này sẽ được biểu diễn dưới dạng code python như sau :

```
self.obj = sum(data.E[i, j] * self.X[u, i, j] + data.OC[i, j] * self.X[u, i, j]
| | | | for i in range(allpoints) for j in range(allpoints) for u in range(data.ND))
```

Tiếp theo chúng ta sẽ biểu diễn các ràng buộc của bài toán bằng các bất phương trình tuyến tính của các biến quyết định trên code của python :

```
self.cst_8 = ((self.s[i] - self.s[j] + data.N * self.X[u, i, j]) <= data.N - 1
| | | | for u in range(data.ND) for i in range(data.N) for j in (data.NEI[i][1:]) if i != j)

self.cst_81 = (sum(self.X[u, i, j] for j in range(allpoints)) == 1
| | | | for i in range(allpoints) for u in range(data.ND))

self.cst_82 = (sum(self.X[u, i, j] for i in range(allpoints)) == 1
| | | | for j in range(allpoints) for u in range(data.ND))

self.cst_83 = [sum(self.X[u, i, j] for j in (data.NEI[i][1:] + list(range(data.N, allpoints))))==1
| | | | for i in range(data.N) for u in range(data.ND)]

self.cst_85 = (self.X[u, i, i] == 0 for i in range(allpoints) for u in range(data.ND))

self.cst_11 = (sum(data.E[i, j] * self.X[u, i, j] for i in range(allpoints) for j in range(allpoints)) <= data.EU[u]
| | | | for u in range(data.ND))
```

Biến: Xác định các biến quyết định, bao gồm cả biến liên tục và biến nguyên :

```
allpoints = data.N + data.NumDepot
self.X = self.cpl_model.binary_var_cube(data.ND, allpoints, allpoints, name=f'X')
self.s = self.cpl_model.continuous_var_list(data.N, name=f's')
```

3.2.2 Giải mô hình HCPP

Sử dụng các phần mềm tối ưu hóa chuyên dụng để giải quyết mô hình MILP. Ở đây em sẽ sử dụng CPLEX để giải mô hình trên , CPLEX là phần mềm tối ưu hóa, tên của nó bắt nguồn từ simplex method và C programming language, mặc dù ngày nay phần mềm này đã có sử dụng phương pháp điểm trong và sử dụng cả ngôn ngữ C++, C# và Java. Phần mềm này có thể giải quyết được các bài toán quy hoạch nguyên, bài toán quy hoạch tuyến tính với kích thước cực lớn, bài toán quy hoạch toàn phương và quy hoạch lồi.

Phần mềm được phát triển bởi Robert E. Bixby và bán thông qua CPLEX Inc Optimization, tổ chức được mua lại bởi ILOG vào năm 1997; ILOG sau đó đã được mua lại bởi IBM vào tháng 1 năm 2009

Phần mềm sẽ cung cấp giải pháp tối ưu cho bài toán, bao gồm giá trị của các biến quyết định và giá trị tối ưu của hàm mục tiêu.

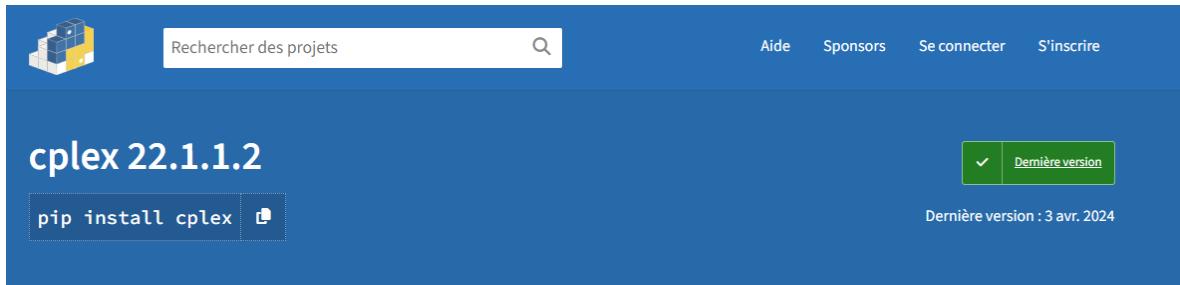
Để sử dụng CPLEX chúng ta cần tải CPLEX về :

Nếu bạn muốn giải nhiều biến dữ liệu thì cần liên hệ với CPLEX để được cung cấp phần mềm cũng như API để sử dụng.



Hình 3.6 Hình ảnh phần mềm CPLEX

Cách khác bạn có thể sử dụng bản miễn phí từ git : *pip install cplex*



Sau khi đã cài đặt được CPLEX chúng ta sẽ cài docplex để có thể import Model : *pip install docplex*

```
from docplex.mp.model import Model
cpl_model = Model(name='MILP_CPP')
```

để sử dụng model CPLEX chúng ta cũng cần viết thêm các hàm cần thiết :

```
def add_constraints(self, *constraints):
    for index, constraint in enumerate(constraints):
        self.cpl_model.add_constraints(constraint,
names=f'constraint{index}_')

def solve(self, time_limit=sys.maxsize):
    self.cpl_model.set_time_limit(time_limit)
    self.solution = self.cpl_model.solve(clean_before_solve=True)
    if self.solution == None:
        print('Fails')
    else:
        print('Solve successfull')

def print_results(self):
    print('Objective function value: %.2f' %
self.cpl_model.objective_value)
    print(self.cpl_model.solve_details)

def print_solution(self):
```

```

    self.cpl_model.print_solution()

def print_information(self):
    self.cpl_model.print_information()

def print_lp_string(self):
    print(self.cpl_model.export_as_lp_string())
    self.cpl_model.export_as_lp('D:/CPP')

def clear_model(self):
    self.cpl_model.clear()

def get_results(self):
    return {'optimal_value':self.cpl_model.objective_value,
            'time':self.cpl_model.solve_details.time,
            'gap':self.cpl_model.solve_details.gap}

```

Sau khi đã khởi tạo và có được các biến data chúng ta bắt đầu giải mô hình :

```

model = MILP0(DATA)

model.print_lp_string()

model.solve()

model.print_solution()

way = model.getsolution(Area, points, len(NEI) - 1)

model.exportToJson(pathmissionjson, way, att=droneAltitude,
speed=droneSpeed)

```

3.4 Tổng hợp kết quả

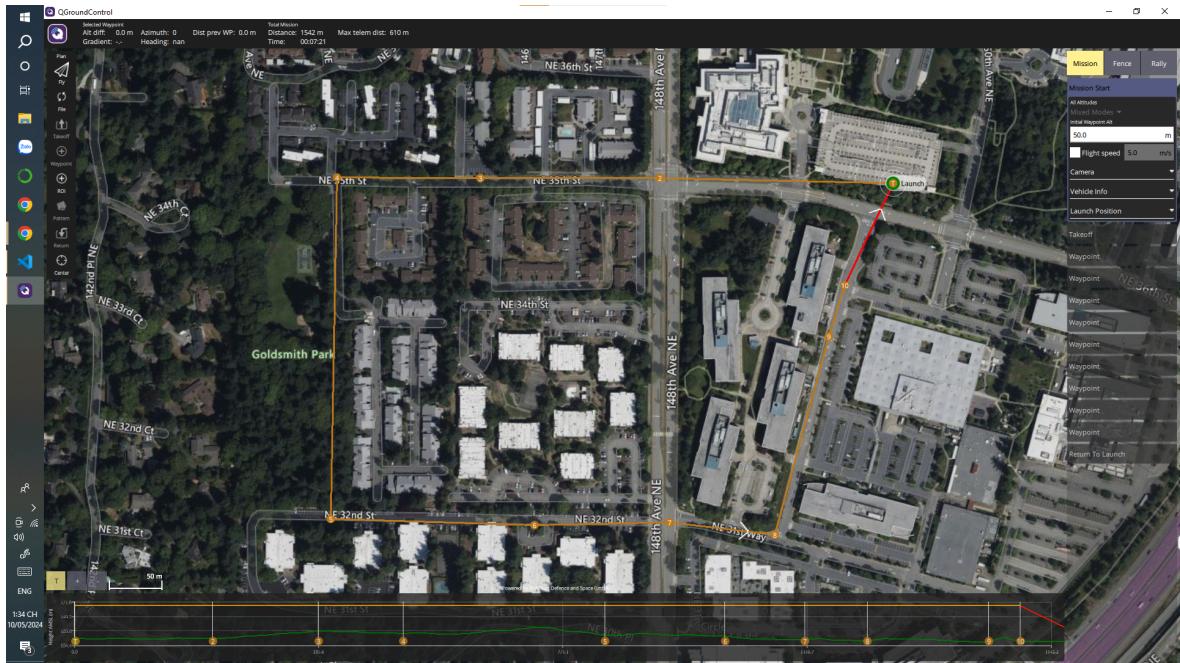
Sau quá trình giải mô hình chúng ta tiến hành phân tích kết quả thu được để đảm bảo tính hợp lý và chính xác.

Kết quả đầu ra sau khi giải mô hình chúng ta sẽ được các giá trị của biến x ví dụ như :

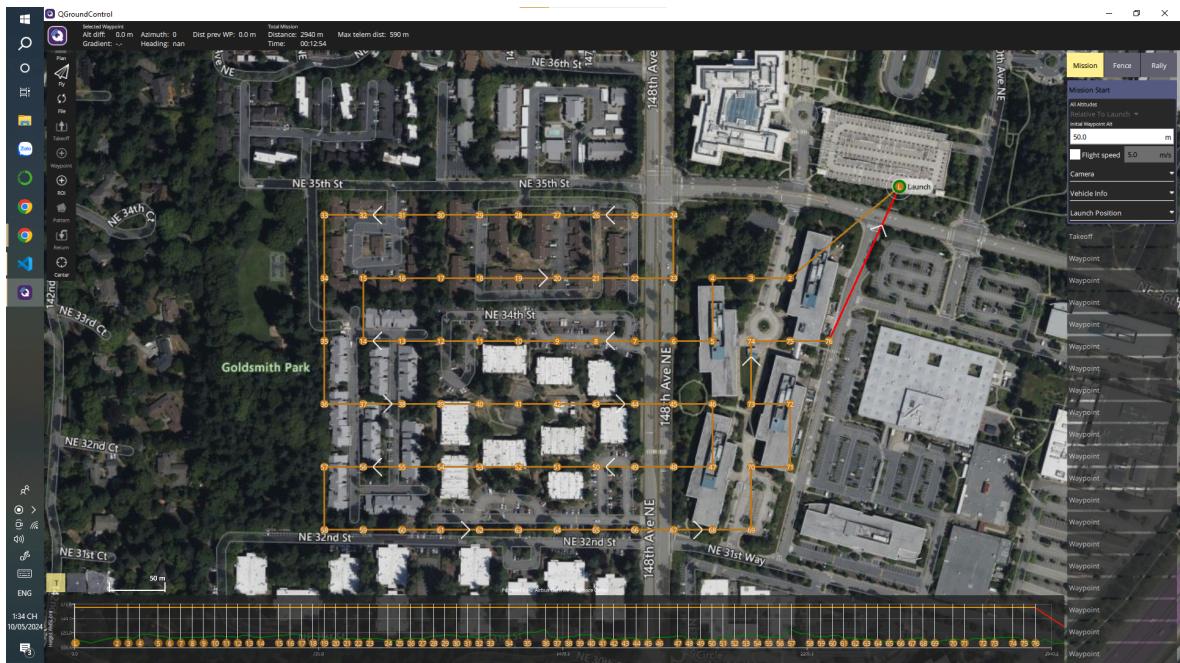
```
X_0_0_6=1
X_0_1_0=1
X_0_2_1=1
X_0_3_2=1
X_0_4_3=1
X_0_5_4=1
X_0_6_7=1
X_0_7_12=1
X_0_8_9=1
X_0_9_15=1
X_0_10_11=1
X_0_11_5=1
X_0_12_13=1
X_0_13_14=1
X_0_14_8=1
X_0_15_18=1
X_0_16_10=1
X_0_17_16=1
X_0_18_19=1
X_0_19_20=1
X_0_20_17=1
```

Sau khi đã thu được biến x, ta sẽ biết được thứ tự di chuyển lần lượt của các điểm, tiếp theo chúng ta cần tạo ra file *.plan giống như file ban đầu nhưng thêm các điểm đã được tính toán để phục vụ quá trình chạy sau này.

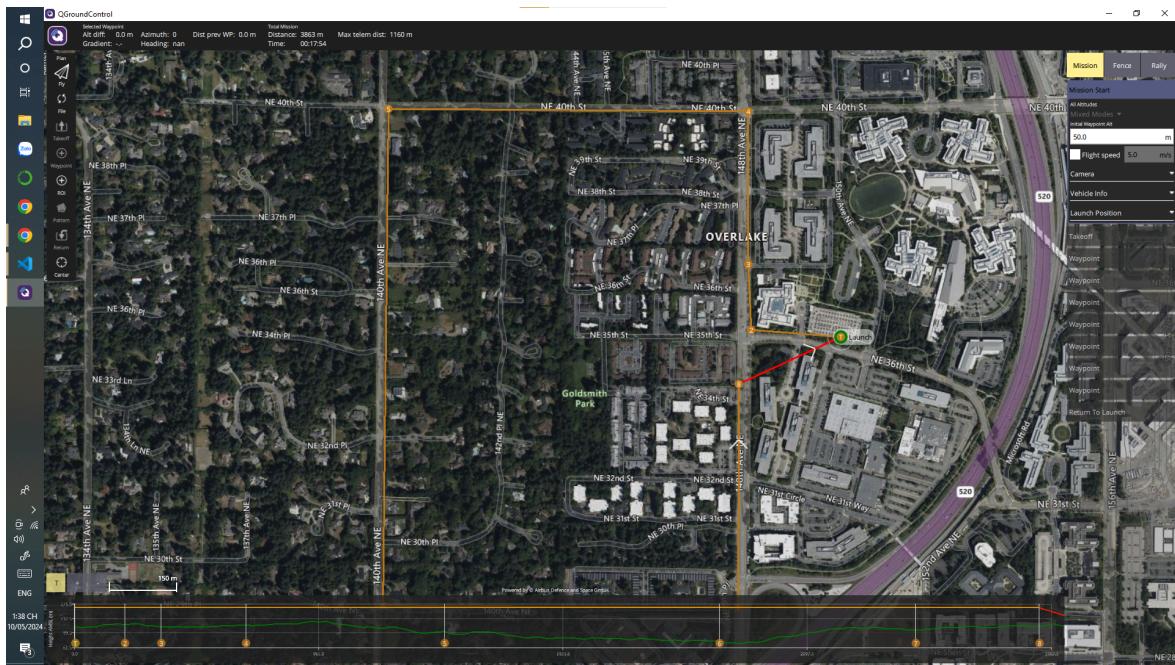
Dưới đây là 1 số hình ảnh biểu diễn kết quả sau quá trình giải mô hình và đã được chuyển sang dạng plan và hiển thị lên ứng dụng :



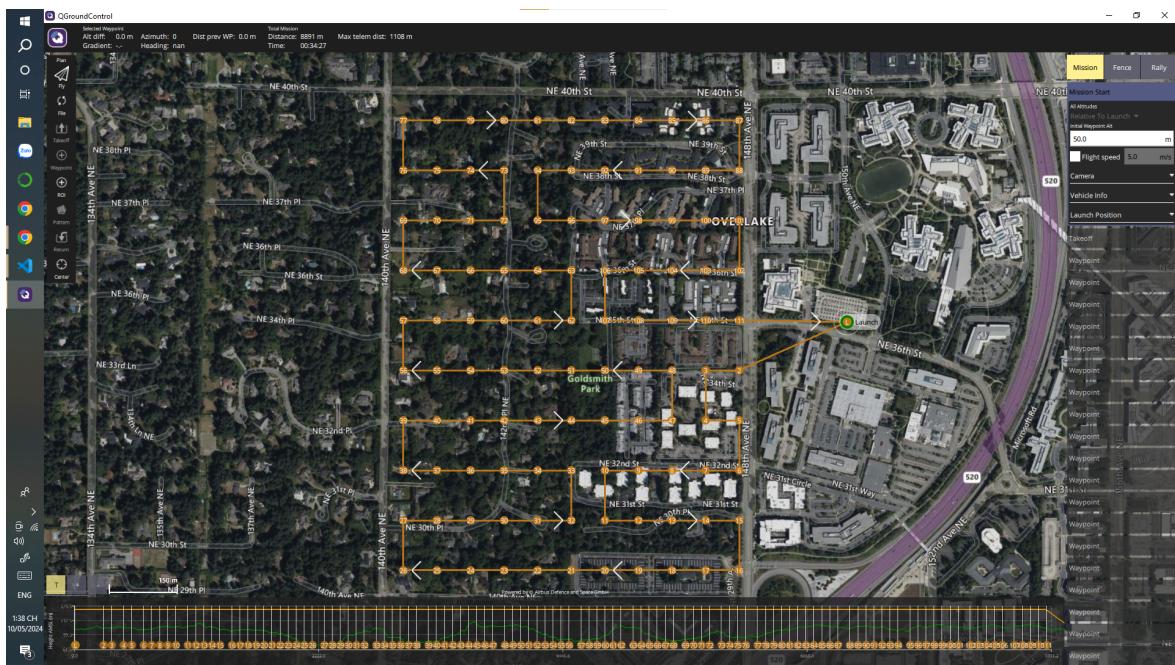
Hình 3.7 Hình ảnh đầu vào của bài toán : Input-1



Hình 3.8 Hình ảnh đầu ra sau khi giải bài toán : Output-1



Hình 3.9 Hình ảnh đầu vào của bài toán : Input-2



Hình 3.10 Hình ảnh đầu ra sau khi giải bài toán : Output-2

CHƯƠNG 4 : MÔ PHỎNG BẰNG MICROSOFT AIRSIM

4.1. Giới thiệu về các công cụ mô phỏng

Để mô phỏng quá trình chạy của Drone trên máy tính, chúng ta có thể sử dụng một số công cụ phổ biến và được sử dụng rộng rãi trên toàn cầu.

Dưới đây là một số công cụ mà chúng ta có thể kể đến:

- Airsim: Đây là một công cụ mô phỏng Drone phát triển bởi Microsoft. Airsim cung cấp một môi trường mô phỏng chân thực và linh hoạt, cho phép người sử dụng tương tác với Drone thông qua API và cung cấp dữ liệu cảm biến và camera mô phỏng.
- PX4-Autopilot: Đây là một phần mềm mã nguồn mở giúp điều khiển Drone. PX4-Autopilot cung cấp một bộ công cụ mô phỏng để kiểm tra và xem trước các hành vi của Drone trong môi trường ảo. Nó hỗ trợ các chức năng điều khiển tự động và cung cấp một giao diện đồ họa để xem dữ liệu mô phỏng.
- QGroundControl: Đây là một ứng dụng điều khiển Drone mã nguồn mở và cung cấp giao diện đồ họa trực quan. QGroundControl cho phép người sử dụng xem và kiểm soát Drone trong quá trình mô phỏng. Nó hỗ trợ nhiều chức năng như xem trên bản đồ, điều khiển từ xa, và hiển thị dữ liệu từ các cảm biến.

Việc sử dụng các công cụ trên sẽ giúp chúng ta có trải nghiệm bay gần với thực tế và khám phá các khía cạnh khác nhau của quá trình chạy mô phỏng Drone. Tuy nhiên, cần lưu ý rằng do hạn chế của phần cứng và phần mềm, việc mô phỏng một số yếu tố vật lý như gió, nhiễu sóng,... có thể không được tái hiện chính xác trong quá trình mô phỏng.

Chúng ta sẽ cùng đi tìm hiểu về các ứng dụng được sử dụng trong quá trình chạy mô phỏng dưới đây.

4.2 Giới thiệu về Microsoft Airsim

Trong tương lai, drone và xe tự lái sẽ có thể xử lý được nhiều hơn các tình huống thực tế nhờ môi trường huấn luyện ngày càng tinh vi, chuyên nghiệp của nền tảng mô phỏng không gian thực AirSim do Microsoft phát triển.

Thực tế ảo không chỉ hỗ trợ đào tạo con người. Nhờ tính trung thực và chất lượng ngày càng được cải thiện, thực tế ảo giờ đây sẽ hỗ trợ con người huấn luyện robot một cách dễ dàng.



Hình 4.1 Hình ảnh mô phỏng AirSim

Hồi tháng Hai năm nay, bộ phận Microsoft Research từng giới thiệu một nền tảng mới có tên AirSim, giúp các nhà phát triển drone tích hợp hệ thống robot và tự động hóa.

Nền tảng Aerial Informatics and Robotics (AirSim) của Microsoft cung cấp hình dạng mô phỏng và các công cụ thực tế cho các nhà thiết kế hay nhà phát triển tạo ra lượng dữ liệu lớn phục vụ huấn luyện robot và drone. AirSim ứng dụng các tiến bộ trong tính toán và đồ họa, kết hợp giữa nhận thức và tính chất vật lý để tạo ra các mô phỏng chính xác, thực tế.

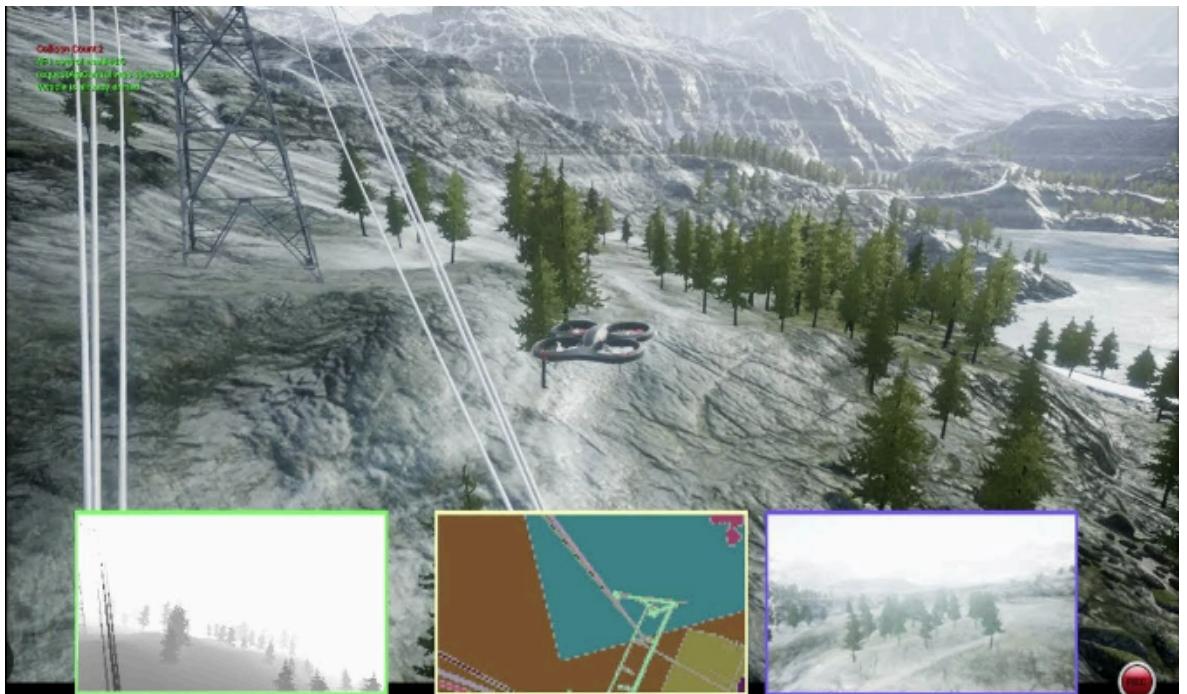
Môi trường đào tạo xe tự lái ngày càng được mở rộng từ thành phố tới ngoại ô

Phiên bản mới của AirSim gồm nhiều môi trường và hình ảnh mô phỏng xe hoàn toàn mới. Bên cạnh đó cũng có rất nhiều kịch bản mới đã sẵn sàng đưa vào thử nghiệm. Việc có thể tạo ra môi trường mô phỏng chân thực giúp hạn chế xây dựng các nền tảng phần cứng đắt tiền, cung cấp khối lượng dữ liệu lớn, khả năng thử nghiệm, đánh giá kết quả cũng như mở rộng nghiên cứu ít tốn nguồn lực hơn.



Hình 4.2 Hình ảnh mô phỏng AirSim : môi trường

AirSim sẽ mô phỏng môi trường đô thị dưới dạng 3D khá chi tiết với nhiều điều kiện đa dạng, bao gồm đèn giao thông, công viên, hồ và các công trình xây dựng. Người dùng có thể kiểm tra hệ thống dưới dạng khu phố, ngoại ô hay các khu công nghiệp. Mô phỏng môi trường sẽ gồm hơn 12km đường, băng qua khoảng 20 dãy phố.



Hình 4.3 Hình ảnh mô phỏng AirSim

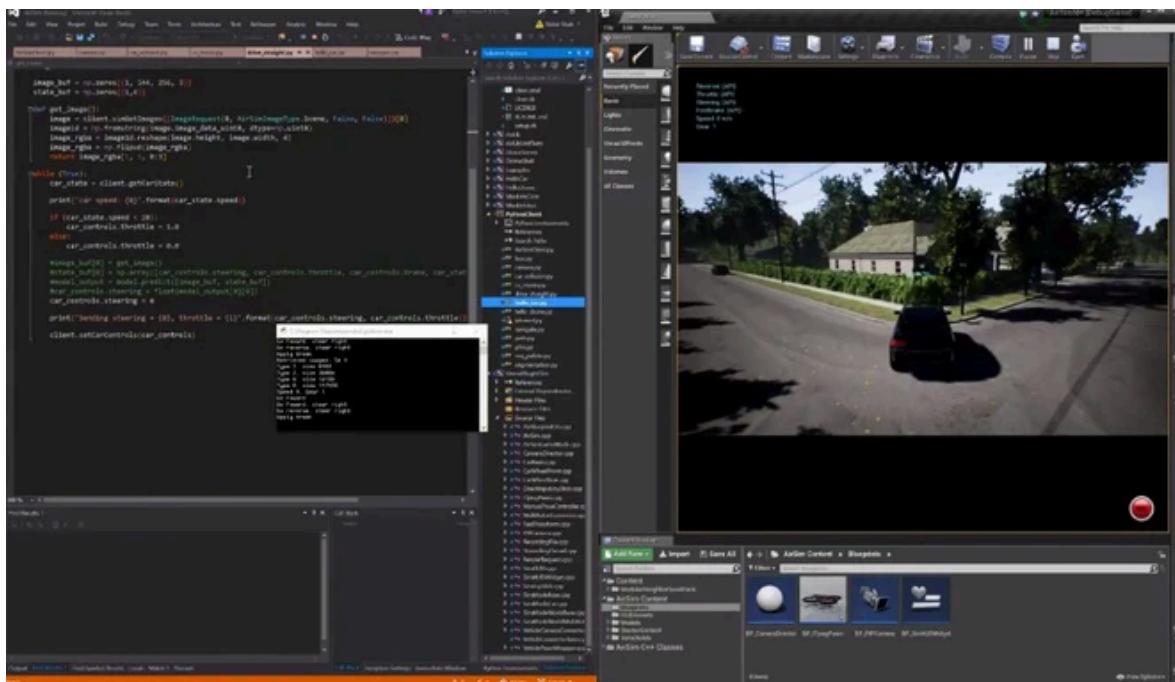
AirSim được phát triển giống như một plugin cho Unreal Engine, một công cụ phổ biến để phát triển game. Điều này có nghĩa, mô phỏng của xe sẽ tách rời với môi trường đang vận hành.

Người dùng có thể tạo ra một môi trường cụ thể, chẳng hạn đường ở thành phố hoặc nông thôn, và sau đó chỉ cần đưa vào plugin AirSim để kiểm tra thuật toán tự lái hoạt động ra sao trong các môi trường đó. Việc mở rộng AirSim cũng cho phép các nhà phát triển có thể kết hợp thêm cảm biến hoặc các loại động cơ mới.



Hình 4.4 Hình ảnh mô phỏng AirSim

AirSim sẽ cung cấp API hỗ trợ nhiều loại ngôn ngữ máy tính phổ biến như C hay Python. Điều này giúp AirSim trở nên thân thiện hơn các công cụ máy học khác.



Hình 4.5 Hình ảnh mô phỏng AirSim : API

Phiên bản AirSim mới cũng bao gồm nhiều tính năng và cải tiến, ví dụ công cụ thử nghiệm phương tiện vận tải hàng không. Microsoft sẽ giúp mọi người tham gia vào các mô phỏng drone bằng cách thêm một bộ điều khiển tích hợp có tên simple_flight, qua đó giúp đơn giản hóa quá trình thiết lập.

Phiên bản mới nhất của AirSim hiện đã có sẵn trên nền tảng GitHub dưới dạng mã nguồn mở.

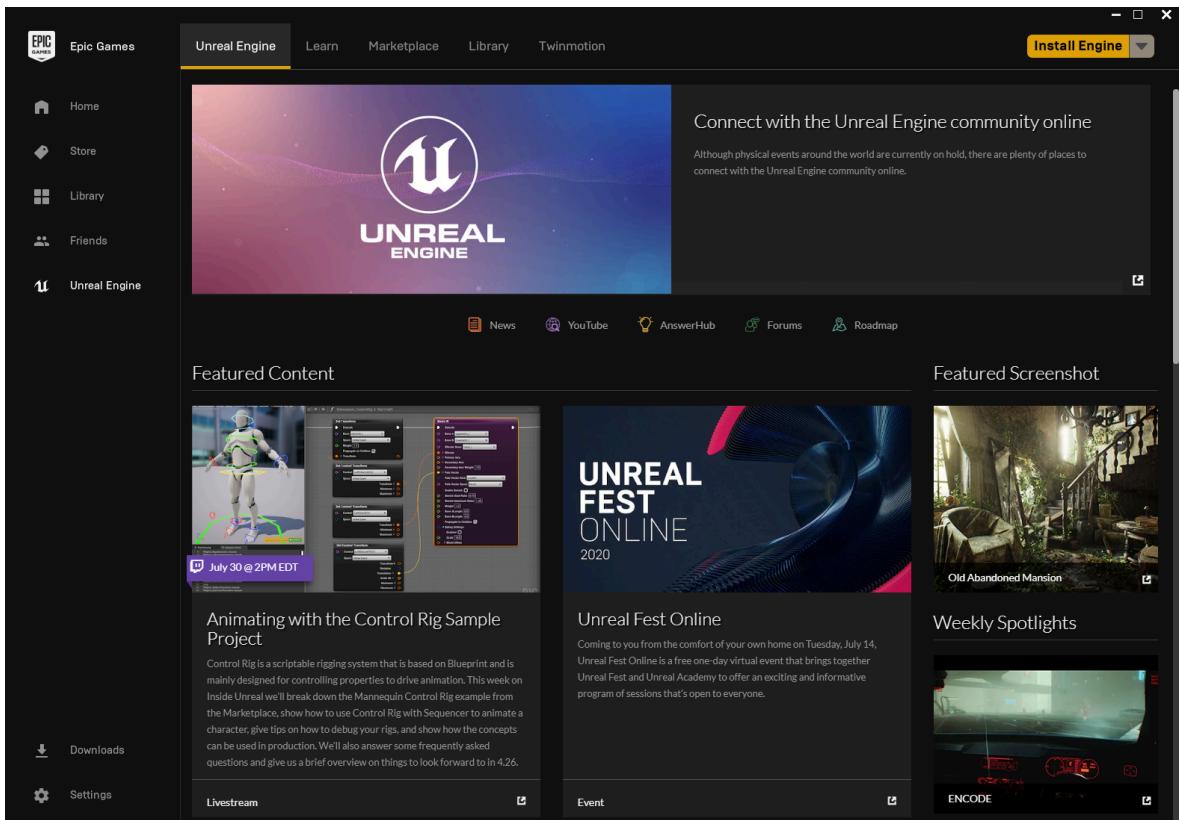
Trong bản phát hành thương mại tương lai, Microsoft hy vọng sẽ kịp bổ sung thêm nhiều cảm biến, hoặc các thay đổi vật lý phù hợp cho xe, mô hình hóa thời tiết hay tạo ra các môi trường thực tế mang tính chi tiết cao hơn.

Xây dựng AirSim trên Windows

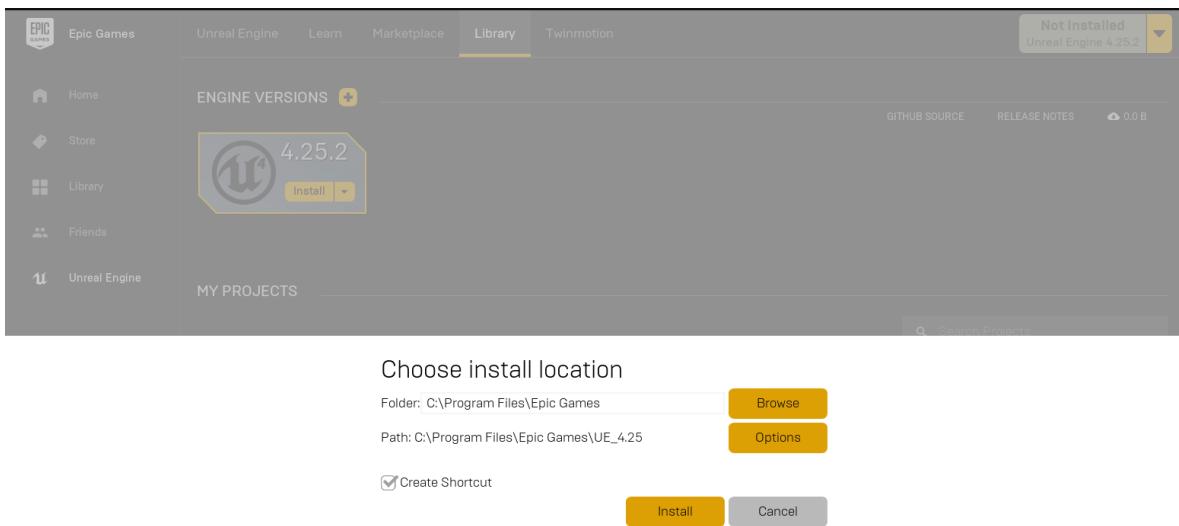
- Bước 1 : Cài đặt Unreal Engine

Tải xuống Epic Games Launcher. Mặc dù Unreal Engine là nguồn mở và tải xuống miễn phí nhưng vẫn cần phải đăng ký.

Chạy Epic Games Launcher, mở tab Unreal Engine ở khung bên trái. Nhấp vào nút Cài đặt ở trên cùng bên phải, nút này sẽ hiển thị tùy chọn tải xuống Unreal Engine ≥ 4.27 . Chọn vị trí cài đặt phù hợp với nhu cầu của bạn, như trong hình bên dưới. Nếu bạn đã cài đặt nhiều phiên bản Unreal thì hãy đảm bảo phiên bản bạn đang sử dụng được đặt thành phiên bản hiện tại bằng cách nhấp vào mũi tên xuống bên cạnh nút Khởi chạy cho phiên bản đó.



Hình 4.6 Hình ảnh ứng dụng Epic Games Launcher



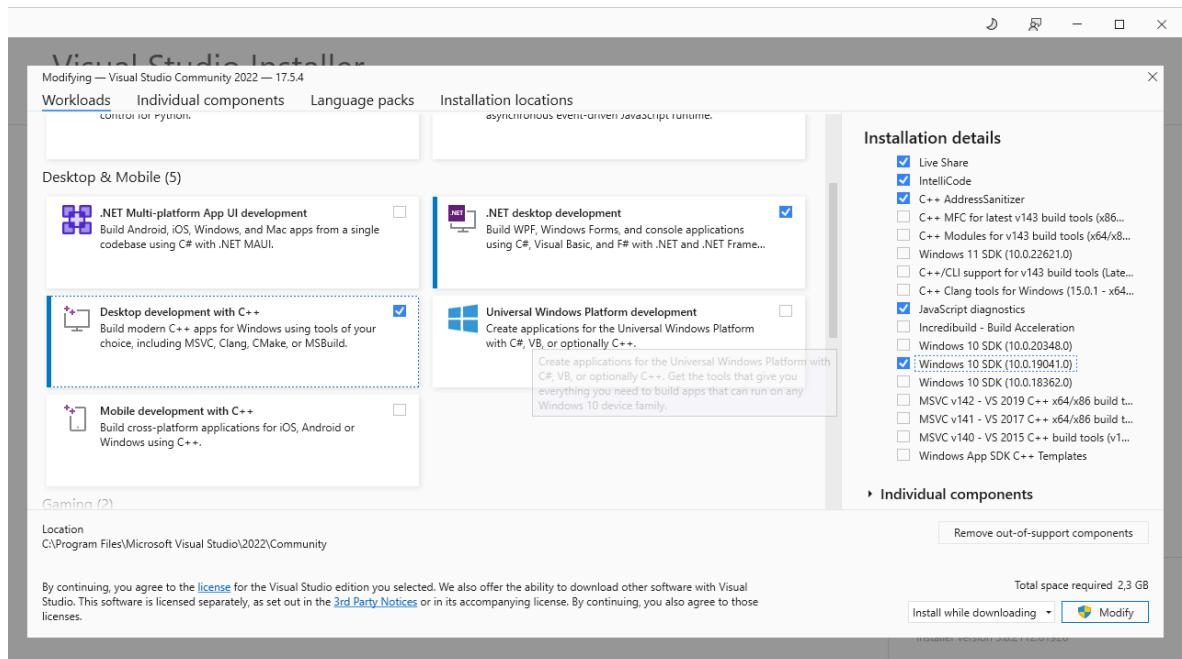
Hình 4.7 Hình ảnh ứng dụng Epic Games Launcher : Cài đặt Unreal Engine

Lưu ý: Nếu bạn có dự án UE 4.16 trở lên, vui lòng xem hướng dẫn nâng cấp để nâng cấp dự án của bạn.

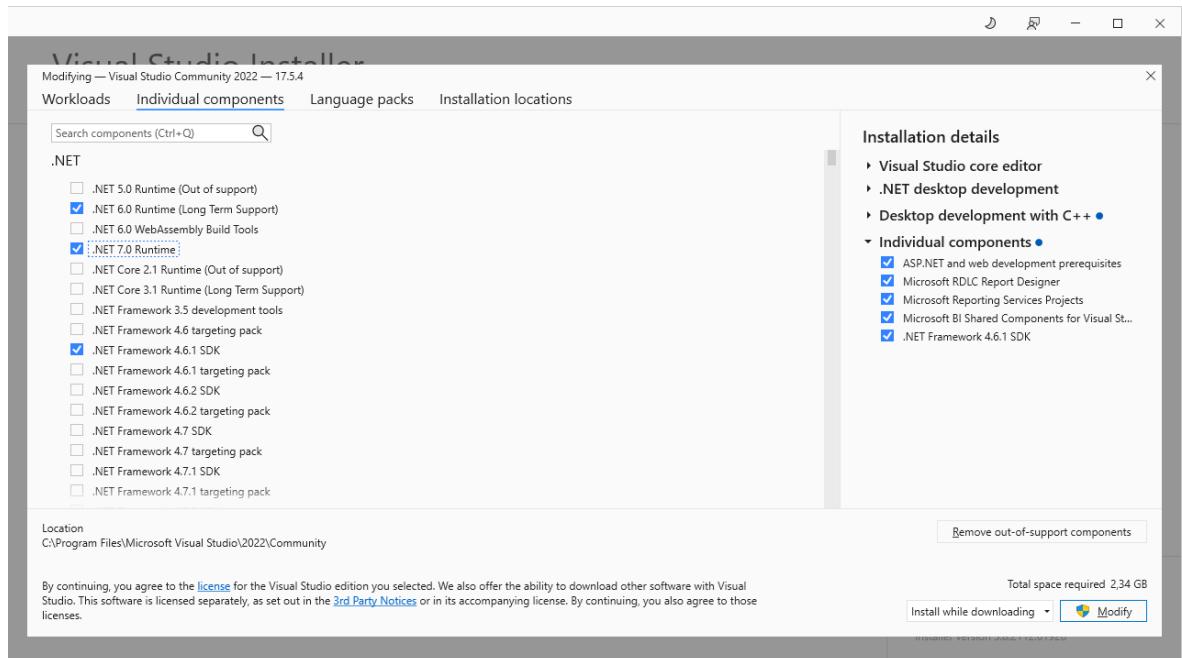
- Bước 2 : Xây dựng AirSim

Cài đặt Visual Studio 2022. Đảm bảo chọn Desktop Development with C++ và Windows 10 SDK 10.0.19041 (nên chọn theo mặc định) và chọn .NET

Framework SDK mới nhất trong tab 'Individual Components' trong khi cài đặt VS 2022.



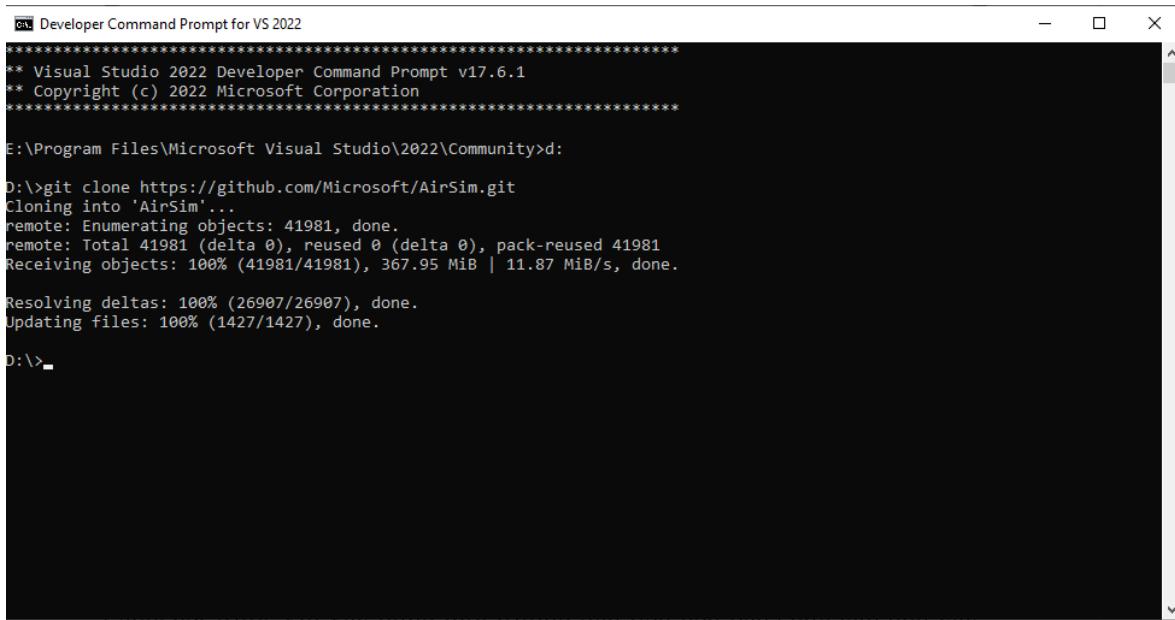
Hình 4.8 Hình ảnh ứng dụng Visual Code 2022 : Installation Workloads



Hình 4.9 Hình ảnh ứng dụng Visual Code 2022 : Installation Individual components

Khởi chạy Developer Command Prompt for VS 2022

Sao chép repo: `git clone https://github.com/Microsoft/AirSim.git` và truy cập thư mục AirSim bằng `cd AirSim`.



```

Developer Command Prompt for VS 2022
=====
** Visual Studio 2022 Developer Command Prompt v17.6.1
** Copyright (c) 2022 Microsoft Corporation
=====

E:\Program Files\Microsoft Visual Studio\2022\Community>d:
D:\>git clone https://github.com/Microsoft/AirSim.git
Cloning into 'AirSim'...
remote: Enumerating objects: 41981, done.
remote: Total 41981 (delta 0), reused 0 (delta 0), pack-reused 41981
Receiving objects: 100% (41981/41981), 367.95 MiB | 11.87 MiB/s, done.

Resolving deltas: 100% (26907/26907), done.
Updating files: 100% (1427/1427), done.

D:\>

```

Hình 4.10 Hình ảnh cài đặt AirSim trên Developer Command Prompt for VS 2022

Lưu ý: Nói chung không nên cài đặt AirSim trong ổ C. Điều này có thể khiến tập lệnh bị lỗi và yêu cầu chạy VS ở chế độ Quản trị viên. Thay vào đó hãy sao chép vào một ổ đĩa khác như D hoặc E.

Chạy *build.cmd* từ dòng lệnh. Điều này sẽ tạo ra các bit plugin sẵn sàng để sử dụng trong thư mục Unreal\Plugins có thể được thả vào bất kỳ dự án Unreal nào.

Bước 3 : Xây dựng Unreal Project

Cuối cùng, bạn sẽ cần một dự án Unreal để tạo môi trường cho phương tiện của bạn. Đảm bảo đóng và mở lại Unreal Engine và Epic Games Launcher trước khi xây dựng môi trường đầu tiên nếu bạn chưa làm như vậy. Sau khi khởi động lại Epic Games Launcher, nó sẽ yêu cầu bạn liên kết các phần mở rộng tệp dự án với Unreal Engine, hãy nhấp vào 'fix now' để sửa. AirSim đi kèm với "Blocks Environment" tích hợp sẵn mà bạn có thể sử dụng hoặc bạn có thể tạo môi trường của riêng mình.

4.3 Giới thiệu về Px4-autopilot

PX4 là phần mềm điều khiển chuyến bay nguồn mở mạnh mẽ và linh hoạt dành cho máy bay không người lái (UAV) và các phương tiện tự hành khác. Hệ thống này cung cấp một bộ công cụ toàn diện cho các nhà phát triển UAV nhằm chia sẻ công nghệ và tạo ra các giải pháp phù hợp cho nhiều ứng dụng đa dạng.



Hình 4.11 Hình ảnh phần mềm PX4-autopilot

Điểm nổi bật của PX4:

- Tiêu chuẩn hóa: PX4 thiết lập tiêu chuẩn cho hỗ trợ phần cứng và ngăn xếp phần mềm UAV, tạo nền tảng cho hệ sinh thái phát triển phần cứng và phần mềm mở rộng và bền vững.
- Tính mô-đun cao: Kiến trúc dựa trên cổng của PX4 cho phép mở rộng hệ thống dễ dàng mà không ảnh hưởng đến hiệu suất hoặc độ ổn định. Các nhà phát triển có thể tích hợp linh hoạt các thành phần mới, đáp ứng nhu cầu cụ thể của từng dự án.
- Phát triển cộng đồng: PX4 được đồng phát triển bởi cộng đồng toàn cầu, đảm bảo hệ thống đáp ứng nhu cầu thực tế và liên tục được cải tiến.
- Dễ sử dụng: PX4 cung cấp API và SDK tối ưu hóa cho các nhà phát triển, giúp họ dễ dàng tích hợp các tiện ích và triển khai các tính năng mới.
- Hỗ trợ thị giác máy tính: PX4 được thiết kế để tích hợp chặt chẽ với thị giác máy tính nhúng, mở ra tiềm năng to lớn cho các ứng dụng tự hành.
- Độ tin cậy cao: Hàng nghìn hệ thống dựa trên PX4 đã được triển khai thành công trên toàn thế giới, trải qua quá trình thử nghiệm nghiêm ngặt để đảm bảo độ an toàn và độ tin cậy.
- Miễn phí và nguồn mở: PX4 được cấp phép miễn phí theo Giấy phép BSD 3 Điều khoản, cho phép sử dụng cho cả mục đích thương mại và phi thương mại.
- Hệ sinh thái phong phú: PX4 đi kèm với hệ sinh thái các thiết bị được hỗ trợ và các tiêu chuẩn cho truyền thông, tích hợp thiết bị ngoại vi và quản lý năng lượng.

- Tính năng an toàn: PX4 tích hợp các tính năng an toàn tiên tiến như hành vi an toàn tự động, hỗ trợ các chế độ quay trở lại và nhiều tính năng khác, đảm bảo an toàn cho UAV trong quá trình vận hành.

Tại sao nên chọn PX4?

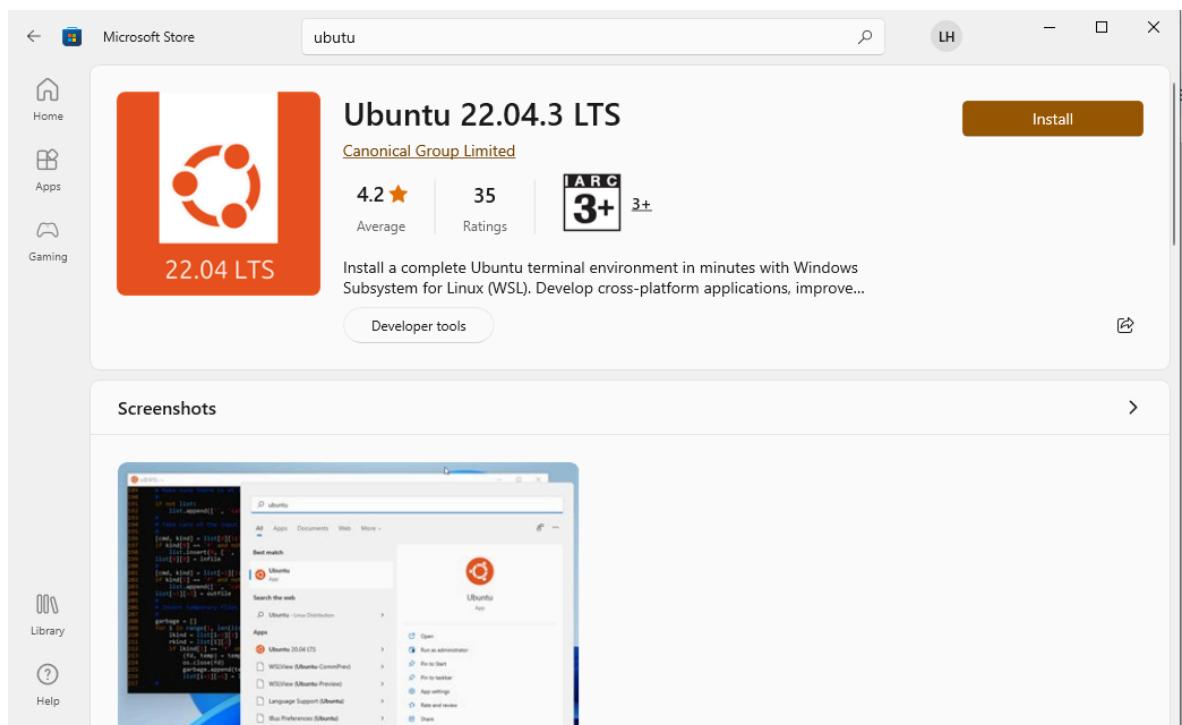
PX4 là lựa chọn lý tưởng cho các nhà phát triển UAV và phương tiện tự hành bởi những ưu điểm vượt trội sau:

- Mở rộng và linh hoạt: PX4 đáp ứng mọi nhu cầu, từ các dự án nghiên cứu nhỏ đến các ứng dụng thương mại quy mô lớn.
- Cộng đồng hỗ trợ mạnh mẽ: Người dùng có thể dễ dàng tiếp cận nguồn tài liệu phong phú và cộng đồng hỗ trợ nhiệt tình.
- Tiết kiệm chi phí: PX4 miễn phí và nguồn mở, giúp giảm chi phí phát triển và vận hành.
- Độ tin cậy cao: PX4 đã được kiểm chứng qua nhiều thử nghiệm thực tế, đảm bảo an toàn và hiệu suất cho hệ thống.
- Tương lai phát triển: PX4 được cập nhật liên tục với các tính năng mới và cải tiến, đảm bảo hệ thống luôn đáp ứng nhu cầu thị trường.

Như vậy ta có thể thấy PX4 là hệ thống điều khiển bay nguồn mở hàng đầu dành cho UAV và phương tiện tự hành. Với tính năng đa dạng, hiệu suất cao và cộng đồng hỗ trợ mạnh mẽ, PX4 là lựa chọn tối ưu cho các nhà phát triển muốn tạo ra những giải pháp sáng tạo và đột phá trong lĩnh vực UAV.

Xây dựng PX4 trên Windows

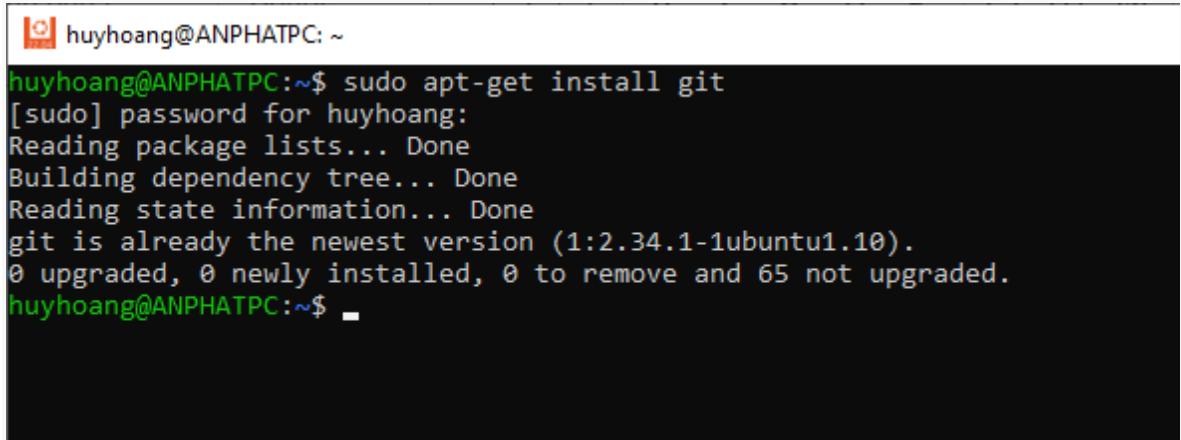
Bước 1: Cài đặt Ubuntu trên Microsoft Store



Hình 4.12 Hình ảnh cài đặt Ubuntu

Bước 2: Xây dựng PX4

Truy cập vào ứng dụng Ubuntu, sau khi thiết lập User Admin tiến hành install git : `sudo apt-get install git`



```
[!] huyhoang@ANPHATPC: ~
huyhoang@ANPHATPC:~$ sudo apt-get install git
[sudo] password for huyhoang:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.34.1-1ubuntu1.10).
0 upgraded, 0 newly installed, 0 to remove and 65 not upgraded.
huyhoang@ANPHATPC:~$ -
```

Tiến hành clone PX4 trên github về :

```
git clone https://github.com/PX4/PX4-Autopilot.git --recursive
```

```
Administrator: ~ % git clone https://github.com/PX4/PX4-Autopilot.git --recursive
Cloning into 'PX4-Autopilot'...
remote: Enumerating objects: 458502, done.
remote: Counting objects: 100% (458502/458502), done.
remote: Compressing objects: 100% (810/810), done.
remote: Total 458502 (delta 999), reused 100% (delta 642), pack-reused 457032
Receiving objects: 100% (458502/458502) | 217.68 MiB / 2.22 MiB/s, done.
   100% (33784/33784) | 1.40 MiB / 1.40 MiB/s, done.
Submodule 'Tools/flightgear_bridge' (https://github.com/PX4/PX4-FlightGear-Bridge.git) registered for path 'Tools/simulation/flightgear_bridge'
Submodule 'Tools/simulation/gazebo-classic/sitl_gazebo-classic' (https://github.com/PX4/PX4-SITL_gazebo-classic.git) registered for path 'Tools/simulation/gazebo-classic/sitl_gazebo-classic'
Submodule 'Tools/simulation/jmavsim/jMavSim' (https://github.com/PX4/PX4-gazebo-models.git) registered for path 'Tools/simulation/gz/jMavSim'
Submodule 'jMavSim' (https://github.com/PX4/jMavSim)
Submodule 'Tools/simulation/jbsim/jbsim_bridge' (https://github.com/PX4/pix/jbsim_bridge.git) registered for path 'Tools/simulation/jbsim/jbsim_bridge'
Submodule 'Tools/simulation/nuttx/nuttx' (https://github.com/PX4/nuttx.git) registered for path 'Tools/simulation/nuttx'
Submodule 'platforms/nuttx/Nuttx' (https://github.com/PX4/Nuttx.git) registered for path 'platforms/nuttx'
Submodule 'src/drivers/nuttx/nuttx' (https://github.com/PX4/Nuttx.git) registered for path 'src/drivers/nuttx/nuttx'
Submodule 'src/drivers/cyphal/legacy_data_types' (https://github.com/PX4/cyphal/legacy_data_types.git) registered for path 'src/drivers/cyphal/legacy_data_types'
Submodule 'src/drivers/cyphal/public_regulated_data_types' (https://github.com/opencyphal/public_regulated_data_types.git) registered for path 'src/drivers/cyphal/public_regulated_data_types'
Submodule 'src/drivers/gps/devices' (https://github.com/PX4/GPSDevices.git) registered for path 'src/drivers/gps/devices'
Submodule 'src/modules/mavlink/mavlink' (https://github.com/PX4/mavlink.git) registered for path 'src/modules/mavlink/mavlink'
Submodule 'src/lib/cdrstream/cyclonedd' (https://github.com/pix/cyclonedd) registered for path 'src/lib/cdrstream/cyclonedd'
Submodule 'src/lib/cdrstream/rosidl' (https://github.com/pix/rosidl) registered for path 'src/lib/cdrstream/rosidl'
Submodule 'src/lib/crypto/libmbedtls' (https://github.com/PX4/libmbedtls.git) registered for path 'src/lib/crypto/libmbedtls'
Submodule 'src/lib/crypto/libmbedtlscrypto' (https://github.com/PX4/libmbedtlscrypto.git) registered for path 'src/lib/crypto/libmbedtlscrypto'
Submodule 'src/lib/crypto/monocypher' (https://github.com/PX4/monocypher.git) registered for path 'src/lib/crypto/monocypher'
Submodule 'src/lib/events/libevents' (https://github.com/mavlink/libevents.git) registered for path 'src/lib/events/libevents'
Submodule 'src/lib/libc' (https://github.com/PX4/libc.git) registered for path 'src/lib/libc'
Submodule 'src/modules/mavlink/mavlink' (https://github.com/mavlink/mavlink.git) registered for path 'src/modules/mavlink/mavlink'
Submodule 'src/modules/zenoh/zenoh-pico' (https://github.com/PX4/zenoh-pico.git) registered for path 'src/modules/zenoh/zenoh-pico'
Cloning into '/home/muyhaog/PX4-Autopilot/tools/simulation/flightgear_bridge'...
remote: Enumerating objects: 382, done.
remote: Counting objects: 100% (37/37), done.
remote: Compressing objects: 100% (37/37), done.
remote: Total 382 (delta 25), reused 36 (delta 22), pack-reused 336
Receiving objects: 100% (382/382), 1.47 MiB | 4.00 MiB/s, done.
   100% (382/382) | 1.47 MiB / 1.47 MiB, done.
Cloning into '/home/muyhaog/PX4-Autopilot/tools/simulation/gazebo-classic/sitl_gazebo-classic'...
remote: Enumerating objects: 9511, done.
remote: Counting objects: 100% (57/57), done.
remote: Compressing objects: 100% (57/57), done.
remote: Total 9511 (delta 188), reused 213 (delta 105), pack-reused 9167
Receiving objects: 100% (9511/9511), 193.56 MiB | 8.55 MiB/s, done.
   100% (9511/9511) | 193.56 MiB / 193.56 MiB, done.
Cloning into '/home/muyhaog/PX4-Autopilot/tools/simulation/gz'...
remote: Enumerating objects: 387, done.
remote: Counting objects: 100% (121/121), done.
remote: Compressing objects: 100% (75/75), done.
remote: Total 387 (delta 77), reused 40 (delta 46), pack-reused 266
Receiving objects: 100% (387/387), 12.35 MiB | 11.40 MiB/s, done.
   100% (387/387) | 12.35 MiB / 12.35 MiB, done.
Cloning into '/home/muyhaog/PX4-Autopilot/tools/simulation/jmavsim/jMavSim'...
remote: Enumerating objects: 3357, done.
remote: Counting objects: 100% (3357/3357), done.
remote: Compressing objects: 100% (67/67), done.
Receiving objects: 100% (3357/3357), 77.08 MiB | 6.44 MiB/s
   100% (3357/3357) | 77.08 MiB / 77.08 MiB, done.

Administrator: ~ %
```

Tiếp tục chạy setup cho hệ thống PX4 :

```
bash ./PX4-Autopilot/Tools/setup/ubuntu.sh --no-sim-tools
```

```
[huyhoang@ANPHATPC: ~]
huyhoang@ANPHATPC:~$ bash ./PX4-Autopilot/Tools/setup/ubuntu.sh --no-sim-tools
Ubuntu 22.04

Installing PX4 general dependencies
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:2 http://packages.osrfoundation.org/gazebo/ubuntu-stable jammy InRelease [4276 B]
Get:3 http://packages.osrfoundation.org/gazebo/ubuntu-stable jammy/main amd64 Packages [109 kB]
Hit:4 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:5 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:7 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [27.2 kB]
Reading package lists...
E: Release file for http://security.ubuntu.com/ubuntu/dists/jammy-security/InRelease is not valid yet
E: Release file for http://archive.ubuntu.com/ubuntu/dists/jammy-updates/InRelease is not valid yet
huyhoang@ANPHATPC:~$
```

Sau quá trình setup xong chúng ta sẽ restart lại Ubuntu :

exit

Vào lại Ubuntu : `cd PX4-Autopilot`

```
make px4_sitl jmavsim
```

```
huyhoang@ANPHATPC:~/PX4-Autopilot$  
huyhoang@ANPHATPC:~/PX4-Autopilot$ cd PX4-Autopilot/  
huyhoang@ANPHATPC:~/PX4-Autopilot$ make px4_sitl jmavsim  
-- PX4 version: v1.15.0-beta1-90-g253208fd4 (1.15.0)  
-- Found PythonInterp: /usr/bin/python3 (found suitable version "3.10.12", minimum required is "3")  
-- PX4 config file: /home/huyhoang/PX4-Autopilot/boards/px4/sitl/default.px4board  
-- PLATFORM posix  
-- ROMFSROOT px4fmu_common  
-- ROOTFSDIR .  
-- TESTING y  
-- ETHERNET y  
-- PX4 config: px4_sitl_default  
-- PX4 platform: posix  
-- PX4 lockstep: enabled  
-- The CXX compiler identification is GNU 11.4.0  
-- The C compiler identification is GNU 11.4.0  
-- The ASM compiler identification is GNU  
-- Found assembler: /usr/bin/cc  
-- Detecting CXX compiler ABI info  
-- Detecting CXX compiler ABI info - done  
-- Check for working CXX compiler: /usr/bin/c++ - skipped  
-- Detecting CXX compile features  
-- Detecting CXX compile features - done  
-- Detecting C compiler ABI info  
-- Detecting C compiler ABI info - done  
-- Check for working C compiler: /usr/bin/cc - skipped  
-- Detecting C compile features  
-- Detecting C compile features - done  
-- cmake build type: RelWithDebInfo  
/usr/bin/python3: Error while finding module specification for 'symforce.symbolic' (ModuleNotFoundError: No module named 'symforce')  
-- Could NOT find gz-transport (missing: gz-transport_DIR)  
-- Looking for gz-transport12 -- found version 12.2.1  
-- Searching for dependencies of gz-transport12  
-- Found Protobuf: /usr/lib/x86_64-linux-gnu/libprotobuf.so (found version "3.12.4")  
-- Config-file not installed for ZeroMQ -- checking for pkg-config  
-- Checking for module 'libzmq' >= 4'  
-- Found libzmq , version 4.3.4  
-- Found ZeroMQ: TRUE (Required is at least version "4")  
-- Checking for module 'uuid'  
-- Found uuid, version 2.37.2  
-- Found UUID: TRUE  
-- Looking for gz-utils2 -- found version 2.2.0  
-- Searching for dependencies of gz-utils2  
-- Searching for <gz-utils2> component [cli]  
-- Looking for gz-utils2-cli -- found version 2.2.0  
-- Searching for dependencies of gz-utils2cli  
-- Looking for gz-msgs9 -- found version 9.5.0  
-- Searching for dependencies of gz-msgs9  
-- Looking for gz-math7 -- found version 7.4.0  
-- Searching for dependencies of gz-math7  
-- Looking for gz-utils2 -- found version 2.2.0  
-- Checking for module 'tinyxml2'  
-- Found tinyxml2, version 9.0.0  
-- Found Java: /usr/bin/java (found version "11.0.22")  
-- ROMFS: ROMFS/px4fmu_common  
Architecture: amd64  
--> CPACK_INSTALL_PREFIX = @DEB_INSTALL_PREFIX@  
-- Configuring done  
-- Generating done  
-- Build files have been written to: /home/huyhoang/PX4-Autopilot/build/px4_sitl_default
```

```

huyhoang@ANPHATPC: ~/PX4-Autopilot
compile:
[javac] Compiling 76 source files to /home/huyhoang/PX4-Autopilot/Tools/simulation/jmavsim/jMAVSIM/out

create_run_jar:
[jar] Building jar: /home/huyhoang/PX4-Autopilot/Tools/simulation/jmavsim/jMAVSIM/out/product.jar

copy_res:
[copy] Copying 22 files to /home/huyhoang/PX4-Autopilot/Tools/simulation/jmavsim/jMAVSIM/out/resource

BUILD SUCCESSFUL
Total time: 4 seconds
[989/990] launching px4 jmavsim_iris (SYS_AUTOSTART=10017)

  _/\_ \_\_ /_/
 |  / \ V / / / |
 | / / \ \ / \_ |
 | \ / \ \ \ \_ |
 | \ \ \ \ \_ |

px4 starting.

INFO [px4] startup script: /bin/sh etc/init.d-posix/rcS 0
env SYS_AUTOSTART: 10017
INFO [param] selected parameter default file parameters.bson
INFO [param] selected parameter backup file parameters_backup.bson
  SYS_AUTOCONFIG: curr: 0 -> new: 1
  SYS_AUTOSTART: curr: 0 -> new: 10017
  CAL_ACC0_ID: curr: 0 -> new: 1310988
  CAL_GYRO0_ID: curr: 0 -> new: 1310988
  CAL_ACC1_ID: curr: 0 -> new: 1310996
  CAL_GYRO1_ID: curr: 0 -> new: 1310996
  CAL_ACC2_ID: curr: 0 -> new: 1311004
  CAL_GYRO2_ID: curr: 0 -> new: 1311004
  CAL_MAG0_ID: curr: 0 -> new: 197388
  CAL_MAG0_PRIO: curr: -1 -> new: 50
  CAL_MAG1_ID: curr: 0 -> new: 197644
  CAL_MAG1_PRIO: curr: -1 -> new: 50
  SENS_BOARD_X_OFFSET: curr: 0.0000 -> new: 0.0000
  SENS_DPREC_OFFSET: curr: 0.0000 -> new: 0.0010
INFO [dataman] data manager file './dataman' size is 7872608 bytes
INFO [init] jMAVSIM simulator
INFO [simulator_mavlink] using TCP on remote host localhost port 4560
WARN [simulator_mavlink] Please ensure port 4560 is not blocked by a firewall.
INFO [simulator_mavlink] Resolved host 'localhost' to address: 127.0.0.1
INFO [simulator_mavlink] Waiting for simulator to accept connection on TCP port 4560
Buildfile: /home/huyhoang/PX4-Autopilot/Tools/simulation/jmavsim/jMAVSIM/build.xml

make_dirs:
compile:
create_run_jar:
copy_res:

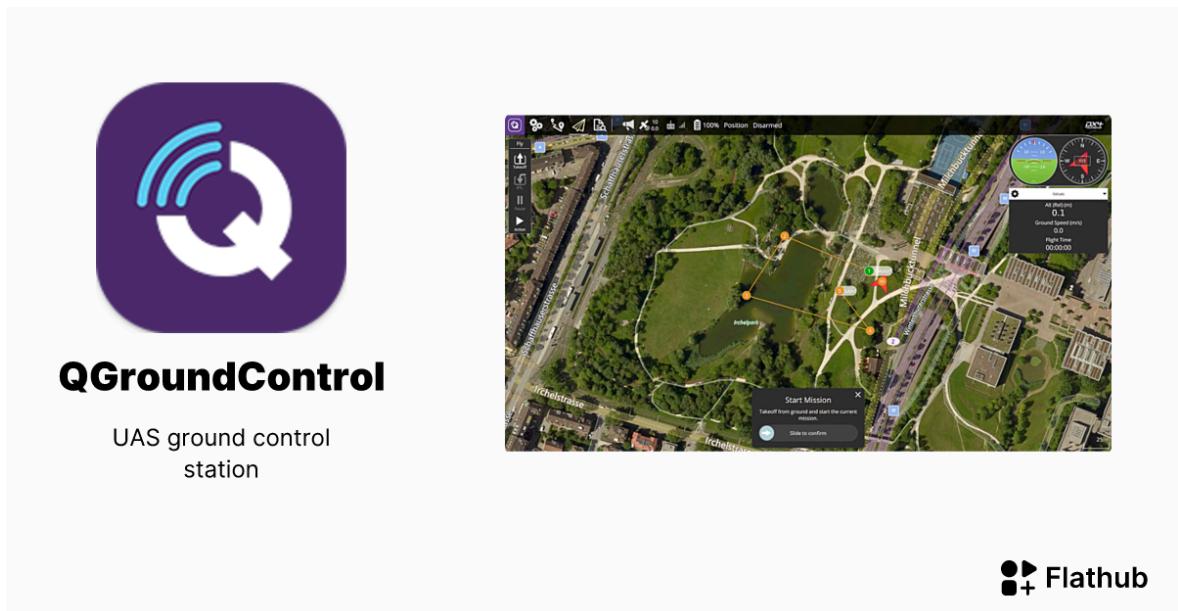
BUILD SUCCESSFUL
Total time: 0 seconds
Options parsed, starting Sim.
Starting GUI...
INFO [simulator_mavlink] Simulator connected on TCP port 4560.
Init MAVLink
INFO [lockstep_scheduler] setting initial absolute time to 1715298598475000 us

```

4.4 Giới thiệu về QgroundControl

QGroundControl cung cấp khả năng điều khiển chuyến bay đầy đủ và thiết lập phuorong tiện cho các phuorong tiện chạy bằng động cơ PX4 hoặc ArduPilot. Nó cung cấp cách sử dụng dễ dàng và đơn giản cho người mới bắt

đầu, đồng thời vẫn cung cấp hỗ trợ tính năng cao cấp cho người dùng có kinh nghiệm.



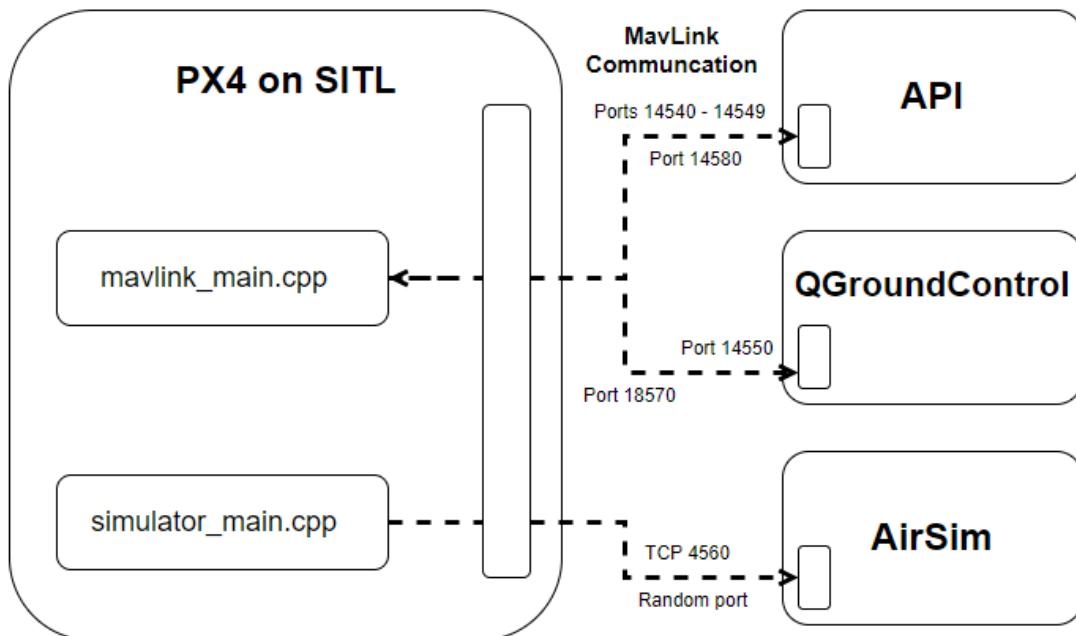
Flathub

Hình 4.13 Hình ảnh phần mềm QGroundControl

Các tính năng chính:

- Thiết lập/cấu hình đầy đủ cho xe chạy bằng ArduPilot và PX4 Pro.
- Hỗ trợ chuyến bay cho các phương tiện chạy PX4 và ArduPilot (hoặc bất kỳ chế độ lái tự động nào khác giao tiếp bằng giao thức MAVLink).
- Lập kế hoạch nhiệm vụ cho chuyến bay tự trị.
- Hiển thị bản đồ chuyến bay hiển thị vị trí phương tiện, đường bay, điểm tham chiếu và các thiết bị của phương tiện.
- Trình xem 3D hiển thị bản đồ 3D của môi trường (tệp .osm), mô hình 3D của phương tiện (hiện chỉ có nhiều cánh quạt) và quỹ đạo 3D của nhiệm vụ (bao gồm cả các điểm tham chiếu).
- Truyền phát video với lớp phủ hiển thị nhạc cụ.
- Hỗ trợ quản lý nhiều phương tiện.
- QGC chạy trên nền tảng Windows, OS X, Linux, thiết bị iOS và Android.

4.5 Phương thức kết nối



Hình 4.14 Hình ảnh phương thức kết nối giữa các ứng dụng

PX4-autopilot, Microsoft AirSim và QGroundControl là ba công cụ quan trọng trong lĩnh vực mô phỏng và điều khiển UAV. Mỗi công cụ có chức năng riêng biệt, nhưng khi kết hợp với nhau, chúng tạo thành hệ thống mô phỏng UAV hoàn chỉnh và hiệu quả. Dưới đây là trình bày chi tiết về phương thức kết nối giữa ba công cụ này:

1. PX4-autopilot:

Chức năng: Là phần mềm điều khiển bay nguồn mở cho UAV, chịu trách nhiệm điều khiển động cơ, cảm biến và các thành phần khác của UAV.

Kết nối với AirSim: PX4 kết nối với AirSim thông qua giao thức MAVLink, cho phép AirSim cung cấp dữ liệu mô phỏng về môi trường xung quanh cho PX4, và PX4 gửi lệnh điều khiển đến AirSim để điều khiển UAV mô phỏng.

2. Microsoft AirSim:

Chức năng: Là trình mô phỏng môi trường thực tế cho UAV, tạo ra môi trường 3D ảo mô phỏng các yếu tố như địa hình, thời tiết, giao thông và các vật thể khác.

Kết nối với PX4: AirSim kết nối với PX4 thông qua giao thức MAVLink. AirSim nhận dữ liệu điều khiển từ PX4 và sử dụng dữ liệu này để điều khiển UAV mô phỏng trong môi trường ảo. AirSim cũng gửi dữ liệu mô

phỏng về môi trường xung quanh cho PX4, chẳng hạn như vị trí, hướng, tốc độ và cảm biến của UAV.

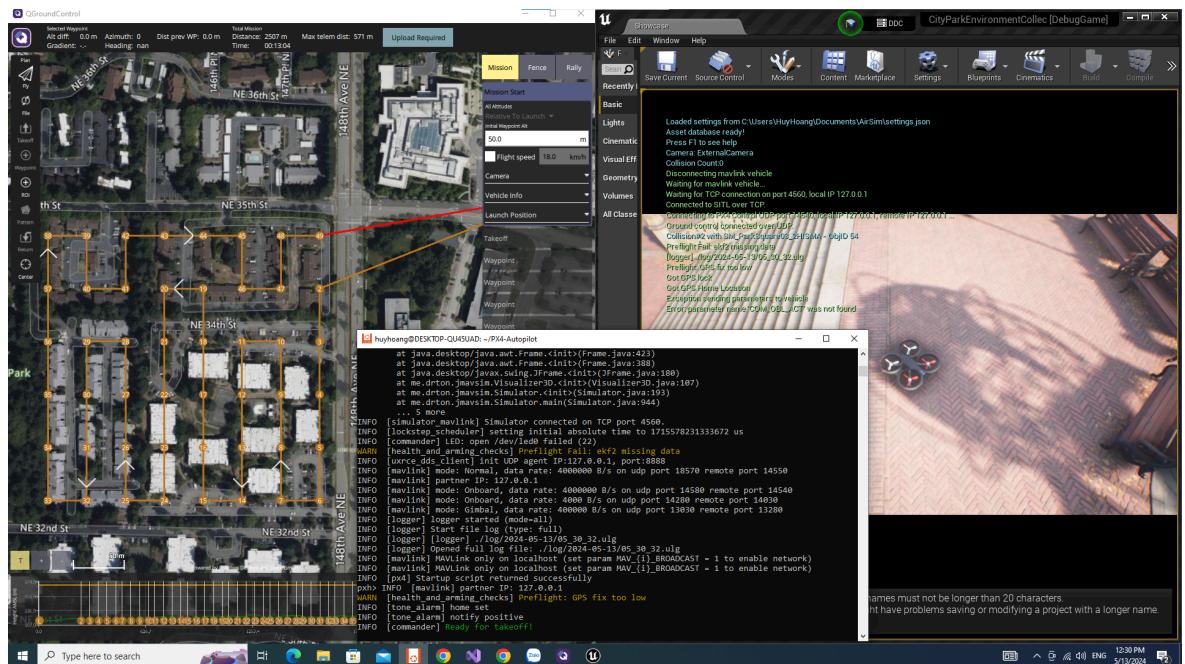
3. QGroundControl:

Chức năng: Là giao diện điều khiển đồ họa cho PX4, cung cấp cho người dùng một môi trường trực quan để theo dõi trạng thái UAV, điều khiển UAV bằng tay, cấu hình các tham số điều khiển và thực hiện các nhiệm vụ bay tự động.

Kết nối với PX4: QGroundControl kết nối với PX4 thông qua giao thức MAVLink. QGroundControl nhận dữ liệu trạng thái từ PX4 và hiển thị thông tin này trên giao diện người dùng. Người dùng có thể sử dụng QGroundControl để gửi lệnh điều khiển đến PX4, cấu hình các tham số điều khiển và thực hiện các nhiệm vụ bay tự động.

4.6 Kết quả chạy mô phỏng

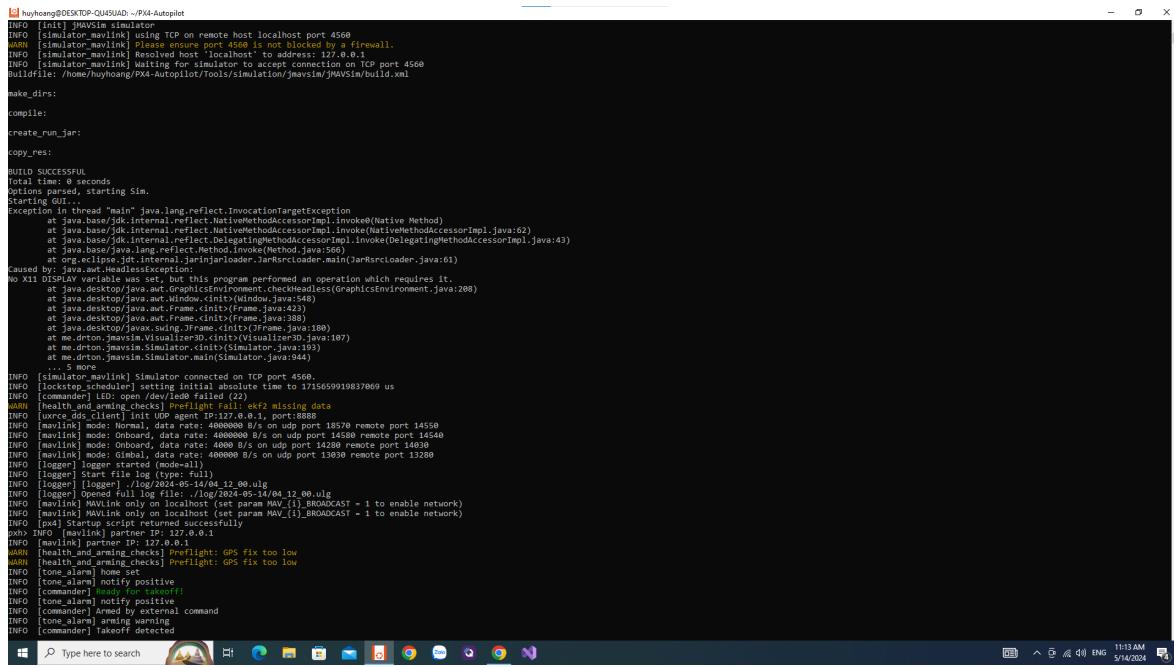
Khi chạy mô phỏng, chúng ta khởi động 3 ứng dụng là PX4, QGroundControl và AirSim chạy trên Unreal Engine lên. Sau khi Play AirSim, 3 ứng dụng sẽ được tự động connect với nhau thông qua các cổng mà chúng đã được cấu hình sẵn.



Tại ứng dụng QGroundControl, chúng ta sẽ mở tập tin Plan đã được lưu sau quá trình giải bài toán, nhằm kiểm tra xem Drone có tuân theo lộ trình đã được thiết lập trước hay không. Sau khi tập tin đã được mở, chúng ta cần click vào Upload Required để cho ứng dụng có thể hiểu và nhận dạng điểm

khởi tạo của Drone. Tiếp theo, để Drone cất cánh, chúng ta nhấn nút Fly - Take Off và quan sát quá trình bay của Drone.

Dưới đây là kết quả mô phỏng bay sử dụng AirSim:

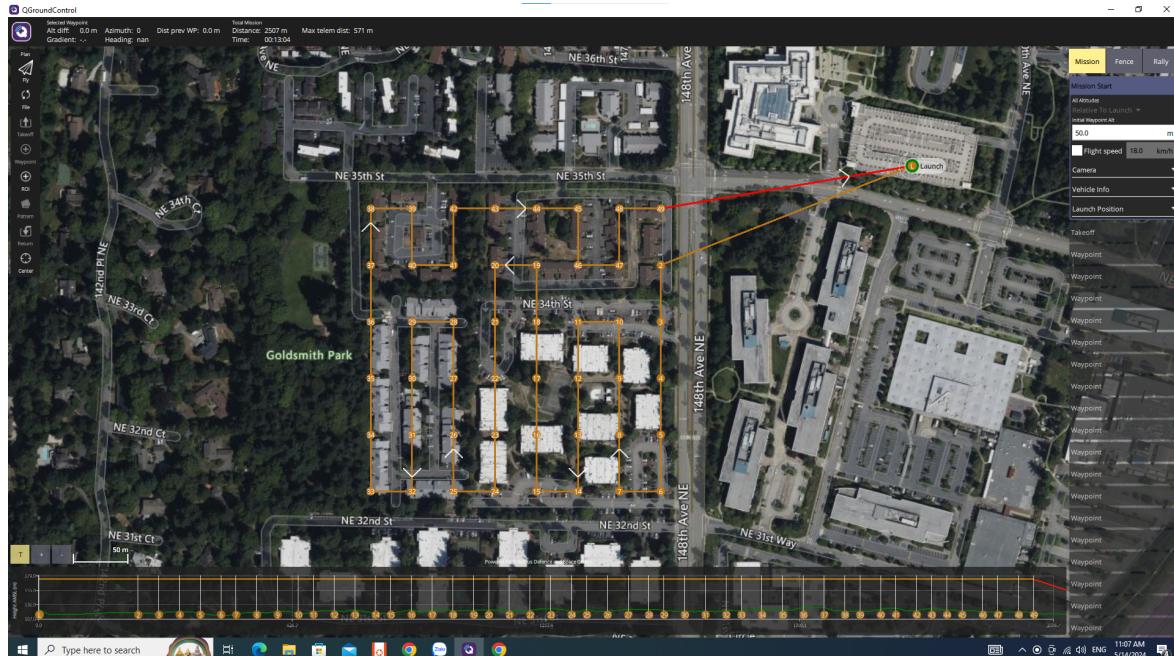


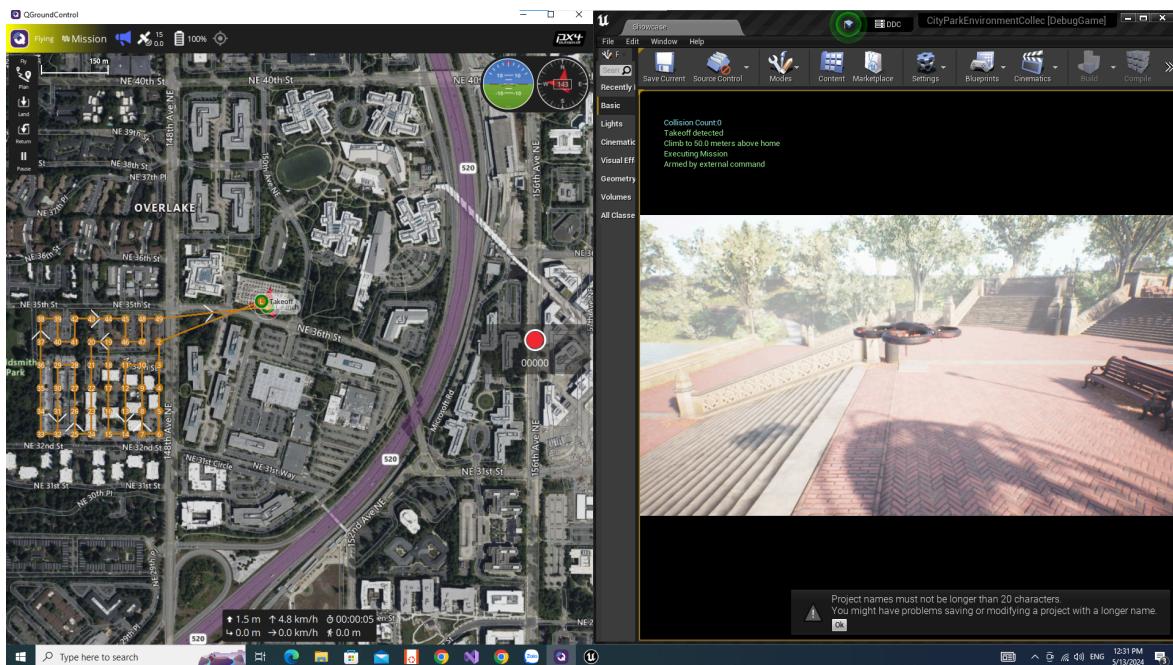
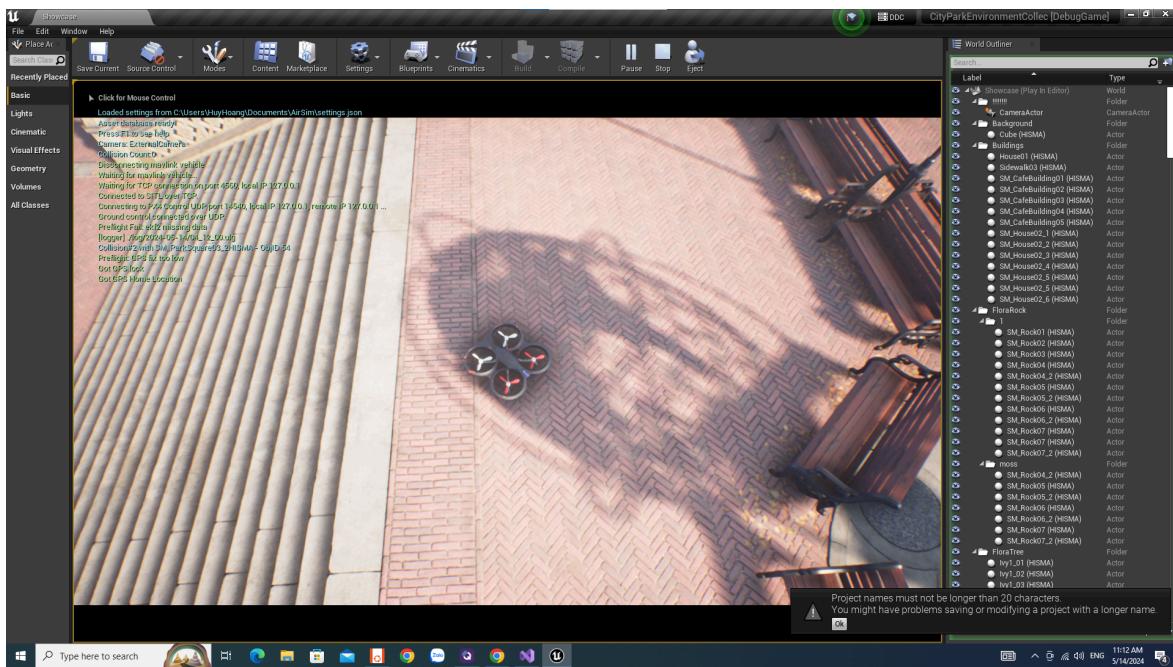
```

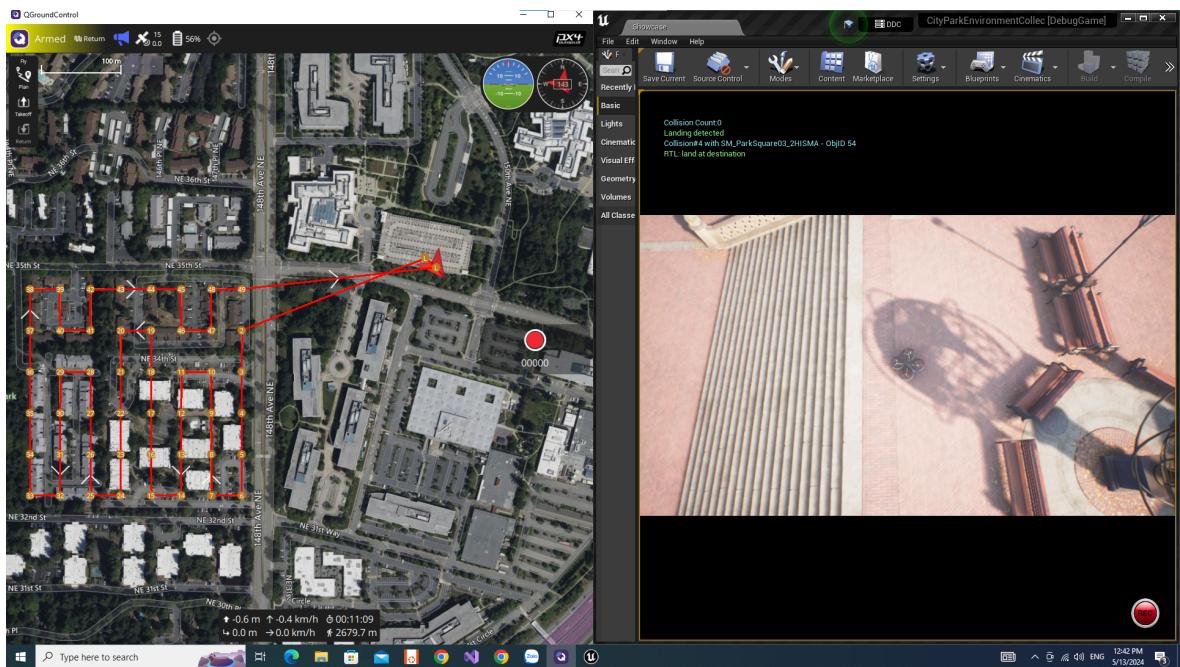
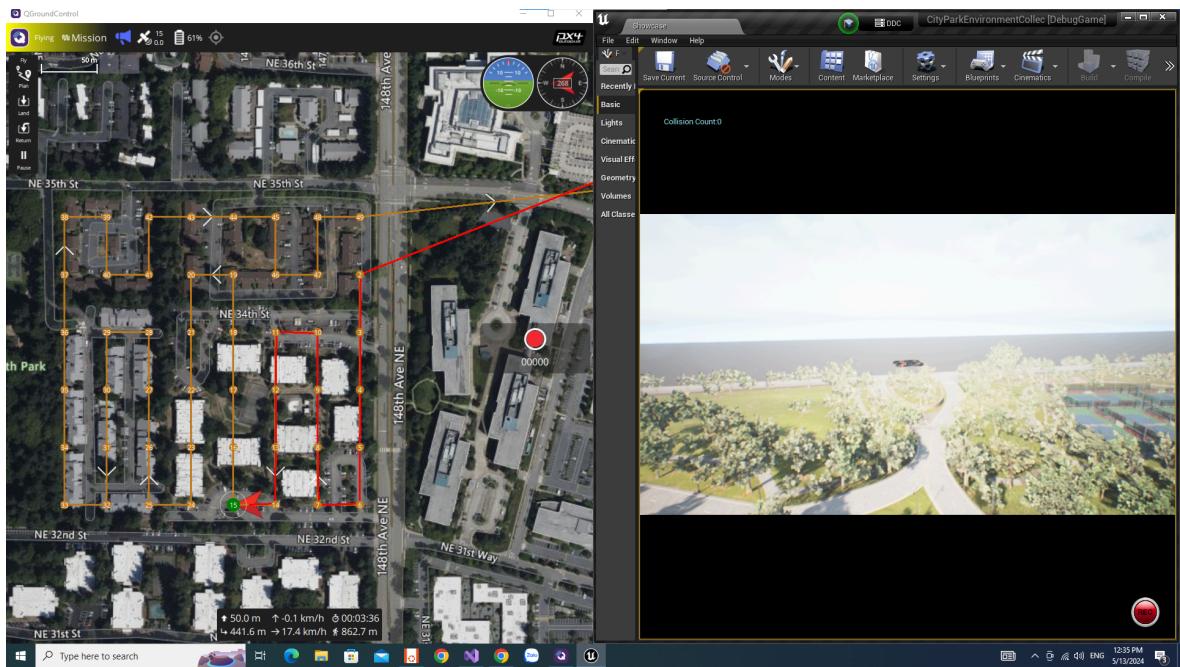
[huyhang@000A:~/PXA-Autopilot]
INFO [main] MAVSim simulator
INFO [simulator.mavlink] Using TCP on remote host localhost port 4560
WARN [simulator.mavlink] Please ensure port 4560 is not blocked by a firewall.
INFO [simulator.mavlink] Waiting for simulator to accept connection on TCP port 4560
INFO [simulator.mavlink] Waiting for simulator to accept connection on TCP port 4560
Buildfile: /home/huyhang/PXA-Autopilot/tools/simulation/jmavsim/JMAVSIM/build.xml

make_dirs:
compile:
create_run.jar:
copy_res:
BUILD SUCCESSFUL
Total time: 0 seconds
Position used: Starting Sim.
Starting GUI...
Exception in thread "main" java.lang.reflect.InvocationTargetException
    at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
    at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.base/java.lang.reflect.Method.invoke(Method.java:566)
    at org.eclipse.jdt.internal.jarinjarloader.JarRsrcLoader.main(JarRsrcLoader.java:61)
Caused by: java.awt.HeadlessException:
No X11 Display Manager: This program performed an operation which requires it.
    at java.desktop/java.awt.GraphicsEnvironment.checkHeadless(GraphicsEnvironment.java:208)
    at java.desktop/java.awt.Window.<init>(Window.java:548)
    at java.desktop/java.awt.Frame.<init>(Frame.java:384)
    at java.desktop/java.awt.Frame.<init>(Frame.java:388)
    at java.desktop/javax.swing.JFrame.<init>(JFrame.java:188)
    at me.drton.jmavsim.Visualizer3D.<init>(Visualizer3D.java:107)
    at me.drton.jmavsim.Visualizer3D.<init>(Visualizer3D.java:103)
    at me.drton.jmavsim.Simulator.<main>(Simulator.java:944)
    ... 5 more
INFO [simulator.mavlink] Simulator connected on TCP port 4560.
INFO [lockstep_scheduler] Setting initial absolute time to 1715659919837069 us
INFO [commander] LED: open /dev/led0 failed (22)
INFO [mavlink] mode: Offboard
INFO [mavlink] mode: Offboard, link lost, trying to connect again
INFO [mavlink] mode: Offboard, link lost, trying to connect again
INFO [mavlink] mode: Offboard, link lost, trying to connect again
INFO [mavlink] mode: Offboard, link lost, trying to connect again
INFO [mavlink] mode: Offboard, link lost, trying to connect again
INFO [mavlink] mode: Offboard, link lost, trying to connect again
INFO [mavlink] mode: Offboard, link lost, trying to connect again
INFO [mavlink] mode: Offboard, link lost, trying to connect again
INFO [logger] logger started (mode=ALL)
INFO [logger] [log/2024-05-14/04_12_00.ulg]
INFO [logger] Opened full log file: ./log/2024-05-14/04_12_00.ulg
INFO [logger] Setting log level to INFO
INFO [mavlink] MAVLink only on localhost (set param MAV_1[BROADCAST = 1 to enable network])
INFO [mavlink] MAVLink only on localhost (set param MAV_1[BROADCAST = 1 to enable network])
INFO [px4] Startup script returned successfully
INFO [mavlink] partner_id: 100, 0.1
INFO [mavlink] partner_id: 127.0.0.1, 0.1
WARN [health_and_arming_checks] preflight: GPS fix too low
WARN [health_and_arming_checks] preflight: GPS fix too low
INFO [tone_alarm] notify positive
INFO [commander] Ready for takeoff!
INFO [commander] Armed by external command
INFO [tone_alarm] arming warning
INFO [commander] Takeoff detected

```







KẾT LUẬN

Bài báo cáo đồ án tốt nghiệp trên đã trình bày một cái nhìn tổng quan về mô hình MILP (Mixed Integer Linear Programming) và ứng dụng của nó trong việc tối ưu hóa hiệu suất từ đó đã giải quyết được bài toán “Lập kế hoạch đường đi bao phủ cho thiết bị bay không người lái”. Qua quá trình thực hiện đề tài, em đã có cơ hội trải nghiệm thực tế trong việc xây dựng bài toán, xử lý dữ liệu đầu vào và thiết lập các ràng buộc để đạt được kết quả tối ưu.

Tuy nhiên, em nhận thấy rằng vẫn còn một số hạn chế trong công việc của mình. Một trong số đó là khả năng tránh chướng ngại vật và xử lý cho nhiều drone. Do thời gian và tài nguyên hạn chế, em chưa thể hoàn thiện các phần này trong đề tài của mình.

Về sau em sẽ tiếp tục nghiên cứu và hoàn thiện đề tài trong tương lai. Em sẽ tiếp tục tìm hiểu và áp dụng các phương pháp và kỹ thuật mới để giải quyết vấn đề tránh chướng ngại vật và xử lý tối ưu cho nhiều drone. Em hi vọng rằng việc nghiên cứu này sẽ đóng góp vào sự phát triển của lĩnh vực CPP và mang lại những ứng dụng thực tiễn trong thực tế.

Đề tài này đã đem lại những kiến thức quý giá và kinh nghiệm thực tiễn cho em trong quá trình thực hiện đồ án tốt nghiệp. Mặc dù còn nhiều hướng nghiên cứu và cải tiến tiềm năng, em đã nhận thức được những giới hạn và cam kết tiếp tục nỗ lực để nâng cao đề tài của mình trong tương lai.

Tài Liệu Tham Khảo

- [1]. Sung-Won Cho , Jin-Hyoung Park , Hyun-Ji Park and Seongmin Kim : Multi-UAV Coverage Path Planning Based on Hexagonal Grid Decomposition in Maritime Search and Rescue, 27 December 2021
- [2]. Gamil Ahmed, Tarek Sheltami1,AshrafMahmoud and Ansar Yasar : Energy-Ecient UAVs Coverage Path Planning Approach , 29 March 2022.
- [3]. Website hướng dẫn cách cài đặt AirSim. URL:
https://microsoft.github.io/AirSim/build_windows. Lần truy cập gần nhất 13/05/2024.
- [4]. Website hướng dẫn cách cài đặt PX4. URL:
https://microsoft.github.io/AirSim/px4_build/. Lần truy cập gần nhất 13/05/2024.
- [5]. Website hướng dẫn cách cài đặt QGroundControl. URL:
<https://qgroundcontrol.com/>. Lần truy cập gần nhất 13/05/2024.
- [6]. Website hướng dẫn cách cài đặt CPLEX. URL:
<https://www.ibm.com/products/ilog-cplex-optimization-studio>. Lần truy cập gần nhất 13/05/2024.