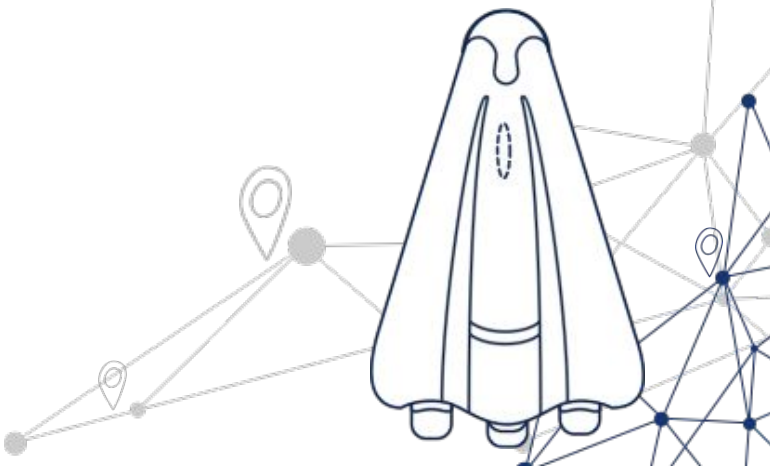
ROS NAVIGATION WORKSHOP

DAY 2

obodroid
Corporation Limited



หุ่นยนต์เริ่มเดินอัตโนมัติ

กันแล้วใช่ไหมล่ะ

เรื่องของการ

ปรับแต่งค่าต่างๆ

เราต้อง ควบคุมมันให้ได้ !!!

obodroid
Corporation Limited

ROS Navigation Parameter

1. Parameter แบบที่ต้องปรับก่อนเริ่มทำงาน [Static]
   - ใส่ใน yaml, launchfile สถานเดียวเลย
   - ปรับพวกค่าสำคัญๆ ที่เปลี่ยนระหว่างทำงานไม่ได้
   - ถ้าจะแก้ ต้องปิด Node และเปิดใหม่

2. Parameter แบบที่สามารถปรับระหว่างทำงานได้ [Dynamic]
   - เรียกว่า Dynamic Parameter (Dynamic Reconfigure param)
   - ใส่ก่อนทำงานก็ได้ (แบบด้านบน)
   - ปรับด้วย rqt_reconfigure ระหว่างทำงานก็ได้

ROS Navigation Parameter

1. Parameter แบบที่ต้องปรับก่อนเริ่มทำงาน [Static]
   - ใส่ใน yaml, launchfile สถานเดียวเลย
   - ปรับพวกค่าสำคัญๆ ที่เปลี่ยนระหว่างทำงานไม่ได้
   - ถ้าจะแก้ ต้องปิด Node และเปิดใหม่

แก้ไฟล์
ก่อนเริ่มเปิด Node

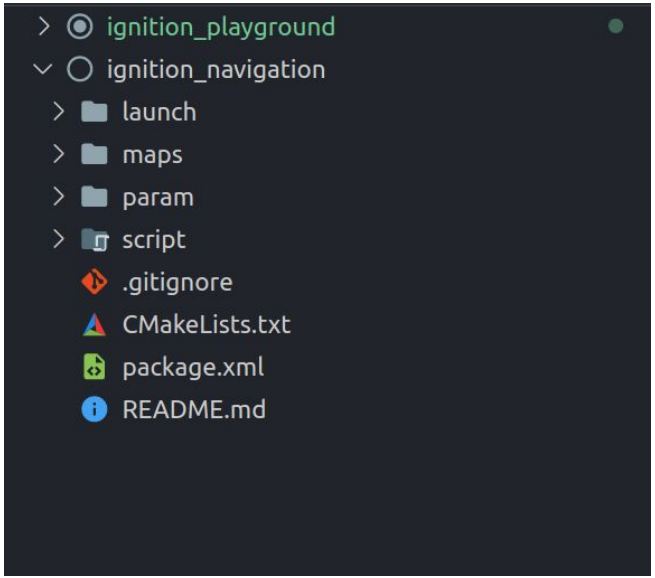
2. Parameter แบบที่สามารถปรับระหว่างทำงานได้ [Dynamic]
   - เรียกว่า Dynamic Parameter (Dynamic Reconfigure param)
   - ใส่ก่อนทำงานก็ได้ (แบบด้านบน)
   - ปรับด้วย rqt_reconfigure ระหว่างทำงานก็ได้

แก้ระหว่างที่
RUN NODE ได้

ROS Package Structure

- โดยปกติ ROS Package มักจะมีการเก็บ CONFIG Parameter ต่างๆ ใน Folder ประมาณนี้
  - config
  - param

- ใน ROS จะเก็บ Parameter เป็น YAML ไฟล์



YAML ?

มันอ่านว่าไงฟระ

เป็นการเก็บ Config แบบ
param_name: value
โดยสามารถซ้อนชั้นกันได้

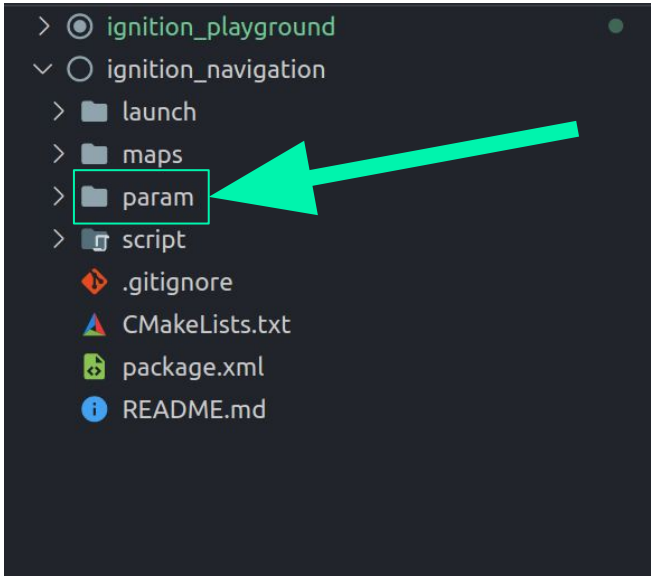
```yaml
# HELLO YAML
# COMMENT

parameter_int: 10
parameter_double: 3.141
parameter_bool: True
parameter_text: "HELLO IT'S ME"

parameter_array: [[a,b], [c,d]]

key_value_array:
  - {sub_param_a: test_value_1, sub_param_b: blabla}
  - {sub_param_a: test_value_2, sub_param_b: boxbox}

parameter_nested_top:
  nested_inside: 5
  nested_again:
    inside: 10
```

ROS Package Structure

- โดยปกติ ROS Package มักจะมีการเก็บ CONFIG Parameter ต่างๆ ใน Folder ประมาณนี้
  - config
  - param

- ใน ROS จะเก็บ Parameter เป็น YAML ไฟล์

- ใน Navigation Stack Template ของ Obodroid เราเก็บ CONFIG Parameter ใน folder param



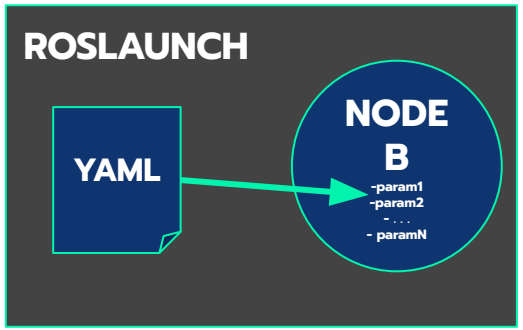
ROS Parameter : STATIC PARAM Loading

เมื่อพูดถึง Node ต่างๆใน ROS

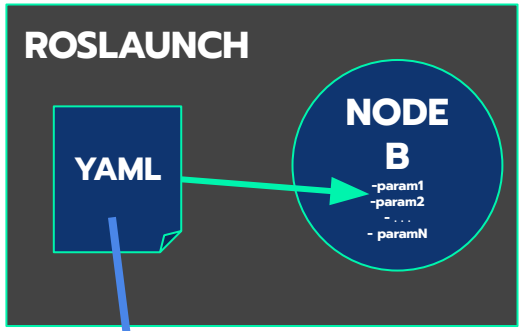
- เราสามารถเขียนโปรแกรม ให้ Set Parameter จากด้านนอก Node เข้าไปใช้ใน Node ได้ (ตอนจังหวะจะเริ่มเปิด Node)
  > **rosrun** package_a node_a param1:=999

- แต่ถ้าเรามี Parameter เยอะมากๆ หลายๆตัว
  - เก็บไว้ใน file YAML แล้วโหลดใส่ Node ด้วย **roslaunch**
    > param1:=1
    > param2:=True
    > . . . . . เยอะมาก

YAML
param1: 1
param2: True
. . . .
paramN: “text”



ROS Parameter : STATIC PARAM Loading

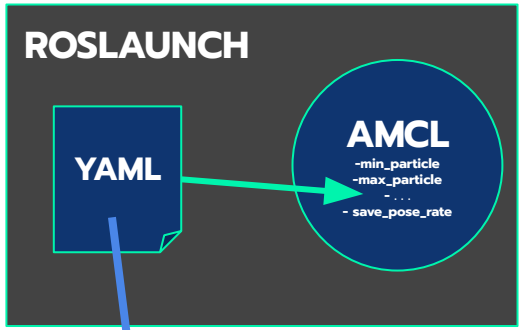



```xml
<?xml version="1.0"?>
<launch>
    <node pkg="my_package" type="executable" name="my_node_name">
        <!-- PARAMETER GOES HERE -->
        <rosparam file="example_param.yaml" command="load" />
    </node>
</launch>
```

YAML
param1: 1
param2: True
. . . .
paramN: "text"

```yaml
# example_param.yaml
param1: 1
param2: True
#........
paramN: "text"
```

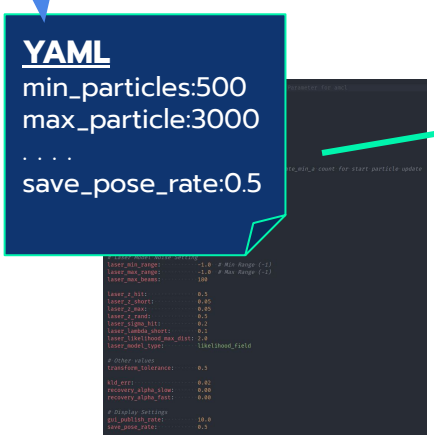
ROS Parameter : STATIC PARAM Loading



amcl.launch

แบบใส่ทีละอันก็มีนะ

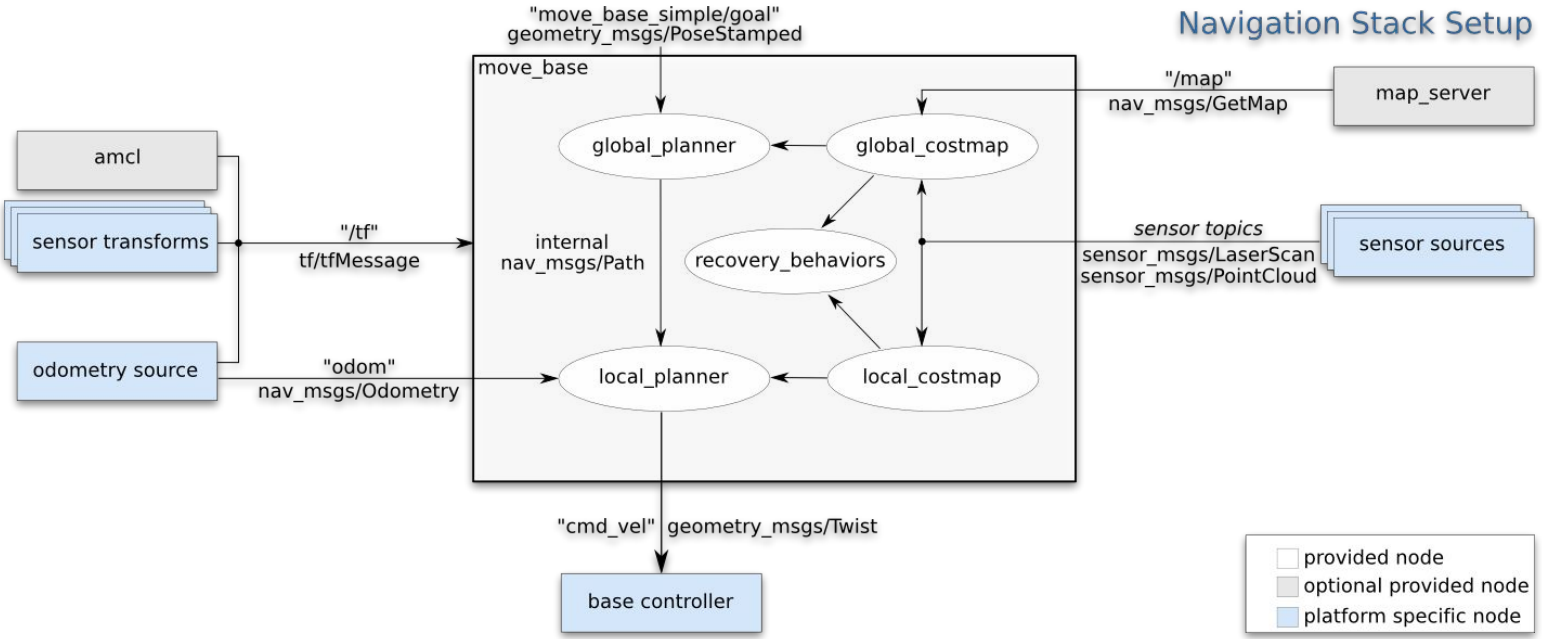
```
Theppasith N, a day ago | 1 author (Theppasith N)
<?xml version="1.0"?>
<!--
    AMCL Node Launching
    - Launch AMCL with custom yaml parameter loading
    Author : Theppasith N. <theppasith.n@obodroid.com>
-->
<launch>
    <param name="use_sim_time" value="True" />
    <node pkg="amcl" type="amcl" name="amcl">
        <!-- Starting Position in map frame -->
        <param name="initial_pose_x" value="-12.636" />
        <param name="initial_pose_y" value="-10.974" />
        <param name="initial_pose_a" value="3.14" />
        <!-- Topic Remapping (rename) -->
        <remap from="scan" to="scan" />
        <!-- Load YAML as Private Param for this node -->
        <rosparam file="$(find ignition_navigation)/param/amcl/amcl_params.yaml" command="load"/>
    </node>
</launch>      Theppasith N, 3 days ago • Add Parameter for amcl
```



ยัดทั้งหมดเข้าไปใน node ทีเดียว แบบง่ายๆ
ด้วย roslaunch - “rosparam” tag

ระวัง : บรรทัดล่างทับบรรทัดบนได้นะ ถ้าซ้ำกัน

ROS Navigation Stack : Components



ROS Navigation Stack : Components (Nodes)

รู้ว่าตัวเองอยู่ตรงไหน

AMCL

เดินไปจุดหมายด้วยตัวเอง

MOVE_BASE

MAP_SERVER

โหลดไฟล์รูป -> แผนที่

AMCL : Adaptive Monte-Carlo Localization

INPUT

TF DATA

LiDAR 2D DATA

ODOMETRY DATA

MAP (OCCUPANCY GRID)

รู้ว่าตัวเองอยู่ตรงไหน

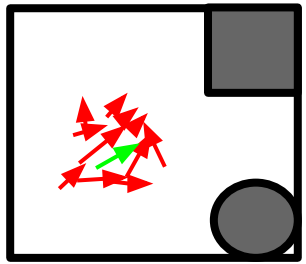
AMCL

OUTPUT

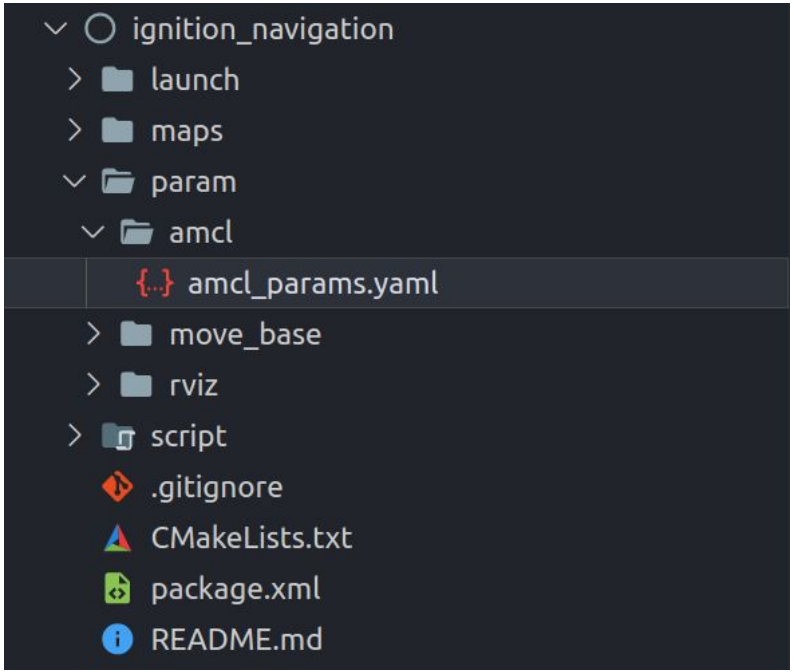
Correction TF
(map->odom)

AMCL Pose

AMCL Particle Cloud
(ไว้ดูลักษณะการทำงาน)



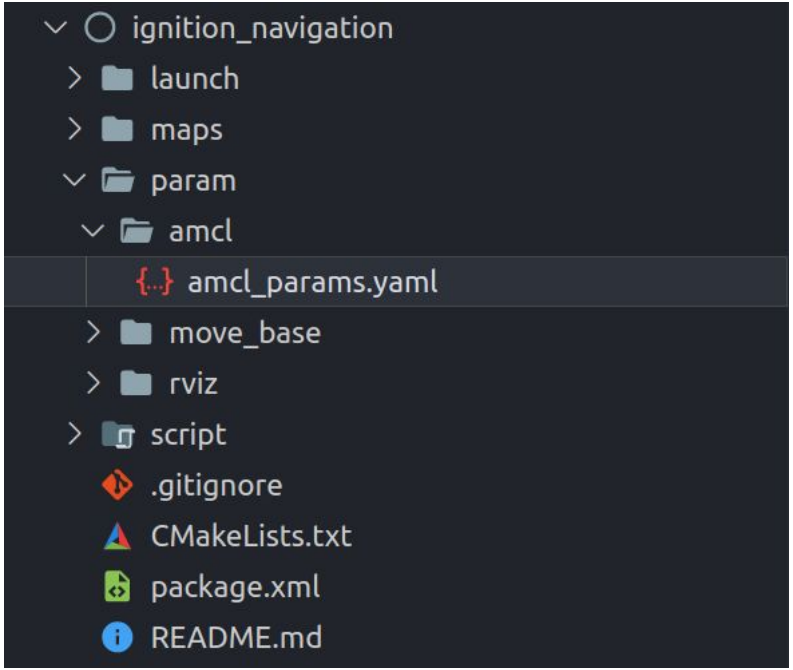
AMCL : Parameter



AMCL : Parameter

แบ่ง Config ออกเป็นกลุ่มๆดังนี้

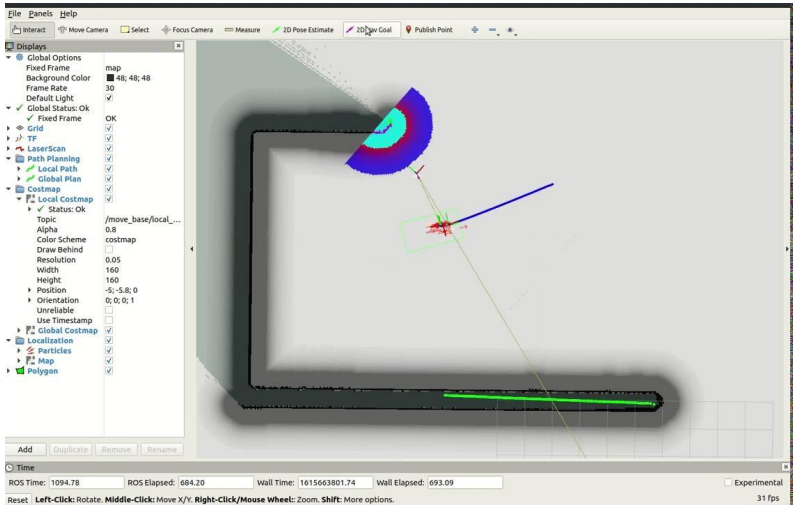
- Filter Parameter
  เกี่ยวกับการเกิด ลูกศร (Particle) ต่างๆ
- Laser Model Parameter
  เกี่ยวกับการเทียบและให้คะแนน laser
- Odometry Model Parameter
  เกี่ยวกับ ลักษณะการ resampling
  (โปรยลูกศรให้เป็นรูปต่างๆ)

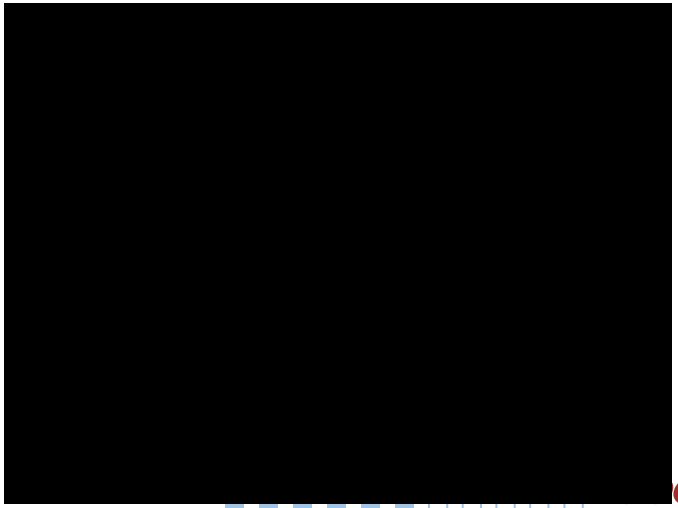


AMCL : Filter Parameter

- Min_particles ปริมาณ ลูกศรที่น้อยที่สุดที่เราจะมีได้
- max_particles ปริมาณ ลูกศรมากที่สุดที่เราจะมีได้

ใช้บอกจำนวน particle ที่เราจะโปรยลง map มากไปก็เปลืองพลังประมวลผล น้อยไปก็อาจบอกตำแหน่งผิด





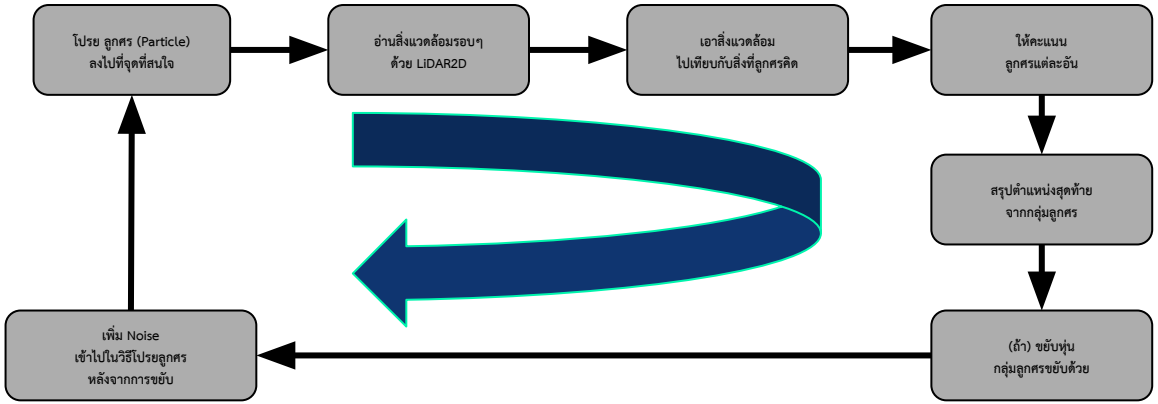
AMCL : Filter Parameter

- Update_min_d
  ถ้าหุ่นเคลื่อนที่ ได้ระยะ d
  จะเริ่มนับ 1 step

- Update_min_a
  ถ้าหุ่นหมุน องศา a
  จะเริ่มนับ 1 step

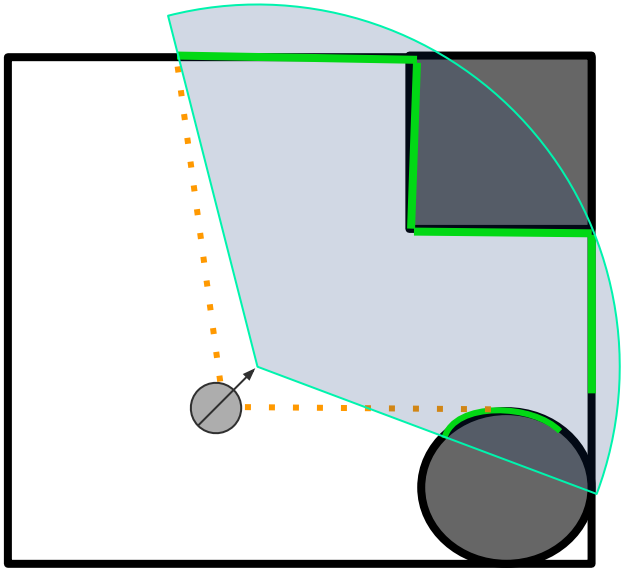
- Resample_interval
  กี่ STEP ถึงจะเริ่มกระบวนการ

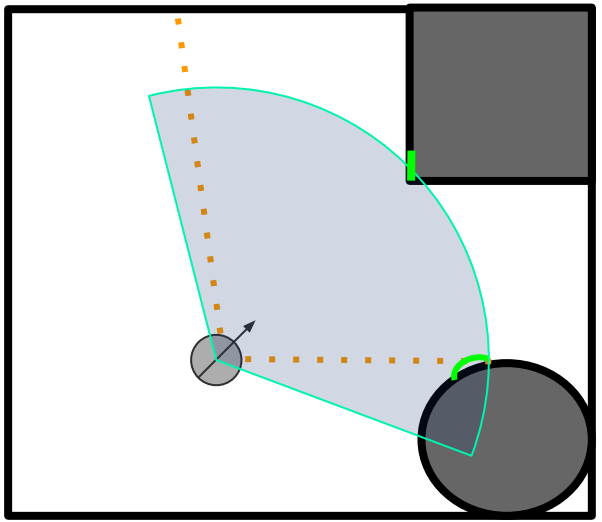
- transform_tolerance
  ให้คำตอบของ AMCL เผื่อเวลาไป
  นานขนาดไหน ปกติค่านี้จะเปลี่ยน
  ตามพลังประมวลผล



AMCL : Laser Model Parameter

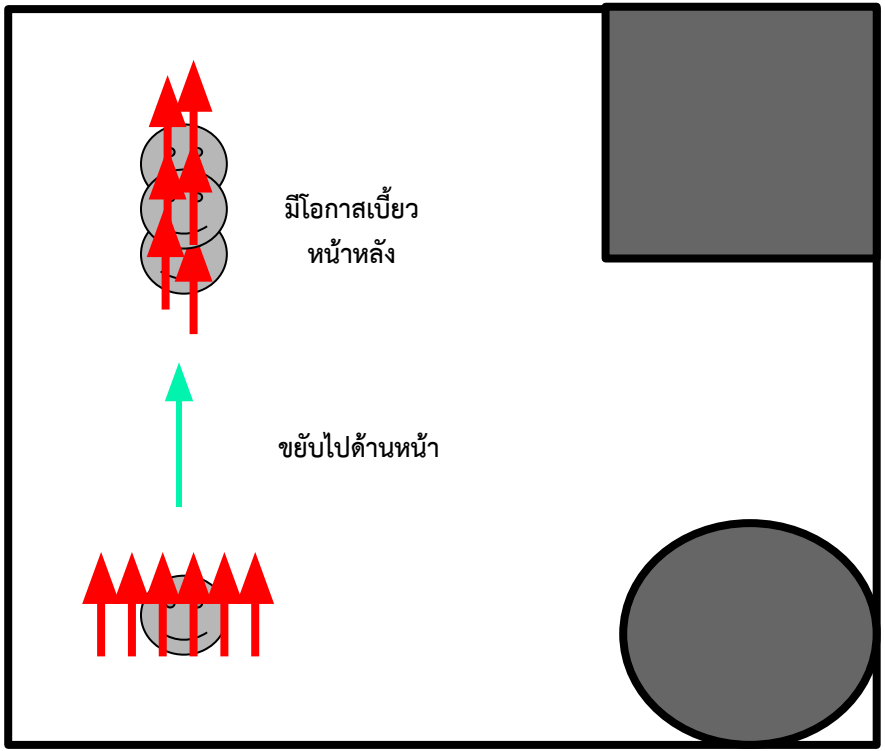
- Laser_min_range ระยะเลเซอร์ที่จะใช้ ใกล้สุดเท่าไร
- laser_max_range ระยะเลเซอร์ที่ใช้ ไกลสุดเท่าไร
- laser_max_beams





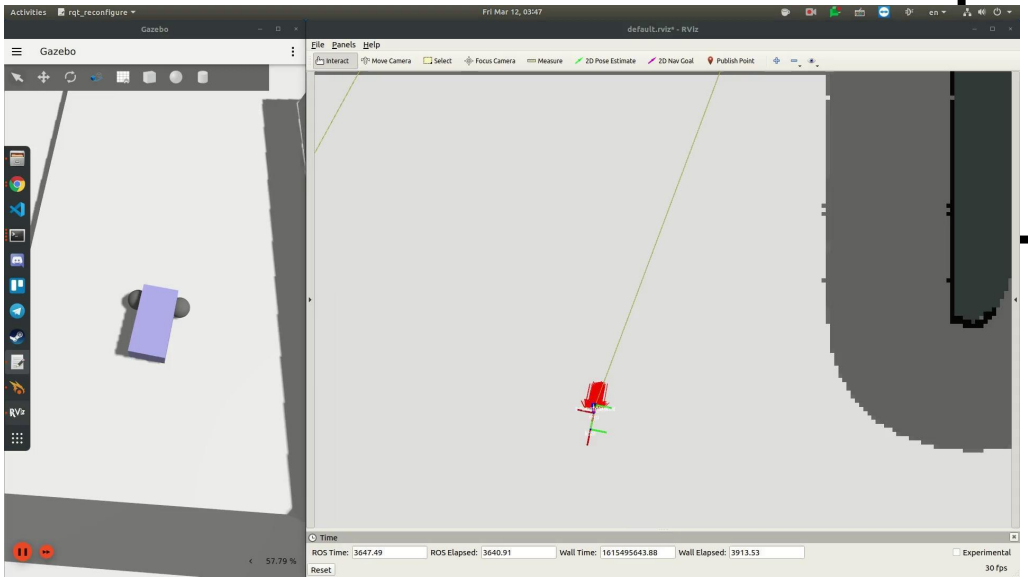
AMCL : Odometry Model Parameter

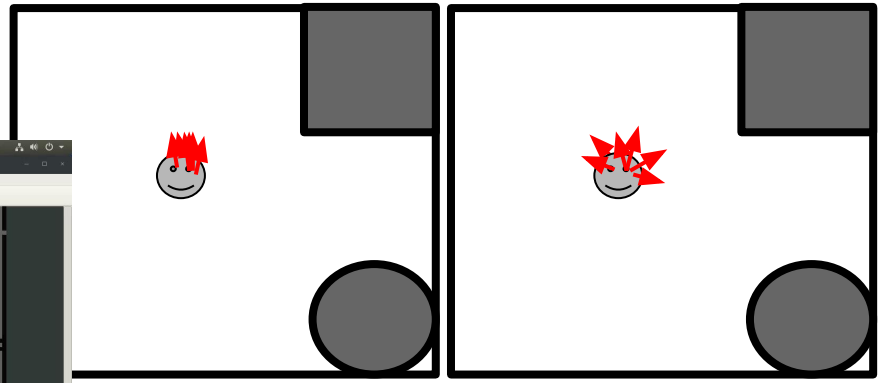
- Odom_model_type
  ใช้เป็น diff-drive(หุ่น2ล้อ) ไปก่อน
- odom_alpha1
- odom_alpha2
- odom_alpha3
- odom_alpha4
- odom_frame_id : “odom”
- base_frame_id: “base_footprint”
- global_frame_id: “map”



AMCL : Odometry Model Parameter

- odom_alpha1

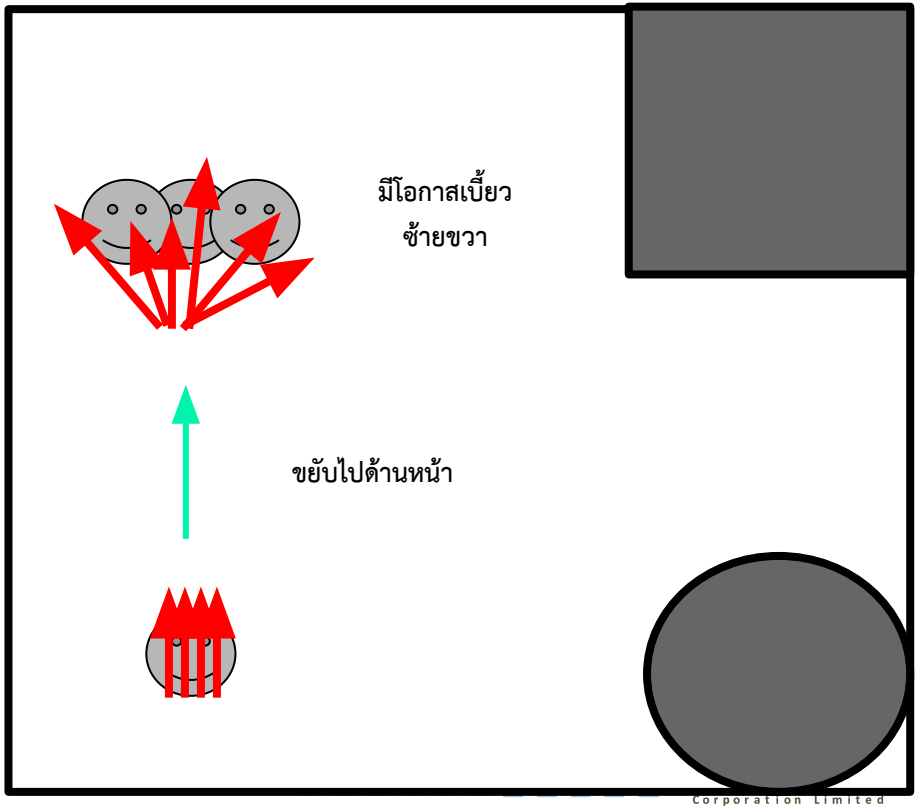




หมุนอยู่กับที่
มีโอกาสหมุนขาด-เกิน

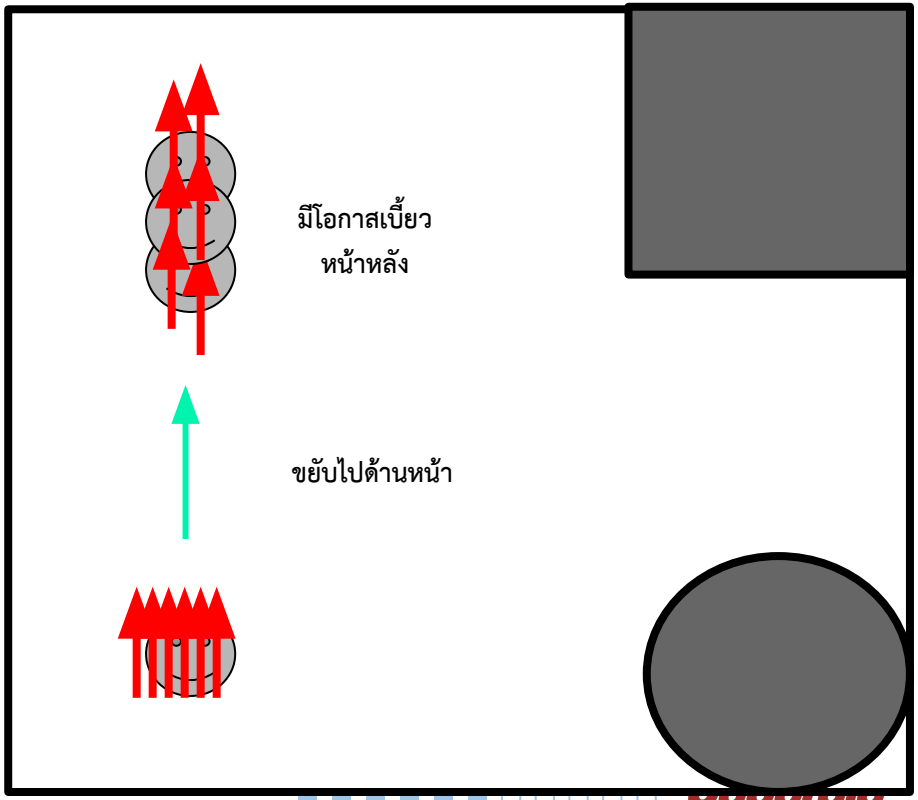
AMCL : Odometry Model Parameter

- odom_alpha2



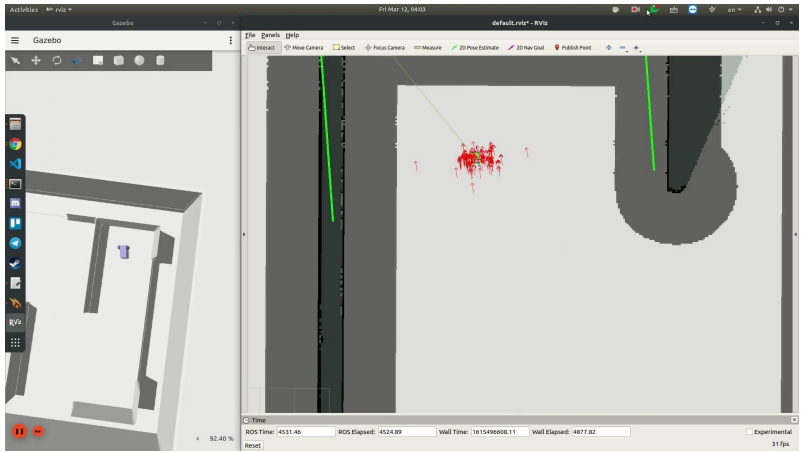
AMCL : Odometry Model Parameter

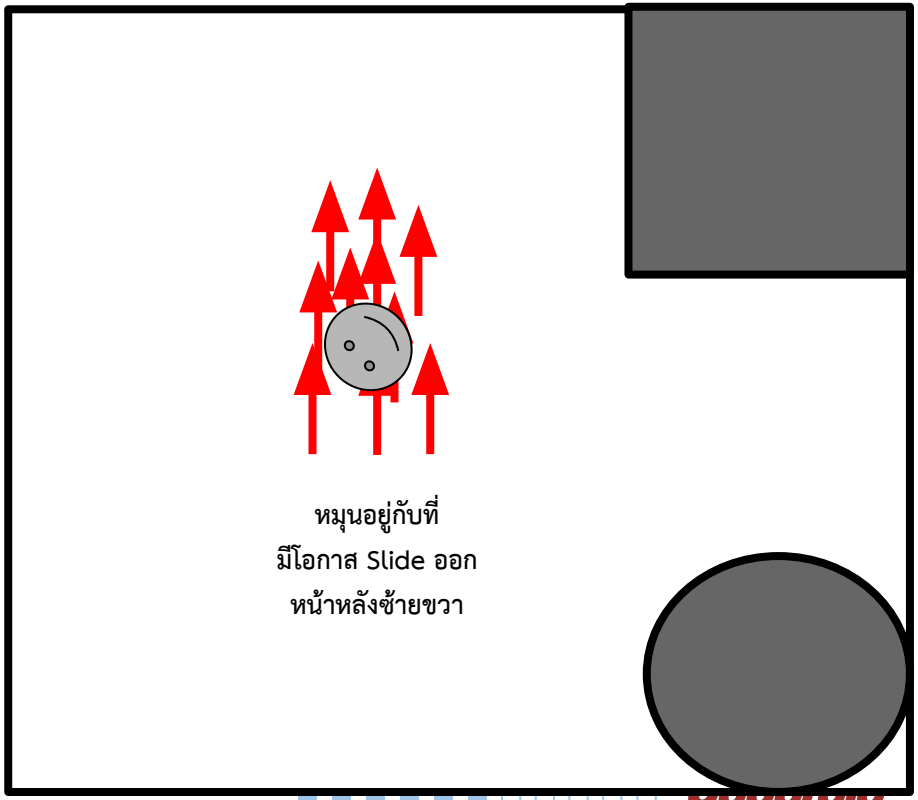
- odom_alpha3



AMCL : Odometry Model Parameter

- odom_alpha4





AMCL Tuning

- ปรับค่าแบบ Dynamic ได้ ด้วยการเปิด

> rosrun rqt_reconfigure rqt_reconfigure

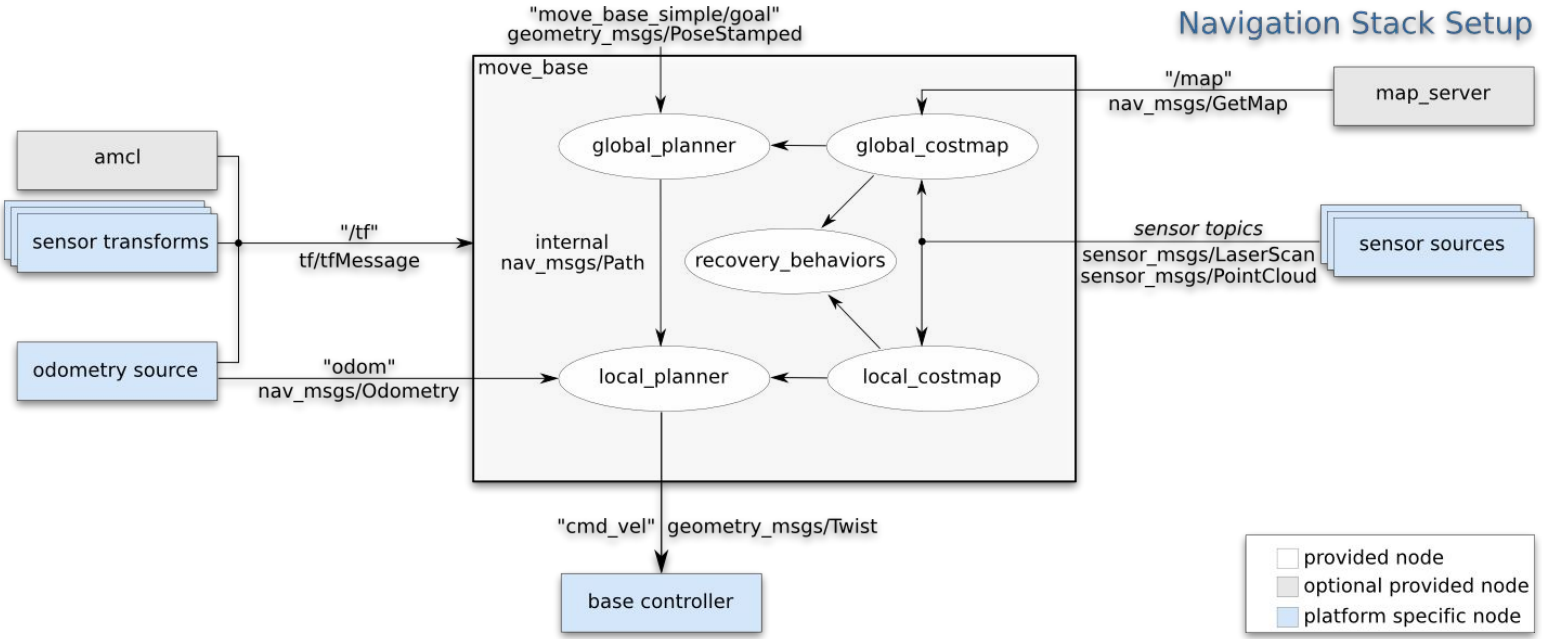
- เลือก Module จากด้านข้าง
  - ในกรณีนี้ เลือก AMCL เพื่อเปิดขึ้นมาดู ว่ามี Config ตัวไหนที่ปรับ ระหว่างทำงานได้

AMCL Tuning

ให้น้องๆ ลองเล่น AMCL ซัก 30-45 นาที

move_base

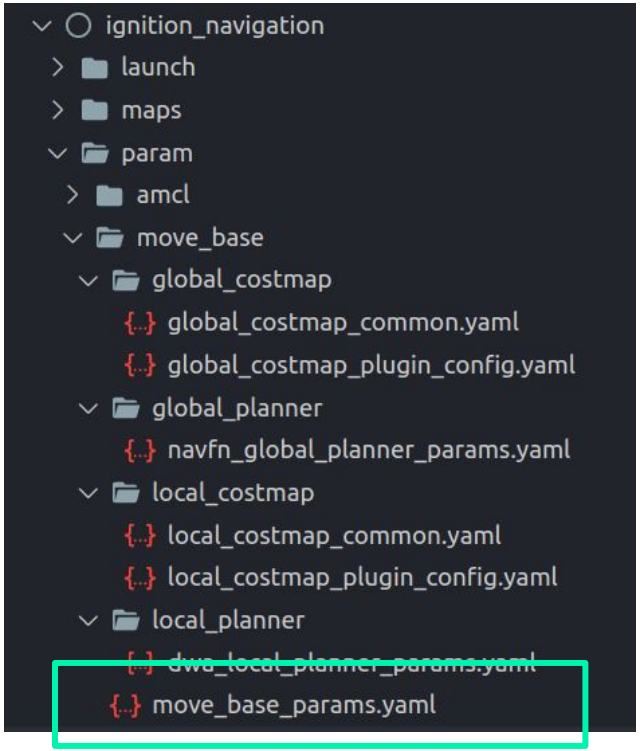
Move Base


Navigation Stack Setup
"move_base_simple/goal"
geometry_msgs/PoseStamped
move_base
global_planner
global_costmap
"/map"
nav_msgs/GetMap
map_server
amcl
sensor transforms
"/tf"
tf/tfMessage
internal
nav_msgs/Path
recovery_behaviors
sensor topics
sensor_msgs/LaserScan
sensor_msgs/PointCloud
sensor sources
odometry source
"odom"
nav_msgs/Odometry
local_planner
local_costmap
"cmd_vel"
geometry_msgs/Twist
base controller
provided node
optional provided node
platform specific node

Move Base : Parameters

Move_base_param

เก็บเกี่ยวกับ General Config ของ move_base

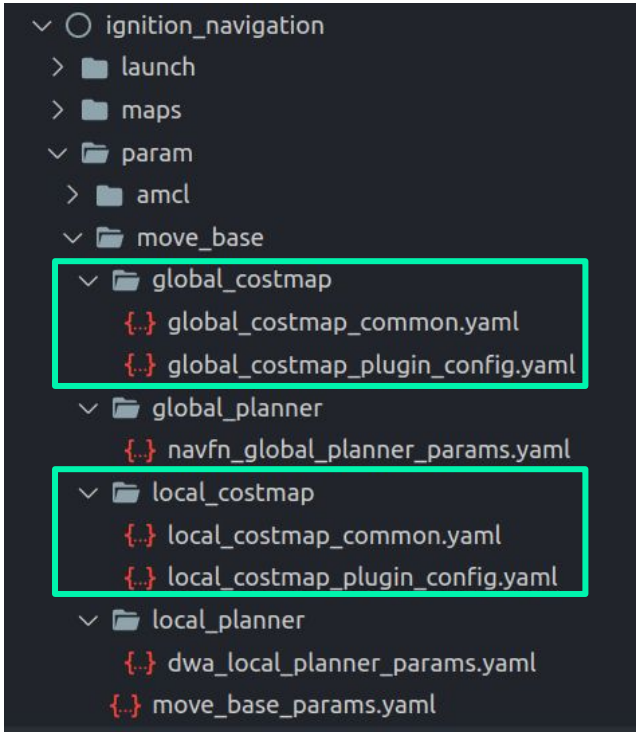


Move Base : Parameters

- **Costmap 2D Parameter**
  เราได้วาง template ไว้เป็น 2 Files

  1. **Common Parameter**
  2. **Plugin config**

- Global Costmap Parameter
- Local Costmap Parameter



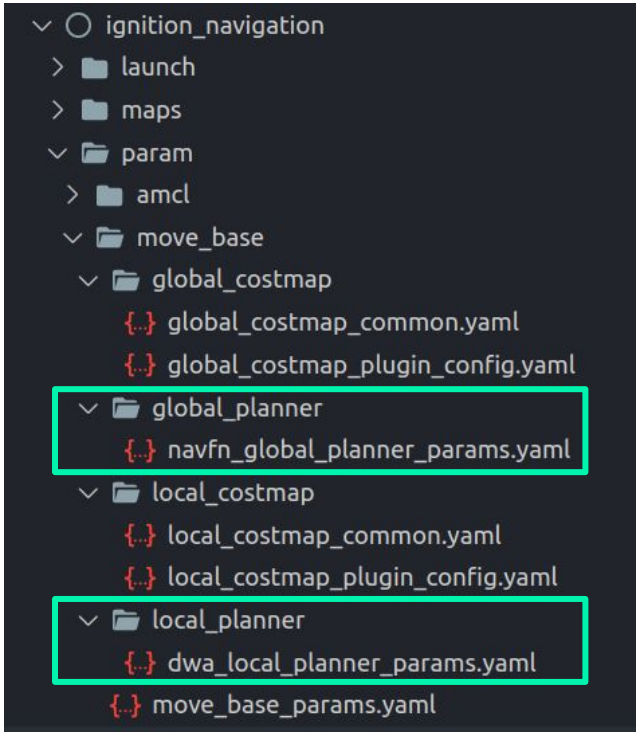
Move Base : Parameters

- Global Planner Parameter

  เกี่ยวกับการเลือกเส้นทางและจุดหมาย

- Local Planner Parameter

  เกี่ยวกับการคุมลักษณะการเดินตามเส้นทาง



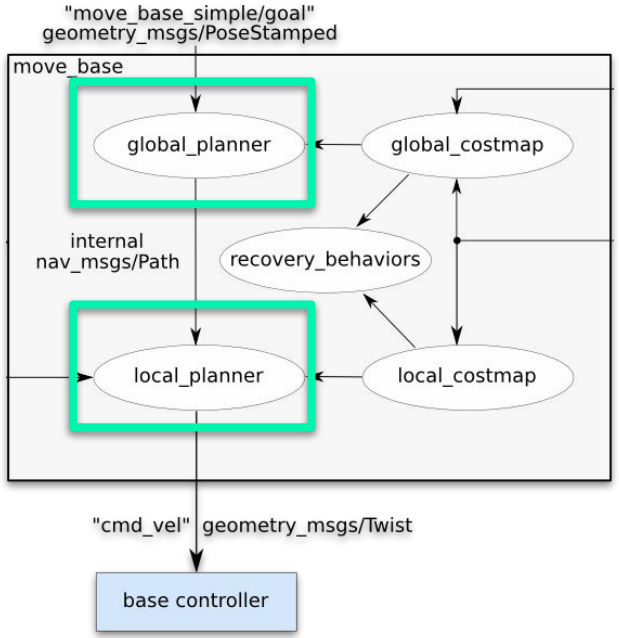
Move Base General Parameters

เริ่มจากเลือกประเภทของ PLANNER
ทั้ง Global และ Local Planner

- base_local_planner: "dwa_local_planner/DWAPlannerROS"
- base_global_planner: "navfn/NavfnROS"

เลือกประเภทของ Recovery Behavior

- recovery_behaviors : default



Move Base General Parameters

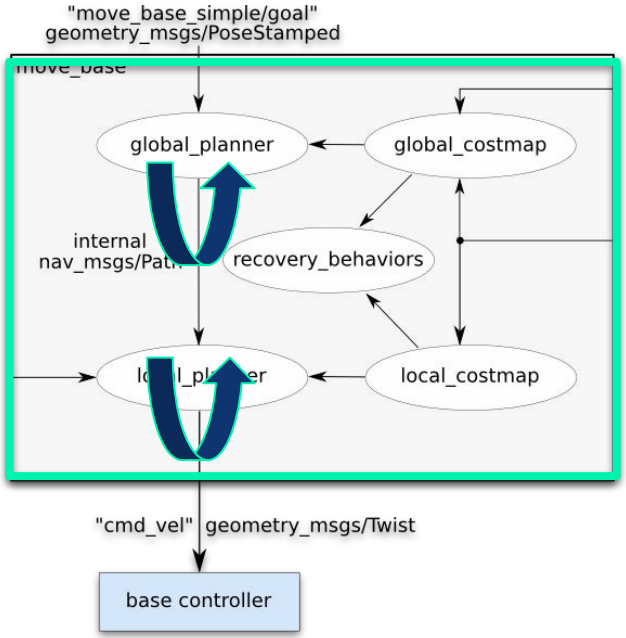
Planning

- controller_frequency
  จะให้ Local Planner ทำงานที่ความเร็วเท่าไร

- controller_patience
  ให้ Local Planner อดทนนานขนาดไหน
  ก่อนที่จะบอกว่า ~ ไม่ไหวแล้ว ~ ยอมแพ้
  ขอ Global Path ใหม่

- planner_frequency
  จะให้ Global Planner ทำงานที่ความเร็วเท่าไร

- Planner_patience
  ให้ Global Planner อดทนนานขนาดไหน
  ก่อนที่จะบอกว่า ~ ไม่ไหวแล้ว ~ ยอมแพ้
  Mission aborted



Move Base General Parameters

Recovery Behaviors

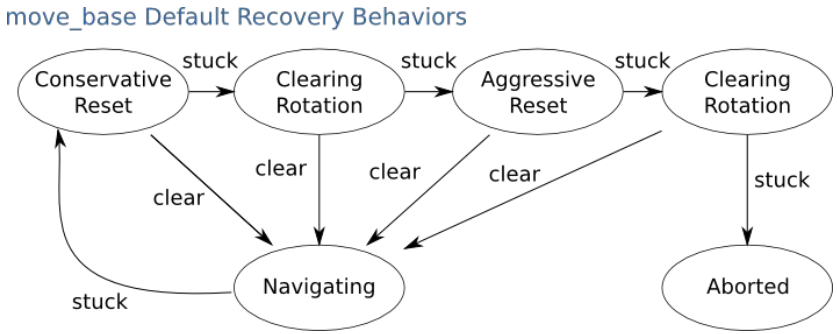
- recovery_behavior_enabled

  จะเปิดใช้งาน recovery behavior

- conservative_reset_dist

  เคลียร์ costmap รอบตัวหุ่นยนต์ไปห่างมากมั้ยยย

- clearing_rotation_allowed



ทดลองเล่น move_base general parameter

ทดลองปรับ planner_frequencyและ Controller

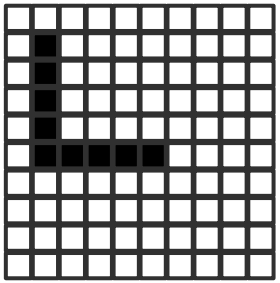
ดู และสังเกตผลลัพธ์

ว่าหุ่นเตอบสนองต่อเส้นทางยังไง

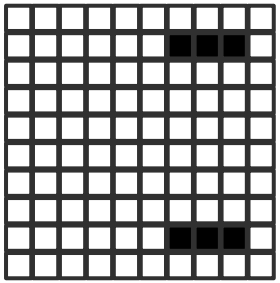
costmap

Costmap 2D : Parameters

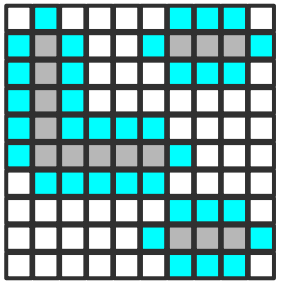
STATIC LAYER → OBSTACLE LAYER → INFLATION LAYER → MASTER GRID



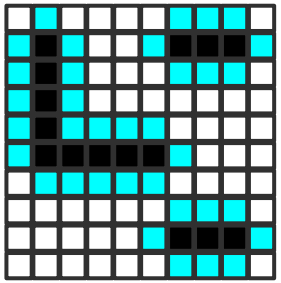
เป็นแผนที่เปล่าๆ ที่ได้จาก map_server



หุ่นยนต์ใช้ Sensor Detect และ วาด Obstacle



สร้างระยะระวัง จาก ช่องสีดำที่มีในแผนที่



ผลลัพธ์สุดท้าย ก่อนเอาไปใช้งาน

Costmap 2D : Common Parameters

- global_frame
- robot_base_frame
- update_frequency
- publish_frequency
- footprint
- robot_radius
- footprint_padding
- static_map
- rolling_window -> Global/Local Costmap

Costmap 2D: Layer

Setting ของระบบ Costmap Layer จะทำงานแบบนี้

1. เราเลือกว่าจะเรียงลำดับ Layer ยังไงก่อน
   - ignition_navigation
     file : xxxx_costmap_common.yaml
     - หัวข้อ plugins
       - ใส่ว่าจะมี layer อะไรบ้าง และเป็นชนิดอะไร
       - เรียงจากบนลงล่าง

2. เรากำหนดค่าประจำแต่ละ Layer ว่าจะมีลักษณะเป็นยังไงบ้าง
   - Ignition_navigation
     File : xxxx_costmap_plugins_config.yaml
       - แยกรายละเอียด ไปแต่ละชั้นย่อยๆ

Costmap 2D : Common Parameters

global_costmap_common.yaml

user, a day ago | 2 authors (Theppasith N and others)

```yaml
global_costmap:
  # Costmap Type (costmap, voxel)
  map_type: costmap

  # Frame Setup
  global_frame: map
  robot_base_frame: base_footprint

  # Update Rate
  update_frequency: 3.0
  publish_frequency: 1.0
  transform_tolerance: 0.5

  # Global/Local Costmap
  resolution: 0.10 # m/px
  static_map: true
  rolling_window: false
  # width: 8  # m
  # height: 8 # m

  # C-Space Setting
  footprint: [[1.00571, 0.5], [1.00571, -0.5], [-1.00571, -0.5], [-1.00571, 0.5]]
# robot_radius: 1.5
  footprint_padding: 0.05

  # Costmap Layering (Top to Bottom)
  plugins:
    - {name: static_layer,          type: "costmap_2d::StaticLayer"}
    - {name: obstacle_layer,        type: "costmap_2d::VoxelLayer"}
    - {name: inflation_layer,       type: "costmap_2d::InflationLayer"}

```

local_costmap_common.yaml

You, seconds ago | 3 authors (Theppasith N and others)

```yaml
local_costmap:
  # Costmap Type (costmap, voxel)
  map_type: costmap

  # Frame Setup
  global_frame: odom
  robot_base_frame: base_footprint

  # Update Rate
  update_frequency: 10.0
  publish_frequency: 3.0
  transform_tolerance: 0.5

  # Global/Local Costmap
  resolution: 0.05  # m/px
  static_map: false
  rolling_window: true
  width: 8  # m
  height: 8 # m

  # C-Space Setting
  footprint: [[1.00571, 0.5], [1.00571, -0.5], [-1.00571, -0.5], [-1.00571, 0.5]]
# robot_radius: 1.5
  footprint_padding: 0.05

  # Costmap Layering (Top to Bottom)
  plugins:
    - {name: static_layer,          type: "costmap_2d::StaticLayer"}
    - {name: obstacle_layer,        type: "costmap_2d::VoxelLayer"}
    - {name: inflation_layer,       type: "costmap_2d::InflationLayer"}

```

global_costmap_common.yaml

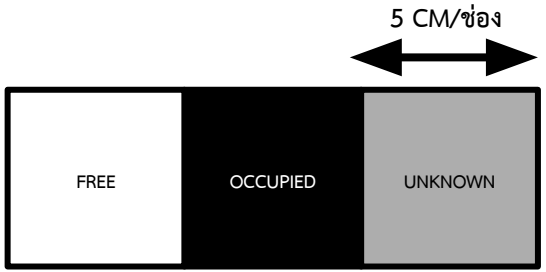
```yaml
global_costmap:
  # Costmap Type (costmap, voxel)
  map_type: costmap

  # Frame Setup
  global_frame: map
  robot_base_frame: base_footprint

  # Update Rate
  update_frequency: 3.0
  publish_frequency: 1.0
  transform_tolerance: 0.5

  # Global/Local Costmap
  resolution: 0.10 # m/px
  static_map: true
  rolling_window: false
  # width: 8  # m
  # height: 8 # m

  # C-Space Setting
  footprint: [[1.00571, 0.5], [1.00571, -0.5], [-1.00571, -0.5], [-1.00571, 0
# robot_radius: 1.5
  footprint_padding: 0.05

  # Costmap Layering (Top to Bottom)
  plugins:
    - {name: static_layer,          type: "costmap_2d::StaticLayer"}
    - {name: obstacle_layer,        type: "costmap_2d::VoxelLayer"}
    - {name: inflation_layer,       type: "costmap_2d::InflationLayer"}
```

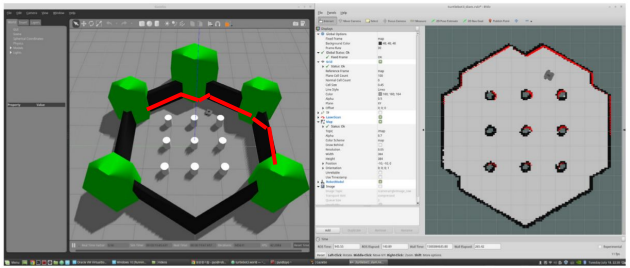
global_costmap_plugin_config.yaml

```yaml
global_costmap:

  # Layer 1 - Static
  static_layer:
    enabled:              true

  # Layer 2 - Obstacle
  obstacle_layer:
    enabled:              true
    origin_z:             0.0
    max_obstacle_height:  1.5    # Max Obstacle
    z_resolution:         0.3    # Resolution
    z_voxels:             5      # max_obstacle_height = z_resolution * z_voxel
    unknown_threshold:    15
    mark_threshold:       0
    combination_method:   1
    track_unknown_space:  false # True needed for disabling global path planning
    obstacle_range:       2.0       Theppasith N, a day ago • New Parameter Struc
    raytrace_range:       2.1
    publish_voxel_map:    false
    observation_sources:  lidar_scan

    # Observation Sources detail here
    lidar_scan:
      data_type:          LaserScan # From Sensor msgs
      sensor_frame:       lidar
      topic:              scan
      marking:            true
      clearing:           true
      min_obstacle_height: 0.0
      max_obstacle_height: 2.0
      inf_is_valid:       true
      observation_persistence: 0.0  #Secs

  # Layer 3 - Inflation
  inflation_layer:
    enabled:              true
    cost_scaling_factor:  3.0 # exponential rate at which the obstacle cost drop
    inflation_radius:     1.5 # max. distance from an obstacle at which costs ar
```

Static Map Layer Parameters

- map_topic
- unknown_cost_value
- first_map_only
- track_unknown_space





Real World Occupancy Grid

Obstacle Map Layer Parameters

Sensor

- observation_sources
  - sensor_frame
  - observation_persistence
  - expected_update_rate
  - data_type
  - clearing
  - marking
- observation_sources
  - max_obstacle_height
  - min_obstacle_height
  - inf_is_valid

Obstacle Map Layer Parameters

Global Filtering

- max_obstacle_height
- **obstacle_range**
- **raytrace_range**

Obstacle Map Layer Parameters

Obstacle Costmap

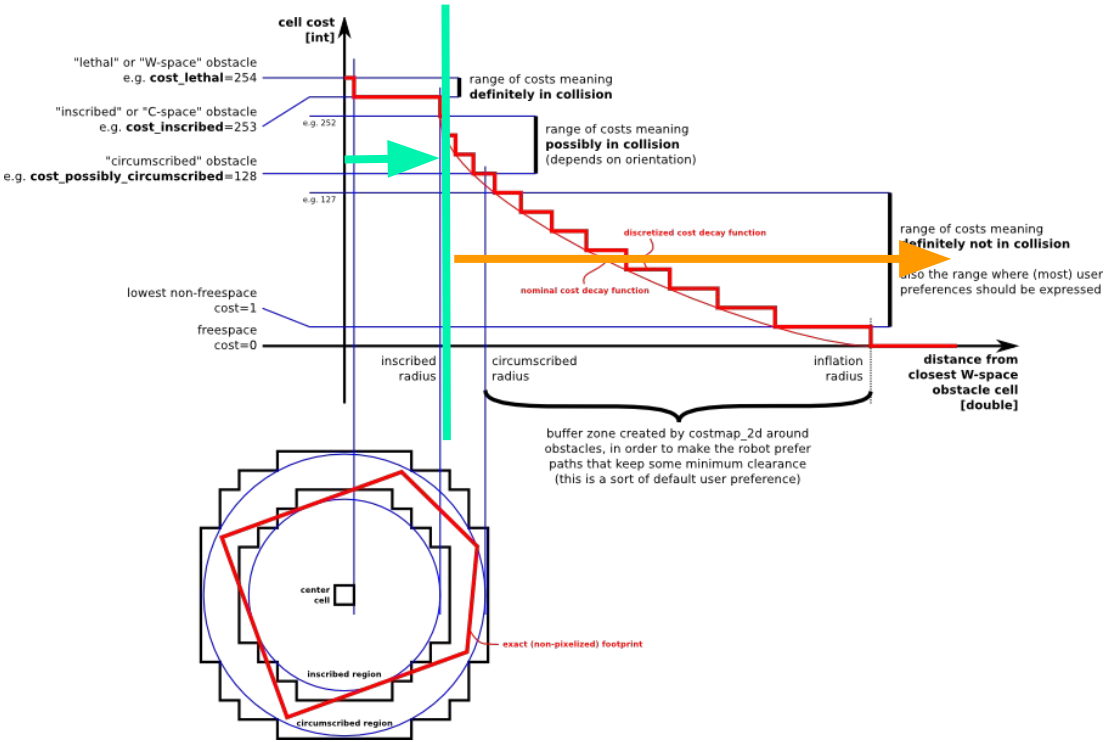
- track_unknown_space - มอง Unknown Space ให้เป็น Obstacle
- footprint_clearing_enabled - ลบของที่อยู่ในตัวเองออก ถ้ามองเห็น

Inflation Layer Parameters

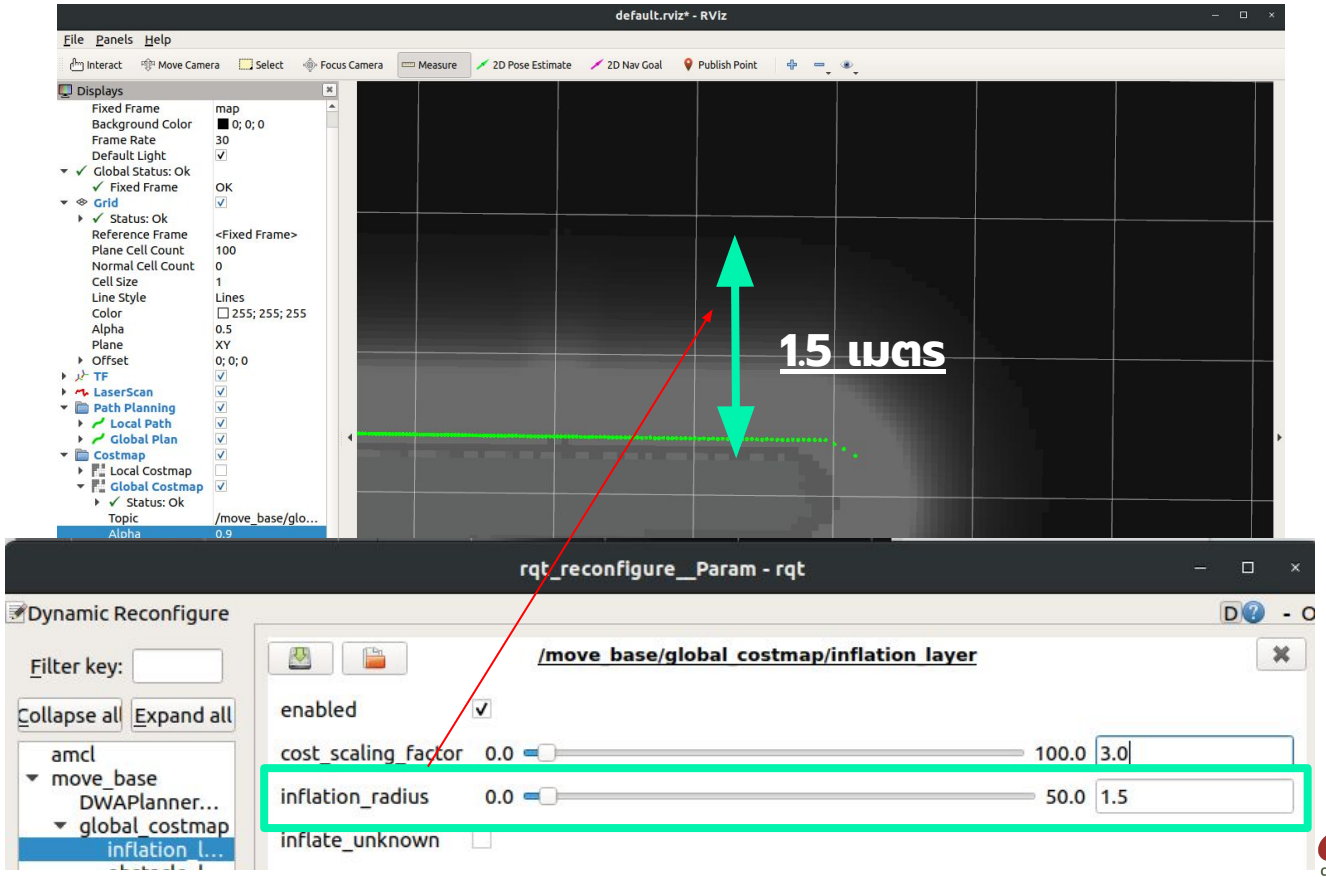
- Inflation_radius
  ระยะระวังของหุ่นกับ Obstacle

  ยิ่งเยอะ หุ่นจะพยายามห่าง

- cost_scaling_factor
  default: 3
  ปรับลักษณะโค้งของการ inflate ให้ โค้งลงไวหรือช้า



Global Costmap Parameters

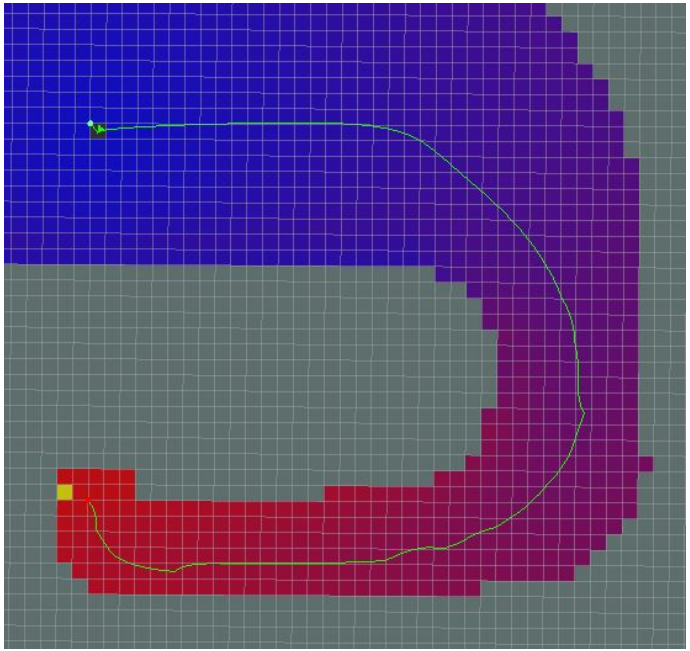


obodroid Corporation Limited

Global and Local Costmap Parameters

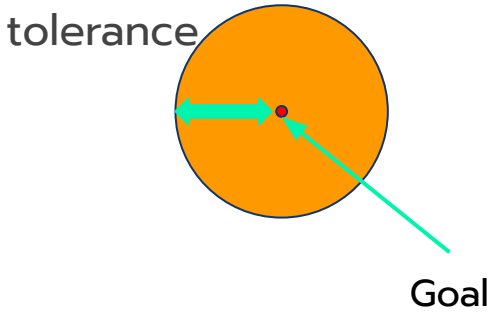
มาลองเล่น Costmap กัน

Global Planner Parameters



Global Planner Parameters

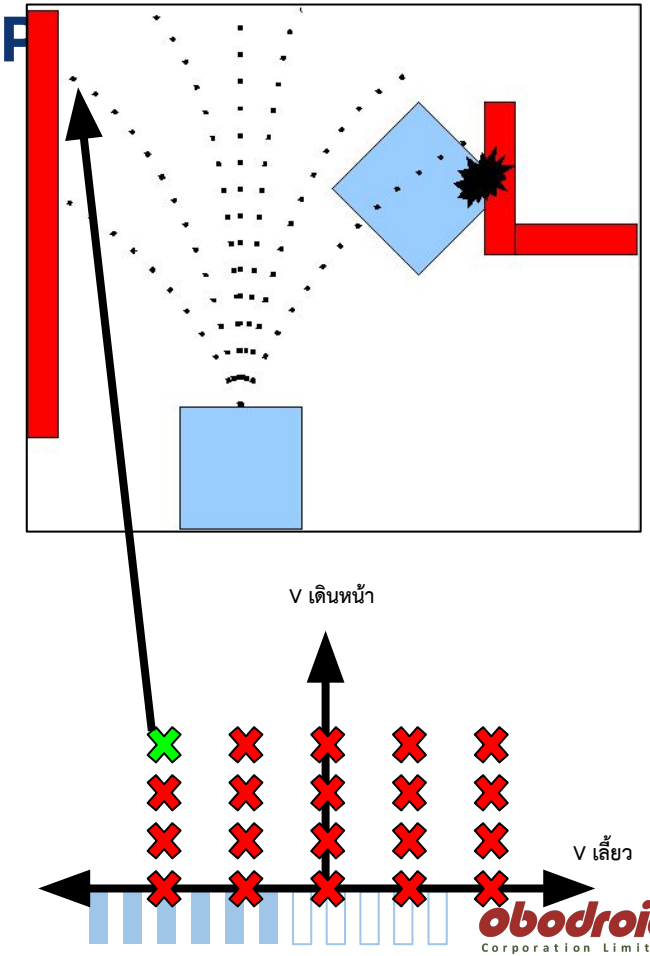
- allow_unknown
  - อนุญาตให้ Plan เข้าไปในส่วนที่เป็น Unknown Space ได้
- default_tolerance



DWA Local Planner P

Robot Configuration

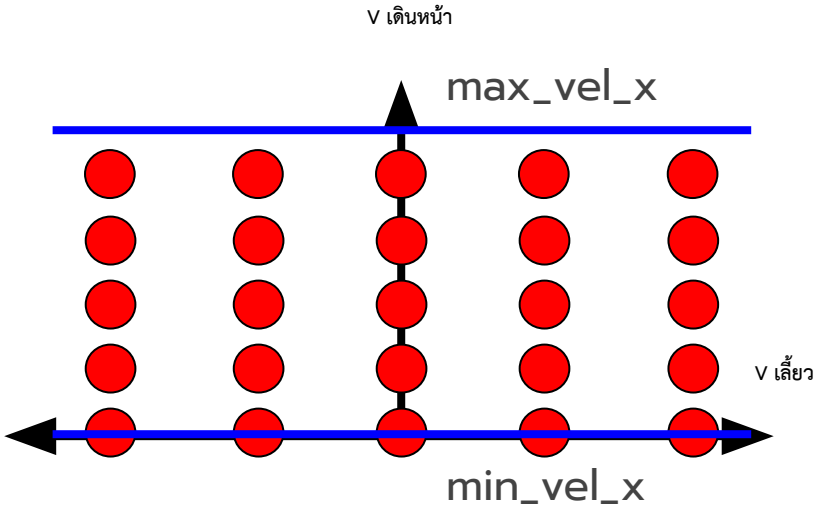
- max_vel_x
- min_vel_x
- acc_lim_x
- acc_lim_th
- max_vel_trans, min_vel_trans
  - ถ้าหุ่นเคลื่อนที่ทางอื่นได้ ตัวนี้จะคำนวณ Vx และ Vy
  - ใน Model หุ่นของเราในงานนี้ ให้ใส่เหมือน max_vel_x, min_vel_x
- max_rot_vel
- min_rot_vel



DWA Local Planner Parameters

Robot Configuration

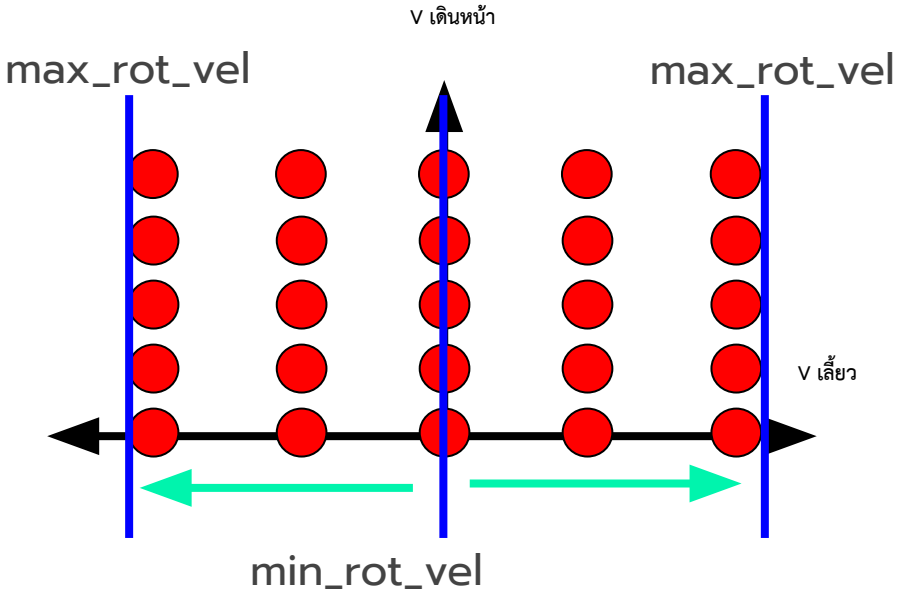
- max_vel_x
- min_vel_x
- acc_lim_x
- acc_lim_th
- max_vel_trans, min_vel_trans
  - ถ้าหุ่นเคลื่อนที่ทางอื่นได้ ตัวนี้จะคำนวณ Vx และ Vy
  - ใน Model หุ่นของเราในงานนี้ ให้ใส่เหมือน max_vel_x, min_vel_x
- max_rot_vel
- min_rot_vel



DWA Local Planner Parameters

Robot Configuration

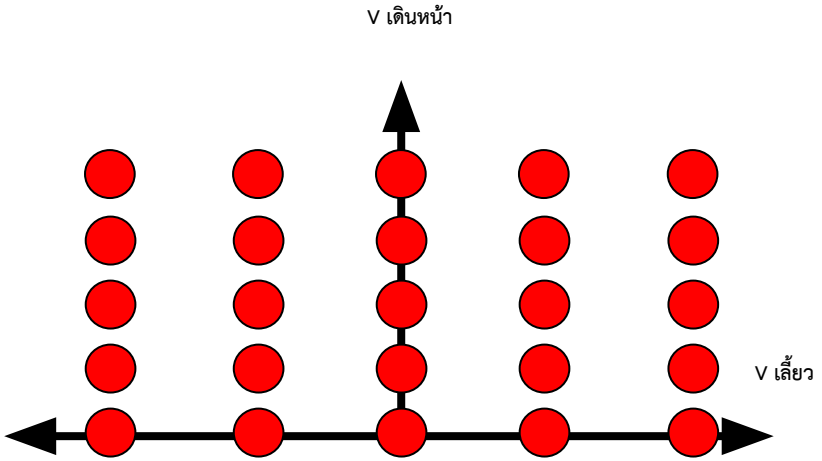
- max_vel_x
- min_vel_x
- acc_lim_x
- acc_lim_th
- max_vel_trans, min_vel_trans
  - ถ้าหุ่นเคลื่อนที่ทางอื่นได้ ตัวนี้จะคำนวณ Vx และ Vy
  - ใน Model หุ่นของเราในงานนี้ ให้ใส่เหมือน max_vel_x, min_vel_x
- max_rot_vel
- min_rot_vel



DWA Local Planner Parameters

Robot Configuration

- max_vel_x
- min_vel_x
- acc_lim_x
- acc_lim_th
- max_vel_trans, min_vel_trans
  - ถ้าหุ่นเคลื่อนที่ทางอื่นได้ ตัวนี้จะคำนวณ Vx และ Vy
  - ใน Model หุ่นของเราในงานนี้ ให้ใส่เหมือน max_vel_x, min_vel_x
- max_rot_vel
- min_rot_vel



DWA Local Planner Parameters

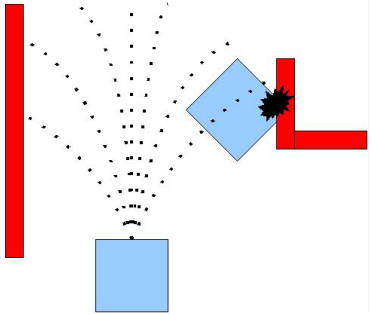
Goal Tolerance

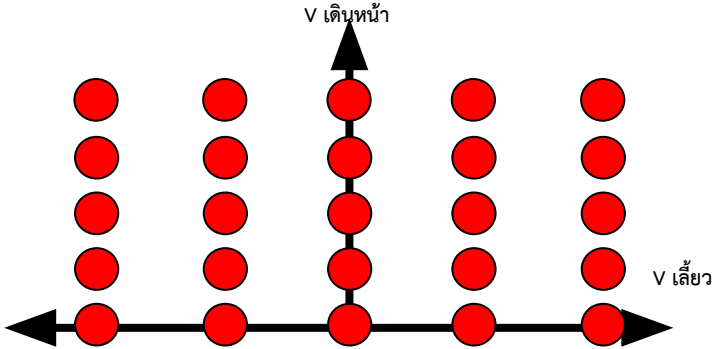
- yaw_goal_tolerance
  - ให้หุ่นยนต์เข้า goal หันทิศไม่ตรงที่ตั้งไว้แค่ไหน
- xy_goal_tolerance
  - ให้หุ่นยนต์เข้า goal ไม่ตรงจุดได้ไกลแค่ไหน

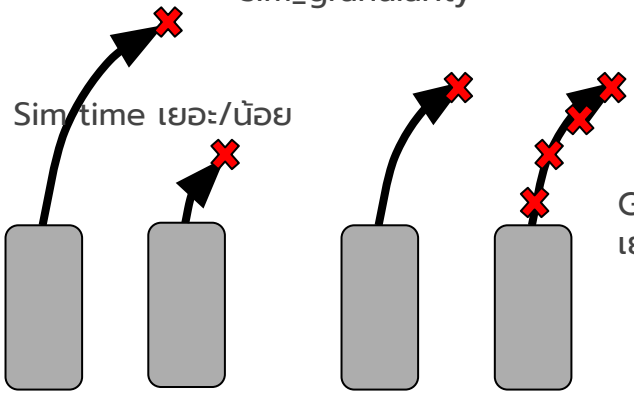
DWA Local Planner Parameters

Forward Simulation

- controller_frequency
- vx_samples
- vth_samples
- sim_time
- sim_granularity



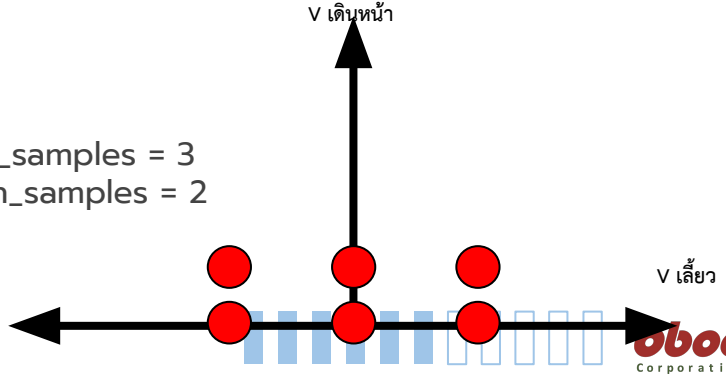




Sim time เยอะ/น้อย

Granularity
เยอะ x ออกมา 4 อัน

vx_samples = 3
vth_samples = 2

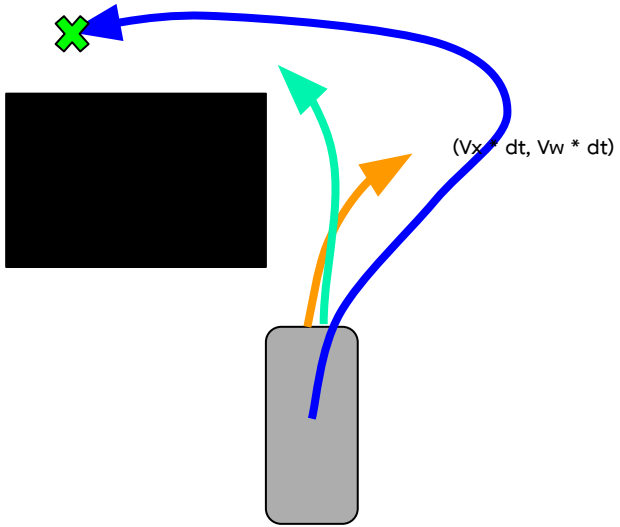


obodroid
Corporation Limited

DWA Local Planner Parameters

Trajectory Scoring

- Path_distance_bias
  เพิ่มความพยายามพยายามตาม global path
- goal_distance_bias
  เพิ่มความพยายามสร้างทางที่เข้าใกล้ Goal
- occdist_scale
  เพิ่มความพยายามออกห่างจาก Obstacle



DWA Local Planner Parameters

Oscillation Prevention

- Oscillation_reset_dist
  ถ้าหุ่นได้ความเร็วออกมา แต่ทำให้ระยะทางไม่ขยับ เกินระยะทางเล็กๆนี้
  ซักพักจะถือว่า Local Plan ไม่สำเร็จ

Bag file

- Bag file เป็นรูปแบบไฟล์ของ ROS ที่ใช้เก็บ ROS message
- เอาไว้ใช้เก็บข้อมูลต่างๆระหว่างที่ ROS ทำงาน
- เราสามารถนำข้อมูลนั้นมาเล่นซ้ำ แล้วแสดงผลดูสิ่งที่เกิดขึ้น กี่รอบก็ได้

Basic Bags Command: Record

- บันทึก ROS message ทั้งหมด

  > rosbag record -a

- บันทึก ROS message ตาม topic ที่เลือก

  > rosbag record /topic_name

- บันทึก ROS message ตาม topic ที่เลือกและตั้งชื่อไฟล์

  > rosbag record -O file_name /topic_name

Basic Bags Command: Play

- เล่น bag file ที่เลือก

  > rosbag play file_name

- เล่น bag file ที่เลือกโดยเริ่มตามเวลาที่กำหนด

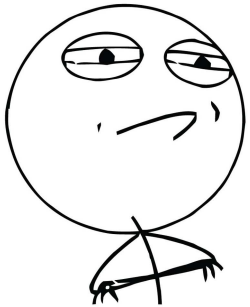
  > rosbag play file_name -s SEC

- เล่น bag file ที่เลือกด้วยความเร็วที่กำหนด

  > rosbag play file_name -r Rate

CHALLENGE ประจำวันน ~

ทีมไหนเดินได้ไวสุดชนะ !!!!
มีของแจกด้วยนะอิ



CHALLENGE ACCEPTED

DAY2 - Challenge “The Flash”

เรามีด่านให้ทุกคนสามารถ RUN Navigation Stack ได้

ภายใน 1 ชั่วโมงนี้

ทีมไหนสามารถเดินไปถึงจุดเป้าหมายได้ไวที่สุด จะได้รับรางวัล. . .บางอย่าง

สามารถชนสิ่งกีดขวางยังไงก็ได้

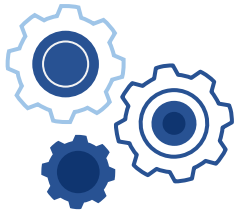
แต่ไหนๆก็จะชนะแล้ว คิดว่า ไม่ชน ก็เก๋าดีใช่ไหมล่ะ !!

ทีมไหน Setup เสร็จแล้ว จะให้ TA เข้าไปจับเวลา และ TA จะเอาเวลามา Update ที่ Scoreboard ด้านหน้านี้

DAY2 - Challenge “The Flash”

THIS IS THE SCOREBOARD

TEAM	TIME
TA สุดหล่อ	10 วินาที (จอกๆปะวะ)



แล้วกลับมาพบกับ Simulation

ต่อใน WEEK 2

:)