**Kính thưa thầy giáo, cô giáo và các bạn sinh viên**,  
Em là **Vũ Trung Đức**, sau đây em xin trình bày **khóa luận tốt nghiệp** về đề tài:  
**"Nghiên cứu mô phỏng hệ thống trình diễn ánh sáng drone, sử dụng kỹ thuật tối ưu quỹ đạo chuyển động"**, được **hướng dẫn bởi thầy Phạm Duy Hưng**.

Phần trình bày của em bao gồm **4 phần chính**:

**Phần thứ nhất – Giới thiệu chung**

**Phần thứ hai – Cơ sở lý thuyết**

**Phần thứ ba – Thiết kế hệ thống**

Và cuối cùng là **Thực nghiệm và đánh giá**

Đầu tiên em xin trình bày phần **giới thiệu chung**, tại đây em sẽ đưa ra **vấn đề** từ đó xác định **mục tiêu nghiên cứu** của đề tài.

Những năm gần đây, **trình diễn ánh sáng bằng drone** dần **thay thế pháo hoa truyền thống** nhờ sự:

**An toàn** (không gây cháy nổ)

**Thân thiện với môi trường**

**Dễ dàng tái sử dụng**

Một số buổi trình diễn như: **“Rực rỡ Thăng Long năm 2025”** hay **buổi tập duyệt drone cho ngày 30/4–1/5**.  
Tuy nhiên, việc triển khai hệ thống **trong thực tế lại khá phức tạp**, do:

**Chi phí tốn kém** (một chiếc drone có giá khoảng **20 triệu đồng**)

Nhiều **sự cố do nhiễu sóng**, **lỗi định vị**, hoặc **va chạm giữa các drone** dẫn đến **rơi hàng loạt UAV**

Chính vì vậy, **việc mô phỏng trước khi thực hiện trở thành một đề tài đáng được chú ý**.  
Để giải quyết vấn đề, ta cần trả lời **3 câu hỏi chính**:

**Làm sao để tối ưu quỹ đạo của drone?**

**Làm sao để drone hạn chế va chạm trong quá trình di chuyển?**

**Làm sao để xây dựng mô phỏng chính xác?**

Đề tài của em có **4 mục tiêu chính**:

**Xây dựng mô phỏng 3D trên nền tảng Unity**

**Tích hợp các thuật toán điều khiển**

**Thiết kế giao diện người dùng**

**Mô phỏng và kiểm thử với 85 drone**

Phần thứ hai, em xin trình bày về **tổng quan hệ thống UAV** và **các thuật toán được sử dụng** như:

**Thuật toán Hungarian** cho **phân nhiệm**

**APF** cho **tránh va chạm**

**MEC** cho **tối ưu chuyển động**

Trên đây là hình ảnh hệ thống trình diễn ánh sáng UAV:

Phần cứng bao gồm UAV nhỏ gọn có tích hợp đèn led, được định vị bằng cảm biến GNSS RTK, và nhận lệnh từ trung tâm điều khiển mặt đất (GCS) hoặc remote control.

Phần mềm là GCS thực hiện lập kế hoạch trình diễn, phân nhiệm, điều hướng drone.

Drone và GCS giao tiếp thông qua: Wifi hoặc MAV\_Link.

Ta có hàm tính tổng chi phí di chuyển của drone, trong đó i là thứ tự drone, j[i] là đích đến của drone tương ứng. Bài toán phân nhiệm có nhiệm vụ tìm một phương án sao cho tổng quãng đường đi là ngắn nhất.

Em sử dụng thuật toán Hungarian, với ma trận đầu vào là chi phí drone đến các mục tiêu. Thuật toán thực hiện phân công nhiệm vụ cho từng UAV với tổng quãng đường đi là nhỏ nhất. Độ phức tạp là O(n^3).

Trên đây là sơ đồ khối cho thuật toán Hungarian với 5 bước chính:

Bước 1: Từ ma trận đầu vào ta tiến hành chuẩn hóa hàng.

Bước 2: Ta thực hiện chuẩn hóa cột.

Bước 3: Sau khi thực hiện chuẩn hóa hàng và cột ta tiến hành tìm che phủ số 0 tối thiểu, nếu không tìm được số lượng số 0 phù hợp ta tiến hành Bước 4: Điều chỉnh ma trận và quay lại xử lý trên ma trận mới, ngược lại t thực hiện Bước 5: Xác định phương án.

Trong trường hợp không có vật cản, quãng đường nối giữa drone và đích đến là ngắn nhất, tuy nhiên trong trình diễn, drone có thể gặp các vật cản bên ngoài hoặc chính các drone khác. Để xử lý tình huống này, em có áp dụng thuật toán APF. Trong quá trình di chuyển, hướng di chuyển của drone bị ảnh hưởng bởi hai lực: Lực hút từ đích (Fatt) và tổng hợp lực đẩy từ các vật cản (Frep) với môi trường thế hấp dẫn được xác định bởi công thức, trong đó ka là hệ số hấp dẫn, rô là vector hướng từ drone tới đích. Môi trường thế đẩu được xác định bởi công thức, trong đó kr là hệ số đẩy, R là bán kính vùng kiểm tra.

Tuy nhiên, trong một số trường hợp đặc biệt tổng hợp lực đẩy ngược hướng với lực hút, dẫn đến drone bị đứng lại hoặc có thể va chạm với nhau. Trên hình ảnh là 2 ví dụ minh họa.

Để xử lý điều này, hệ thống cần tích hợp tối ưu cục bộ, khi có nguy cơ va chạm, drone gửi tín hiệu về bộ điều khiển, bộ điều khiển tiến hành quét các drone nằm trong phạm vi nguy hiểm rồi thực hiện phân nhiệm lại với những drone này, kết quả ta thu được hình dưới.

Đối với các kịch bản hình ảnh động, ta có thể xác định được đích đến tiếp theo của drone thay vì thực hiện lại thuật toán Hungarian. Ta có công thức: vị trí tiếp theo = vị trí hiện tại + vector chuyển động. Từ đó giúp giảm độ phức tạp tính toán từ O(n^3) xuống còn O(1)

Phần 3, thiết kế hệ thống với 3 nội dung: Kiến trúc tổng thể, việc tích hợp thuật toán và Giao diện ADI điều phối kịch bản.

Về kiến trúc tổng thể, hệ thống em xây dựng gồm 2 modun, modun xây dựng kịch bản và modun trình diễn drone. Dữ liệu đầu ra của modun adi sẽ là dữ liệu đầu vào của modun trình diễn drone.

Modun ADI gồm các thao tác cơ bản như thêm, xóa, sắp xếp kịch bản.

Tại Modun trình diễn drone sau khi nhận dữ liệu từ modun ADI thực hiện 3 bước:

Bước thứ nhất: đọc dữ liệu kịch bản đồng thời đánh dấu id cho từng drone.

Mỗi lần chuyển đổi kịch bản, thuật toán Hungarian được áp dụng để phân nhiệm cho từng drone.

Trong quá trình thực hiện, bộ quản lý đợi thông báo từ drone từ đó thực hiện chuyển frame đối với kịch bản hình động, đối với kịch bản tĩnh tiến hành chuyển sang kịch bản tiếp theo.

Tại trình điều khiển drone, tiến hành nhận dữ liệu từ bộ quản lý về đích đến, màu sắc hiển thị.

Trong quá trình di chuyển, drone xử lý va chạm với các drone khác, nếu có va chạm drone sẽ gửi thông báo về bộ quản lý để xử lý cục bộ. Tiếp đến drone xử lý di chuyển, thực hiện thuật toán APF với frame đầu tiên, nếu drone đang trong trạng thái lỗi drone được cộng thêm một vector random để ổn định. Đối với frame tiếp theo drone thực hiện di chuyển theo vector hướng đã xác định trước đó.

Cuối cùng, em xin trình bày về kết quả và đánh giá của hệ thống.

Em đã tiến hành chạy thử hệ thống trên 4 kịch bản, … với 2 trường hợp, trường hợp có sử dụng tối ưu cục bộ đường màu xanh và trường hợp thứ 2 không sử dụng tối ưu cục bộ đường màu cam. Từ đồ thị thứ nhất, ta nhận thấy ở trường hợp 1 khi drone có nguy cơ va chạm, sau một khoảng thời gian drone kéo dãn khoảng cách nhanh chóng đối chiếu sang đồ thị thứ 2 về mật độ drone, khi mật độ drone lên cao sau đó đã giảm, ngược lại đối với trường hợp 2, drone bị mắc kẹt một khoảng thời gian dẫn đến thời gian thực hiện bị chậm lại trong thực tế điều này rất nguy hiểm do vấn đề về năng lượng. Về quãng đường di chuyển, đối với 4 kịch bản trong đó có 2 kịch bản mật độ drone lớn nên tỉ lệ va chạm ít, như trên sơ đồ chương trình xử lý cục bộ khoảng 4 lần trong suốt quá trình tuy nhiên vẫn có sự tối ưu, trung bình mỗi drone giảm từ 154.22m xuống còn 153.81m tổng quãng đường giảm: 13108.64 xuống 13073.68m.

Qua đó em nhận thấy việc tối ưu cục bộ giúp xử lý tình huống va chạm chính xác, giúp thời gian thực hiện đảm bảo và tổng quãng đường di chuyển là tối ưu.

Cuối cùng, đề tài đã đạt được các mục tiêu nghiên cứu được đề cập trước đó: Thực hiện mô phỏng với 85 drone, trên nhiều kịch bản, áp dụng các thuật toán vào hệ thống và có thử nghiệm và đánh giá.

Về hướng phát triển trong tương lai,

Thứ nhất hiện tại drone chưa có hệ thống tính toán năng lượng, chính vì vậy việc tích hợp mô hình năng lượng là điều cần thiết. Bên cạnh đó việc áp dụng trí tuệ nhân tạo giúp drone tự xác định được quỹ đạo tối ưu khi có sự thay đổi địa hình, và cuối cùng để có thể trình diễn nhiều kịch bản quy mô hệ thống cần được cải tiến.

Trên đây là video về chương trình.

Phần trình bày của em đến đây là kết thúc, em xin chân thành cảm ơn quý thầy cô và các bạn đã lắng nghe. Em xin gửi lời cảm ơn trân thành tới thầy Phạm Duy Hưng đã giúp em trong quá trình nghiên cứu. Em rất mong nhận được các góp ý quý báu từ thầy cô và các bạn để có thể hoàn thiện đề tài tốt hơn trong tương lai.