

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN ĐIỆN TỬ - VIỄN THÔNG



BÁO CÁO GIỮA KỲ
ĐIỆN TỬ TƯƠNG TỰ II
DỊCH TÀI LIỆU

Nhóm 49
Sinh viên 1 Tạ Hoàng Việt
20172916
Sinh viên 2 Hồ Huy Hoàng
20172581

Hà Nội, 06/2021

Mục lục

Danh sách hình vẽ

Danh sách bảng

Giới thiệu

Chương 1

Khám phá phối hợp

1.1 Giới thiệu

Việc thăm dò và lập bản đồ các khu vực rộng lớn là một lĩnh vực nghiên cứu tích cực trong lĩnh vực robot trên không. Nó bao gồm việc xây dựng mô hình 3D đại diện cho không gian làm việc khi rõ bốt tiến triển bên trong nó. Gần đây, chủ đề quan trọng trong vấn đề thăm dò là việc triển khai hợp tác nhiều robot hứa hẹn nâng cao hiệu suất so với thăm dò bằng robot đơn lẻ.

Trong chương này, chúng tôi giải quyết vấn đề của các chiến lược thăm dò hợp tác mà không có kiến thức ưu tiên về môi trường. Chúng tôi sẽ cố gắng trả lời câu hỏi: Tiếp theo mỗi robot nên di chuyển ở đâu?

1.2 Công việc có liên quan

Gần đây, một số công trình đã đề xuất các giải pháp thăm dò bằng cách sử dụng các đội nhiều robot để giảm thời gian nhiệm vụ và tăng khả năng mở rộng [**bautin2012strategie**], [**jensen2013rolling**], [**yan2014team**]. Do đó, thách thức là phải có sự hợp tác chặt chẽ giữa các đại lý trong hệ thống trong khi duy trì liên lạc [**rooker2007multi**]. Vì vậy, các phương pháp tiếp cận hiện có có thể được tập trung hóa, nghĩa là, một robot trong thiết bị chịu trách nhiệm giao các mục tiêu [**burgard2000collaborative**]. Trong [**schmuck2017multi**], một máy chủ trung tâm với nhiều tài nguyên tính toán được sử dụng để nhận, xử lý, tối ưu hóa và gửi lại thông tin cho các rô bốt khác. Các tác phẩm khác, chẳng hạn như [**yuan2010cooperative**], [**sheng2006distributed**], sử dụng các phương pháp tiếp cận phân tán trong đó mỗi robot chọn mục tiêu của riêng mình. Một xu hướng khác là chuyển từ hành vi khám phá cá

nhân sang hành vi hợp tác khi rõ bốt không thể hội tụ đến mức tối thiểu cục bộ với tỷ lệ đáp ứng [**wu2012robust**]. Trong [**konolige2003map**], các tác giả đề xuất xem xét bốn khả năng có tính đến tương tác của rõ bốt để xây dựng bản đồ phân tán. Các tình huống liên quan là: không tương tác (các robot không nằm trong phạm vi giao tiếp của nhau), tạo giả thuyết (có sự tương tác giữa các robot nhưng chúng không biết vị trí tương ứng của mình), giả thuyết xác minh (có sự tương tác giữa các robot và họ đề xuất giả thuyết về vị trí của mình) và khám phá phối hợp (có sự tương tác giữa các robot và chúng chia sẻ bản đồ của mình).

1.2.1 Chuyển nhượng rõ bốt đến mục tiêu

Thuật toán ánh xạ được thực hiện trong khi rõ bốt cố gắng tiếp cận mục tiêu. Vì vậy, để lập bản đồ môi trường điện tử, mục tiêu cần được lựa chọn cẩn thận. Có rất nhiều chiến lược gán mục tiêu cho một robot tới một mục tiêu. Phần lớn chúng tập trung và sử dụng hàm chi phí để tính toán tiện ích của việc đạt được mục tiêu.

Trong nhiệm vụ Greedy [**yamauchi1998frontier**], mỗi robot chọn một mục tiêu tùy thuộc vào chức năng chi phí của nó mà không cần phối hợp với các robot khác. Do đó, một mục tiêu có thể được truy cập bởi các robot khác nhau. Để giải quyết vấn đề này, có thể loại bỏ các mục tiêu đã chọn trước khi được người khác xem xét chuyển nhượng bằng cách sử dụng tính năng Phát sóng về tính đủ điều kiện tại địa phương (BLE) [**werger2000broadcast**] còn được gọi là Phép gán lặp lại. Tuy nhiên, phương pháp này không nhất thiết tạo ra giải pháp tối ưu vì nó phụ thuộc vào thứ tự của robot.

Phương pháp K-mean [**solanas2004coordinated**] bao gồm việc phân chia môi trường để khám phá thành các khu vực có cùng số lượng rõ bốt, sau đó, chỉ định một rõ bốt đến khu vực gần nhất nơi nó sẽ chọn mục tiêu từ các biên giới tùy thuộc vào hàm chi phí.

Phương pháp Hungary [**deb1999multi**] đề nghị bởi [**kuhn2005hungarian**], giải quyết công việc giao nhiệm vụ được viết dưới dạng $n \times n$ ma trận C ở $c_{i,j}$ là chi phí của nhiệm vụ j giao cho công nhân i . Bài tập tối ưu được tìm thấy với độ phức tạp về thời gian $O(n^3)$. Thuật toán này yêu cầu số lượng công nhân bằng số lượng nhiệm vụ không thể được đảm bảo. Ngoài ra, các rõ bốt hoặc mục tiêu tương ứng có thể được thêm vào để đáp ứng giả định và được bỏ qua sau đó trong lựa chọn.

Các tác giả trong [**nanjanath2006dynamic**] trình bày một phương pháp dựa trên đấu giá để giao nhiệm vụ cho một nhóm rõ bốt. Việc phân phối các nhiệm vụ được thực hiện bằng hình thức đấu giá ngược giá lần đầu có nghĩa là người bán đấu giá là người mua. Mỗi nhiệm vụ được bán đấu giá theo mức độ ưu tiên. Sau đó, đấu giá viên lựa chọn người trả giá tốt nhất và giao nhiệm vụ cho người trả giá tương ứng. Thuật toán này thích nghi tốt với các môi trường động, nơi các chướng ngại vật bất ngờ có thể ngăn cản rõ-bốt tiếp cận mục tiêu.

Trong [**zhao1996genetic**], [**leigh2007using**], một thuật toán di truyền được sử dụng để giao nhiệm vụ cho robot một cách tối ưu. Đây là một bài toán khó NP vì nó được coi là một bài toán đóng gói thùng đa loại hai chiều tổng quát.

Các tác giả trong [**faigl2012goal**] ppfan đổi một cách tiếp cận mới được gọi là Bài toán nhân viên bán hàng đi du lịch nhiều nơi (MTSP) để phân công robot với mục tiêu trong bài toán khám phá nhiều robot. Nó bao gồm việc phân cụm một môi trường, xác định chi phí khoảng cách TSP cho mỗi cặp cụm robot, chỉ định mục tiêu từ cụm không trống và xác định mục tiêu cuối cùng cho các cụm trống. Cách tiếp cận này được so sánh với phương pháp tham lam, lặp đi lặp lại và phương pháp Hungarian. MTSP đưa ra các kết quả cạnh tranh về tổng số yêu cầu tính toán và số lượng rõ bốt ngày càng tăng.

Trong [**kulich2015comparison**], việc so sánh một số chiến lược phân công được sử dụng cho hệ thống nhiều robot được thực hiện. Các chiến lược được so sánh bao gồm phương pháp Hungarian, Greede, phương pháp BLE và phân cụm K-mean. Kết quả cho thấy rằng phương pháp Hungarian tốt hơn các phương pháp khác trong phần lớn các trường hợp. Không giống như phương pháp Hungarian, phép gán lặp đi lặp lại có thể được thực hiện trong một môi trường phân tán. Ngoài ra, các phương pháp Hungarian nặng về mặt tính toán so với thuật toán Greedy đơn giản được ưu tiên trong kịch bản áp dụng.

1.2.2 Chức năng tiện ích

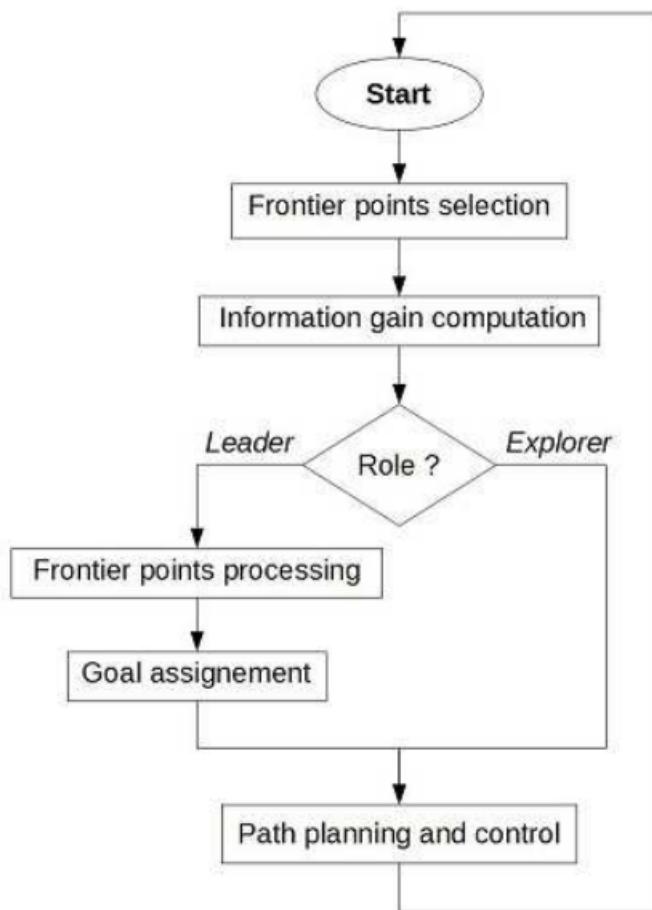
Hầu hết các bài tập robot đến mục tiêu được dựa trên chức năng tiện ích mà các lợi ích mà lợi ích mà robot phải đạt được mục tiêu này, có tính đến mục tiêu của nhiệm vụ [**burgard2000collaborative**]. Công việc đề xuất trong [**benavides2016multi**] trình bày một chức năng tiện ích mới có tính đến chi phí di chuyển đến mục tiêu và tiện ích kết nối. Điều này cho phép

giao dịch O FF giữa việc giảm thiểu số lượng thời gian thăm dò và kết nối. Để tăng tốc độ vận tốc, các tác giả trong [cieslewski2017rapid] Đề xuất một kỹ thuật lựa chọn biên giới nhanh chóng để chọn mục tiêu từ lượt xem của robot. Cách tiếp cận này giảm thiểu thời gian nhiệm vụ tổng thể bằng cách giảm thiểu sự thay đổi vận tốc của robot. Tuy nhiên, cách tiếp cận này làm tăng tổng chiều dài đường dẫn đi. Trong [heng2015efficient], Tối đa hóa mô hình tái tạo được ưa chuộng trong thời gian nhiệm vụ. Hơn nữa, cách tiếp cận được đề xuất giải quyết đồng thời các vấn đề thăm dò và bảo hiểm để tối đa hóa tính đầy đủ của mô hình tái tạo. Trong khi ở [simmons2000coordination], Mục đích là tối đa hóa các tiện ích của các mục tiêu giảm thiểu tiềm năng chồng chéo trong việc tăng thông tin giữa các thành viên của FL EET. Tiện ích đạt được mục tiêu phụ thuộc về cơ bản về mục đích của nhiệm vụ trong khi tính đến một số ràng buộc bổ sung như thời gian, tính đầy đủ của bản đồ, cảm biến hạn chế và phạm vi giao tiếp, hoặc số lượng robot.

1.3 Đề xuất chiến lược thăm dò

1.3.1 Tổng quát

Trong một hệ thống đa UAV, chiến lược thăm dò cần phải hợp tác để tối đa hóa sự nghiêm túc của FFI. Mục tiêu chính được đề xuất ở đây, là hợp tác chọn các khu vực cụ thể để được khám phá đồng thời bằng cách sử dụng phương pháp tiếp cận dựa trên biên giới. Thông thường, điều này được thực hiện bằng cách chọn mục tiêu ứng viên và gán chúng cho mỗi robot một cách được tối ưu hóa. Nhân vật ?? cho thấy quá trình thăm dò đường ống được đề xuất được thực hiện bởi mỗi UAV.

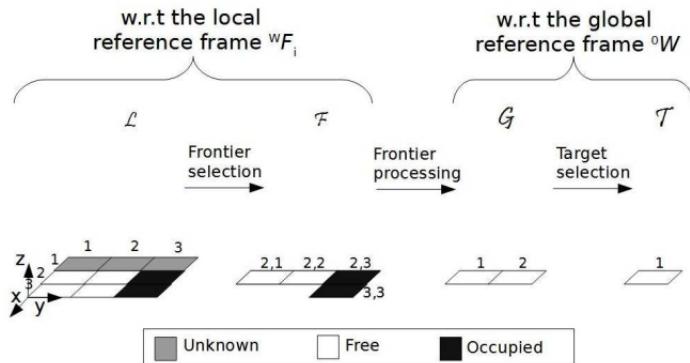


Hình 1.1: Đề xuất đường ống quá trình thăm dò.

Mỗi UAV chọn các điểm biên giới của bản đồ cục bộ được xây dựng trong bước SLAM.Sau đó, nó tính toán thông tin tương ứng của các điểm này.Nếu vai trò của UAV là một *explorer*, nó sẽ thụ động chờ cho đến khi nhận được hướng dẫn;khác, nếu nó hoạt động như một *leader*, Nó sẽ xử lý các điểm biên giới được thu thập, và sẽ chỉ định một mục tiêu cho mỗi robot trong nhóm.Dựa ra một mục tiêu, UAV sẽ lên kế hoạch cho một đường dẫn cụ thể để đạt được nó.

Trong các bước này, cấu trúc bản đồ phát triển từ bản đồ lưới 2D dự kiến thu được từ mô-đun bản đồ và bản đồ, thông qua các điểm biên giới trong

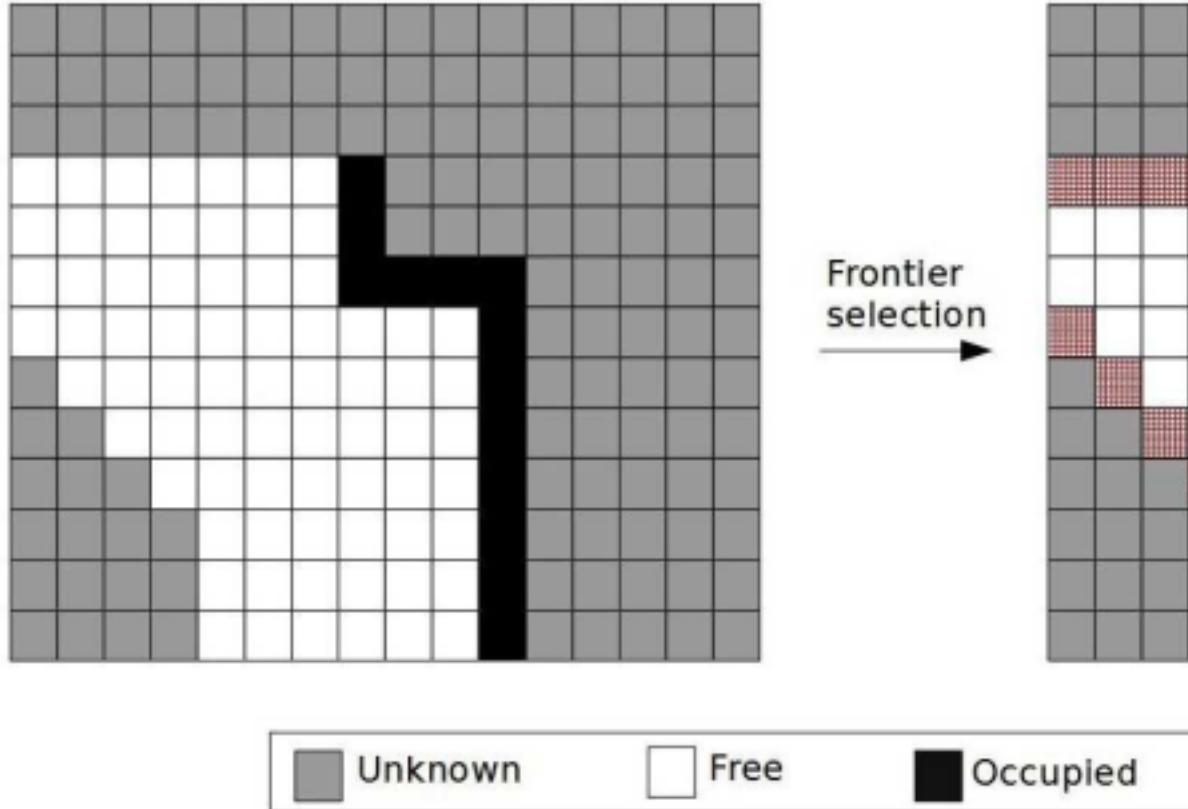
quá trình lựa chọn biên giới, sau đó các mục tiêu ứng cử viên trong bước xử lý biên giới, đến một tập các mục tiêu được chọn để đạt được. Sự phát triển của cấu trúc bản đồ được minh họa trong hình ???. Thuật toán ?? mô tả các bước chính được thực hiện trong quá trình thăm dò.



Hình 1.2: Tiến hóa cấu trúc bản đồ trong quá trình thăm dò: Từ các ô 2D \mathcal{L} đến Ô biên giới 2D. \mathcal{F} để ứng cử viên biên giới / điểm \mathcal{G} (mục tiêu ứng viên) đến 2D ô mục tiêu / điểm \mathcal{T} .

1.3.2 Lựa chọn điểm biên giới

Quá trình lựa chọn biên giới được sử dụng để định nghĩa biên giới của các vùng bị giới hạn bởi các chướng ngại vật hoặc không gian không xác định (xem hình ??).



Hình 1.3: Các tế bào biên giới / điểm lựa chọn bản đồ lưới phòng chiếm 2D.

Các tế bào biên giới. $\mathbf{f}_{i,j} \in \mathcal{F}$ được chọn từ tập hợp các ô $\mathcal{L}(\mathcal{F} \subset \mathcal{L})$ sao cho họ là:

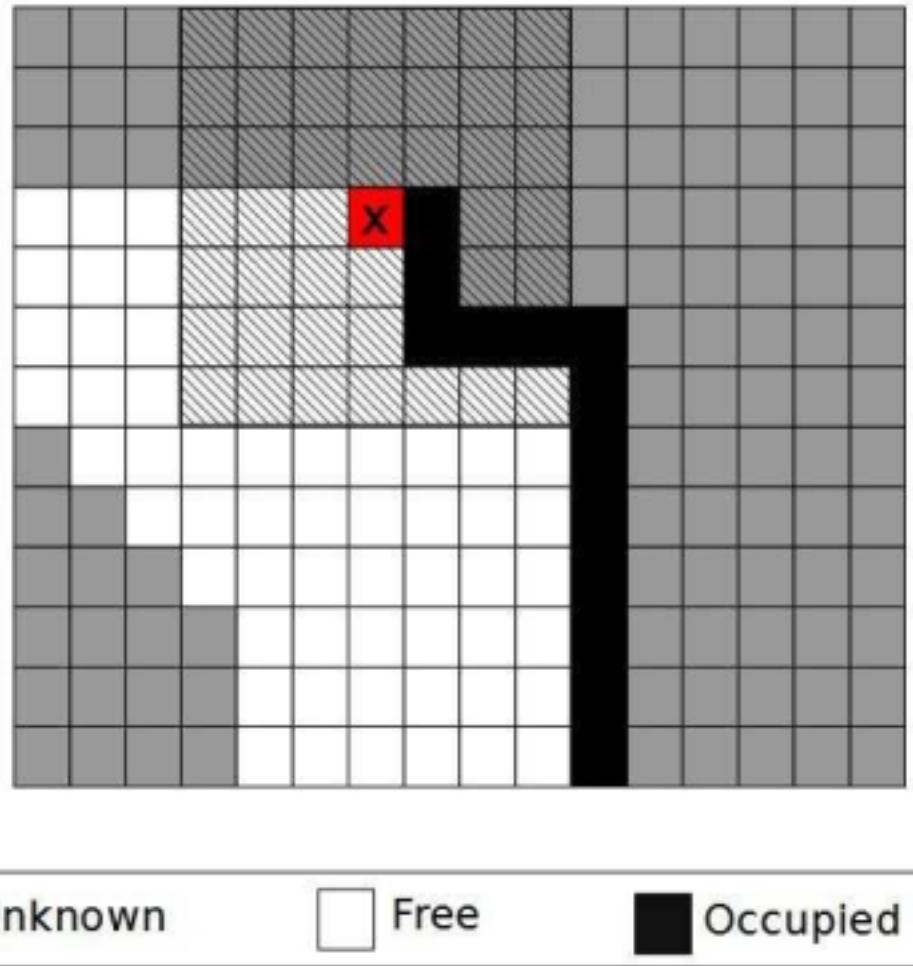
- Miễn phí \mathbf{l}_f và liền kề không rõ.
- Dán nhãn là chiếm đóng \mathbf{l}_o . Các ô bị chiếm đóng \mathbf{l}_o được coi là các tế bào biên giới để có thể thực hiện xử lý biên giới trong bước tiếp theo. Chúng không thể được chọn làm mục tiêu và sẽ bị loại bỏ sau.

Tại hình ??, Các ô biên giới là: $\mathbf{l}_f(2,1), \mathbf{l}_f(2,2), \mathbf{l}_f(2,3)$, và $\mathbf{l}_f(3,3)$. Vì vậy, đối với một cụm \mathcal{C} chứa UAV_i , các biên giới là: $F = \{\mathbf{f}_{i,1}(2,1), \mathbf{f}_{i,2}(2,2), \mathbf{f}_{i,3}(2,3), \mathbf{f}_{i,4}(3,3)\}$

1.3.3 Thông tin đạt được tính toán

Trong các phương pháp thăm dò dựa trên biên giới, chỉ có các tế bào liền kề với những tế bào chưa biết có thể được thực hiện dưới dạng các điểm biên giới ứng cử viên và có khả năng sẽ được chọn làm mục tiêu. Qua đó, mức tăng thông tin được liên kết với từng người trong số chúng để ước tính tiện ích tiếp cận từng biên giới. Việc đạt được thông tin tương ứng này có thể được thực hiện trong cách phân cách dirent erent tùy thuộc vào mục đích nhiệm vụ. Tác giả trong [burgard2005coordinated] Đề xuất sử dụng chức năng xác suất để giảm giá trị không đổi được chỉ định có tính đến khoảng cách tương đối với tư thế của UAV. Chiến lược này là chung và không tính đến các ô được cập nhật được cập nhật. Cách tiếp cận được đề xuất trong [heng2015efficient] FF ECTS, để đạt được thông tin, số lượng ô không xác định và không bị chặn trong chế độ xem FrusM của mục tiêu. Phương pháp này phụ thuộc vào ước tính thực tế của thông tin đạt được khi đến thăm tư thế được coi là. Tuy nhiên, nó đòi hỏi nhiều tính toán hơn.

Trong chiến lược được đề xuất, mức tăng thông tin được phân bổ để nó không xác định lượng các ô không xác định xung quanh mục tiêu (xem hình ??). Nó là một giá trị phi số liệu đếm số lượng các ô được dán nhãn là không xác định l u từ 48 ô xung quanh điểm biên giới.



Hình 1.4: Tính toán thông tin đạt được điểm Frontier X (màu đỏ). Các tế bào nở đại diện cho 48 tế bào xung quanh của điểm biên cương x. Số đạt được thông tin của x là: $I(x) = 25$.

1.3.4 Xử lý điểm biên giới

Tất cả các điểm biên giới. $\mathbf{f}_{i,j} \in \mathcal{F}$ của UAV_i. Trong cụm \mathcal{C} với $i \in [1..n_c]$, được thu thập. Điểm trong \mathcal{F} sau đó được xử lý bằng thuật toán ?? để có được các điểm biên giới ứng cử viên được coi là mục tiêu ứng cử viên $\mathbf{t}_k \in \mathcal{G}$ với $k \in [1..n_g]$ (Xem hình ??) [H] **Input:** Điểm biên giới $\mathbf{f}_{i,j} \in \mathcal{F}$ của

UAV_i với $i \in [1..n_c]$.

Output: Candidate targets \mathcal{G}

- 1: $p_u = \bigcup_{i=1}^{n_c} \mathbf{f}_{i,j}$.
- 2: $p_i = \bigcap_{i=1}^{n_c} \mathbf{f}_{i,j}$.
- 3: $\mathcal{G} = p_u \setminus p_i$.
- 4: Xóa các điểm biên giới chương ngại vật $\mathbf{f}_{i,j}(x, y) = \mathbf{l}_o(x, y)$ from \mathcal{G} .
- 5: **return** \mathcal{G} .

Hình ?? Hiển thị một ví dụ về xử lý biên giới với hai bản đồ của UAV để có được các mục tiêu ứng cử viên.Các điểm biên giới của chương ngại vật $\mathbf{f}_{i,j}(x, y) = \mathbf{l}_o(x, y)$ – Được dán nhãn là chiếm đóng - chỉ được giữ để tính toán giao điểm của các điểm biên cương.Chỉ các ô Frontier miễn phí \mathbf{l}_f có thể được coi là mục tiêu ứng cử viên.

Khi sử dụng điểm biên giới cục bộ thay vì bản đồ cục bộ, quy trình biên giới thay thế quy trình khớp bản đồ trong đó mục tiêu là để xóa các khu vực chồng chéo.Do đó, trong bước xử lý biên giới, các điểm biên giới thuộc các khu vực chồng chéo bị xóa.Do đó, sử dụng điểm biên giới cho phép tiết kiệm bộ nhớ quan trọng.Để tính các điểm biên giới thuộc về các khu vực chồng chéo (các bước 1, 2 và 3 trong thuật toán ??), Chúng tôi đề xuất hai cách tiếp cận với các hình dạng lồi và các giả định hình dạng lõm.

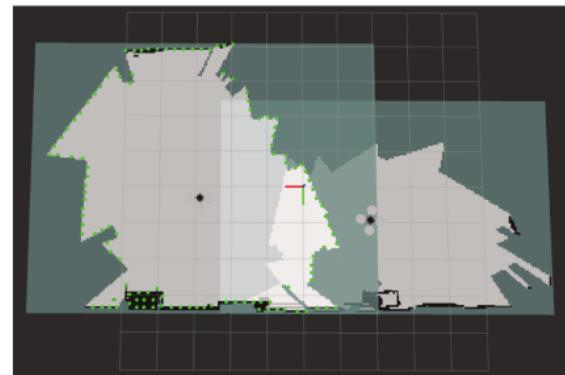
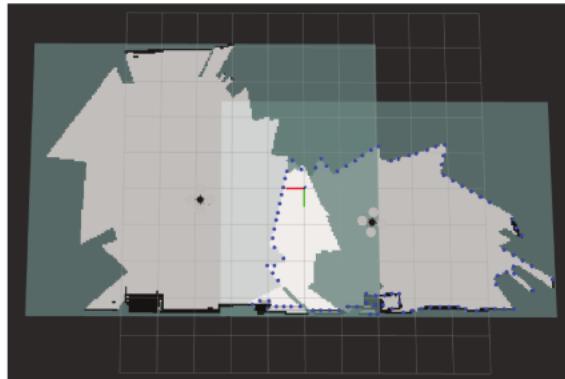
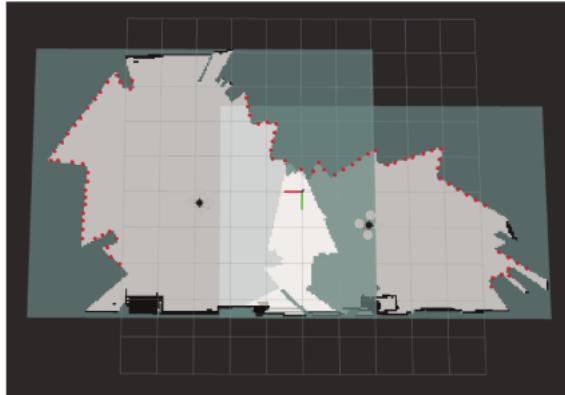
Cách tiếp cận đầu tiên: Bản đồ hình convex

Trong cách tiếp cận này, các hình dạng bản đồ được coi là convex, đơn giản hóa tính toán giao nhau trong thuật toán ???. Hình ?? Hiển thị hai ví dụ về việc áp dụng thuật toán xử lý biên giới trong khi giả sử các hình lồi với hai và ba bản đồ của UAV.

Thuật toán này tương đối dễ dàng để áp dụng, nhưng kết quả cho thấy rằng nó không hoạt động tốt.Giả định lồi dẫn đến một số điểm biên giới chồng chéo sai.Do đó, một số khu vực chưa biết sẽ không được truy cập vì các điểm tương ứng của chúng được gỡ bỏ và do đó, sẽ không được chỉ định là mục tiêu.

Cách tiếp cận thứ hai: Bản đồ hình dạng lõm

Trong cách tiếp cận thứ hai, chúng tôi đưa ra giả định về các hình dạng lõm cho bản đồ của UAV để tính toán giao điểm của chúng trong thuật toán ???. Hình ?? Hiển thị hai ví dụ về việc áp dụng thuật toán xử lý biên giới trong khi giả sử các hình lõm với hai và ba bản đồ của UAV.

(a) Frontiers $f_{1,j}$ of UAV₁(b) Frontiers $f_{2,j}$ of UAV₂(c) Candidate targets \mathcal{G}

Hình 1.5: Ví dụ xử lý điểm biên giới: (a) và (b) đại diện cho các điểm biên giới của UAV₁ và UAV₂, tương ứng. (c) đại diện cho các mục tiêu ứng cử viên. Quá trình này bao gồm loại bỏ các điểm biên giới thuộc về các khu vực chồng chéo, và những điểm nằm liền kề với các chướng ngại vật.

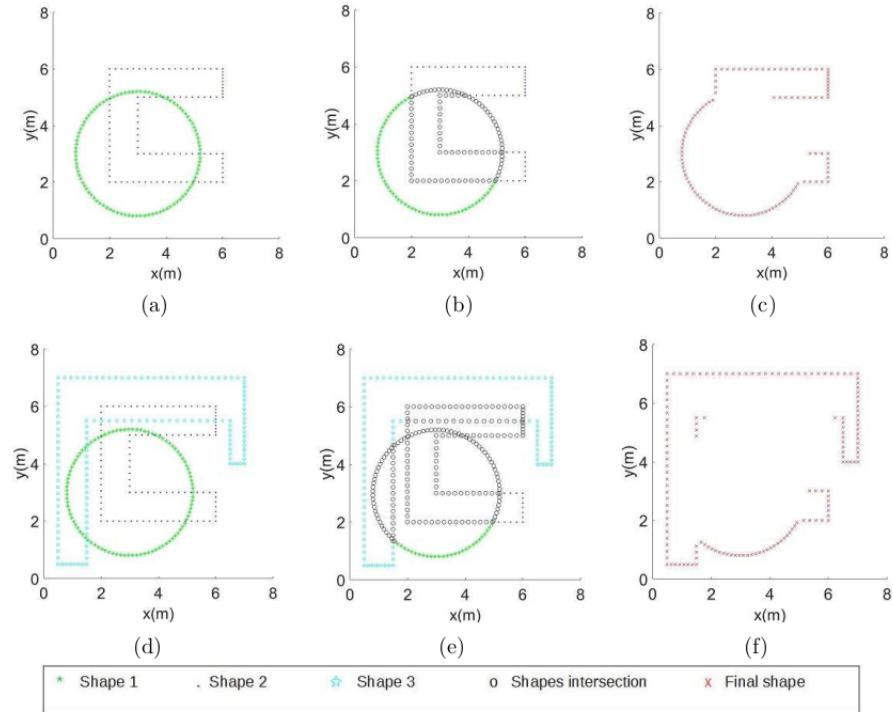
Algorithm 2 Thuật toán tính toán giao nhau với giả định hình dạng lồi.

```

Input: Vect1, Vect2
Output: Vect_Final

1: for  $i \in Vect1$  do
2:    $inf = 0, sup = 0$ 
3:   for  $j \in Vect2$  do
4:     if  $Vect1(i).x < Vect2(j).x$  then
5:        $inf ++;$ 
6:     else
7:        $sup ++;$ 
8:     end if
9:   end for
10:  if  $inf > 0$  and  $sup > 0$  then
11:    Vect_inter.push_back(Vect1(i))
12:  end if
13: end for
14: for  $i \in Vect\_inter$  do
15:    $inf = 0, sup = 0;$ 
16:   for  $j \in Vect2$  do
17:     if  $|Vect\_inter(i).x - Vect2(j).x| < 0.5$  then
18:        $inf ++;$ 
19:     else
20:        $sup ++;$ 
21:     end if
22:   end for
23:   if  $inf > 0$  and  $sup > 0$  then
24:     Vect_Final.push_back(Vect_inter(i))
25:   end if
26: end for

```



Hình 1.6: Điểm biên giới xử lý Wile giả định hai hình dạng bắn đồ lồi trong dòng đầu tiên và ba trong dòng thứ hai.(a) và (d) biểu thị hai và ba hình dạng bắn đồ lồi, tương ứng;(b) và (e) đại diện cho các điểm biên cương theo chồng chéo (giao lộ);và (c) và (f) đại diện cho các điểm biên giới thu được sau khi xử lý (hình dạng nal).

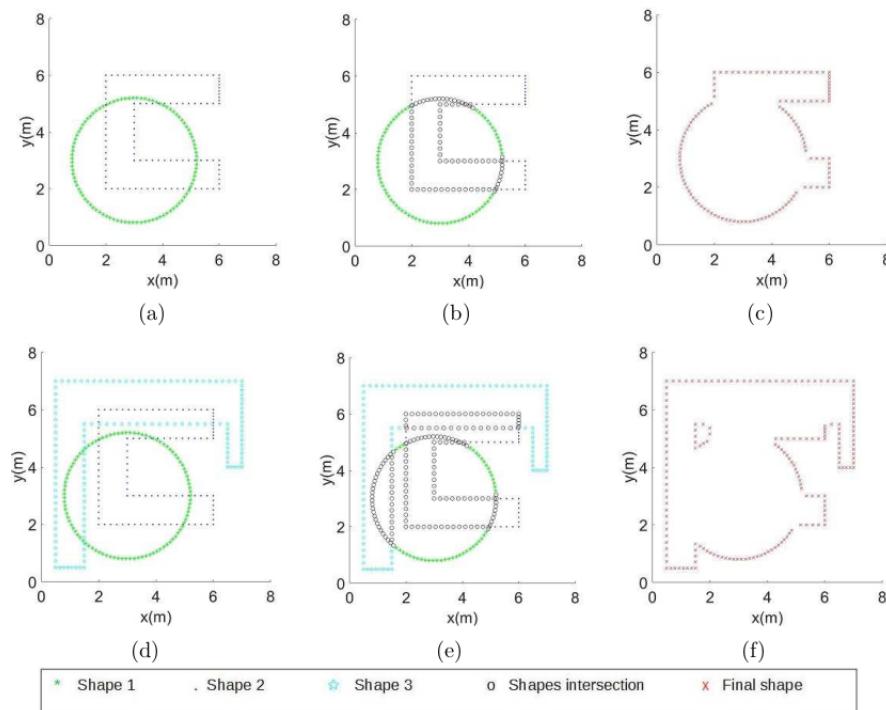
Algorithm 3 Thuật toán tính toán giao nhau với giả định hình dạng lồi.

```

Input: Vect1, Vect2
Output: Vect_Final

1: for  $i \in Vect1$  do
2:    $inf = 0, sup = 0$ 
3:   for  $j \in Vect2$  do
4:     if  $Vect1(i).x < Vect2(j).x - 0.5$  then
5:        $inf ++;$ 
6:     else if  $Vect1(i).x > Vect2(j).x + 0.5$  then
7:        $sup ++;$ 
8:     else
9:        $eq ++;$ 
10:    end if
11:   end for
12:   if  $inf > 0$  and  $sup > 0$  or  $eq > 0$  then
13:     Vect_inter.push_back(Vect1(i))
14:   end if
15: end for
16: for  $i \in Vect\_inter$  do
17:    $inf = 0, sup = 0, eq = 0;$ 
18:   for  $j \in Vect2$  do
19:     if  $|Vect\_inter(i).y - Vect2(j).y| < 0.5$  then
20:       if  $Vect\_inter(i).x < Vect2(j).x - 0.5$  then
21:          $inf ++;$ 
22:       else if  $Vect1(i).x > Vect2(j).x + 0.5$  then
23:          $sup ++;$ 
24:       else
25:          $eq ++;$ 
26:       end if
27:     end if
28:   end for
29:   if ( $inf > 0$  and  $sup > 0$ ) or  $eq > 0$  then
30:     Vect_Final.push_back(Vect_inter(i))
31:   end if
32: end for

```



Hình 1.7: Điểm biên giới xử lý Wile giả định hai hình dạng bản đồ lõm trong dòng đầu tiên và ba trong dòng thứ hai.(a) và (d) biểu thị hai và ba hình dạng bản đồ lõi, tương ứng;(b) và (e) đại diện cho các điểm biên cương theo ch่อง chéo (giao lộ);và (c) và (f) đại diện cho các điểm biên giới thu được sau khi xử lý (hình dạng nal).

Kết quả cho thấy chỉ các điểm biên giới trong các khu vực ch่อง chéo được loại bỏ.Ngay cả khi một số điểm biên giới mơ hồ được đặt trong / bên trong hình dạng toàn cầu - có thể gây nhầm lẫn -, chúng không bị xóa.Thuật toán này, giả sử các hình dạng lõm, thực hiện tốt hơn so với trước đó giả định hình dạng lõi.Do đó, đối với chế biến biên giới, hình dạng của bản đồ địa phương được giả định lõm.

Chức năng tiện ích

Chức năng tiện ích được đề xuất trong EQ. ?? Nhằm mục đích tăng đồng thời tỷ lệ khu vực được khám phá và để giảm khoảng cách của từng UAV đến mục tiêu tương ứng.Chức năng này cũng xem xét trung bình khoảng cách giữa mỗi robot trong nhóm và mục tiêu này để tối đa hóa khoảng cách

giữa các robot.

$$\mathbf{U}(UAV_i, t_j) = \mathbf{I}(t_j) \exp(-\lambda \cdot (dmin(\mathbf{P}_i, \mathbf{t}_j) + \frac{n_c - 1}{d_{tot}})), \quad (1.1)$$

Nơi UAV tôi là robot được coi là, $\mathbf{t}_j \in \mathcal{G}$ và $\mathbf{I}(\mathbf{t}_j)$ tương ứng là mục tiêu ứng viên và mức tăng thông tin tương ứng của nó, $\lambda \in [0, 1]$ là một tham số Trade-off, n_c là số lượng UAV trong cụm \mathcal{C} , and $d_{tot} = \sum_{k=1, k \neq i}^{n_c} (dmin(\mathbf{P}_k), \mathbf{t}_j)$ là tổng của khoảng cách tối thiểu từ UAV_k's tư thế \mathbf{P}_k đến mục tiêu ứng viên j . Chức năng tiện ích được đề xuất được truyền cảm hứng từ [heng2015efficient] và nó đã được trình bày trong các tác phẩm của chúng tôi [mahdoui2017cooperative],[mahdoui2018cooperative]. Hàm này thực hiện giao dịch OF FF giữa khám phá nhanh chóng và một bản đồ chính xác bằng cách sử dụng tham số điều chỉnh λ . Từ hình ??, Chúng ta có thể nhận thấy rằng lớn hơn λ , Khoảng cách ít quan trọng hơn d_{tot} . Do đó, một cách chính xác được ưa chuộng về việc thăm dò nhanh chóng và ngược lại.

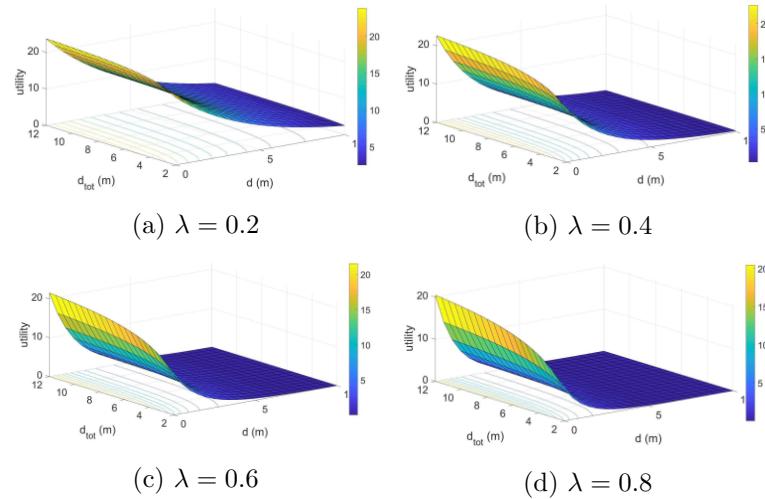
Về trường hợp đa UAV, chức năng tiện ích dựa trên trung bình khoảng cách hàng xóm.Như thể hiện trong hình ??, với một thông tin đạt được $\mathbf{I}(\mathbf{t}_j) = 25$ và ba UAV trong cụm ($n_c = 3$); Một khoảng cách ngày càng tăng của UAV đến mục tiêu sẽ làm giảm chức năng tiện ích.Trong khi đó, khoảng cách trung bình lớn hơn so với các UAV khác w.r.t.đến mục tiêu, tiện ích càng nhiều.Vì vậy, chức năng có xu hướng chọn mục tiêu gần nhất với UAV được coi là;Nhưng đồng thời, xa nhất từ những người khác.

Trong trường hợp một UAV duy nhất, chức năng tiện ích (xem EQ. ??) Có xu hướng chọn mục tiêu gần nhất với mức tăng tối đa thông tin:

$$\mathbf{U}(UAV_i, \mathbf{t}_j) = \mathbf{I}(\mathbf{t}_j) \exp(-\lambda \cdot (dmin(\mathbf{P}_j, \mathbf{t}_j))), \quad (1.2)$$

Thông số $d_{tot} = \sum_{k=1, k \neq i}^{n_c} (dmin(\mathbf{P}_k, \mathbf{t}_j))$ đại diện cho tổng khoảng cách tối thiểu giữa mục tiêu \mathbf{t}_j và các vị trí của hàng xóm \mathbf{P}_k với $k \in [1..n_c] \setminus i$. Vì thế nếu \mathbf{t}_j Có các UAV lân cận quá xa, chức năng tiện ích sẽ tăng lên, vì vậy T J có nhiều khả năng được chọn.

Mục tiêu trong chức năng tiện ích là tối đa hóa d_{tot} . Chúng tôi phân biệt hai trường hợp d_{tot} có thể quá gần bằng 0 hoặc bằng nó:



Hình 1.8: Hành vi chức năng tiện ích: $\mathbf{I}(t_j) = 25, n_c = 3, d_{tot} = \sum_{k=1, k \neq i}^{n_c} (d_{min}(\mathbf{P}_k, \mathbf{t}_j))$. Khoảng cách trung bình của khán UAVs d_{tot} có giá trị tối thiểu khác với số 0 kể từ khi n_c khác với 0 quá.

- Không có hàng xóm. Trong trường hợp này, có một UAV trong FL EET: $n_c = 1$. Chức năng tiện ích trong EQ. ?? được sử dụng.
- Mục tiêu ứng cử viên \mathbf{t}_j và các vị trí của hàng xóm \mathbf{P}_k với $k \in [1..n_c] \setminus i$ gần như bối rối (quá gần nhau). Trong trường hợp này \mathbf{t}_j ít có khả năng được chọn.

Tuy nhiên, tham số này d_{tot} có thể có giá trị cao khi:

- Có một UAV quá xa \mathbf{t}_j .
- Có một số UAV với khoảng cách tương đối ngắn từ \mathbf{t}_j .

Vì vậy, để tránh những tình huống mơ hồ này, trung bình của d_{tot} được xem xét bằng cách sử dụng $\frac{n_c}{d_{tot}}$.

1.3.5 Quy trình phân công mục tiêu

Để thực hiện phân công UAV-to-nhóm mục tiêu thích hợp, tiện ích tiếp cận từng biên giới ứng cử viên được xem xét. Quá trình phân công mục tiêu được mô tả trong Algo. ??.

Cho mỗi UAV_i , Các tiện ích để đạt được tất cả các mục tiêu ứng cử viên được tính toán.Mục tiêu \mathbf{t}_g tối đa hóa tiện ích được tính toán và, bài tập $\theta(\text{UAV}_i, \mathbf{t}_g)$ được thực hiện.Sau đó, các mục tiêu ứng cử viên còn lại $\mathcal{G} \setminus \mathcal{T}$ được lên kế hoạch để tránh để chọn một mục tiêu quá gần \mathbf{t}_g , trong lần lặp tiếp theo.Cũng thế, \mathbf{t}_g được gỡ bỏ khỏi \mathcal{G} để ngăn chặn việc chỉ định cùng một mục tiêu cho di ff robot erent erent.Quá trình chuyển nhượng này được thực hiện cho các UAV có sẵn trong \mathcal{C} theo cách tuần tự cho đến khi nhận được tất cả các mục tiêu được chỉ định $\mathbf{t}_g \in \mathcal{T}$ với $g \in [1..n_t]$ (Xem hình ??)

Algorithm 4 Thuật toán gán mục tiêu.

Input: Mục tiêu ứng viên $\mathbf{t}_k \in \mathcal{G}, k \in [1..n_g]$ và thông tin tương ứng của họ đạt được $\mathbf{U}(\mathbf{t}_k)$, poses \mathbf{P}_i của tất cả các robot trong cụm được coi là \mathcal{C} .

Output: $\theta(\text{UAV}_i, \mathbf{t}_g)$ phân công UAV_i với mục tiêu g .

- 1: $\mathcal{T} = \emptyset$
 - 2: **while** Không có mục tiêu cho UAV_i **do**
 - 3: Tính toán tiện ích tương ứng của nó để đạt được mỗi mục tiêu ứng cử viên còn lại $\mathbf{U}(\text{UAV}_i, \mathbf{t}_k)$ với $\mathbf{t}_k \in \mathcal{G} \setminus \mathcal{T}$.
 - 4: $t_g = \text{argmax}_{t_k \in \mathcal{G} \setminus \mathcal{T}} (\mathbf{U}(\text{UAV}_i, \mathbf{t}_k))$.
 - 5: Lên lịch để đạt được thông tin của các ứng cử viên còn lại $\mathbf{t}_k \in \mathcal{G} \setminus \mathcal{T}$.
 - 6: $\mathcal{T} = \mathcal{T} \cup t_g$.
 - 7: **end while**
 - 8: **return** $\theta(\text{UAV}_i, \mathbf{t}_g)$ assignment.
-

Stop condition

Quá trình lựa chọn mục tiêu được thực hiện bởi mỗi cluster/group-leader (if $n > n - c$) hoặc đội tàu- leader (if $n = n_c$). Nhiệm vụ này nhằm mục đích, hợp tác, phân phối robot trong môi trường để khám phá đồng thời di ff erent khu vực không xác định.Miễn là các điểm biên giới ứng cử viên vẫn có sẵn, leader tiếp tục gán mục tiêu cho explorers và họ cố gắng đạt được mục tiêu được giao.Khi mà leader Thông báo rằng không có mục tiêu ứng cử viên nào còn lại, điều đó có nghĩa là tất cả các môi trường đã được khám phá thành công và nhiệm vụ được thực hiện.Vì vậy, nó phải gửi lại cho explorers một sự thừa nhận để ngăn chặn họ giả định mất giao thông.

Loop rate

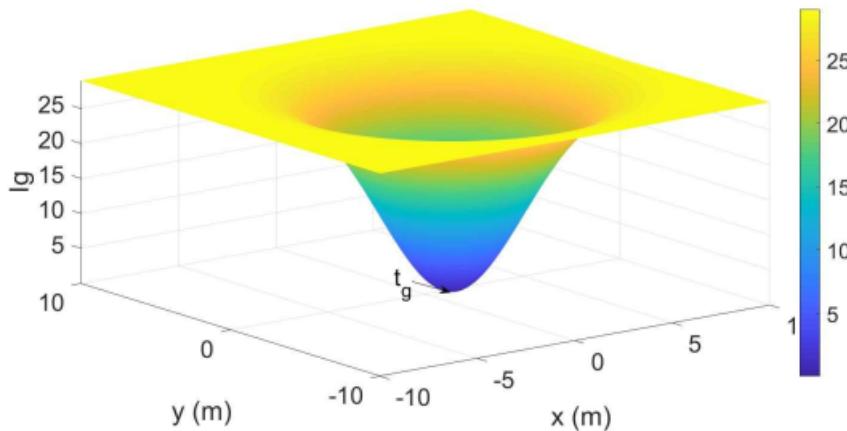
Quá trình phân công mục tiêu được thực hiện tại mỗi vòng lặp.Tần suất gán các mục tiêu ảnh hưởng đến thời gian và sự nghiêm trọng của FFI của nhiệm vụ.Trong một cách tiếp cận phân tán, ngay khi UAV đạt được mục tiêu hiện tại, nó sẽ chọn một mục tiêu mới mà không cần tham khảo ý kiến người khác.Trong một cách tiếp cận tập trung, UAV đầu tiên để đạt được mục tiêu hiện tại phải đợi cho đến khi những người khác đạt được mục tiêu tương ứng.Đây có thể là một vấn đề ngay khi một trong số họ thất bại hoặc rời khỏi nhiệm vụ.Một khả năng khác là bắt đầu gán các mục tiêu sau khi UAV đạt được mục tiêu.Nhưng điều này có thể tạo ra các nhiệm vụ không đầy đủ.Trong chiến lược đề xuất, tần suất phân công hoặc tỷ lệ vòng lặp r là fì ned tùy thuộc vào trung bình thời gian để đạt được một mục tiêu (Xem Eq. ??).

$$r \in \left[\frac{s}{v_{i,max}}, \frac{s}{v_{i,min}} \right] \quad (1.3)$$

Ở s là phạm vi cảm biến tối đa và v_i là vận tốc của UAV.

Lên lịch để đạt với ược thông tin

Tăng thông tin của từng mục tiêu ứng cử viên còn lại $\mathbf{I}(\mathbf{t}_k)_{t-1}$ tại thời điểm $t-1$ với $\mathbf{t}_k \in \mathcal{G} \setminus \{t_g\}$, thuộc về phạm vi $[r_{min}, r_{max}]$, được lên kế hoạch tại thời điểm t tùy theo khoảng cách của nó w.r.t.mục tiêu t_g , sử dụng Eq. ???. Hình ?? đại diện cho một ví dụ về hình dạng chức năng được sử dụng để lên lịch để đạt được thông tin.Ví dụ cho thấy một hàm Gaussian với biên độ tương ứng với giá trị tối da thông tin;một trung tâm tương ứng với vị trí mục tiêu $(\mathbf{t}_g(x), \mathbf{t}_g(y))$; và σ_x và σ_y nó lan rộng blob trong x và y trục, tương ứng.



Hình 1.9: Ví dụ về chức năng hình dạng để lên lịch để đạt được thông tin($\mathbf{I}(\mathbf{t}_k)_{t-1} = 29$, $(\mathbf{t}_g(x), \mathbf{t}_g(y)) = (1.25, -0.275)$, $\sigma_x = \sigma_y = 3$).

$$\mathbf{I}(\mathbf{t}_k)_t = \mathbf{I}(\mathbf{t}_k)_{t-1} \left(1 - \exp\left(-\left(\frac{(\mathbf{t}_k(x) - \mathbf{t}_g(x))^2}{2.\sigma_x^2} + \frac{(\mathbf{t}_k(y) - \mathbf{t}_g(y))^2}{2.\sigma_y^2}\right)\right)\right), \quad (1.4)$$

Ở $\mathbf{t}_k(x)$ and $\mathbf{t}_k(y)$ là tọa độ mục tiêu ứng cử viên còn lại; $\mathbf{t}_g(x)$ và $\mathbf{t}_g(y)$ là tọa độ mục tiêu; và σ_x và σ_y là những lây lan của blob. Khoảng cách của điểm biên giới nhỏ hơn \mathbf{t}_k w.r.t. mục tiêu \mathbf{t}_g , sự tăng thông tin càng nhỏ. Khi giảm mức tăng thông tin, các mục tiêu ứng cử viên ít có khả năng được chọn và do đó, robot đảm bảo một khoảng cách nhất định trong số các mục tiêu trong tương lai của họ.

1.3.6 Lập kế hoạch và kiểm soát đường dẫn

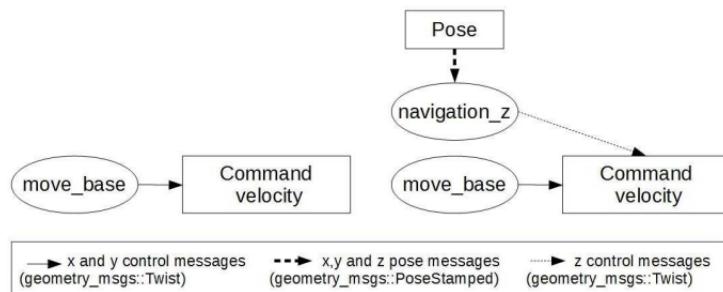
AS được giải thích trong Phần 5.2 của Chương 1, UAV được giả định là điều hướng trong môi trường 2D đơn giản với A* FI XED z giá trị. Khối 6 Trong Hình 1.6 chịu trách nhiệm lên kế hoạch đường dẫn đến mục tiêu đã chọn và cố gắng tiếp cận nó. Những nhiệm vụ này được đảm bảo bởi move cẩn cứ¹ gói.

Đối với nhiệm vụ điều hướng, mỗi UAV duy trì một công cụ lập kế hoạch địa phương và toàn cầu cùng với một costmap địa phương và toàn cầu. COSTMAP là một lưới ô 2D \mathcal{L} . Với bổ sung trong fl ation bao gồm việc truyền các

¹Nguồn: http://wiki.ros.org/move_base

giá trị chi phí từ các tế bào bị chiếm đóng và giảm chúng bằng khoảng cách.CostMap toàn cầu có kích thước của bản đồ của UAV trong khi đó, costmap địa phương có một cửa sổ di chuyển kích thước fixxed.Đưa ra một điểm khởi đầu - tư thế hiện tại - và điểm cuối - Mục tiêu được chỉ định - trong Costmap toàn cầu, công cụ lập kế hoạch toàn cầu tạo ra một kế hoạch sử dụng chức năng điều hướng được tính bằng thuật toán Dijkstra [dijkstra1959note]. Nó bao gồm sau các tế bào miễn phí liền kề cho đến khi đạt được mục tiêu.Sau đó tính đến chi phí địa phương, công cụ lập kế hoạch cục bộ tạo ra các lệnh vận tốc cho cản cứ di động của UAV.Một hành vi quay trở lại cũng được thực hiện khi cần thiết để xóa chế độ xem của robot.

Mục tiêu được chỉ định bởi *leader* được đảm bảo là thuộc về một khu vực không xác định bằng cách sử dụng chiến lược thăm dò.Quá trình lập kế hoạch quỹ đạo được thực hiện tại địa phương trên mỗi robot.Và vì UAV không trao đổi bản đồ địa phương của họ cũng như hợp nhất chúng, họ có khả năng xem lại các khu vực đã khám phá trong khi theo đường dẫn theo kế hoạch.Để giảm thiểu các vùng chồng chéo này trong quá trình điều hướng, ưu tiên được đưa ra cho các điểm biên giới $f_{i,j}$ là một mục tiêu cho UAV_i ngoài UAV_k với $k \neq i$. Điều này giúp UAV duy trì cùng một hướng trong quá trình thăm dò.Các move Gói cơ sở là một ngăn xếp điều hướng 2D.Tuy nhiên, để tránh trôi dạt trên z Trục, một lệnh điều khiển được thêm vào để giữ tĩnh z độ cao (Xem hình ??).



Hình 1.10: Từ ngăn xếp điều hướng 2D (trái) sang 3D (phải).Các *navigation z* là một quá trình ROS được phát triển để cung cấp một lệnh điều khiển trên trục Z.

1.4 Kết quả và thảo luận

Mô phỏng đã được thực hiện để đánh giá chiến lược thăm dò được đề xuất.Các thử nghiệm bổ sung trong khi sử dụng bản địa hóa tương đối đã được thực hiện để đo lường hiệu suất hệ thống.Các mô phỏng được thực hiện bằng Hệ điều hành Robot (ROS) chạy trên máy I7 Linux 2.60GHz.Đối với mô phỏng bốn cánh quạt, mô hình ar-drone² Được trang bị máy ảnh RGB-D theo điều kiện tìm kiếm phía trước, được sử dụng.Một môi trường không xác định được tạo ra bằng cách sử dụng *Gazebo* giả lập.Số lượng robot được sử dụng để đánh giá được giới hạn trong ba, tuy nhiên, kiến trúc hệ thống được đề xuất không bị hạn chế với số robot xed.

1.4.1 Điều chỉnh thông số

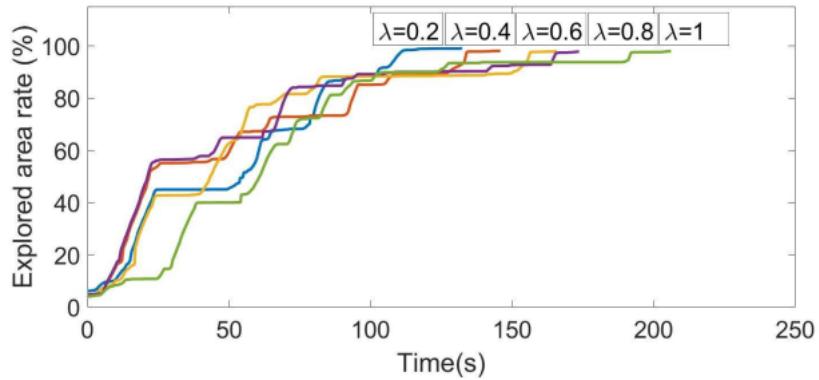
Đối với một đánh giá EF FF của Chiến lược thăm dò, chúng tôi sẽ chạy một số thử nghiệm để đặt các thông số đầy đủ nhất có liên quan.

Tham số đánh đổi λ

Chức năng tiện ích (xem Eq. ??) được sử dụng trong chiến lược thăm dò có thể được điều chỉnh, sử dụng một tham số thương mại off λ , giữa thăm dò nhanh và tìm kiếm chi tiết về bản đồ.

Hình ?? Hiển thị di chuyển chạy trong khi thay đổi tham số này.Bằng cách tăng lên λ , Thông tin thu được khi đạt được mục tiêu được ưa chuộng về khoảng cách và do đó, chi phí cho nó và ngược lại.Vì vậy, khi nào λ là nhỏ, khoảng cách di chuyển là nhỏ và vì vậy thời gian thăm dò.Mặc dù, một số lần trong nhiệm vụ, giá trị cao của λ được nhận thấy để đạt được tỷ lệ thăm dò cao hơn so với những người nhỏ hơn.

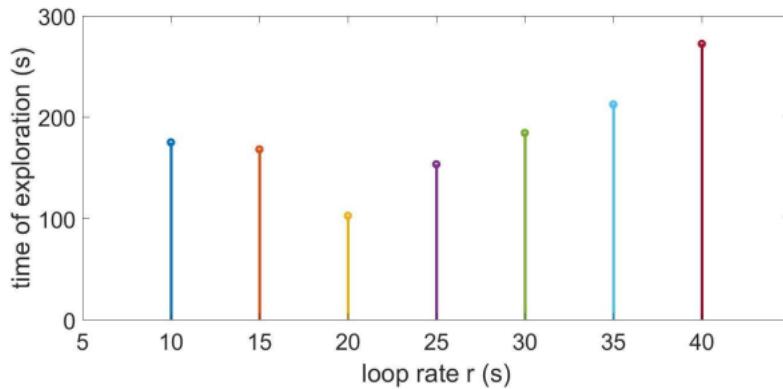
²Nguồn: http://wiki.ros.org/ardrone_autonomy



Hình 1.11: Tác động của việc thay đổi tham số thương mại λ qua thời gian thăm dò.

Tỷ lệ vòng lặp r

Tỷ lệ tần số hoặc vòng lặp r Nhiệm vụ mục tiêu cũng có thể là một hiệu suất thời gian thăm dò của FF. Các giá trị của r thay đổi để tính đến vận tốc robot \mathbf{v}_i và phạm vi tối đa của cảm biến s . Tác động của việc thay đổi tỷ lệ vòng lặp được đánh giá trong hình ??.



Hình 1.12: Thời gian khám phá trong khi thay đổi tỷ lệ vòng lặp r .

Dựa ra một vận tốc robot $\mathbf{v}_i = \{0.1, 0.3\} m.s^{-1}$ và một phạm vi cảm biến tối đa $s = 4m$, Tốc độ vòng lặp biến dạng trong $r \in [10, 40]$. Tham số này không nên quá nhỏ để cho phép robot đạt được mục tiêu; Cũng không quá lớn để ngăn chặn thời gian chờ đợi dài cho nhiệm vụ mục tiêu tiếp theo.

Thông số chung

Tùy thuộc vào kết quả trong hình ?? và hình. ??, tương ứng, λ được đặt thành 0.2 và r đến 20s Để tối đa hóa tỷ lệ khu vực được khám phá trong khi giảm thiểu thời gian nhiệm vụ.Các tham số mô phỏng được tóm tắt trong bảng ??.

Bảng 1.1: Thông số chung.

Parameter	Value
RGB-D PROIZONTAL FOV	$\pi/3$
Tham số đánh đổi λ	0.2
Phạm vi tối đa RGB-D $s(m)$	4
Khoảng cách tối thiểu giữa các biên giới $d(m)$	0.3
Dộ phân giải lưới chiếm dụng (m)	0.05
Phạm vi để lên lịch $lg[\sigma_x, \sigma_y](m)$	[3, 3]
Tỷ lệ vòng lặp $l(s)$	20
Kích thước môi trường (m^2)	8×8
Vận tốc tuyến tính. $v_i(m.s^{-1})$	[0.1, 0.3]
Vận tốc góc cạnh $w_i(rad.s^{-1})$	[0.1, 0.3]

1.4.2 Biểu diễn chiến lược khám phá

Chiến lược thăm dò được đề xuất đã được đánh giá về mặt phân phối robot trong môi trường, tỷ lệ chồng chéo, thời gian thăm dò và toàn bộ khoảng cách di chuyển của mỗi robot.

Bản đồ tiền hóa trong nhiệm vụ

Trong khi đạt được mục tiêu được giao nhiệm vụ tương ứng, mỗi robot phụ trách việc tạo bản đồ lưới chi tiết của khu vực đã truy cập để có được bản đồ toàn cầu về môi trường.Hình ?? minh họa sự tăng trưởng của bản đồ lưới chiếm dụng 3D được xây dựng lại tại Di Ff Erent Times trong nhiệm vụ.Lưu ý rằng để đạt được 99% của phạm vi bảo hiểm, rất nhiều thời gian được chi tiêu.

Hình ?? Hiển thị sự phát triển của bản đồ lưới địa phương 2D dự kiến tương ứng của hai robot trong một nhiệm vụ thăm dò hợp tác xã.Bản đồ lưới 2D dự kiến toàn cầu cũng được tạo và đại diện để đánh giá (quy trình khớp lưới chiếm dụng được giới thiệu trong thuật toán 2.1 trong Chương 2 được sử dụng).Vị trí ban đầu của robot là (1, 0, 0) cho UAV₁ và (1, -3, 0)

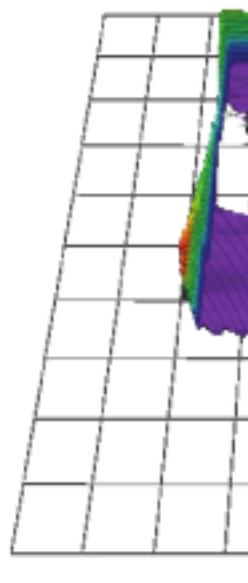
cho UAV₂. Mặc dù có vị trí ban đầu tương đối chặt chẽ, chiến lược được đề xuất sẽ dễ dàng lan robot để UAV₁ phụ trách bên trái của môi trường và UAV₂ của bên phải.

Frontier điểm tiến hóa trong nhiệm vụ

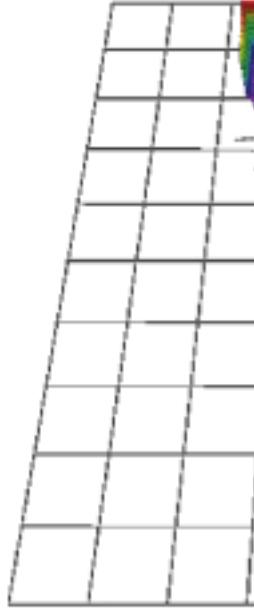
Mục tiêu được chọn từ các điểm biên giới ứng cử viên mà các cạnh của một môi trường không được khám phá trước đó.Những ứng cử viên này được chọn từ các điểm biên giới Nal của mỗi UAV trong fleet (xem hình ??). Trong nhiệm vụ thăm dò, kích thước bản đồ địa phương tăng lên, dẫn đến số lượng điểm biên giới cục bộ ngày càng tăng.Khi bắt đầu thăm dò, số lượng điểm biên giới ứng cử viên tăng lên, nhưng ngay khi thăm dò tiến hóa kịp thời, số lượng của họ giảm.Vào cuối nhiệm vụ, khi tất cả các môi trường được khám phá, không nên để lại điểm biên giới ứng cử viên.

Quỹ đạo của UAV trong nhiệm vụ

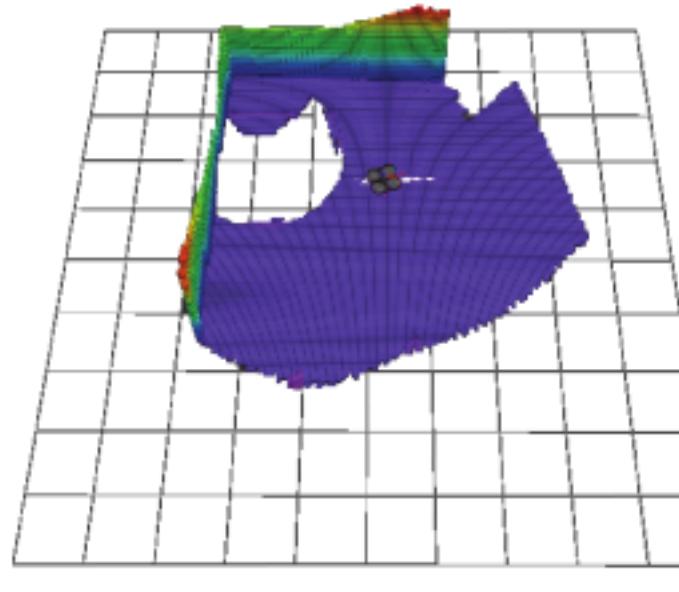
Hình ?? Hiển thị bản đồ được khám phá với các quỹ đạo bằng cách sử dụng một, hai và ba UAV.UAV cố gắng khám phá môi trường đầy đủ trong khi tránh các khu vực đã khám phá.Một cách hợp tác, mỗi UAV phụ trách truy cập một khu vực bằng cách đạt được mục tiêu thuộc về một môi trường không khám phá.Những mục tiêu này được chỉ định bởi Leader Ngay cả khi, ngay cả khi các tư thế ban đầu của UAV tương đối gần, E FF vượt qua chúng vào các khu vực chưa biết.Bản đồ toàn cầu bao gồm sự chồng chéo của tất cả các bản đồ địa phương của tất cả các UAV.



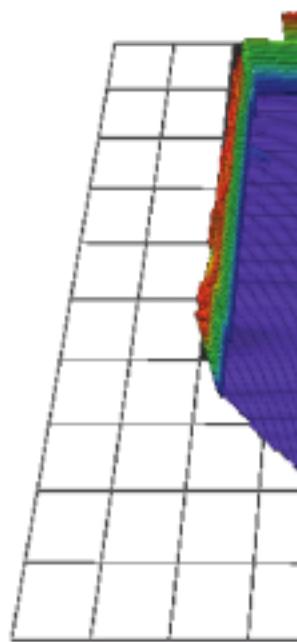
(a) 3% tỉ lệ (5s).



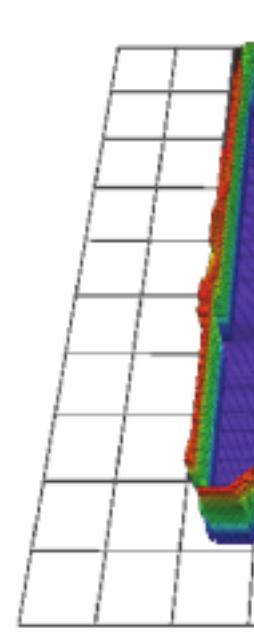
(b) 20% tỉ lệ (28s).



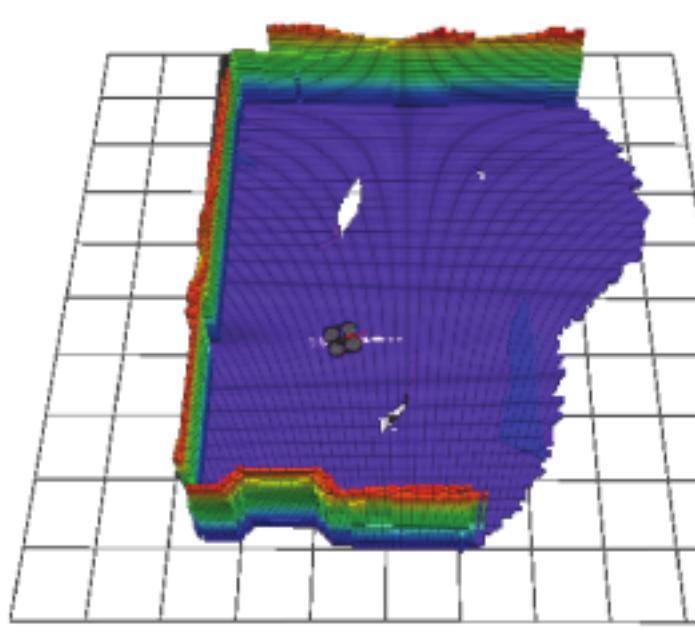
(c) 31% tỉ lệ (50s).



(d) 40% tỉ lệ (69s).

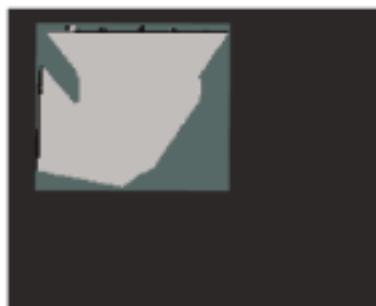


(e) 50% tỉ lệ (75s).

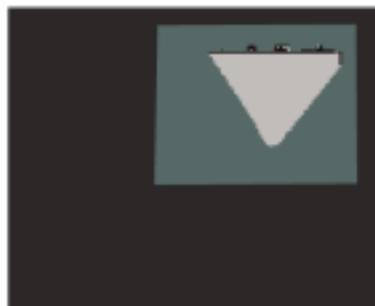


(f) 60% tỉ lệ (83s).

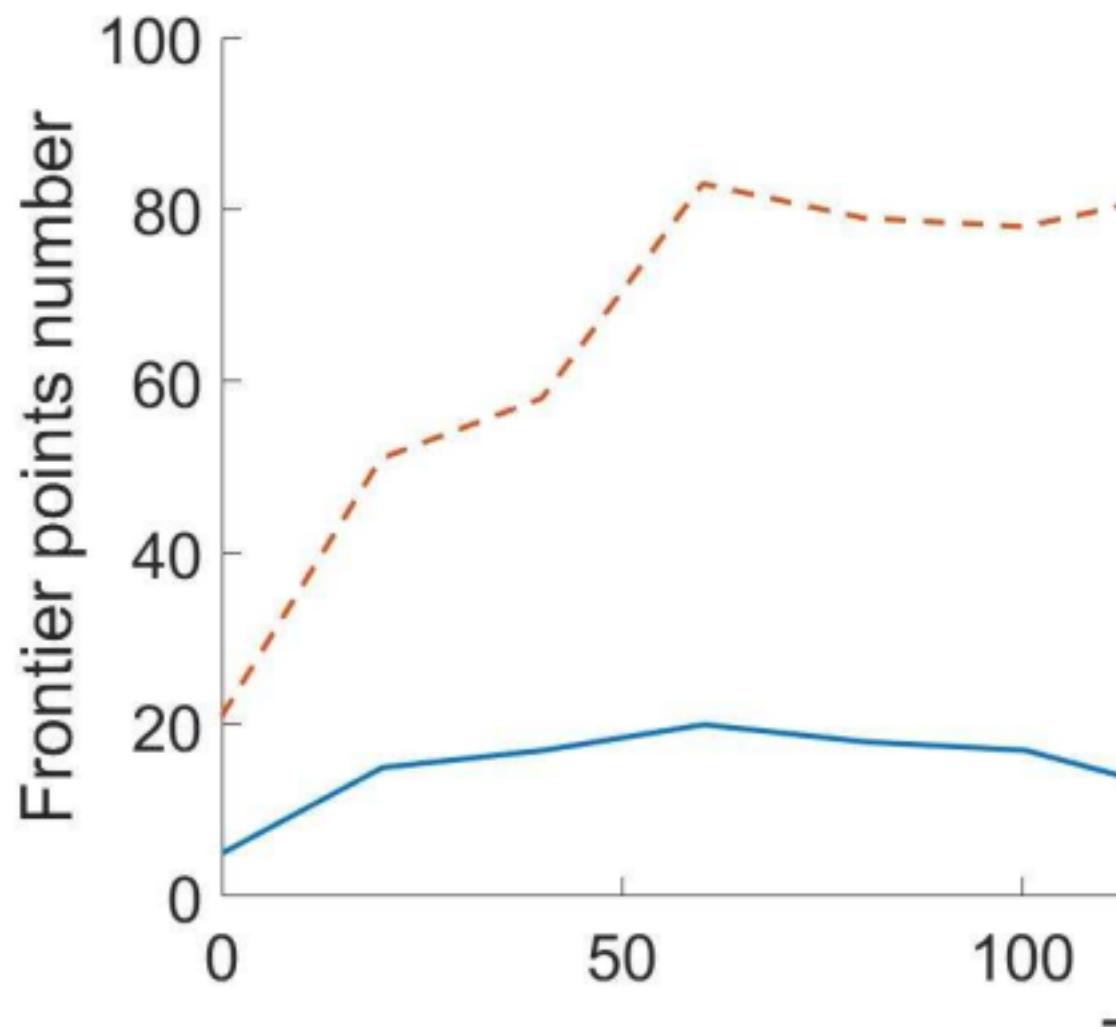
Local map 1



Local map 2



G



(a) Một UAV.





(a) Một UAV.

Đánh giá nhiệm vụ mục tiêu: Phân phối robot trong môi trường

Quá trình phân công mục tiêu được thực hiện theo thuật toán được mô tả trong Phần 3.6. Tuy nhiên, sau khi chỉ định mục tiêu cho robot đầu tiên trong danh sách, cùng một mục tiêu hoặc một mục tiêu khác gần với nó có thể được gán cho robot thứ hai trong danh sách. Để khắc phục những vấn đề này, mức tăng thông tin của các mục tiêu ứng cử viên còn lại được lên lịch. Điều này cho phép loại bỏ một mục tiêu đã được chỉ định và giữ một khoảng cách nhất định giữa mục tiêu mới và phần trước được chỉ định.

Giả sử rằng một mục tiêu được gán cho robot đầu tiên trong danh sách cụm. Hình ?? Hiển thị mục tiêu được chọn cho robot thứ hai khi một bài tập tuần tự được thực hiện:

- Không cần xử lý điểm biên giới hơn nữa (xem hình ??). Do đó, mục tiêu tương tự được gán cho hai robot khác nhau.
- Trong khi loại bỏ mục tiêu được gán từ các điểm biên giới ứng cử viên còn lại (xem Hình ??). Do đó, mục tiêu thứ hai tương đối gần với người đầu tiên được chỉ định.
- Trong khi lên lịch trình thông tin tăng sau mỗi lần gán mục tiêu (xem Hình ??). Do đó, các mục tiêu được chỉ định được cách nhau vào môi trường. Lợi ích thông tin được lên lịch theo lịch trình Eq. ?? . Giá trị thông tin tăng lên với khoảng cách đến mục tiêu ứng viên \mathbf{t}_g .

Quá trình phân công mục tiêu đòi hỏi có thể không tối ưu vì nó phụ thuộc vào thứ tự của robot trong danh sách. Ví dụ, giả sử robot UAV_i và UAV_j có cùng một nhiệm vụ mục tiêu tốt nhất t k sao cho nó đạt được tiện ích tối đa so với các điểm biên giới ứng viên, nơi Eq. ?? và Eq. ?? được xác minh.

$$\mathbf{t}_k = \operatorname{argmax}_{\mathbf{t}_m} \mathbf{U}(UAV_i, \mathbf{t}_m) \quad (1.5)$$

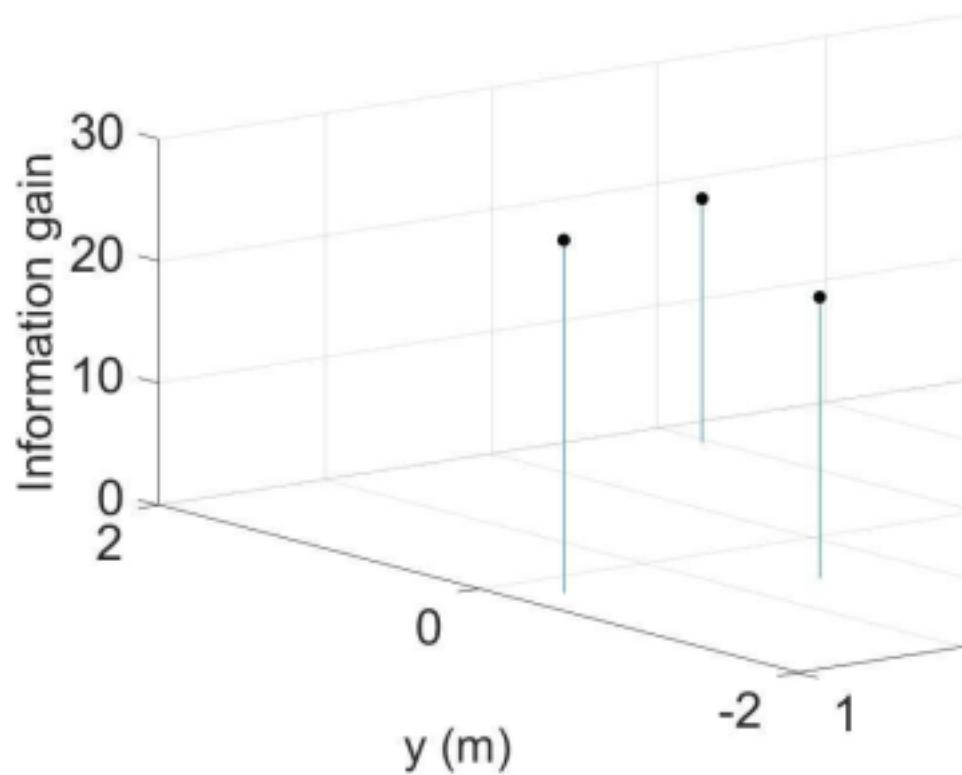
với $\mathbf{t}_m \in \mathcal{G}$.

$$\mathbf{t}_k = \operatorname{argmax}_{\mathbf{t}_n} \mathbf{U}(UAV_j, \mathbf{t}_n) \quad (1.6)$$

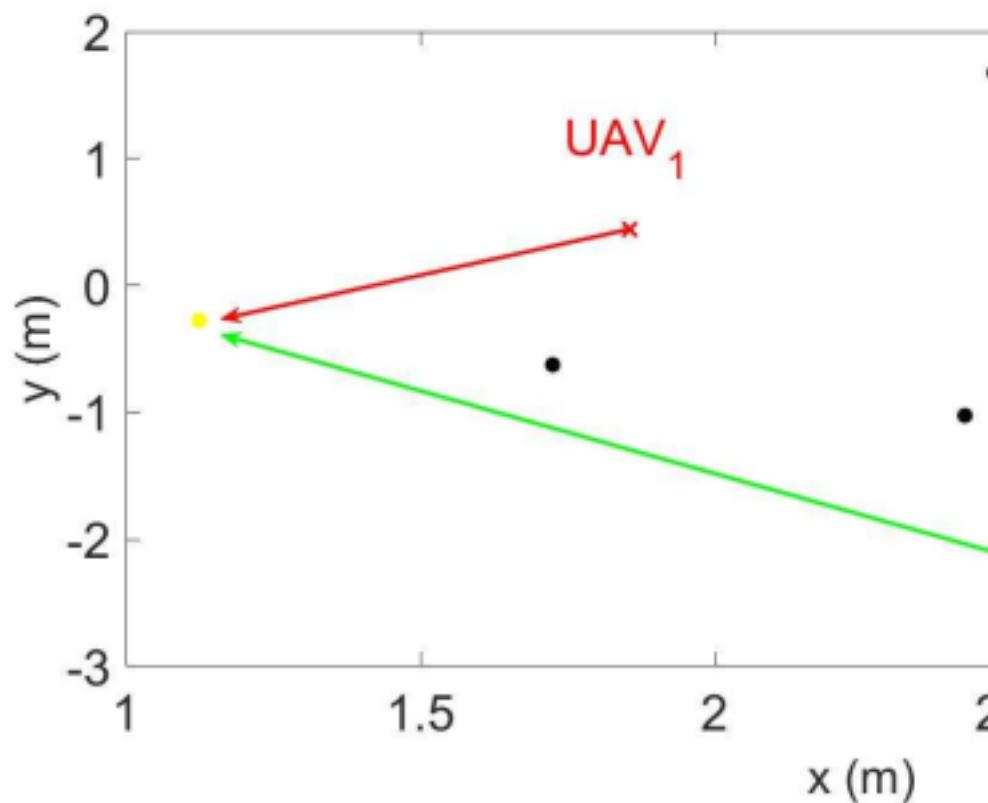
với $\mathbf{t}_n \in \mathcal{G}$.

UAV_i có một điểm biên giới ứng cử viên khác \mathbf{t}_l với:

$$U(UAV_i, \mathbf{t}_k) > U(UAV_i, \mathbf{t}_l) > U(UAV_j, \mathbf{t}_k) \quad (1.7)$$



(a) Mục tiêu ứng viên.



(b) Trường hợp 1.

Vì vậy, giải pháp tối ưu sẽ là gán \mathbf{t}_l đến UAV_i và \mathbf{t}_k đến UAV_j. Nhưng nếu UAV_i là đầu tiên trong danh sách, \mathbf{t}_k được gán cho nó và một điểm biên lai khác với ít tiện ích hơn \mathbf{t}_k , được chỉ định cho UAV_j. Do đó, giải pháp với việc phân công mục tiêu tuần tự không phải lúc nào cũng tối ưu.

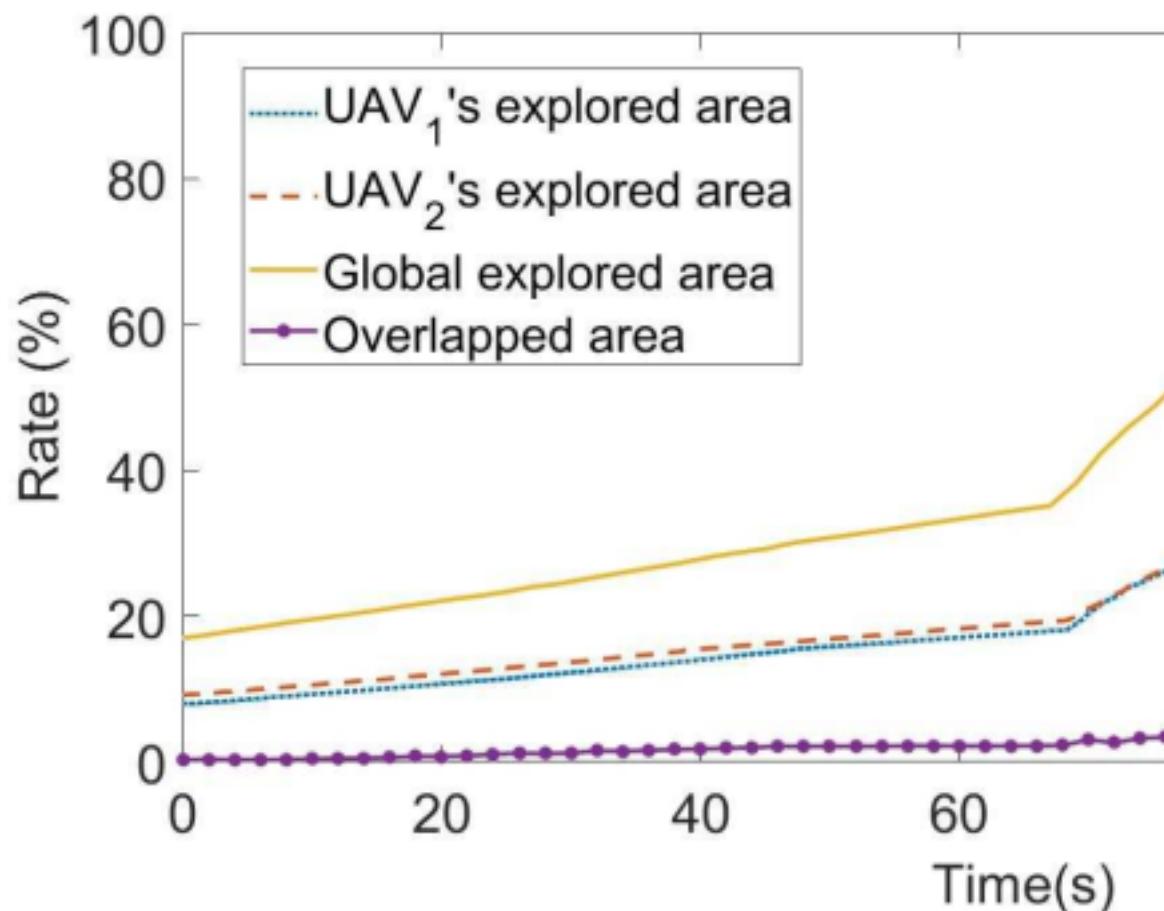
Để khắc phục vấn đề này, tất cả các số kết hợp có thể xảy ra $\frac{n_g!}{n_g!(n_g-n_c)!}$ với n_g số lượng mục tiêu ứng cử viên và n_c số lượng robot, cần phải được xem xét. Điều này tăng đáng kể thời gian tính toán bằng cách tăng số lượng robot. Do đó, trong thuật toán được đề xuất, nhiệm vụ tuần tự được ưa chuộng về việc tính toán tất cả các hoán vị có thể.

Khám phá đánh giá tỷ lệ không gian

Đánh giá này nhằm định lượng lượng không gian khám phá trong nhiệm vụ. Hình ?? cho thấy tỷ lệ không gian được khám phá được thực hiện bởi mỗi UAV trong FL EET. Càng nhiều UAV trong môi trường, tỷ lệ thăm dò càng ít yêu cầu. Một robot không cần phải tiếp tục khám phá một khu vực nếu nó đã được khám phá bởi một khu vực khác. Do đó, thời gian nhiệm vụ giảm đáng kể.

Đánh giá tỷ lệ chồng chéo

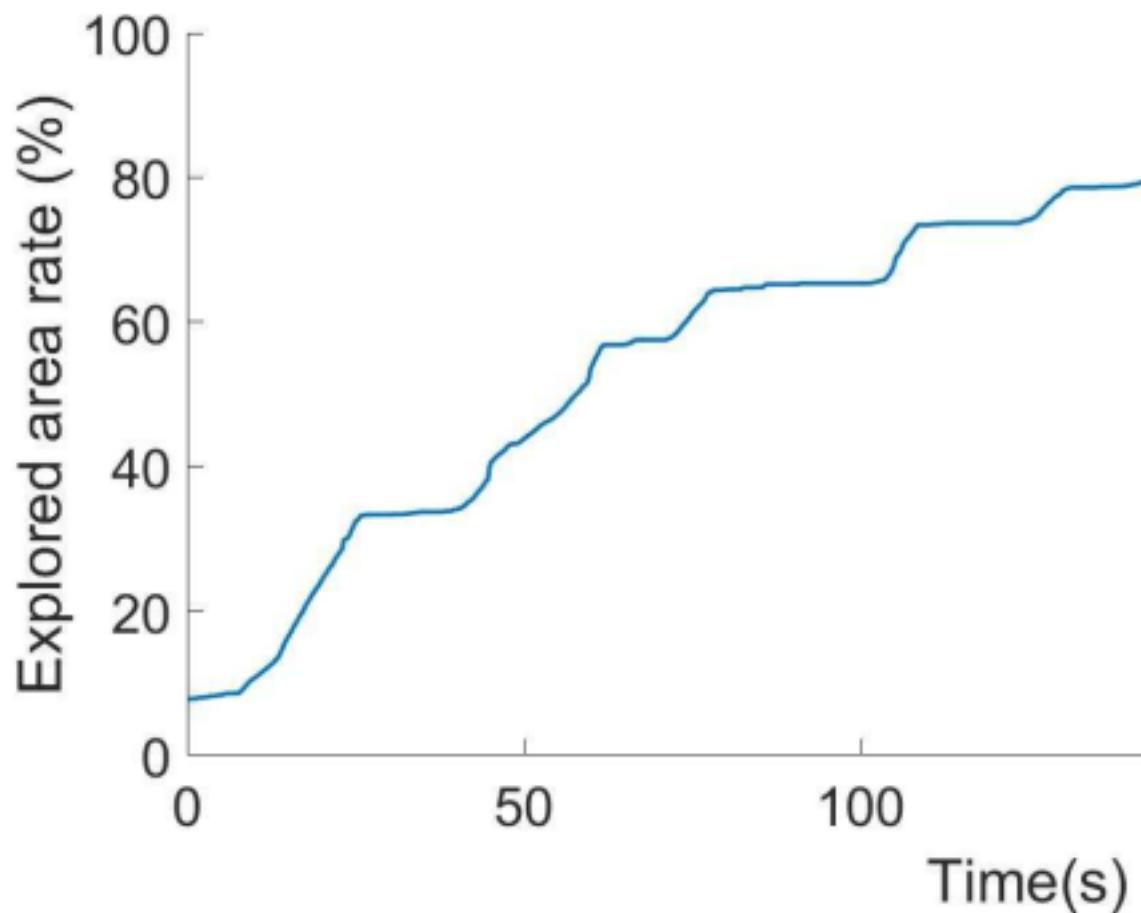
Việc sử dụng một quy trình phân công mục tiêu arteactive e ff sẽ giới hạn sự chồng chéo được tạo. Trong hình ??, Sự phát triển thời gian của sự chồng chéo được đánh giá bằng hai robot hợp tác. Sự chồng chéo trải qua một sự gia tăng đáng kể vào cuối cuộc thám hiểm để đạt được 33%. Điều này được giải thích bởi sự gần gũi của các bản đồ địa phương vào cuối nhiệm vụ chính xác là bản đồ lưới toàn cầu.



Hình 1.18: Tỷ lệ khu vực khám phá và chồng chéo bằng cách sử dụng hai UAV hợp tác.

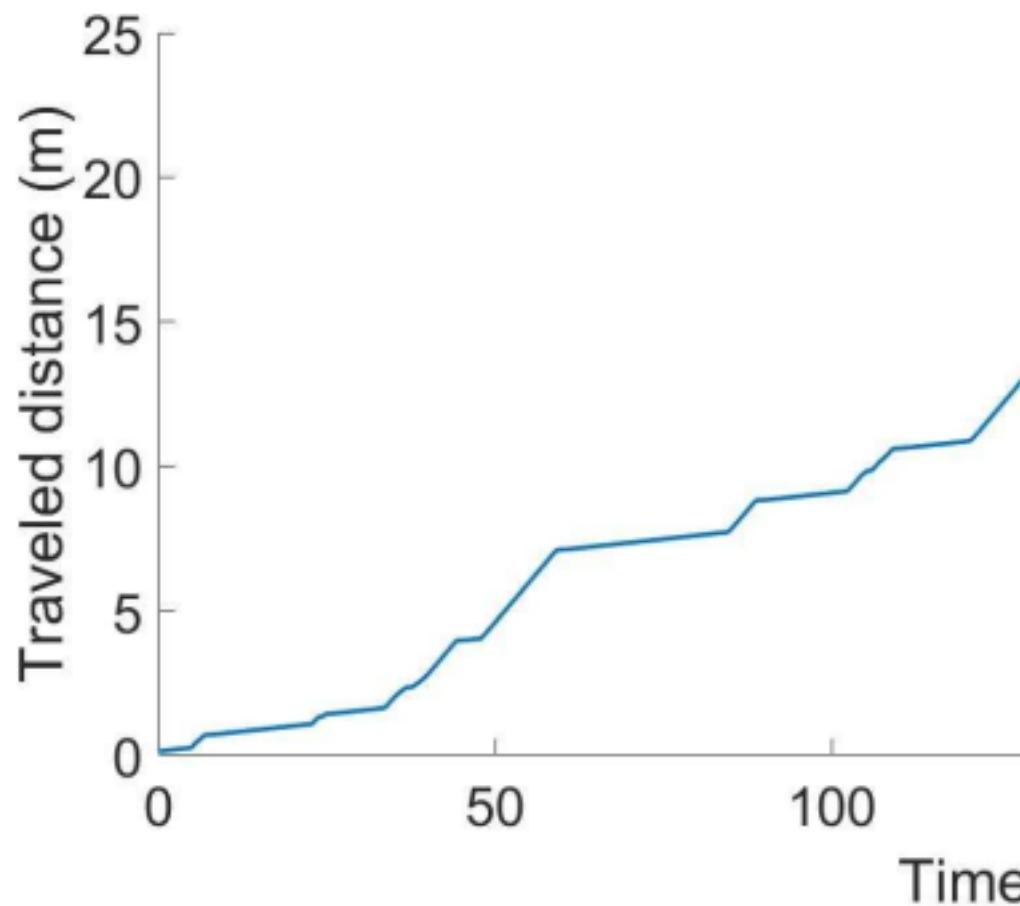
Đánh giá khoảng cách di chuyển

Đến đây đánh giá hiệu suất chiến lược thăm dò về khoảng cách di chuyển của mỗi UAV, diễn ra với một, hai và ba UAV đã được tiến hành trong đó tỷ lệ khu vực được khám phá gần như 99%. Hình ?? cho thấy khoảng cách được di chuyển bởi mỗi UAV trong FL EET trong nhiệm vụ.



(a) Một UAV.

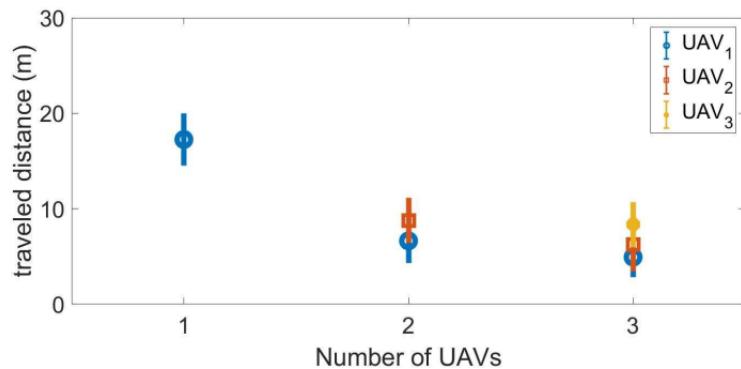




(a) Một UAV.



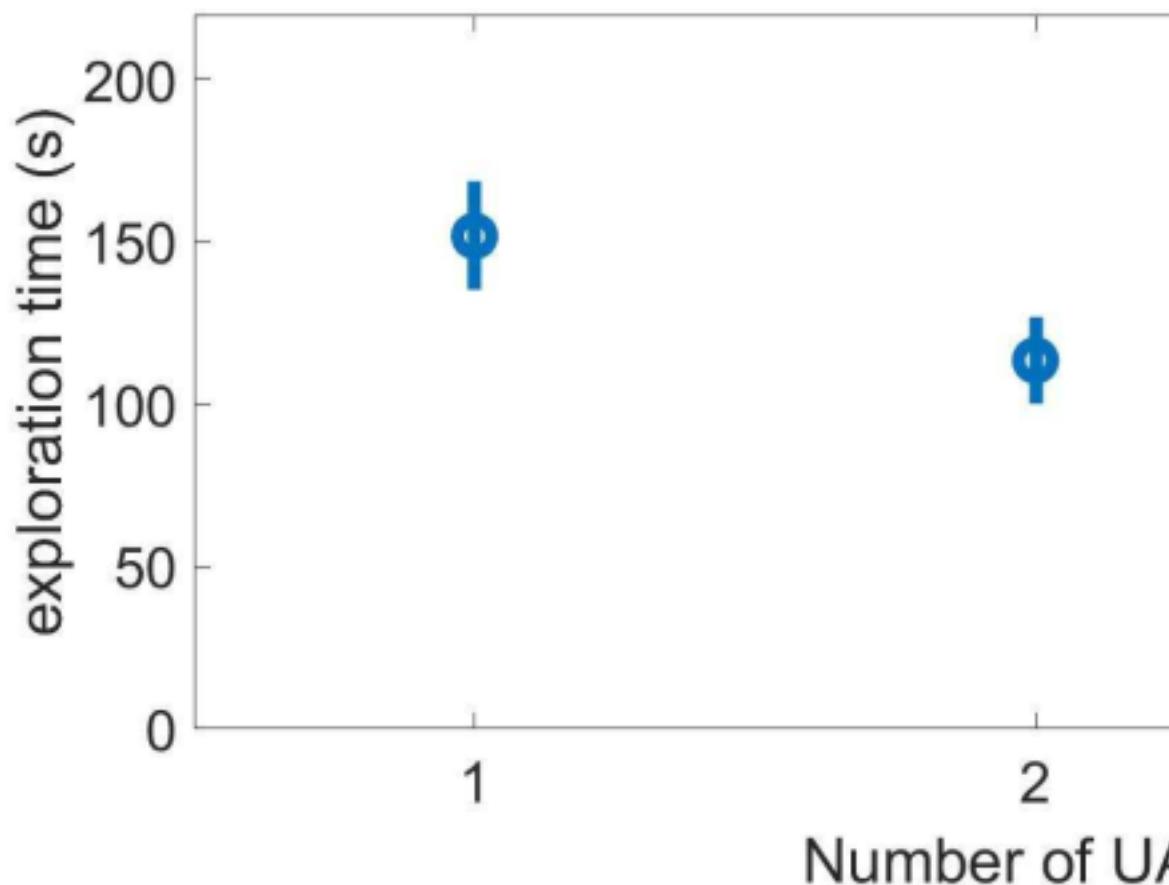
Các khoảng cách được di chuyển bởi mỗi UAV bằng cách sử dụng một, hai và ba UAV trong FL EET được so sánh trong hình ???. Khoảng cách này giảm với số lượng UAV. Khoảng cách trung bình được di chuyển bởi mỗi UAV bị giảm bởi 55% cho 2 UAV và bởi 62% cho 3 UAV. Lỗi của khoảng cách di chuyển được giảm nhẹ từ một đến hai và ba UAV.



Hình 1.21: Di du lịch đánh giá khoảng cách.

Đánh giá thời gian khám phá

Đối với đánh giá thời gian thăm dò, di ff erent chạy đã được thực hiện bằng cách sử dụng UAV 1, 2 và 3. Hình ?? Cho thấy rằng trung bình của thời gian thăm dò giảm khi số lượng robot trong EET tăng. Lỗi tính toán cũng giảm. Thời gian bị giảm bởi 25% cho 2 UAV và bởi 30% cho 3 UAV. Thời gian và khoảng cách thăm dò không được chia cho 2 hoặc 3 khi nhân với 2 hoặc 3 số lượng robot, tương ứng. Trong các mô phỏng này, vị trí ban đầu của robot là: (1, 0, 0) cho một UAV; (1, 0, 0) và (1, -3, 0) cho hai UAV; và (1, 1, 0), (1, -1, 0) và (1, -3, 0) cho ba UAV.

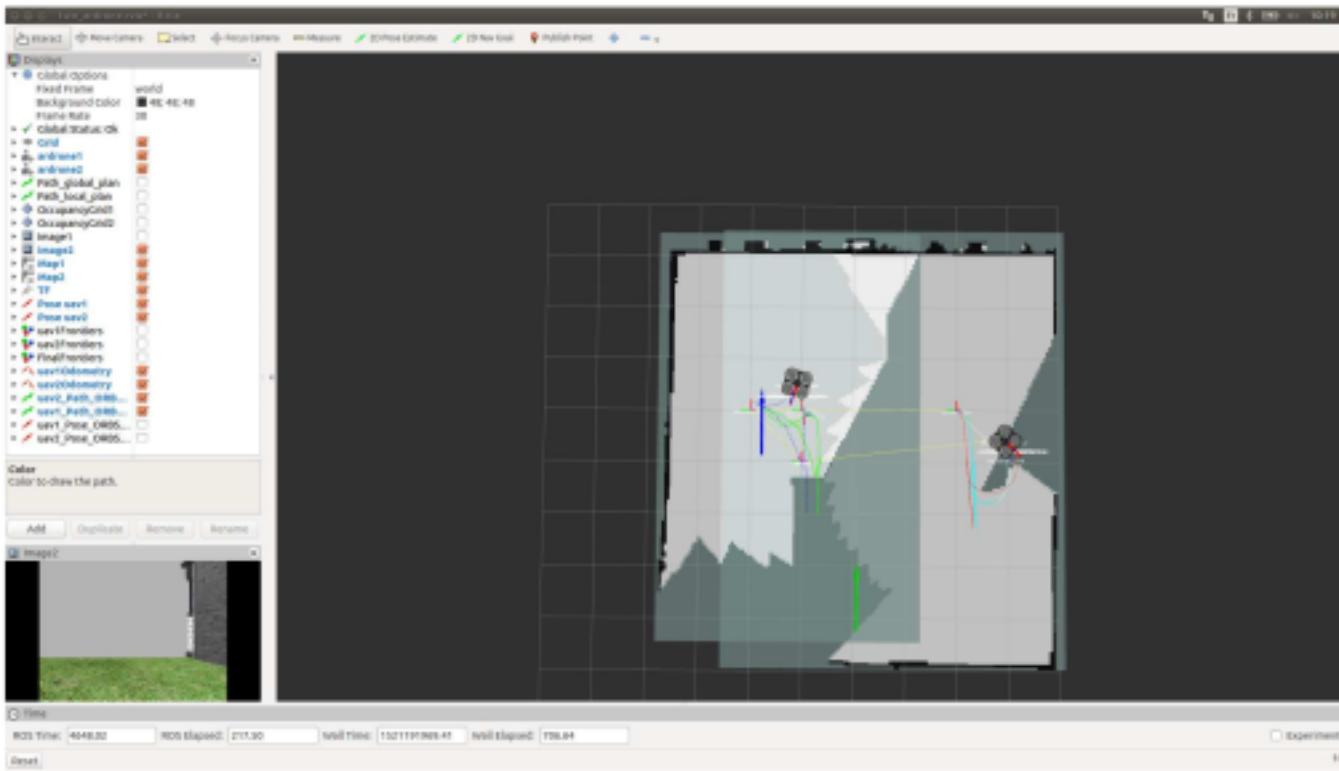


Hình 1.22: Thời gian thăm dò trung bình.

Các kết quả được trình bày trong Phần 4.2 được đánh giá mà không có bản địa hóa tương đối. Vì vậy, để có một kịch bản thực tế khó khăn hơn, chạy với thuật toán bản địa hóa tương đối đã được thực hiện để đánh giá các màn trình diễn hệ thống bằng SLAM.

1.4.3 Nhiệm vụ khám phá bằng thuật toán bản địa hóa tương đối

Hướng tới một kịch bản thực tế hơn, cách tiếp cận ORB-SLAM2 (được giới thiệu trong Phần 5 của Chương 2) đã được triển khai để thực hiện bản địa hóa tương đối (xem Hình ??).



Hình 1.23: Hai điều hướng UAV bằng ORB-SLAM2.Hình ảnh Rviz (trái) hiển thị, đối với mỗi UAV, bản đồ lưới chiếm dụng 2D được xây dựng, quỹ đạo ước tính và sự thật tương ứng.Đám mây điểm (giữa) đại diện cho việc tái thiết thưa thớt môi trường được thực hiện bởi mỗi UAV.Và, các điểm đánh dấu màu xanh lá cây (phải) đại diện cho các tính năng được tính bằng mỗi UAV để thực hiện bản địa hóa.

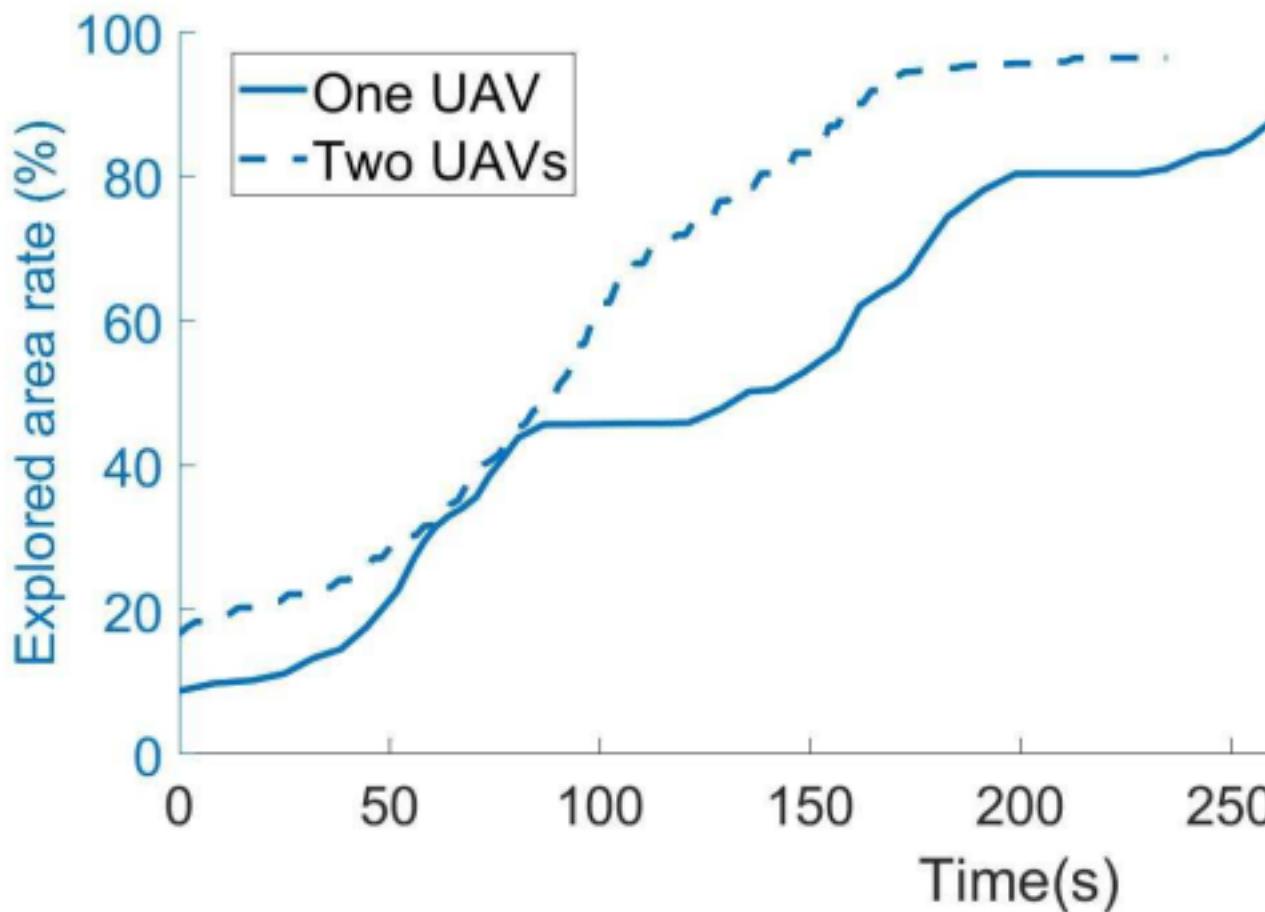
Mỗi robot thực hiện Slam nơi nó xây dựng bản đồ của nó trong khung tham chiếu cục bộ của nó WF_i , và ước tính tư thế tương đối của nó \mathbf{p}_i bên trong nó. Sau đó, thiết bị thực hiện khám phá hợp tác bằng cách sử dụng một số thông tin cụ thể được trao đổi giữa các UAV. Tuy nhiên, những thông tin này phải nằm trong một hệ quy chiếu chung. Do đó, những thông tin như tư thế \mathbf{p}_i và các điểm biên cương $\mathbf{f}_{i,j}$ nhất thiết phải biến thành khung tham chiếu toàn cầu 0W trước khi được trao đổi.Vì thế *leader* làm cho tất cả các tính toán cần thiết và gửi lại cho *explorers* các mục tiêu trong 0W . Khi robot nhận được mục tiêu được giao, nó sẽ biến nó thành WF_i để

lên kế hoạch cho một con đường đến nó.

Để thực hiện một phép chuyển đổi từ tham chiếu cục bộ ${}^W F_i$ đến toàn cầu một. ${}^0 W$, Robot phải biết - ít nhất là - tư thế ban đầu của nó w.r.t. ${}^0 W$. Như được giải thích trong Phần 5.2 của Chương 1, khung tham chiếu toàn cầu của môi trường được khởi tạo sao cho nó trùng với khung tham chiếu cục bộ của nhóm RST-leader trong đội đó là UAV₁. Trong ví dụ được coi là phương trình ??.

$${}^0 W \equiv {}^W F_1, \quad (1.8)$$

Sau đó, bằng cách phát hiện robot này bằng các thẻ được gắn trên đó, các robot khác có thể ước tính biến đổi tương ứng của chúng thành ${}^{F_j} [\mathbf{R} \ \mathbf{t}]_{F_1}, j \in [2..n_c]$. Để đánh giá mô phỏng, thông tin về biến đổi - được tính toán trong khi phát hiện thẻ - được giả định sẽ được biết đến. Hình ?? Cho thấy sự phát triển tỷ lệ thăm dò trong nhiệm vụ thăm dò trong khi sử dụng ORB-SLAM2 làm phương pháp nội địa hóa tương đối. Thời gian truyền bá sử dụng ORB-SLAM2 bị giảm bởi 43%. Đối với 2 UAV thay vì 1 UAV.



Hình 1.24: Khám phá sự tiến hóa tỷ lệ khu vực trong nhiệm vụ thăm dò với một và hai UAV trong khi thực hiện ORB-SLAM2 bởi mỗi UAV.

Thời gian thăm dò khi sử dụng bản địa hóa tương đối (xem hình ??) tương đối quan trọng so với việc khám phá mà không có slam (xem hình ??) Vì vận tốc đã giảm đáng kể.

1.5 Phản kết luận

Trong chương này, chúng tôi đã trình bày một trạng thái của nhiệm vụ thăm dò đa robot bao gồm chiến lược được sử dụng để chỉ định robot cho mục

tiêu và một chức năng tiện ích được áp dụng để ước tính sự quan tâm của việc tiếp cận nó.Sau đó, chúng tôi đã giới thiệu một chiến lược thăm dò dựa trên nhóm-leader quyết định. Bài tập robot-mục tiêu được thực hiện bằng cách sử dụng chức năng tiện ích mới lạ.Hàm này tạo ra một giao dịch giao dịch giữa khám phá nhanh và nhận bản đồ lối chi tiết, và cũng có tính đến khoảng cách của mỗi robot trong nhóm từ bộ mục tiêu chưa được khám phá.Ngoài ra, chúng tôi đề xuất để lên lịch để đạt được thông tin để e FFI tiêu hóa tiêu hóa UAV vào môi trường.Hơn nữa, chiến lược được thông qua trao đổi các điểm biên giới thay vì toàn bộ bản đồ địa phương.

Kết quả cho thấy chiến lược thăm dò hợp tác đề xuất giảm thiểu thời gian thăm dò toàn cầu theo 25% cho 2 UAV và bởi 30% cho 3 UAV, đồng thời giảm thiểu khoảng cách trung bình di chuyển của mỗi UAV bởi 55% cho 2 UAV và bởi 62% cho 3 UAV.Hơn nữa, chiến lược được đánh giá bằng thuật toán nội địa hóa tương đối nơi giảm thời gian thăm dò 43% Đối với 2 UAV thay vì 1 UAV.

Chương 2

Giao tiếp giữa các robot

2.1 Giới thiệu

Một chủ đề quan trọng trong các hệ thống nhiều robot là giao tiếp giữa các robot. Tính năng này được sử dụng để điều phối và chia sẻ thông tin cụ thể. Một đội có thể được triển khai trong một số nhiệm vụ trong các khu vực tương đối khác biệt và thù địch. Tuy nhiên, những môi trường này có thể không có cơ sở hạ tầng sẵn có, ngoài các liên kết truyền thông không lý tưởng.

Chương này nhấn mạnh một trong những điểm thách thức nhất trong MRS, đó là giao tiếp giữa các robot. Vấn đề này có thể được giải quyết từ các quan điểm khác nhau; nhưng, chúng tôi đã lựa chọn nghiên cứu hai vấn đề nhỏ, đó là: định dạng mạng, cấu trúc liên kết mạng và chiến lược cho sự mạnh mẽ của MRS.

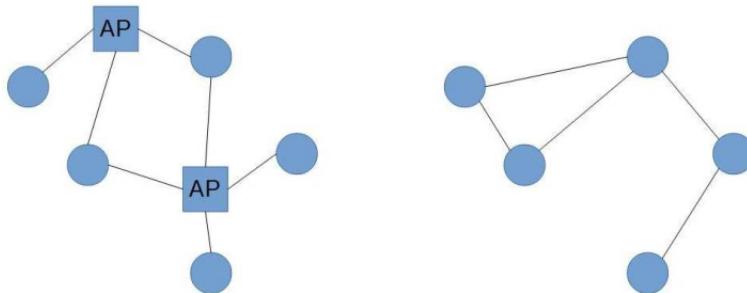
2.2 Phân loại mạng

Mạng được sử dụng để liên kết giữa các thực thể khác nhau và để thiết lập giao tiếp có thể có giữa chúng. Một số loại mạng tồn tại và có thể được phân loại theo nhiều cách khác nhau.

2.2.1 Chế độ cơ sở hạ tầng so với chế độ không có cơ sở hạ tầng

Để quản lý cơ sở hạ tầng mạng, các chế độ hiện có có thể được phân loại thành hai loại chính [bekmezci2013flying],[hayat2015experimental].

Loại đầu tiên là chế độ cơ sở hạ tầng, có thể được gọi là chế độ Điểm truy cập (AP) và loại thứ hai là chế độ không có cơ sở hạ tầng, gọi là Ad Hoc mode (Xem Hình ??). Trong chế độ AP, giao tiếp UAV-to-UAV được thực hiện thông qua cơ sở hạ tầng (AP, bộ định tuyến, v.v.), không giống như chế độ Ad Hoc trong đó các nút giao tiếp trực tiếp với nhau. Chế độ Ad Hoc còn được gọi là chế độ ngang hàng. Mạng Ad Hoc có thể phát triển thành một chế độ ngang hàng khác được gọi là mạng lưới bằng cách cho phép khả năng đa bước. Thật vậy, mạng Ad-Hoc không có bất kỳ khả năng cố hữu nào cho đa bước. Các nút giao tiếp với nhau khi chúng nằm trong phạm vi giao tiếp của nhau. Trong khi đó, trong mạng lưới, các nút có thể giao tiếp trực tiếp hoặc thông qua một hoặc nhiều nút trung gian.



Hình 2.1: Chế độ hạ tầng (trái) và chế độ không có hạ tầng (phải).

Bảng 2.1: Chế độ cơ sở hạ tầng so với chế độ không có cơ sở hạ tầng.

	Chế độ cơ sở hạ tầng	Chế độ không có cơ sở hạ tầng
Ưu điểm	Giao tiếp đáng tin cậy	Kết nối trực tiếp với nhau, dễ thiết lập, mạnh mẽ với lỗi nút, cho phép mở rộng và điều chỉnh trong cấu trúc liên kết mạng.
Nhược điểm	Phản ứng đắt, phức tạp, phạm vi bị hạn chế .	Thừa trong kết nối mạng, khó bảo trì.
Tiêu chuẩn	802.11	802.11, 802.15, 802.16

Ban đầu, robot phải thực hiện nhiệm vụ của chúng trong môi trường không có cơ sở hạ tầng mạng. Trên thực tế, chúng trao đổi thông tin khi

chúng hoạt động như một bộ định tuyến và như một AP. Vì vậy, hệ thống MRS có thể được xem như một mạng Ad Hoc di động theo quan điểm giao tiếp.

2.2.2 Phân loại mạng Ad Hoc

Mạng Ad Hoc là mạng được sử dụng phổ biến để quản lý giao tiếp đa robot. Với mạng này, mỗi robot có thể di chuyển tự do và chuyển tiếp các gói tin đến và từ mỗi robot khác tùy thuộc vào phương thức phân phối dữ liệu [**bouachir2014conception**]. Mạng này được đặc trưng bởi chất lượng dịch vụ phức tạp, trong đó giao thức định tuyến xác định đường đi tối ưu của thông tin có tính đến sự thay đổi thường xuyên của cấu trúc liên kết.Thêm vào đó, sẽ rẻ hơn nếu nhận ra loại mạng này thay vì các loại khác (vô tinh, di động, v.v.).

Mạng Ad Hoc có thể được phân loại thành ba loại chính là Mobile Ad hoc NETwork (MANET), Vehicle Ad hoc NETwork (VANET) và Flying Ad hoc NETworks (FANET). Nói chung, đối với hệ thống đa UAV, mô hình FANET được sử dụng. Bảng ?? mô tả chi tiết mỗi đặc điểm của danh mục [**bekmezci2013flying**], [**maistrenko2016experimental**], phân tích dự đoán của chúng tôi.

Bảng 2.2: So sánh giữa MANET, VANET, FANET và mô hình mạng của chúng tôi.

Đặc điểm	MANET	VANET	FANET	Mô hình mạng của chúng tôi
Tính di động	Người ở một số địa hình nhất định	Phương tiện ở cao tốc	Máy bay trong kế hoạch 3D	3D
Mức độ di động	+	+	++	++
Chế độ di động	Điểm cách ngẫu nhiên với hướng và tốc độ ngẫu nhiên	Có khả năng dự đoán cao trên đường	Không được xác định trước (mô hình chuyển động ngẫu nhiên của UAV, mô hình dựa trên pheromone)	Có thể dự đoán
Thay đổi cấu trúc liên kết	+	+	++	++
Khoảng cách giữa các nút	+	+	++	+
Mật độ nút	++	++	+	+
Mô hình truyền thanh	Sự hiện diện hiếm hoi của đường ngầm	Sự hiện diện hiếm hoi của đường ngầm	Sự hiện diện thường xuyên của đường ngầm	Luôn có mặt của đường ngầm
Công suất tiêu thụ và công suất tính toán thời gian sống	+	+	++	++
Tiêu chuẩn	IEEE 802.11, 802.15, 802.15.4, 802.16 and 802.20	IEEE 802.11p	IEEE 802.11	?

So sánh cho thấy FANET là mạng gần nhất với kỳ vọng của mô hình của chúng tôi. Do đó, trong số các tiêu chuẩn hiện có cho loại mạng này, IEEE 802.11b và IEEE 802.11g cung cấp tốc độ dữ liệu cao và thường được sử dụng cho các hệ thống nhiều robot.

2.2.3 Phân loại cấu trúc liên kết mạng

Cấu trúc liên kết mạng xác định điểm đầu, điểm cuối và đường dẫn của thông tin trong mạng. Để cho các điểm mục tiêu, cấu trúc liên kết xác định tất cả các con đường có thể để đạt được chúng.

Từ quan điểm phần mềm, mạng chủ yếu có thể được phân loại theo ba cấu trúc liên kết (Xem Hình ??¹):



Hình 2.2: Mạng tập trung (trái), phi tập trung (giữa) và mạng phân tán (phải).

- Mạng tập trung: Đây là cấu trúc liên kết được sử dụng nhiều nhất [[morgenthaler2012uavnet](#)], [[forster2013collaborative](#)]. Tất cả các nút - còn được gọi là các trạm - được kết nối với nhau thông qua một trung tâm tập trung. Đó là khi một nút cần gửi một thông tin đến một nút khác, dữ liệu nhất thiết phải đi qua nút trung tâm để được chuyển đến đích của nó. Nếu nút trung tâm không hoạt động hoặc không thể truy cập được, không có dữ liệu nào có thể được truyền đi, điều này làm cho cấu trúc liên kết này dễ bị lỗi nút.
- Decentralized network: This topology is considered as a set of centralized networks connected together [[konolige2003map](#)], [[brand2014stereo](#)]. One central hub is employed for every little group of nodes to send and receive the information. If a hub fails, nodes which are directly relied to it become inaccessible.

¹Source: From user 1983 on Wikipedia. Licensed CC BY-SA

- Distributed network: This topology contains nodes that are connected to each others. Unlike the centralized topology, several paths are possible to reach one destination. Also, nodes are not affected by one node's failure. This allows to increase the survivability and to decrease the vulnerability of the network. The distributed topology is attractive to ensure reliable network [**cunningham2010ddf**]. Nevertheless, it is quite difficult to reach and to realize in reality, and the simulation are not fully distributed [**waharte2009coordinated**], [**cameron2009collaborative**], [**scherer2015autonomous**]. This is because all processing and decision making have to be on-board the robot whichs requires an important memory and a heavy computation.

2.3 Network typology

The typology of the network represents its type and, accordingly, the characteristics introduced.

2.3.1 Related work

MRS communication is a critical problem to tackle in the robotics community. In the last decade, the great progress in wireless technologies and MRS has created a growing interest for inter-robot communication. Hence, different methods have been proposed.

To face disaster scenarios such as fire in an industrial warehouse, authors in [**witkowski2008ad**] propose to use Wireless LAN, Bluetooth and ZigBee to form an Ad- Hoc network. Some robots of the fleet form the network infrastructure to support communication.

Authors in [**morgenthaler2012uavnet**] use two network standards to build a multi-UAV system. A IEEE 802.11s wireless mesh network is build using UAVs carrying mesh nodes directly connected to the flight electronics. Each of these mesh nodes acts as an AP in order to form an IEEE 802.11g.

A methodology that profiles the wireless links for Linux-based networked aerial vehicles is presented in [**kuschnig2012profiling.**] The approach includes two options to collect link quality information and to monitor the 802.11 wireless interface.

An evaluation of the standard 802.11a used between an UAV and an AP

is performed in [**kuschnig2012profiling**]. Results show that the Received Signal Strength (RSS) for the downlink decreases with the distance to the AP but remains at an acceptable level as well as the throughput.

Knowing that the visible light spectrum is much wider than the Radio Frequency spectrum, it seems interesting to uptake Visible Light Communication (VLC) based on Light Emitting Diodes (LEDs). IEEE has developed the 802.15.7 standard for short range communication using visible light. Authors in [**wang2014openvlc**] introduce a bidirectional communication using Open VLC. This solution hardware needs a Beagle Bone Black (BBB) board and a font transceiver that employs a single LED to both transmit and receive. The solution is implemented on Linux driver that communicates directly with the LED front-end and the Linux networking stack. The main idea to reuse the same LED for both Transmitting (TX mode) and Receiving Light signal (RX mode) is that when a node is transmitting data, the other node can expect the LOW symbol of the bit 0 and makes use of this time to switch his mode from RX to TX and transmit data.

The Zigbee – based on IEEE 802.15.4 – is a protocol of high level which is particularly useful for short communication range and low consumption. Several UAVs' application use this network [**asadpour2014routing**]. Communication can be made under three frequency bands depending on the chipset type: 868MHz for a bit rate of 20 kbps, 915MHz for 40 kbps and 2.4GHz for 250 kbps.

Authors in [**vidal2015multi**] adopt different sized groups composed of tactical UAVs and UAVs moving in delimited geographical areas. Virtualization techniques are used to adapt the upgrade of any function, which allows the flexibility in heterogeneous services.

In a search and rescue mission, for example, authors in [**scherer2015autonomous**] propose to do real-time streaming between UAVs over a large distance. The mission is divided into five principle phases: The pre-planning to define the best paths in the search area allowing to reduce the time of the mission then searching by just following the predefined way-points; the detection of the target and sending the new plans to the others, followed by the repositioning by setting a multi hop link to evaluate the situation by the base station and finally, the streaming of the video.

The multi-UAV, moving in a three dimensional space, need sophisticated

transmission systems to ensure flexibility. In [**scherer2015autonomous**], an omni-directional isotropic antenna is fixed on the base station and on all the deployed heterogeneous UAVs. For data exchange, the standard IEEE 802.11s mesh technology is used.

Authors in [**hayat2015experimental**] present a performances comparison between the standard IEEE 802.11n and the IEEE 802.11ac in a multi hop network. Both networks were experimented in indoor and outdoor environment, and in two mode, the access point mode (AP)/infrastructure mode and the mesh mode regarding throughput and fairness. In indoor experiments, for an infrastructure mode, the 802.11ac shows improved performances in both TCP and UDP throughput, and packet loss for UDP traffic. In outdoor experiments, for infrastructure mode, the throughput of 802.11n is three time higher than that of 802.11a however, the link quality drops more steeply. For mesh mode, 802.11n achieves higher throughput at close range but drops faster as soon as date rate gets higher and the range longer. The recorded throughput for mesh network is lower than infrastructure mode due to the longer inter-packet transmission times.

To face issues of maintenance connectivity, collision avoidance, robustness to failure and area coverage improvement, authors in [**ghedini2018toward**] propose a novel model that provides more efficient network topologies.

In [**harms2017development**], a new communication layer is proposed to deal with networks that requires a high bandwidth. For that, some mechanism are used to buffer messages, to compress data or to react to unexpected situations.

Table ?? lists some standards used in multi-robot systems. For each standard, we detail the concerned layer in the OSI model, its characteristics, the resulting performances, and the hardware and software used for experiments.

Bảng 2.3: Examples of some standards used in MRS.

Work	Standard	Layer	Characteristics	Performances	Hardware	Software
[Scherer et al., 2015]	IEEE 802.11s mesh technology	Network layer	Compatible with mesh mode and AP mode	Range: 100m, communication delay: 5ms, Throughput: 10-20Mbit/s	UAVs + laptop + wifi module	Middleware Robot Operating System ROS
[Hayat et al., 2015]	IEEE 802.11n	PHY and MAC layer	High throughput in both mode (AP and mesh), acceptable degree of fairness	Outdoor mode + single hop; Range: 500m; TCP throughput: 35Mbit/s (50m); FB: 40Mhz	Two UAVs + laptop + Compex WLE300NX 802.11abgn mini PCIe modules	Ubuntu Linux Kernel 3.2. with ath9k driver
	IEEE 802.11ac	PHY and MAC layer	Do not support mesh mode	Outdoor mode + single hop; Range: 500m; TCP throughput: 10Mbit/s (50m); FB: 80Mhz	Two UAVs + laptop + Compex WLE900N518 802.11ac 5Ghz miniPCIe modules	Ubuntu Linux Kernel 3.2. with ath10k driver

[Kuschnig et al., 2012]	IEEE 802.11a	PHY and MAC layer	RSS throughput decreases with the distance but still at an acceptable level	Outdoor over campus Range: 150m*150m; 10m; Throughput: 54Mbit/s (theo) and 27Mbit/s (prac); FB: 5Ghz	AP: Netgear WNDR3700 + Atheros AR9280 based wireless cards + UAV with Intel Processor + SparkLAN WPEA110N wireless card + antenna WIMO 18720.11 for UAV and AP	Linux-based OpenWRT Backfire 10.03.1RC5
[Asadpour et al., 2014]	IEEE 802.11n multi hop	PHY and MAC layer	Long convergence time and high routing overhead (with BATMAN protocol for Network Layer)	In-flight experiment; Throughput for 200m: 5.95Mbit/s; Convergence time (20-100m): 28s; Routing overhead (20-100m): 10msg/s	Two Arducopter + GB + WLAN IEEE 802.11n + XBEE-PRO	

[Morgen-thaler et al., 2012]	IEEE 802.11s mesh nodes for UAV to UAV and IEEE 802.11g for UAV to AP	network layer	In single hop, flying UAVs reach higher throughput than UAVs in the ground. These results are performed with the location based position mode which are lower than those with the signal strength positioning mode. These performances are higher than in multi-hop.	Single hop (1UAV altitude 3-5m and AP = 75m) in location positioning mode; TCP throughput = 6.5Mbit/s + in signal strength positioning mode; TCP throughput = 8.1Mbit/s	One UAVNet quadcopters Professional Mesh OM1P + two notebooks	Linux 2.6.37.6 Kernel generated by ADAM (embedded Linux distribution) + driver ath5k
[Muzaffar and Yannaz, 2014]	IEEE 802.11ab	PHY and MAC layer	Throughput decreases with the increase of the number of nodes	Range: up to 1000m × 1000m × 50m; Throughput: 54Mbit/s (prac); 5Ghz FB:	Simulation: UAVs + Ground Station	Omnnet++

[Wang et al., 2014]	IEEE 802.15.7	PHY and MAC layer	Bidirectional transmission, matching filtering and timing error recovery can increase the communication range and stability	<i>One hop:</i> throughput: 1.6kb/s, packet loss ratio 5%; <i>Two hop:</i> throughput 0.65kb/s, packet loss ratio 15%	Embedded BBB board + LED front end	Linux Kernel 3.8.13
---------------------	---------------	-------------------	---	--	------------------------------------	---------------------

2.3.2 Network standards and protocols

Since the exchange of information is necessary for coordination, the network used for multi-UAV communication has to ensure that the information circulates properly throughout nodes. Consequently, we have to overcome some issues such as:

- Node failure: When the link is broken or defected, the network must find a path to reach the node.
- Topology changes: Nodes move in the environment and generate changes in the topology that the network have to deal with rapidly.
- Communication bandwidth: UAVs possess certain information to exchange with other and need some bandwidth that supports the data exchange.

Taking into account the above cited issues, and to ensure a good integration between the visual SLAM and the communication, the distributed and cooperative wireless mesh network seems to be the most adequate network typology. It allows to introduce the following advantages:

- Improving the network reliability because infrastructureless network offers several paths, so that information may reach the destination even if only one link is broken.
- Allowing self scalability of the network by a quick adaptation to topology changes.
- Enabling rapid deployment with lower cost back-haul.
- Providing easily coverage in areas which have difficult access.
- Saving battery life due to its lower power consumption.
- Enabling to face a growing number of UAVs in the fleet.

Among the existing mesh network standards, the 802.11s amendment, related to the MAC layer, is interesting for MRS applications. It is an Extended Service Set network that supports broadcast, multicast and unicast communication. It contains the Hybrid Wireless Mesh Protocol (HWMP) as the default routing protocol. The HWMP is a hybrid routing protocol inspired by the AODV (Ad-hoc On-demand Distance Vector: an on demand and reactive portion) and the tree based protocol (a proactive portion). Thereby, it

contains the advantages of the reactive protocol since it prepares the routing table when nodes change their position, and thus provides the safer path. On the other hand, it contains the advantages of the proactive protocol since the routing table is ready which allow to save time when needed. In addition to the default routing protocol HWMP, the 802.11s mesh network supports other protocols like Optimized Link State Routing (OLSR), Better Approach to Mobile Ad hoc Networking (BATMAN), Wireless Distribution System (WDS), Open Shortest Path First (OSPF) and BABEL. According to [wang2010experimental], BATMAN proved better performances than HWMP and OLSR. Thus, experiments using HWMP and BATMAN have been performed to point out the saved data.

BATMAN protocol definition The BATMAN² is a proactive routing protocol inspired by AODV and OLSR. It is supported by multi-hop Ad Hoc mesh networks. It uses different approaches to route selection node by periodically sending OriGinator Messages (OGM) to neighbors with node information to next hop and destination because the routing decisions are distributed on nodes. In this protocol, each node decides for the next hop and not for the whole route so nodes do not use or even know the topology of the network. In the case of detecting other nodes, BATMAN protocol finds the best route to them. It also keeps track of new nodes and informs its neighbors about their existence.

2.3.3 Result and discussion

For the evaluation of the networked system using BATMAN³ protocol, we use heterogeneous nodes composed of three laptops: 2.40GHz dual core Linux machine, 2.27GHz i3 Linux machine, 2.50GHz i5 Linux machine and a Parrot AR-Drone 2.0 (See Figure ??).

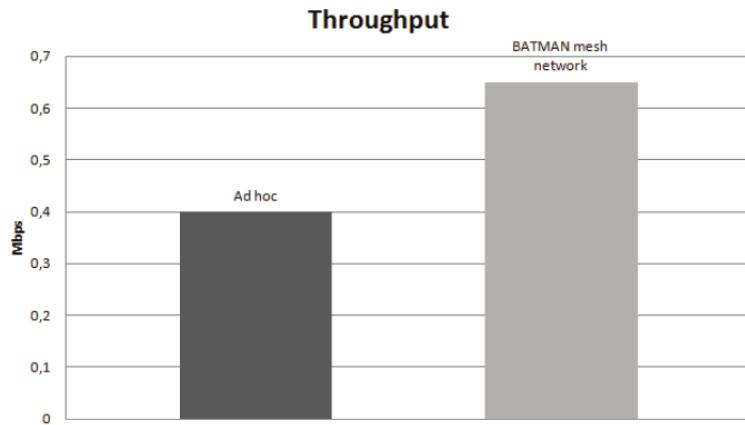
²Source: <https://www.open-mesh.org/projects/open-mesh/wiki>

³The BATMAN-adv version used for the testbed is available since 2013



Hình 2.3: Mesh network illustration between three laptops and one drone.

We simulate the broadcast of data between nodes in both Ad Hoc and mesh network with BATMAN protocol to underline the saved data. The drone was controlled from the laptop using a cross compilation. First, we perform an Ad Hoc network between endpoints and broadcast the data from node A to node B, C and D in the network. Then, we broadcast – in the same conditions – from node A to B, C and D with the BATMAN mesh network protocol. Results in Figure ?? show that the throughput achieved an average of 0.4 Mbits/s; whereas, the throughput evaluated in mesh network with BATMAN protocol, achieved an average of 0.65 Mbits/s. The BATMAN mesh network protocol improves by 1,5 times the throughput of the network compared to a basic Ad Hoc network.



Hình 2.4: Broadcast testbed throughput result in Ad Hoc and BATMAN mesh network protocol.

2.4 Network topology and strategy for MRS robustness

The topology of the network defines the geometric properties of nodes (robots in our case). The proposed strategy in this work represents the behavior adopted to face critical situations and to achieve a MRS robustness.

2.4.1 Related work

The challenge in MRS communication is to maintain reliable network during the mission in order to make the robots able to perform cooperative exploration [**rooker2007multi**], [**gupta2015survey**]. The strategy used for exploration affects the exchange of data among robots including type, destination and frequency.

The information exchanged between a robot and a server may be key-frames and map points [**schmuck2017multi**], or only features of selected key-frames and relative-pose estimates among robots and ground station [**forster2013collaborative**]. But mostly, robots exchange their local copies of the map and their poses [**fox2006distributed**], [**bresson2015general**], [**schuster2015multi**].

The amount of data exchanged may rapidly increase in size, which can cause network congestion and data loss. In order to reduce the bandwidth requirements, authors in [**mohanarajah2015cloud**] propose to send only compressed key-frames and updated key-frame poses. Also, authors in [**cunningham2010ddf**] propose a Decentralized Data Fusion-Smoothing And Mapping (DDF-SAM) approach, where each robot propagates towards other robots, its condensed local graph in order to achieve scalability and robustness to node failure.

Most works deal with the problem of communication while assuming an ideal network or aim at keeping team members within range of one another in order to focus their attention on higher level problems [**scherer2015autonomous**], [**burgard2005coordinated**].

Considering communication loss and/or limited bandwidth helps to prevent from mission failure and to ensure a more realistic scenario. Indeed, in real scenarios, many issues can arise such as having a distance among robots that exceeds the communication range, losing major information in a broken communication link, losing precious time in sending information due to lim-

ited bandwidth, etc. The exploration strategy has to take into account the mentioned issues to avoid mission failure in real world scenario. Some works began to tackle the exploration problem while considering communication limitations [**couceiro2014darwinian**], [**schmuck2017multi**].

In [**dai2018quality**], the aim is to sense a geometrically complex environment by assigning targets to robots when the spatial and temporal resolutions are satisfied. This approach uses a min-max energy path planning algorithm that obeys to a deadline time.

In this work, we make the choice to let UAVs exchange with each other only frontier points, robot poses, and assigned targets. This exchange happens at each iteration while considering UAVs' role, which are adapted according to the network topology. This adaptation allows also to cope with communication limitations.

2.4.2 Inter-robot communication approach

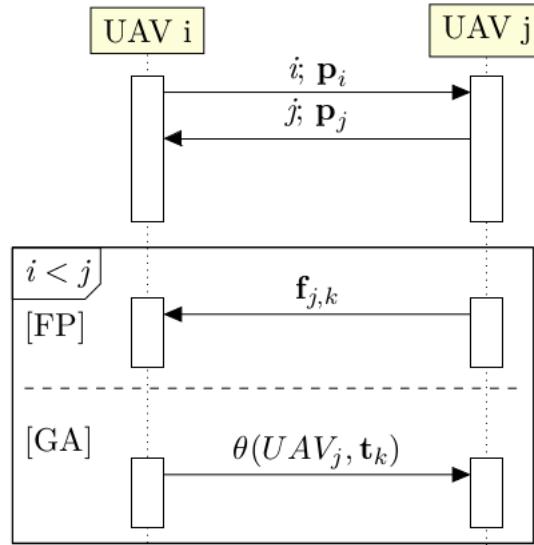
Interactions among members of the fleet are important especially in the exploration missions in order to prevent UAVs to explore the same regions, and to allow them to cooperatively discover the unknown areas more rapidly and in an optimized manner. However, inter-UAV communication is a challenging issue that requires to answer some practical questions: Which kind of data nodes must exchange? How often data should be shared? Should we consider a multi-hop data exchange? If so, how to identify the endpoints of the data exchange? How to cope with communication limitations? These questions are addressed in the following subsections.

Multi-UAV interaction and data exchange

In the proposed cooperative exploration strategy, local frontier points $\mathbf{f}_{i,j} \in \mathcal{F}$, current pose \mathbf{p}_i , and current target point \mathbf{t}_m are exchanged instead of the whole copy of the local map. This is expected to produce a considerable reduction of exchanged data volume, and, consequently, memory consumption. The sequence diagram⁴ in Figure ?? details (timing and information) the messages exchanged between two UAVs. UAV_i, with $i \in [1..n_c]$ and UAV_j with $j \in [2..n_c](i < j)$ are robots in the cluster \mathcal{C} . They forward their respective id number and current poses \mathbf{p}_i and \mathbf{p}_j . Since $i < j$, the explorer

⁴This diagram uses Unified Modeling Language's sequence diagram notation.

UAV_j sends to the selected *leader* UAV_i its local frontier points $\mathbf{f}_{j,k}$ during the frontier processing (FP) step.



Hình 2.5: Data flow between two robots. The FP and GA stand, respectively, for frontier processing and goal assignment.

Then, the *leader* performs the goal assignment process (GA) and sends back to UAV_j the selected target point \mathbf{t}_m . The cited example represents two UAVs in the cluster \mathcal{C} . In case of multiple UAVs in \mathcal{C} , the same sequences will be performed among one *leader* and multiple *explorers*.

Exploration strategy to face communication loss

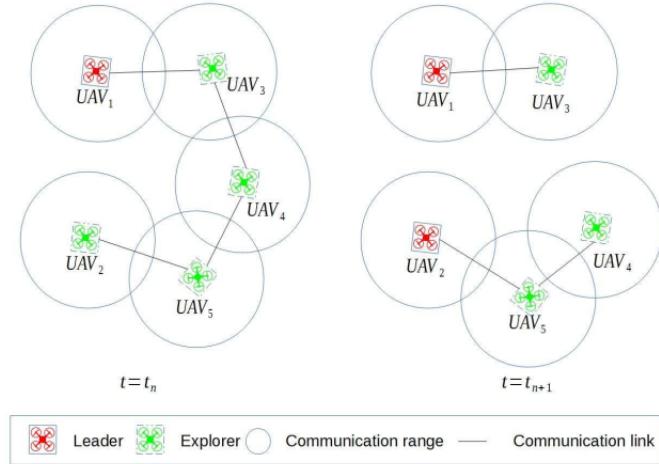
In the proposed system, considering the communication limitations is important to ensure the mission continuity. In case of losing contact with the *leader* due to communication failure or UAV getting stuck, another *leader* is self-selected in the next iteration so that the mission can continue. In Figure ??, at $t = t_n$, the fleet is composed of one cluster where UAVs are able to communicate with each other. One *leader* handles the decisions for others. At $t = t_{n+1}$, the communication link fails between UAV₃ and UAV₄. The fleet is divided into two clusters with one *leader* each.

Particular case In case of losing contact with the *leader* and before another one is selected, *explorers* let a timer τ expires while waiting for target assignment. If no target is received, the *explorer* selects its own target according to local information.

Using this strategy, as long as – at least – one UAV exists in the fleet, the mission will continue until all the bounded environment is explored (no candidate frontier points are left).

Data exchange strategy discussion

In the proposed strategy, data flow exchange is repeated at each iteration while taking into account network topology changes to define clusters. The starting points and endpoints are defined according to these roles. The UAV's role also specifies the type of exchanged data. In addition to the exchanged current pose \mathbf{p}_i and *id* number i , if the UAV $_i$ is an *explorer*, it would passively share information about itself and its surrounding environment with the leader (frontier points $\mathbf{f}_{i,j} \in \mathcal{F}$); else, its role would be to send targets to visit to the *explorers* (target points $\mathbf{t}_k \in \mathcal{G}$).



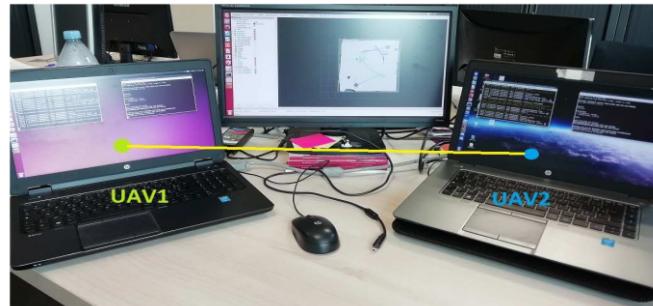
Hinh 2.6: Role evolution in limited communication range.

The proposed strategy ensures a mission continuity in case of communication loss. Nevertheless, the UAV may explore regions already explored by other nodes, since no local maps are exchanged nor fused to keep track of

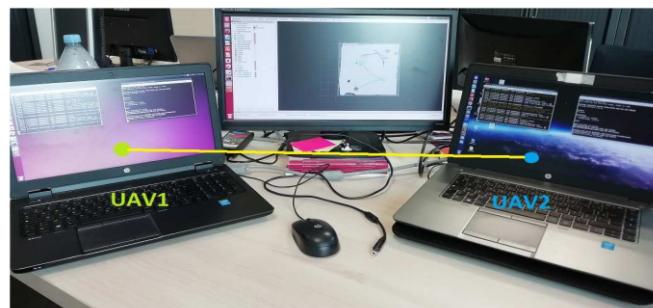
visited areas. Thus, in case of communication loss, the mission accomplishment is favored over consumption minimization of resources, such as time and battery.

2.4.3 Results and discussion

As UAVs are equipped with IEEE 802.11b,g wireless card, we set up an infrastructureless network within the set of robots to quantify the data exchange among members of the fleet, as well as, to determine the performance of the robot network. Runs with 2 and 3 UAVs were performed (See Figure ??). The network was composed of two 2.60GHz i7 Linux machines and a 2.50GHz i7 Linux machine. The number of robots used for evaluation is limited to three, however, the proposed system architecture is not constrained to a fixed number of robots.



(a) Two UAVs (Laptops)



(b) Three UAVs (Laptops).

Hình 2.7: Ad Hoc network illustration during exploration mission.

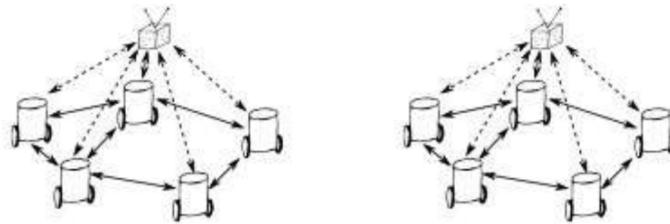
Network setting: From one to multiple machines

When running multiple UAVs on a single machine (like in previous simulations in chapter 3), one ROS master is responsible of managing the intra and inter-processes communication using publisher/subscriber. In case of multiple machines, two configurations are possible:

- One ROS core for multiple machines (See Figure ??): Even with multi-UAV on different machines, one ROS master/core can be adopted by specifying the machine running the ROS core. In this case, the SoS will manage inter-processes communication the same way as if they are on the same machine. The exception is that a real world communication is used instead of a shared memory for a multi-UAV running on a single machine.
- Multiple ROS cores for multiple machines (See Figure ??): when running different ROS cores on different machines, each UAV manages its own master. In this multi-cores system – also called multi-master system –, a synchronization among these masters needs to be done.

Nonetheless for both cases, the virtual environment to be explored in *Gazebo* has to be the same so that UAVs explore the same environment at the same time. Specifically, this means that the IP client of *Gazebo* has to match the IP server of the machine running the *Gazebo* world by setting *GAZEBO_MASTER_URI*.

When running a single ROS core for multiple UAVs, a reliable network is needed, else ROS processes would not work properly when the network connection is unstable. Also, running multi-master system is more realistic because in real world, each UAV has to be functional, independent and cooperative to achieve the mission objectives, which is, in this work, the exploration of an unknown environment. A multi-master system represents a distributed system configuration, which has different advantages such as scalability to fault tolerance.



(a) One ROS core for multiple machines.
(b) Multiple ROS cores for multiple machines.

Hình 2.8: Two ROS configurations in multiple machines case. Image from [andre2014coordinated].

Multi-master systems require synchronization. For this, different packages exist such as the *multi-master fkie* package⁵ which allows unicast as well as multicast transmissions using UDP protocol. The *wifi_comm* package⁶ implements the Optimized Link State Routing (OLSR), but can be used with different routing algorithms. The *recon_multimaster* package⁷ is a centralized multi-master system that implements building blocks around the ROS communication layer but do not implement communication itself. Authors in [andre2014coordinated] propose a distributed approach where each robot runs a master managing its local communication using the *Adhoc_communication* package⁸ where AODV protocol is implemented. Another multi-master package is the *Nimbro_network* package⁹ which offers a robust transport of ROS topics and services over unreliable networks. The above cited multi-master packages are not an exhaustive list and other synchronization packages exist.

For simplicity and as a first approach, the *multi-master fkie* package¹⁰ is used to run the adopted multi-core system. This package allows us to both use and synchronize multiple cores using the default protocol UDP. For ROS topics data exchange, TCP protocol is used. For an effective evaluation especially concerning the time, clock synchronization needs to be ensured. Network Time Protocol (NTP) is used to synchronize laptops within a few

⁵Source: http://wiki.ros.org/multimaster_fkie

⁶Source: http://wiki.ros.org/wifi_comm

⁷<http://wiki.ros.org/rocon>

⁸Source: http://wiki.ros.org/adhoc_communication

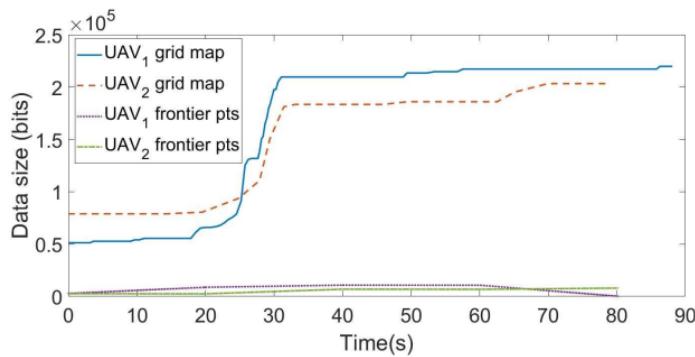
⁹Source: https://github.com/AIS-Bonn/nimbro_network

¹⁰Source: http://wiki.ros.org/multimaster_fkie

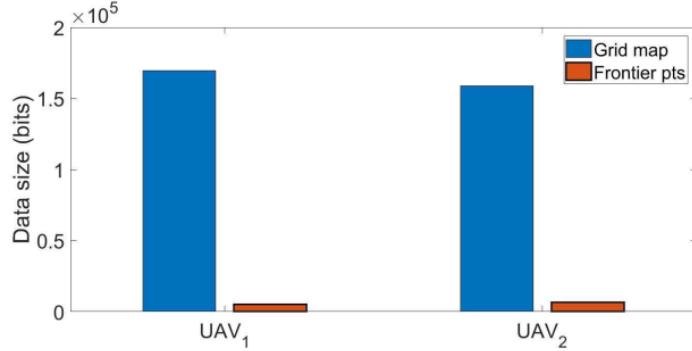
milliseconds of Coordinated Universal Time (UTC).

Exchanged data size evaluation

The first evaluation aims at pointing out the amount of exchanged data by sharing local frontier points instead of local grid maps (See Figure ??).



(a) Evolution of the amount of exchanged data during the mission.



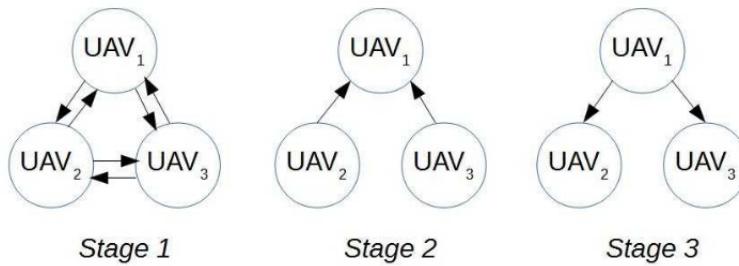
(b) Average of the amount of exchanged data during the mission.

Hinh 2.9: Data size when UAVs exchange a whole copy of their local grid map versus frontier points of it.

Figure ?? shows that the size of grid maps increases consequently in time compared to the size of frontier points that is almost constant during the mission. According to results in Figure ??, the size of data saved, when exchanging frontier points instead of grid maps, is almost divided by 10.

Exchanged data average time evaluation

Depending on the size and frequency of the exchanged data, the time allocated for communication may increase with the increasing number of robots. Thus, evaluations of time behavior and its potential impact on the exploration performances have been conducted. Figure ?? shows the network topology evolution during data exchange.



Hình 2.10: Network topology evolution during a loop with three cooperative UAVs.

The information is exchanged in three stages and they are the following:

- Stage 1: The *id* number and current poses \mathbf{p}_i with $i \in [1..n_c]$.
- Stage 2: The frontier points $\mathbf{f}_{i,j}$ with $i \in [1..n_c]$ and $j \in [1..n_i]$.
- Stage 3: The target points assignment $\theta(UAV_i, \mathbf{t}_k)$ with $i \in [1..n_c]$ and $k \in [1..n_c]$.

Table ?? shows the average time spent in data exchange during exploration. A slight increase in the computed average time occurs when increasing the number of robots. The time spent in communication is relatively negligible compared to the total time of exploration.

Bảng 2.4: Communication module timings.

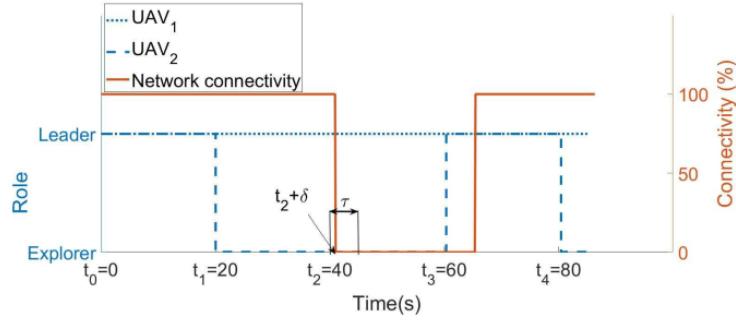
UAVs		Time spent in stage 1 (s)			Time spent in stage 2 (s)			Time spent in stage 3 (s)			Time for exploration (s)
		UAV ₁	UAV ₂	UAV ₃	UAV ₁	UAV ₂	UAV ₃	UAV ₁	UAV ₂	UAV ₃	
Two UAVs	UAV ₁	θ	0.136 ± 0.139		θ	θ		θ	0.022 ± 0.017		120, 1
	UAV ₂	0.065 ± 0.068			0.026 ± 0.008			θ	θ		
Three UAVs	UAV ₁	θ	0.056 ± 0.065	0.575 ± 0.769	θ	θ	θ	0.335 ± 0.407	0.765 ± 0.678		86
	UAV ₂	0.107 ± 0.111		0.483 ± 0.678	0.185 ± 0.244	θ	θ	θ	θ		
	UAV ₃	0.267 ± 0.165	0.616 ± 0.549		0.251 ± 0.109	θ	θ	θ	θ		

Network interruption evaluation

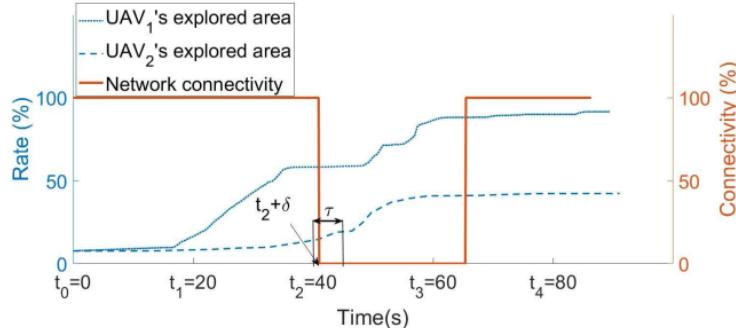
To evaluate the system behavior during communication failure, the network connectivity has been voluntarily interrupted during the exploration. Figure ?? shows the robot's role and the exploration rate performance when the network connectivity is interrupted and then recovered.

The system performs neighbor discovering, role selection and target assignment at each loop of $t_{t+1} = t_i + i.r$ with $t_0 = 0s$ and $r = 20s$. In Figure ??, at $t = t_0$, both robots begin with a *leader* role. When discovering each other (at $t = t_1$), UAV₁ selects itself as leader and UAV₂ becomes explorer. Consequently, UAV₁ assigns a target to UAV₂. UAV₂ receives the target and attempts to reach it. At $t = t_2 + \delta$, the connectivity is voluntarily interrupted, just after the role selection but before the target information is assigned to UAV₂. After a time period τ , UAV₂ selects a target taking into account its own local data. In the next loop (at $t = t_3$), since the connectivity is still interrupted, UAV₂ finds no neighbors and selects itself as a *leader*. Both robots perform exploration independently, that is, without cooperation. Shortly after $t = t_3$, the connectivity is re-established. Thus, robots are able to cooperate again and UAV₂ takes over the role of *explorer*. In Figure ??, even after the network connectivity is interrupted, the exploration continues to be performed by both UAVs. When the connectivity is

re-established, the *leader* collects the frontier points and performs frontier processing where it finds that no candidates targets are remaining, that is, all the environment is now explored. Therefore, the mission is accomplished.



(a) UAV's role along with network connectivity.



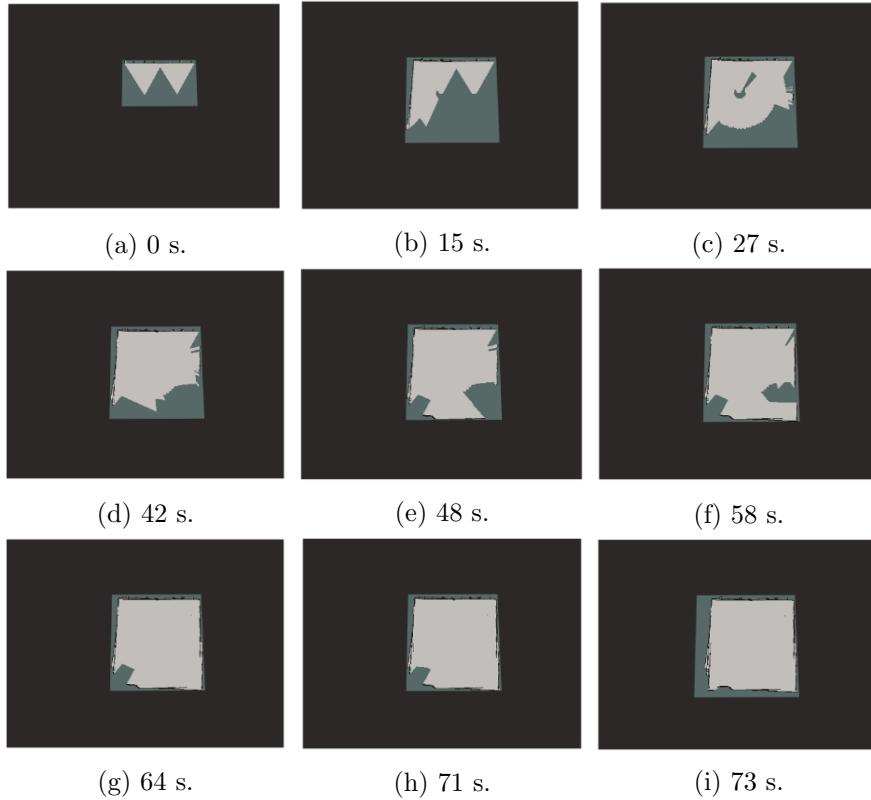
(b) Exploration rate along with network connectivity.

Hình 2.11: Two cooperative robots exploration along with network connectivity.

Global map evaluation

The exploration mission aims at creating a global map of the environment, in an efficient manner. Hence, we want to evaluate the obtained global map during a cooperative mission running with infrastructureless network.

Figure ?? illustrates the growth of the reconstructed global 2D occupancy grid map during the exploration mission.



Hinh 2.12: Global reconstructed 2D occupancy grid map during exploration mission with two cooperative UAVs.

2.5 Conclusion

In this chapter, we addressed two subproblems of the MRS network. The first one deals with the network typology that is the characteristics that should be ensured by the network; whereas, the second one points out to the geographic distribution of nodes in the network, called topology. We also propose to face critical situations.

The typology of the proposed network is based on adopting a mesh network along with BATMAN protocol. Results of broadcasting data performed in testbed with heterogeneous nodes including three laptops and one AR-Drone show that BATMAN mesh network protocol improves the throughput by 1.5 times compared to basic Ad Hoc network.

In a limited communication range, the topology of the network varies during the mission. Consequently, the UAVs' behaviors are adapted by always updating their roles to leader or *explorer* during the mission. Even with an *explorer* role, the UAV is able to continue its mission if the network experiences some issues. Furthermore, we propose to exchange the frontier points of the local map instead of the whole copy of it, which allows to reduce the shared data volume, and consequently memory consumption. The results of the testbed performed with three UAVs, show that the proposed communication module is able to cope with network limitations. They also show that the proposed strategy uses 10 times less data than a strategy that makes the robots exchange the whole local map.[**dirac**]

Conclusion