

CỘNG ĐỒNG ROS: ROBOT OPERATING SYSTEM

NAVIGATION2

ROS2 FOR EVERYONE

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

Action Server

- Được sử dụng để thực thi những nhiệm vụ tốn nhiều thời gian như navigation.
- Action servers tương tự với một server kinh điển. Một Client sẽ yêu cầu một nhiệm vụ để được hoàn thành và nhiệm vụ này mất nhiều thời gian.
- Bởi vì các nhiệm vụ chạy mất nhiều thời gian, action servers cũng sẽ cung cấp feedback tới các clients. Feedback này được định nghĩa trong file .action cùng với các loại yêu cầu và kết quả.
- Action servers được sử dụng trong navigation2 để giao tiếp với mức độ BT navigator cao nhất qua kiểu NavigateToPose.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

Lifecycle Nodes và Bond

Lifecycle nodes là các nodes đặc biệt trong ROS2, chúng chứa các sự chuyển đổi máy trạng thái cho việc đưa lên và chia nhỏ của ROS2 servers. Điều này giúp trong hành vi xác định của hệ thống ROS trong startup và shutdown.

- Chỉ trong Navigation2, chúng ta sử dụng một wrapper của LifecycleNodes là ***nav2_util LifecycleNode***.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.1 Behavior Trees (BT)

BT đang trở nên phổ biến trong các nhiệm vụ robotics phức tạp. Nó tạo ra một giao diện nhiều tính kết nối và khả năng có thể hiểu cho việc định nghĩa đa bước hay nhiều trạng thái các ứng dụng.

Sử dụng **BehaviorTree CPP V3** như là ***behavior tree library***. Chúng ta có thể tạo ra node plugins cái mà có thể được cấu trúc vào trong một tree, bên trong **BT Navigator**. Nó có khả năng để mang subtree, nghĩa là Navigation2 behavior tree có thể được mang vào trong một cấp độ BT cao hơn.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.2 Navigation Servers

1.2.1 Planner, Controller và Recovery Servers

Đây là 3 action servers được sử dụng để tổ chức một bản đồ của các plugins thuật toán để hoàn thành nhiều nhiệm vụ thay đổi. Chúng cũng tổ chức sự hiện diện môi trường cái mà được sử dụng bởi algorithm plugins tổ các đầu ra tính toán của chúng

Planner và controller servers sẽ được cấu hình ở thời gian chạy với các tên (aliases) và kiểu của các thuật toán để sử dụng.

Đối với recovery servers, mỗi trong số chúng đều chứa các tên riêng, tuy nhiên mỗi plugin sẽ đưa ra action server đặc biệt của riêng nó. Recovery server cũng chứa một costmap subscriber tới local costmap, cái mà nhận các update theo real-time.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.2.2 Planners

Nhiệm vụ của một planner là tính toán một lộ trình hoàn tất một vài chức năng mục tiêu. Lộ trình này cũng có thể được biết như là một route, phụ thuộc vào thuật ngữ và thuật. Ví dụ như tính toán kế hoạch di chuyển từ điểm đầu đến điểm đích.

Planners có thể được viết để:

- Tính toán lộ trình ngắn nhất

- Tính toán lộ trình bao phủ hoàn chỉnh

- Tính toán lộ trình dọc theo các tuyến đường thừa thớt hoặc xác định trước.

Trong Navigation2, planners thường được sử dụng để tính toán một lộ trình tối ưu từ vị trí hiện tại đến một vị trí đích.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.2.3 Controllers

Là cách mà chúng ta theo lộ trình toàn cục được tính toán hay hoàn tất một nhiệm vụ cục bộ. Controller sẽ có sự truy cập tới một môi trường cục bộ để tính toán các nỗ lực kiểm soát khả thi cho base để tuân theo.

Controllers có thể được viết để:

- Theo một lộ trình (path)

- Gắn kết với một charging station bằng việc sử dụng các bộ dò tìm trong odometric frame.

- Lên thang máy

- Tương tác với một công cụ

Trong Navigation2, nhiệm vụ cơ bản của controllers là tính toán một control effort theo như global plan.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.2.4 Recoveries

là một mainstay của hệ thống chấp nhận lỗi. Mục tiêu của recoveries là để đối mặt với các điều kiện không xác định hoặc thất bại của hệ thống và tự động xử lý chúng. Ví dụ như khi robot của chúng ta bị mất path (hoặc quỹ đạo), nó sẽ tự động quay tròn cho đến khi tìm được quỹ đạo mới.

Trong một trường hợp hệ thống thất bại hoàn toàn với việc tìm ra path, một recovery sẽ có thể được thi hành để gọi một operators attention cho sự giúp đỡ. Điều này có thể được làm với email, SMS, Slack, Matrix...

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.2.5 Waypoint Following

Waypoint following là một đặc điểm cơ bản của hệ thống navigation. Nó nói cho hệ thống của chúng ta cách để sử dụng navigation để có đa điểm đích.

nav2_waypoint_follower chứa một chương trình waypoint following với một plugin interface cho các thực thi nhiệm vụ chỉ định.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.3 State Estimation

Trong navigation, có hai sự trao đổi thông tin quan trọng cần phải được cung cấp, theo như các tiêu chuẩn giao tiếp:

Sự chuyển đổi **map** tới **odom**: được cung cấp bởi một hệ thống vị trí (SLAM, localization, mapping).

Sự chuyển đổi từ **odom** tới **base_link**: bởi hệ thống odometry.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.3.1 Standards

Mối quan hệ giữa các sự chuyển hệ tọa độ:

Map → **odom** → **base_link** → [**sensor_frame**]

TF2 là thư viện chuyển đổi thông tin biến thời gian trong ROS2, chúng ta sử dụng để đại diện và đặt được các chuyển đổi đồng bộ về thời gian.

Nó là công việc của hệ thống vị trí toàn cục (GPS, SLAM...) để cung cấp việc chuyển từ **map** → **odom**. Sau đó nó theo quy tắc của hệ thống odometry để cung cấp việc chuyển **odom** → **base_link**. **Base_link** nên static và được định nghĩa trong URDF.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.3.2 Global Positioning: Localization and SLAM

Tối thiểu, global positioning system (GPS, SLAM...) cung cấp việc chuyển **map** → **odom**. Chúng ta cung cấp **amcl**, là công nghệ Monte-Carlo Localization được dựa trên phương pháp lọc phần tử cho việc định vị của một map tĩnh. Chúng ta cũng sử dụng SLAM Toolbox như là thuật toán SLAM mặc định cho việc sử dụng để tạo ra vị trí và phát sinh một map tĩnh.

Các phương pháp này cũng sẽ tạo ra đầu ra khác bao gồm các topics vị trí, maps, hay metadata khác, nhưng chúng phải cung cấp sự chuyển thông tin. Các phương pháp đa vị trí có thể được kết nối với nhau bằng việc sử dụng `robot_localization`.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.3.3 Odometry

Đây là quy tắc của hệ thống đo lường để cung cấp thông tin chuyển từ odom -> base_link. Odometry có thể lấy từ nhiều nguồn khác nhau như LIDAR, RADAR, wheel encoders, IMUs...

Mục đích của odometry là để cung cấp một local frame bằng phẳng, liên tục dựa trên chuyển động robot. Hệ thống vị trí toàn cục sẽ cập nhật các thông tin chuyển liên quan tới global frame.

Robot Localization được sử dụng điển hình cho sự kết hợp này. Nó giúp kết hợp các cảm biến lại với nhau. Điển hình là sự kết hợp của IMUs và wheel encoder.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.4 Environmental Representation

1.4.1 Costmaps và Layers

Costmap là sự thể hiện môi trường hiện có, là một lưới 2D chứa một giá trị trong (unknown, free, occupied hoặc inflated cost). Sau đó nó được tìm kiếm để tính toán một kế hoạch toàn cục hay được làm mẫu để tính toán các local control efforts.

Costmap layers có thể được tạo ra để tìm kiếm và truy vết vật cản trong viễn cảnh cho tránh vật cản sử dụng camera hoặc cảm biến chiều sâu. Chúng có thể được sử dụng để chuyển đổi môi trường 2D hoặc 3D chỉ việc tạo dấu vật cản.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.4.2 Costmap Filters

Là costmap layer được dựa trên phương pháp của việc áp dụng các thay đổi hành vi phụ thuộc không gian vào trong filter masks.

Trong navigation2, nó được thi hành như costmap plugins.

Để tạo ra một filtered costmap và thay đổi hành vi robot trong các khu vực được ghi chú, filter plugin đọc data từ filter masks. Data này được chuyển đổi tuyến tính vào thành đặc điểm bản đồ trong không gian lọc.

I. CÁC KHÁI NIỆM TRONG NAVIGATION2

1.4.3 Other Forms

Nhiều các dạng khác của sự thể hiện môi trường như:

- gradient maps: tương tự với costmaps nhưng biểu thị gradients .
- 3D costmap
- Mesh maps
- Vector space