EPIC: A Lightweight LiDAR-Based UAV Exploration Framework for Large-Scale Scenarios

Khung phương pháp khám phá UAV nhẹ dựa trên LiDAR cho các kịch bản quy mô lớn

Dưới đây là tóm tắt phần **Abstract** từ bài báo **EPIC: A Lightweight LiDAR-Based UAV Exploration Framework for Large-Scale Scenarios**:

Bài báo này giới thiệu **EPIC**, một khung phương pháp khám phá UAV nhẹ dựa trên LiDAR cho các môi trường quy mô lớn. Phương pháp này trực tiếp sử dụng dữ liệu điểm mây LiDAR để khám phá các môi trường rộng lớn, loại bỏ nhu cầu sử dụng các bản đồ lưới chiếm dụng tốn bộ nhớ. **EPIC** sử dụng một bản đồ quan sát mới được trích xuất trực tiếp từ chất lượng điểm mây, giúp tiết kiệm bộ nhớ nhưng vẫn đảm bảo khả năng khám phá đầy đủ. Bài báo cũng đề xuất một phương pháp xây dựng đồ thị topo gia tăng cho việc lập kế hoạch đường đi theo thời gian thực trong môi trường quy mô lớn. Các thử nghiệm mô phỏng và thực tế cho thấy **EPIC** đạt được tốc độ khám phá nhanh hơn và giảm mức tiêu thụ bộ nhớ so với các phương pháp hiện tại.

**Tóm tắt** — Khám phá tự động là một vấn đề cơ bản cho nhiều ứng dụng của các phương tiện bay không người lái (UAV). Gần đây, khám phá dựa trên **LiDAR** đã thu hút sự chú ý đáng kể nhờ khả năng tạo ra các bản đồ điểm mây có độ chính xác cao cho các môi trường quy mô lớn. Mặc dù các điểm mây đã cung cấp đủ thông tin cho việc định vị, nhiều phương pháp khám phá hiện tại vẫn phụ thuộc vào các biểu diễn môi trường bổ sung, thường tốn kém. Sự phụ thuộc này xuất phát từ hai lý do chính: cần phát hiện biên giới hoặc tính toán lợi ích thông tin, quá trình này thường phụ thuộc vào các bản đồ lưới chiếm dụng tốn bộ nhớ, và độ phức tạp tính toán cao của việc lập kế hoạch đường đi trực tiếp trên các điểm mây, chủ yếu do kiểm tra va chạm tốn kém. Để giải quyết những hạn chế này, chúng tôi trình bày **EPIC**, một khung khám phá UAV nhẹ dựa trên LiDAR trực tiếp khai thác dữ liệu điểm mây để khám phá các môi trường quy mô lớn. **EPIC** giới thiệu một bản đồ quan sát mới được trích xuất trực tiếp từ chất lượng của các điểm mây, loại bỏ nhu cầu sử dụng các bản đồ lưới chiếm dụng toàn cầu trong khi vẫn duy trì khả năng khám phá toàn diện. Chúng tôi cũng đề xuất một phương pháp xây dựng đồ thị topo gia tăng hoạt động trực tiếp trên các điểm mây, cho phép lập kế hoạch đường đi theo thời gian thực trong các môi trường quy mô lớn. Nhờ những thành phần này, chúng tôi xây dựng một khung lập kế hoạch phân cấp, tạo ra các quỹ đạo linh hoạt và tiết kiệm năng lượng, giảm đáng kể mức tiêu thụ bộ nhớ và thời gian tính toán so với hầu hết các phương pháp hiện tại. Các mô phỏng và thử nghiệm thực tế rộng rãi chứng minh rằng **EPIC** đạt được việc khám phá nhanh hơn trong khi giảm đáng kể mức tiêu thụ bộ nhớ so với các phương pháp tiên tiến.

**Giới thiệu**

Khám phá tự động là một vấn đề cơ bản cho nhiều ứng dụng của các phương tiện bay không người lái (UAV). Gần đây, khám phá dựa trên **LiDAR** đã thu hút được sự chú ý đáng kể nhờ khả năng tạo ra các bản đồ điểm mây có độ chính xác cao cho các môi trường quy mô lớn. Mặc dù các điểm mây vốn đã cung cấp thông tin hữu ích cho việc định hướng, nhưng nhiều phương pháp khám phá hiện tại vẫn dựa vào các biểu diễn môi trường bổ sung, thường tốn kém. Sự phụ thuộc này xuất phát từ hai lý do chính: cần phát hiện biên giới hoặc tính toán lợi ích thông tin, quá trình này thường phụ thuộc vào các bản đồ lưới chiếm dụng tốn bộ nhớ, và độ phức tạp tính toán cao của việc lập kế hoạch đường đi trực tiếp trên điểm mây, chủ yếu do kiểm tra va chạm tốn kém. Để giải quyết những hạn chế này, chúng tôi trình bày **EPIC**, một khung khám phá UAV nhẹ dựa trên LiDAR trực tiếp khai thác dữ liệu điểm mây để khám phá các môi trường quy mô lớn. **EPIC** giới thiệu một bản đồ quan sát mới được trích xuất trực tiếp từ chất lượng của các điểm mây, loại bỏ nhu cầu sử dụng các bản đồ lưới chiếm dụng toàn cầu trong khi vẫn duy trì khả năng khám phá toàn diện. Chúng tôi cũng đề xuất một phương pháp xây dựng đồ thị topo gia tăng hoạt động trực tiếp trên các điểm mây (cập nhật dần khi có những thay đổi mới diễn ra), cho phép lập kế hoạch đường đi theo thời gian thực trong các môi trường quy mô lớn. Nhờ những thành phần này, chúng tôi xây dựng một khung lập kế hoạch phân cấp, tạo ra các quỹ đạo linh hoạt và tiết kiệm năng lượng, giảm đáng kể mức tiêu thụ bộ nhớ và thời gian tính toán so với hầu hết các phương pháp hiện tại. Các mô phỏng và thử nghiệm thực tế rộng rãi chứng minh rằng **EPIC** đạt được việc khám phá nhanh hơn trong khi giảm đáng kể mức tiêu thụ bộ nhớ so với các phương pháp tiên tiến.

**II. Các công trình liên quan**

**A. Biểu diễn môi trường cho khám phá**

Biểu diễn môi trường là một thành phần cơ bản trong khám phá tự động của UAV, cung cấp thông tin quan trọng để hướng dẫn UAV khám phá các không gian chưa biết. Các phương pháp khám phá dựa trên biên giới truyền thống [1, 5, 11] thường dựa vào bản đồ lưới chiếm dụng toàn cầu đồng nhất hoặc các biến thể của nó, chẳng hạn như **UFOMap** [7], trong đó mỗi ô lưới được dán nhãn là chưa biết, tự do hoặc đã chiếm dụng. Các phương pháp này xác định ranh giới giữa các khu vực chưa biết và tự do là biên giới và điều hướng UAV đến những khu vực này để quan sát. Mặc dù các phương pháp này thực hiện tốt trong các môi trường kín như trong nhà hoặc trong hang động, nhưng khi áp dụng vào các khu vực rộng lớn như địa hình gồ ghề hoặc các khuôn viên trường, chúng tạo ra rất nhiều quỹ đạo không cần thiết. Sự không hiệu quả này là do UAV bị điều hướng để quan sát các khu vực trống rỗng, chẳng hạn như bầu trời. Một số phương pháp mới được đề xuất định nghĩa giao điểm giữa các khu vực chưa biết, tự do và đã chiếm dụng là các **biên giới bề mặt** [12, 13], điều này giúp hướng dẫn UAV khám phá tập trung vào bề mặt của cảnh vật, từ đó tránh được các quỹ đạo không cần thiết. Mặc dù cả hai phương pháp này đảm bảo khám phá toàn diện môi trường, yêu cầu sử dụng bản đồ lưới chiếm dụng lại dẫn đến mức tiêu thụ bộ nhớ cao, gây khó khăn cho tài nguyên trên tàu bị hạn chế.

Một số phương pháp khác sử dụng các phương pháp hình học để biểu diễn biên giới. Ví dụ, Yang và cộng sự [3] sử dụng các đa diện lồi để biểu diễn các khu vực được phát hiện và xác định các đa diện xa làm biên giới. Phương pháp này điều hướng UAV đến các biên giới này và cập nhật chúng theo thời gian. Gao và cộng sự [4] sử dụng biểu diễn đa diện sao lồi của các khu vực đã phát hiện, trong đó các mặt tam giác trên bề mặt của các đa diện sao lồi được định nghĩa là biên giới. Khi UAV di chuyển trong môi trường, phương pháp này tạo ra các đa diện sao lồi mới và loại bỏ các mặt tam giác có các điểm trung tâm đã được quan sát. Mặc dù các phương pháp hình học này mang lại lợi thế lớn về hiệu quả bộ nhớ, nhưng các mô hình hình học đơn giản có thể không nắm bắt được các chi tiết môi trường tinh vi, và chiến lược cập nhật biên giới thô có thể dẫn đến việc khám phá không đầy đủ môi trường.

Phương pháp của chúng tôi, được gọi là **bản đồ quan sát**, tập trung vào các bề mặt của môi trường, tránh việc duy trì bản đồ lưới chiếm dụng toàn cầu và giảm bớt các quỹ đạo không cần thiết, trong khi vẫn duy trì đủ thông tin để khám phá toàn diện các môi trường quy mô lớn.

**B. Xây dựng đồ thị topo**

Tìm kiếm đường đi là rất quan trọng trong khám phá theo thời gian thực, vì nó được thực hiện thường xuyên trong lập kế hoạch đường đi chỉ đạo toàn cầu. Tuy nhiên, việc tìm kiếm ngây thơ trên các bản đồ lưới chiếm dụng hoặc điểm mây có thể không khả thi trong thời gian thực cho các khoảng cách dài. Để giải quyết vấn đề này, một số phương pháp xây dựng đồ thị topo để tìm kiếm đường đi hiệu quả trong các môi trường quy mô lớn.

Musil và cộng sự [8] sử dụng bản đồ octree để biểu diễn các khu vực tự do, bao phủ các khu vực này bằng các hình cầu không va chạm và cập nhật gia tăng đồ thị topo dựa trên sự kết nối của các hình cầu. Zhang và cộng sự [2] biểu diễn không gian khám phá sử dụng bản đồ lưới chiếm dụng đồng nhất và phân vùng không gian một cách gia tăng thành các khu vực tự do và chưa biết không giao nhau dựa trên kết nối voxel từ thông tin chiếm dụng mới nhất, xây dựng một đồ thị kết nối trong đó các trung tâm khu vực hoạt động như các đỉnh và sự kết nối giữa các khu vực xác định các cạnh. Cả hai phương pháp này đều đạt được việc cập nhật gia tăng của đồ thị topo nhưng yêu cầu duy trì các bản đồ lưới chiếm dụng hoặc bản đồ octree bổ sung, dẫn đến mức tiêu thụ bộ nhớ cao trong các môi trường quy mô lớn. Các phương pháp như [9, 10] có thể xây dựng đồ thị topo trực tiếp trên các điểm mây nhưng thiếu khả năng xây dựng gia tăng trong các cảnh phức tạp.

Phương pháp của chúng tôi được lấy cảm hứng từ Zhang và cộng sự [2], phân vùng không gian thành các khu vực độc lập và xây dựng gia tăng đồ thị topo dựa trên kết nối của chúng. Tuy nhiên, khác với phương pháp của họ, chúng tôi xây dựng trực tiếp trên các điểm mây, giảm đáng kể mức tiêu thụ bộ nhớ.

**III. Định nghĩa vấn đề và tổng quan hệ thống**

Vấn đề được giải quyết trong bài báo này là khám phá một không gian 3D chưa biết quy mô lớn **H ∈ R^3** bằng một phương tiện bay không người lái (UAV) để tạo ra bản đồ điểm mây dày đặc.

Khung phương pháp khám phá được đề xuất bao gồm một **biểu diễn môi trường dựa trên quan sát** (IV) và một **khung lập kế hoạch phân cấp theo thời gian thực** (V). Khi nhận được một bản quét điểm mây LiDAR mới, **bản đồ quan sát** **Mobs**, ghi lại chất lượng của các quan sát bề mặt, được cập nhật gia tăng (IV-A). Dựa trên **Mobs**, các biên giới **Sftr**được phát hiện và phân nhóm (IV-B). Để tạo điều kiện cho việc lập kế hoạch đường đi hiệu quả trong các môi trường quy mô lớn, một **đồ thị topo** **G** được xây dựng gia tăng trên các điểm mây (V-A). Các điểm nhìn (**viewpoints**) được chọn chiến lược dựa trên khả năng kết nối topo. Sau đó, một đường dẫn chỉ đạo toàn cầu bắt đầu từ vị trí hiện tại và đi qua tất cả các điểm nhìn được tính toán (V-B). Dựa trên hành trình toàn cầu, các quỹ đạo địa phương hiệu quả về năng lượng và nhanh nhẹn được tạo ra, hướng dẫn UAV đến các điểm nhìn này theo trình tự (V-C). Quá trình khám phá được coi là hoàn thành khi không còn biên giới nào.

**IV. Biểu diễn môi trường khám phá dựa trên quan sát**

Trong phần này, chúng tôi đề xuất một phương pháp biểu diễn môi trường dựa trên quan sát mới cho khám phá UAV, bao gồm hai thành phần chính: **bản đồ quan sát Mobs** và **danh sách các cụm biên giới Lclusters**.

**Mobs** ghi lại chất lượng của các quan sát bề mặt, điều này được xác định bởi khoảng cách quan sát và hướng nhìn của LiDAR. Dựa trên việc đánh giá chất lượng, các bề mặt được quan sát **Sobs** được dán nhãn là quan sát tốt hoặc quan sát kém. Các biên giới **Sfrt**, được định nghĩa là ranh giới giữa các bề mặt quan sát tốt **Swell** và các bề mặt quan sát kém **Spoorly**, được phân nhóm và lưu trữ trong **Lclusters**. Khi nhận được một khung điểm mây mới, cả **Mobs** và **Lclusters** được cập nhật gia tăng.

**A. Xây dựng bản đồ quan sát**

Để hỗ trợ việc truy vấn hiệu quả chất lượng quan sát bề mặt và tránh tiêu thụ bộ nhớ quá mức, chúng tôi sử dụng một bản đồ **băm không gian** [14] để lưu trữ chất lượng của các quan sát bề mặt. Định nghĩa bản đồ này là **bản đồ quan sát Mobs**, mỗi phần tử bên trong là một voxel đại diện cho một mảnh bề mặt nhỏ và được dán nhãn là quan sát tốt hoặc quan sát kém dựa trên chất lượng quan sát. Việc lưu trữ có chọn lọc này, không giống như các bản đồ lưới chiếm dụng tốn bộ nhớ, đảm bảo rằng **Mobs** chỉ tập trung vào các khu vực quanh bề mặt, từ đó giảm thiểu tiêu thụ bộ nhớ mà vẫn giữ đủ thông tin để phát hiện biên giới.

Chất lượng quan sát của mỗi mảnh bề mặt được xác định bởi ràng buộc khoảng cách và ràng buộc hướng nhìn. Xem xét một mảnh bề mặt có tâm tại **ps** và LiDAR được đặt tại **pl**, ràng buộc khoảng cách được coi là thỏa mãn nếu

∥pl−ps∥2≤D

Trong đó **D** là một hằng số được định nghĩa trước để đảm bảo rằng LiDAR đủ gần với mảnh bề mặt để có được quan sát chất lượng cao.

Với chất lượng quan sát cải thiện khi tia LiDAR trở nên vuông góc với mảnh bề mặt, như minh họa trong Hình 2(a), đối với trường hợp 2D, chất lượng của hướng nhìn tỷ lệ nghịch với **|90° − θ|**. Giả sử **l2 ≥ l1**, góc **θ** có thể được tính bằng:

cot⁡θ=l2−l1cos⁡δl1sin⁡δ=1sin⁡δ⋅l2l1−cot⁡δ\cot\theta = \frac{l2 - l1 \cos\delta}{l1 \sin\delta} = \frac{1}{\sin\delta} \cdot \frac{l2}{l1} - \cot\deltacotθ=l1sinδl2−l1cosδ​=sinδ1​⋅l1l2​−cotδ

Trong đó **δ** đại diện cho sự phân tách góc giữa hai tia LiDAR liền kề, một giá trị được xác định bởi cấu hình cơ học của LiDAR. Theo phương trình (2), **|90° − θ|** và **l1/l2** có mối quan hệ nghịch, vì vậy chúng tôi có thể sử dụng **l1/l2** để đánh giá chất lượng hướng nhìn. Chúng tôi coi ràng buộc hướng nhìn được thỏa mãn khi **l1/l2** vượt qua ngưỡng **T**.

Đối với trường hợp 3D, như minh họa trong Hình 2(b), khu vực cảm biến của LiDAR bao gồm một số thể tích hình chóp nhỏ. Xem xét một mảnh bề mặt có tâm tại **pi**, nằm trong một thể tích hình chóp **fi**. Nếu bốn tia tạo thành **fi** thỏa mãn điều kiện đã nêu, chúng tôi coi ràng buộc hướng nhìn được thỏa mãn.

Khi nhận được một khung điểm mây mới, các voxel trong **Mobs** giao với điểm mây sẽ được xác định và thêm vào một hàng đợi cập nhật **Q**. Sau đó, chúng tôi duyệt qua hàng đợi. Đối với mỗi voxel chưa được gán nhãn hoặc đã được gán nhãn là quan sát kém, chúng tôi sẽ cập nhật phân loại của nó thành quan sát tốt nếu nó thỏa mãn cả ràng buộc khoảng cách và hướng nhìn. Nếu không, nó sẽ được gán nhãn là quan sát kém.

**B. Phát hiện và phân cụm biên giới gia tăng**

Để đạt được việc cập nhật biên giới theo thời gian thực, chúng tôi đề xuất một phương pháp phát hiện và phân cụm biên giới gia tăng. Chúng tôi đầu tiên duyệt qua tất cả các voxel được gán nhãn là quan sát kém trong **Q**, và cập nhật phân loại của nó thành biên giới nếu voxel có một voxel láng giềng được gán nhãn là quan sát tốt. Lưu ý rằng tập hợp biên giới **Sftr** là một phần của **Spoorly**. Trong quá trình cập nhật **Mobs** mô tả ở phần IV-A, các voxel có nhãn biên giới được coi như voxel quan sát kém.

Trong khi duyệt, một **hộp bao quanh theo trục (AABB)** **Bupdate** của các voxel đã thay đổi từ biên giới thành quan sát tốt và các voxel đã chuyển từ quan sát kém thành biên giới được tính toán đồng thời. Ngoài việc cập nhật **Mobs**, các biên giới mới được tạo ra trong khung điểm mây hiện tại cũng được thêm vào một tập hợp **Fcurrent**.

Đối với khung điểm mây đầu tiên, chúng tôi áp dụng một phương pháp phân cụm để phân nhóm **Fcurrent** dựa trên khoảng cách Euclid và các vectơ pháp tuyến của bề mặt. Xem xét hai voxel biên giới **fi, fj** có các vị trí trung tâm **pi, pj** và vectơ pháp tuyến **ni, nj** tương ứng. Chúng tôi coi hai voxel này là láng giềng nếu:

∥pi−pj∥2<ϵd∧ni⋅nj>ϵn\|pi - pj\|\_2 < \epsilon\_d \land ni \cdot nj > \epsilon\_n∥pi−pj∥2​<ϵd​∧ni⋅nj>ϵn​

Trong đó **εd** là ngưỡng khoảng cách được định nghĩa trước và **εn** là ngưỡng sự tương đồng của vectơ pháp tuyến.

Quá trình phân cụm bắt đầu bằng cách chọn phần tử đầu tiên từ **Fcurrent**. Bắt đầu từ voxel này, chúng tôi sử dụng thuật toán **Tìm kiếm theo chiều rộng (BFS)** để mở rộng cụm, nhận diện và thêm các voxel láng giềng vào nếu chúng đáp ứng các tiêu chí về khoảng cách và sự tương đồng pháp tuyến đã chỉ định. Quá trình mở rộng kết thúc khi BFS hoàn thành tự nhiên hoặc khi kích thước của **AABB** (hộp bao quanh theo trục) của cụm vượt quá ngưỡng đã định trước. Trong quá trình này, các voxel được thêm vào cụm sẽ đồng thời bị loại bỏ khỏi **Fcurrent**. Sau khi mở rộng một cụm kết thúc, quá trình này được khởi tạo lại với phần tử đầu tiên của **Fcurrent** đã được cập nhật. Quá trình lặp lại cho đến khi **Fcurrent** rỗng. Sau đó, **AABB** của mỗi cụm, các voxel biên giới và các vectơ pháp tuyến tương ứng của chúng được lưu trữ trong **Lclusters** để xử lý tiếp theo.

Đối với các khung điểm mây tiếp theo, chúng tôi đầu tiên xác định tất cả các cụm giao với **Bupdate**, loại bỏ chúng khỏi **Lclusters** và thêm các voxel biên giới của chúng vào **Fcurrent**. Sau đó, chúng tôi thực hiện một quá trình phân cụm mới trên **Fcurrent** và cập nhật **Lclusters** dựa trên kết quả phân cụm.

**V. Lập kế hoạch khám phá phân cấp**

Phương pháp lập kế hoạch khám phá phân cấp của chúng tôi bao gồm ba thành phần chính: một đồ thị topo xây dựng gia tăng **G**, một bộ lập kế hoạch đường dẫn chỉ đạo toàn cầu, và một bộ lập kế hoạch quỹ đạo địa phương.

Để hỗ trợ tìm kiếm đường đi hiệu quả trong các môi trường quy mô lớn, chúng tôi đề xuất một phương pháp xây dựng đồ thị topo gia tăng hoạt động trực tiếp trên các điểm mây. Tận dụng đồ thị topo **G**, bộ lập kế hoạch toàn cầu tính toán các đường dẫn chỉ đạo. Sau đó, bộ lập kế hoạch địa phương tạo ra các quỹ đạo an toàn, mượt mà và khả thi dựa trên các đường dẫn chỉ đạo, từ đó UAV có thể đi qua các biên giới một cách hiệu quả.

**A. Xây dựng đồ thị topo gia tăng**

Đồ thị topo được sử dụng rộng rãi trong khám phá môi trường quy mô lớn để hỗ trợ tìm kiếm đường đi nhanh hơn [2, 3, 8]. Tuy nhiên, hầu hết các phương pháp hiện có xây dựng đồ thị topo dựa trên bản đồ lưới chiếm dụng toàn cầu, điều này có thể gây tốn bộ nhớ. Để giải quyết vấn đề này, chúng tôi đề xuất một phương pháp xây dựng đồ thị topo gia tăng hoạt động trực tiếp trên các điểm mây. Trong phương pháp của chúng tôi, đồ thị topo được biểu diễn dưới dạng **G = (V, E)**, trong đó **V** là các đỉnh đại diện cho các không gian tự do độc lập, và **E** là các cạnh đại diện cho sự kết nối giữa các không gian tự do này. Mỗi cạnh trong **E** gắn với một đường đi không va chạm.

Lấy cảm hứng từ [2], chúng tôi phân vùng toàn bộ không gian khám phá thành các khu vực hình hộp **R**, được quản lý bởi một bản đồ băm không gian. Các đỉnh **V** của đồ thị topo **G** được lưu trữ trong các khu vực tương ứng của chúng. Khi nhận được một khung điểm mây mới, các khu vực giao nhau với các khu vực quan sát của LiDAR được thêm vào danh sách cập nhật **Lregion**. Thêm vào đó, đối với các khu vực lần đầu tiên được thêm vào **Lregion** trong quá trình khám phá, tập hợp các đỉnh tương ứng được khởi tạo là rỗng.

**1) Cập nhật đỉnh**

Đối với mỗi khu vực **ri** trong **Lregion**, chúng tôi sử dụng nhiều hình cầu không va chạm để bao phủ các không gian tự do của nó. Như minh họa trong Hình 3(a) và Hình 3(b), hình cầu không va chạm ban đầu có tâm tại **ri**, với bán kính được xác định bởi khoảng cách đến điểm gần nhất trong bản đồ điểm mây. Nếu hình cầu này bao phủ hoàn toàn khu vực, quá trình bao phủ khu vực **ri** kết thúc. Nếu không, quá trình tiếp tục dựa trên bán kính của hình cầu. Nếu bán kính vượt quá một khoảng cách an toàn được định nghĩa trước, khu vực này sẽ được phân thành các khu vực con nhỏ hơn thông qua các mặt của hình hộp có ba mặt vuông góc với nhau. Nếu không, hình cầu sẽ bị loại bỏ, khu vực sẽ được chia thành tám khu vực con nhỏ hơn theo ba mặt vuông góc đi qua tâm hình cầu.

**2) Cập nhật cạnh**

Tập hợp cạnh **E** được khởi tạo là rỗng. Khi các đỉnh trong **G** được cập nhật, các cạnh nối chúng được xác định từ các đường đi tìm được qua thuật toán tìm kiếm **A\***. Như minh họa trong Hình 4, dựa trên các đỉnh **Vpre** và **Vnew**, chúng tôi phân loại các đỉnh thành ba tập hợp khác nhau:

Vremove=Vpre∖VnewV\_{\text{remove}} = V\_{\text{pre}} \setminus V\_{\text{new}}Vremove​=Vpre​∖Vnew​ Vremain=Vnew∩VpreV\_{\text{remain}} = V\_{\text{new}} \cap V\_{\text{pre}}Vremain​=Vnew​∩Vpre​ Vinsert=Vnew∖VpreV\_{\text{insert}} = V\_{\text{new}} \setminus V\_{\text{pre}}Vinsert​=Vnew​∖Vpre​

Sau đó, chúng tôi duyệt qua **V\_{\text{remove}}**, loại bỏ tất cả các cạnh liên kết với chúng khỏi **G**. Tiếp theo, đối với các đỉnh trong **V\_{\text{remain}}**, chúng tôi kiểm tra tất cả các đường đi không va chạm tương ứng với các cạnh nối chúng. Đối với mỗi đường đi, chúng tôi xác minh xem nó vẫn còn không va chạm trong bản đồ điểm mây cập nhật hay không. Nếu một đường đi không còn không va chạm, chúng tôi thêm cặp đỉnh tương ứng vào hàng đợi **Qedge** và loại bỏ cạnh liên kết khỏi **G**. Sau đó, đối với mỗi **v ∈ V\_{\text{insert}}**, chúng tôi tạo cặp đỉnh giữa **v** và các đỉnh trong các khu vực láng giềng, thêm các cặp này vào **Qedge**.

Sau đó, chúng tôi thực hiện thuật toán **A\*** cho mỗi cặp đỉnh trong **Qedge**. Nếu tìm được đường đi không va chạm, chúng tôi thêm cạnh tương ứng vào **G** và lưu trữ đường đi không va chạm đó. Đồng thời, để tăng tốc quá trình tìm kiếm và tránh các cạnh dư thừa, chúng tôi hạn chế phạm vi tìm kiếm của thuật toán **A\*** trong một khu vực nhỏ, được định nghĩa là một **AABB** (hộp bao quanh theo trục) bao quanh chặt chẽ các điểm bắt đầu và kết thúc, với một khoảng đệm bằng kích thước của một khu vực.

**B. Lập kế hoạch Đường dẫn Hướng dẫn Toàn cục**

Mô-đun này được thiết kế để xác định một chuỗi thăm viếng hiệu quả cho các nhóm biên (frontier clusters) trong *Lclusters* và sau đó tạo ra một con đường hướng dẫn cho bộ lập kế hoạch lộ trình cục bộ.

Vì các chuỗi thăm viếng tối ưu toàn cục thường ưu tiên các cụm gần trước các cụm xa hơn, và xét đến việc lập kế hoạch lại với tần suất cao trong suốt chuyến bay, chỉ một phần nhỏ của con đường hướng dẫn sẽ được thực hiện trước mỗi chu kỳ lập kế hoạch lại. Do đó, ảnh hưởng của việc sắp xếp các cụm xa là giảm bớt. Vì lý do này, chúng tôi sử dụng một phương pháp ước lượng khoảng cách thô để xếp hạng tất cả các nhóm biên trong *Lclusters*, từ đó tạo ra một chuỗi thăm viếng sơ bộ thô. Sau đó, chúng tôi thực hiện điều chỉnh tỉ mỉ hơn đối với K phần tử đầu tiên trong chuỗi này, tạo ra con đường hướng dẫn toàn cục tương ứng. Chiến lược hai giai đoạn này tập trung tài nguyên tính toán vào các cụm gần, giúp cân bằng giữa tối ưu hóa con đường và hiệu quả tính toán, khiến nó đặc biệt phù hợp cho việc thăm dò trong các kịch bản môi trường lớn.

**1) Xếp hạng các Cụm Biên:** Trong suốt quá trình thăm dò, các biên liên tục được tạo ra, mỗi biên đều có thể được tiếp cận thông qua một con đường thẳng từ vị trí UAV tại thời điểm tạo ra nó, vì chúng được UAV nhìn thấy tại thời điểm đó. Dựa trên quan sát này, chúng tôi giới thiệu một chỉ số khoảng cách quay lại để ưu tiên các nhóm biên. Đối với một cụm có trung tâm là *ci*, giả sử *pi* là vị trí của UAV tại thời điểm tạo ra cụm, chỉ số khoảng cách quay lại *di* được định nghĩa như sau:

di=df(pi,pcurrent)+∣∣pi−ci∣∣2d\_i = d\_f(p\_i, p\_{current}) + ||p\_i - c\_i||\_2di​=df​(pi​,pcurrent​)+∣∣pi​−ci​∣∣2​

Trong đó, df(pi,pcurrent)d\_f(p\_i, p\_{current})df​(pi​,pcurrent​) là khoảng cách bay của UAV từ *pi* đến vị trí hiện tại của nó, và ∣∣pi−ci∣∣2||p\_i - c\_i||\_2∣∣pi​−ci​∣∣2​ là khoảng cách Euclid giữa *pi* và *ci*.

Khoảng cách quay lại *di* tính tổng khoảng cách mà UAV phải đi qua để quay lại *ci* bằng cách theo dõi lại hành trình lịch sử của nó. Bằng cách duy trì hồ sơ khoảng cách bay của UAV tại mỗi thời điểm tạo ra cụm và tổng quãng đường bay của nó, chúng tôi có thể tính toán *di* với độ phức tạp thời gian là O(1). Việc tính toán hiệu quả này cho phép chúng tôi nhanh chóng xếp hạng các cụm trong *Lclusters*, từ đó tạo ra một chuỗi thăm viếng toàn cục sơ bộ.

**2) Lập kế hoạch Đường dẫn Hướng dẫn:** Đối với K cụm đầu tiên trong chuỗi, chúng tôi tạo ra một điểm quan sát cho mỗi cụm dựa trên phương pháp đã mô tả trong phần "Viewpoint Generation" (Tạo điểm quan sát), và kết nối các điểm quan sát này như các đỉnh trong đồ thị G. Sau đó, chúng tôi thực hiện tìm kiếm A\* trên G để tính toán khoảng cách giữa các cặp điểm quan sát. Tiếp theo, chúng tôi mô hình hóa vấn đề này như một Bài toán Tối ưu Hành trình Du lịch Không đối xứng (ATSP) và giải quyết nó bằng một thuật toán phù hợp. Giải pháp này mang lại con đường ngắn nhất bắt đầu từ vị trí hiện tại của UAV và đi qua tất cả các điểm quan sát đã chọn, tạo ra con đường hướng dẫn toàn cục.

**3) Tạo điểm quan sát:** Đối với một cụm biên *Cf*, các điểm quan sát ứng viên được lấy mẫu đồng đều trong hệ tọa độ hình trụ với tâm tại trung tâm của cụm, như được minh họa trong Hình 5(a). Sau đó, chúng tôi tính toán khoảng cách từ mỗi điểm quan sát ứng viên đến điểm gần nhất trong bản đồ đám mây điểm. Các điểm quan sát có khoảng cách nhỏ hơn ngưỡng an toàn sẽ bị loại bỏ. Các điểm quan sát còn lại mỗi cái tương ứng với một hình cầu không có va chạm. Chúng tôi sau đó nhóm chúng lại dựa trên sự kết nối của các hình cầu tương ứng của chúng thành các cụm điểm quan sát *Ccv*.

**Giải thích thêm:** Phần này mô tả một phương pháp để xác định các con đường hiệu quả cho UAV khi thăm dò một môi trường lớn. Quá trình này bao gồm việc tìm kiếm các cụm biên (nơi cần thăm dò thêm), tạo ra các điểm quan sát từ các vị trí trong môi trường, và từ đó tính toán con đường UAV nên đi để tối ưu hóa thời gian và năng lượng trong suốt chuyến bay. Hệ thống sử dụng các thuật toán tối ưu hóa để lập kế hoạch các chuyến thăm quan, giúp UAV di chuyển một cách hiệu quả và tránh lãng phí tài nguyên.

**VI. Kết quả**

**A. Chi tiết thực hiện**

Chúng tôi thiết lập **wc = 1.5** trong phương trình (3) và (6). Trong lập kế hoạch hành trình toàn cầu, **ATSP** được giải quyết bằng bộ giải heuristic **Lin-Kernighan-Helsgaun** [33]. Trong việc tinh chỉnh điểm nhìn địa phương, chúng tôi giữ lại tối đa **Nview = 15** điểm nhìn trong mỗi **FIS** và cắt ngắn hành trình toàn cầu trong bán kính **Rrf = 5.0**. Đối với tối ưu hóa quỹ đạo, chúng tôi sử dụng các giá trị **ws,p = 5.0**, **ws,ξ = 2.5**, **wt = 1.0**, **λc = λbs = 10.0**, **λd = 2.0** và **dmin = 0.4**, giải bài toán này bằng một bộ giải tối ưu hóa phi tuyến tổng quát **NLopt2**. **B-spline bậc 3** (pb = 3) được sử dụng để đại diện cho quỹ đạo.

Để đạt được việc khám phá nhanh, một khung bản đồ hiệu quả là rất quan trọng. Trong công trình này, chúng tôi sử dụng một bản đồ **volumetric** [34], đã được áp dụng thành công trong các chuyến bay tự động nhanh [28, 30] trong các cảnh phức tạp. Tương tự như [35], được sử dụng rộng rãi trong khám phá, [34] xây dựng một biểu diễn lưới chiếm dụng của không gian. Trong khi đó, nó cũng duy trì một **ESDF** (Euclidean Signed Distance Field) gia tăng để tạo điều kiện cho việc lập kế hoạch quỹ đạo. Để biết thêm chi tiết về khung bản đồ của chúng tôi, chúng tôi giới thiệu cho người đọc [34].

**B. So sánh chuẩn và phân tích**

Chúng tôi kiểm tra phương pháp của mình trong các mô phỏng. Chúng tôi so sánh phương pháp của mình trong một kịch bản cầu và một kịch bản mê cung lớn. Ba phương pháp được so sánh: **NBVP** [12], **phương pháp biên giới cổ điển** [7] và **phương pháp biên giới nhanh** [1]. Lưu ý rằng không có mã nguồn mở cho **[1]**, vì vậy chúng tôi sử dụng triển khai của mình. Trong tất cả các bài kiểm tra, các giới hạn động học được thiết lập là **vmax = 2.0 m/s** và **ξ̇max = 0.9 rad/s** cho tất cả các phương pháp. Các **FOV** của cảm biến được thiết lập là **[80 × 60] độ** với khoảng cách tối đa là **4.5 m**. Trong cả ba kịch bản, mỗi phương pháp được chạy ba lần với cấu hình ban đầu giống nhau. Thống kê và tiến trình khám phá của bốn phương pháp được hiển thị trong Bảng II và Hình 8.

**1) Kịch bản cầu**

Đầu tiên, chúng tôi so sánh bốn phương pháp trong một không gian **10 × 20 × 5 m³** có chứa một cây cầu, như minh họa trong Hình 7. Kết quả cho thấy chúng tôi đạt được thời gian khám phá ngắn hơn nhiều và độ biến thiên thời gian nhỏ hơn. Hành trình tổng thể của phương pháp của chúng tôi ngắn hơn đáng kể, chủ yếu vì chúng tôi lập kế hoạch hành trình toàn cầu. Đường đi thực hiện mượt mà hơn, vì chúng tôi tinh chỉnh các chuyển động địa phương và tạo ra các quỹ đạo mượt mà. Đồng thời, chúng tôi có thể điều hướng với tốc độ bay cao hơn nhờ vào việc lập kế hoạch quỹ đạo tối thiểu theo thời gian.

**2) Kịch bản mê cung lớn**

Chúng tôi cũng so sánh các phương pháp một cách định lượng trong một môi trường mê cung lớn như trong Hình 9. Không gian khám phá có kích thước **20×80×3 m³**. Trong kịch bản này, tất cả các phương pháp được so sánh mất rất nhiều thời gian để hoàn thành việc bao phủ toàn bộ, do độ phức tạp của cảnh. Tuy nhiên, phương pháp của chúng tôi hoàn thành việc khám phá nhanh hơn trung bình gấp hơn 4 lần. Đường đi thực hiện bởi bốn phương pháp sau khi hoàn thành được hiển thị trong Hình 9. Đặc biệt, phương pháp của chúng tôi khám phá mê cung theo một thứ tự hợp lý hơn, không quay lại các khu vực đã khám phá nhiều lần. Kết quả là, nó tạo ra một con đường bao phủ ngắn hơn nhiều và tỷ lệ khám phá gần như tuyến tính (Hình 8). Hành vi này là nhờ vào kế hoạch toàn cầu, nếu không các khu vực đã biết có thể bị quay lại nhiều lần và làm chậm tiến độ, như các phương pháp so sánh. Lưu ý rằng thời gian tính toán trong mê cung lớn lâu hơn, chủ yếu do cảnh có diện tích lớn hơn, tự nhiên dẫn đến số lượng các cụm biên giới lớn hơn.

**C. Thử nghiệm trong môi trường thực tế**

Để xác minh thêm phương pháp của chúng tôi, chúng tôi tiến hành các thử nghiệm thực tế rộng rãi trong cả môi trường trong nhà và ngoài trời. Trong tất cả các thử nghiệm, chúng tôi thiết lập các giới hạn động học là **vmax = 1.5 m/s**, **amax = 0.8 m/s** và **ξ̇max = 0.9 rad/s**. Lưu ý rằng chúng tôi không sử dụng bất kỳ thiết bị ngoài nào để định vị và chỉ dựa vào bộ ước lượng trạng thái trên tàu.

Đầu tiên, chúng tôi trình bày các thử nghiệm khám phá nhanh trong hai cảnh trong nhà. Cảnh đầu tiên được hiển thị trong Hình 1, trong đó chúng tôi triển khai hàng chục vật cản và UAV phải thực hiện các thao tác 3D để lập bản đồ không gian chưa biết và tránh vật cản đồng thời. Chúng tôi giới hạn không gian cần khám phá trong một hộp có kích thước **10×6×2 m³**. Một bản đồ mẫu và quỹ đạo bay được trình bày trong Hình 1. Cảnh trong nhà thứ hai là một môi trường rộng lớn hơn, bao gồm hai phòng, trong đó một phòng giống như cảnh 1 và phòng kia là một phần của văn phòng chứa bàn và ghế. Không gian này được giới hạn trong hộp **15 × 11 × 2 m³**. UAV bắt đầu khám phá phòng lớn, sau đó tiếp tục đến phòng nhỏ. Cảnh thứ hai, bản đồ trực tuyến tạo ra và quỹ đạo bay được hiển thị trong Hình 10. Lưu ý rằng trong hai cảnh này, UAV bắt đầu ở vị trí có độ bao phủ thấp, vì vậy nó chỉ lập bản đồ một khu vực nhỏ của môi trường vào đầu. Cuối cùng, để xác minh phương pháp của chúng tôi trong các môi trường tự nhiên, chúng tôi tiến hành các thử nghiệm khám phá trong một khu rừng. Kích thước khu vực cần khám phá là **11×10×2 m³**. Môi trường thử nghiệm và các kết quả liên quan được hiển thị trong Hình 11.

**VII. Kết luận**

Trong bài báo này, chúng tôi đề xuất một phương pháp khám phá UAV tự động nhẹ nhàng cho các môi trường quy mô lớn dựa trên LiDAR. Phương pháp này sử dụng **biểu diễn môi trường dựa trên quan sát** trực tiếp từ các điểm mây, loại bỏ nhu cầu sử dụng các bản đồ lưới chiếm dụng tốn bộ nhớ trong khi vẫn duy trì đủ thông tin để khám phá toàn diện. Chúng tôi cũng phát triển một phương pháp xây dựng đồ thị topo gia tăng hoạt động trực tiếp trên các điểm mây, cho phép lập kế hoạch đường đi hiệu quả. Tận dụng các thành phần này, chúng tôi xây dựng một khung lập kế hoạch phân cấp, đạt được thời gian tính toán và mức tiêu thụ bộ nhớ giảm đáng kể so với các phương pháp hiện tại.

Kết quả thử nghiệm cho thấy phương pháp của chúng tôi vượt trội hơn các phương pháp khác trong các kịch bản thử nghiệm lớn và thực tế, với tốc độ khám phá nhanh hơn và hiệu quả sử dụng bộ nhớ tốt hơn. Điều này chứng tỏ tính khả thi của phương pháp trong các môi trường quy mô lớn và phức tạp. Tuy nhiên, một hạn chế của phương pháp là giả định rằng việc ước lượng trạng thái là hoàn hảo, như hầu hết các phương pháp hiện tại. Chúng tôi đánh giá phương pháp của mình trong mô phỏng với định vị chính xác, trong khi sai lệch trong tư thế không được xem xét. Tuy nhiên, sai số trong ước lượng trạng thái là không thể tránh khỏi và cần phải được xem xét. Trong tương lai, chúng tôi dự định sẽ xem xét sự không chắc chắn của ước lượng trạng thái trong phương pháp của mình và đánh giá hiệu suất dưới các sai lệch tư thế.

**Tóm tắt toàn bộ bài báo:**

Bài báo **"EPIC: A Lightweight LiDAR-Based UAV Exploration Framework for Large-Scale Scenarios"** giới thiệu một phương pháp khám phá UAV mới, **EPIC**, được thiết kế để giải quyết những thách thức trong việc khám phá các môi trường quy mô lớn và phức tạp bằng cách sử dụng công nghệ **LiDAR**. Phương pháp này tận dụng dữ liệu điểm mây LiDAR để xây dựng bản đồ quan sát môi trường mà không cần đến các bản đồ lưới chiếm dụng toàn cầu, từ đó giúp tiết kiệm bộ nhớ và giảm tải tính toán. **EPIC** sử dụng một bản đồ quan sát được trích xuất từ chất lượng điểm mây, thay vì phải dựa vào các bản đồ chiếm dụng tốn kém, đồng thời giới thiệu một phương pháp xây dựng đồ thị topo gia tăng để lập kế hoạch đường đi theo thời gian thực. Các thành phần này giúp UAV khám phá môi trường một cách hiệu quả, nhanh chóng và tiết kiệm năng lượng.

Bài báo cũng so sánh **EPIC** với các phương pháp hiện đại và chứng minh rằng phương pháp của họ vượt trội về tốc độ khám phá và mức tiêu thụ bộ nhớ, giúp UAV thực hiện các nhiệm vụ khám phá trong các môi trường quy mô lớn một cách nhanh chóng và hiệu quả hơn.

**Cải tiến công nghệ (Contributions) của bài báo:**

1. **Khung phương pháp nhẹ dựa trên LiDAR**:
   * **EPIC** trực tiếp khai thác dữ liệu điểm mây LiDAR để khám phá môi trường quy mô lớn, loại bỏ việc sử dụng các bản đồ lưới chiếm dụng toàn cầu tốn bộ nhớ mà vẫn duy trì khả năng khám phá toàn diện.
2. **Bản đồ quan sát dựa trên điểm mây**:
   * Phương pháp này sử dụng **bản đồ quan sát** được trích xuất trực tiếp từ chất lượng của các điểm mây, giúp giảm thiểu mức tiêu thụ bộ nhớ mà vẫn cung cấp đủ thông tin để phát hiện biên giới và thực hiện khám phá hiệu quả.
3. **Xây dựng đồ thị topo gia tăng**:
   * **EPIC** giới thiệu một phương pháp **xây dựng đồ thị topo gia tăng** hoạt động trực tiếp trên các điểm mây, cho phép lập kế hoạch đường đi theo thời gian thực và tối ưu hóa quá trình tìm kiếm đường đi trong các môi trường quy mô lớn.
4. **Tối ưu hóa đường đi và quỹ đạo an toàn**:
   * Phương pháp này tạo ra các quỹ đạo linh hoạt và tiết kiệm năng lượng, giúp UAV di chuyển nhanh hơn và hiệu quả hơn trong môi trường phức tạp, đồng thời đảm bảo an toàn trong quá trình khám phá.
5. **Tiết kiệm bộ nhớ và tính toán**:
   * **EPIC** đã giảm đáng kể mức tiêu thụ bộ nhớ và thời gian tính toán so với các phương pháp hiện tại, cho phép UAV thực hiện các nhiệm vụ khám phá trong các môi trường rộng lớn mà không gặp phải các vấn đề về bộ nhớ và hiệu suất tính toán.

**Các cải tiến thể hiện qua các yếu tố:**

* **Hiệu quả bộ nhớ**: Phương pháp này không yêu cầu sử dụng các bản đồ chiếm dụng toàn cầu tốn bộ nhớ, thay vào đó sử dụng **bản đồ quan sát** được tối ưu hóa từ điểm mây.
* **Tối ưu hóa lập kế hoạch**: Việc sử dụng **xây dựng đồ thị topo gia tăng** và **tính toán đường đi theo thời gian thực** giúp giảm tải tính toán và tối ưu hóa việc khám phá.
* **Nâng cao tốc độ khám phá**: Các phương pháp tối ưu hóa quỹ đạo như **tạo ra các quỹ đạo linh hoạt và tiết kiệm năng lượng** giúp UAV khám phá các không gian lớn một cách nhanh chóng và hiệu quả.
* **Thử nghiệm thực tế**: Các thử nghiệm mô phỏng và thực tế cho thấy rằng **EPIC** vượt trội hơn so với các phương pháp hiện tại về tốc độ khám phá và giảm mức tiêu thụ bộ nhớ.