

6.gmapping mapping algorithm

6.gmapping mapping algorithm

6.1 Introduction

6.2 Use

6.2.1 Start

6.2.2 Control the robot

6.2.3 Map save

6.2 Topics and Services

6.3 configuration parameters

6.4 TF transformation

Gmapping: <http://wiki.ros.org/gmapping/>

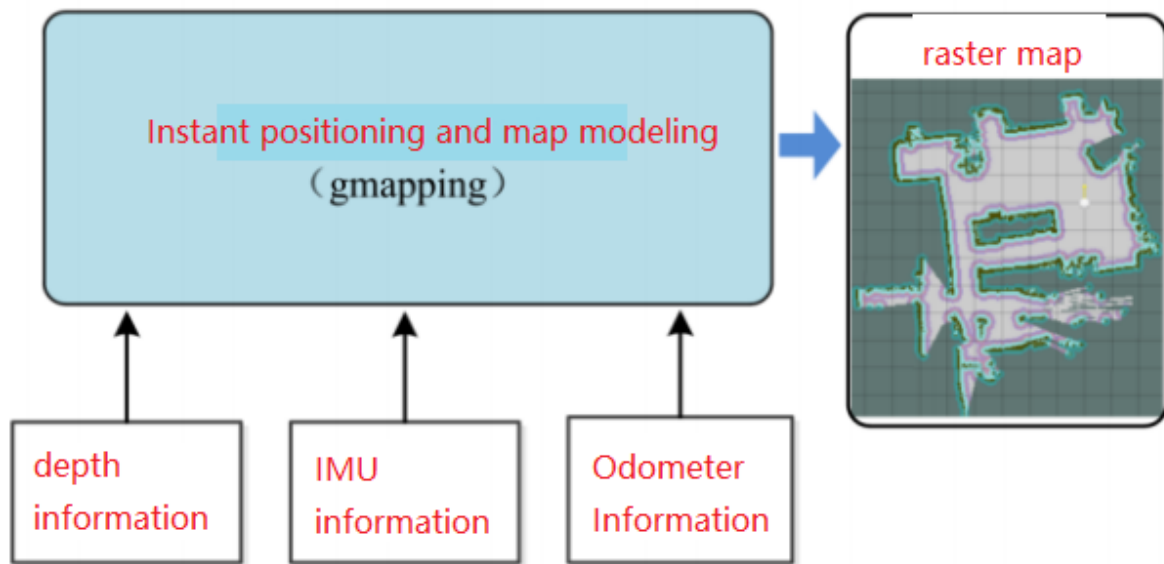
map_server: https://wiki.ros.org/map_server

6.1 Introduction

- gmapping is only applicable to points with less than 1440 2D laser points in a single frame. If the number of laser points in a single frame is greater than 1440, there will be a problem like [[mapping-4] process has died].
- Gmapping is a common open source SLAM algorithm based on the filtering SLAM framework.
- Gmapping is based on the RBpf particle filter algorithm, and the real-time positioning and mapping process are separated, and the positioning is carried out first and then the mapping is carried out.
- Gmapping makes two major improvements to the RBpf algorithm: improved proposal distribution and selective resampling.

Advantages: Gmapping can build indoor maps in real time, requiring less computation and higher accuracy in building small scene maps.

Disadvantage: As the scene grows, the number of particles required increases, because each particle carries a map, so the memory and computation required to build a large map will increase. Therefore, it is not suitable for building large scene maps. And there is no loopback detection, so the map may be dislocated when the loopback is closed. Although increasing the number of particles can make the map closed, it comes at the cost of increasing computation and memory.



6.2 Use

Note: When building a map, the slower the speed, the better the effect (note that if the rotation speed is slower), the effect will be poor if the speed is too fast.

According to different models, you only need to set the purchased model in [.bashrc], X1 (ordinary four-wheel drive) X3 (Mike wheel) X3plus (Mike wheel mechanical arm) R2 (Ackerman differential) and so on. Section takes X3 as an example

Open the [.bashrc] file

```
sudo vim .bashrc
```

Find the [ROBOT_TYPE] parameter and modify the corresponding model

```
export ROBOT_TYPE=X3    # ROBOT_TYPE: X1 X3 X3plus R2 X7
```

6.2.1 Start

Start command (robot side), for the convenience of operation, this section takes [mono + laser + yahboomcar] as an example.

```
roslaunch yahboomcar_nav laser_bringup.launch    # laser + yahboomcar
roslaunch yahboomcar_nav laser_usb_bringup.launch # mono + laser +
yahboomcar
roslaunch yahboomcar_nav laser_astapro_bringup.launch # Astra + laser +
yahboomcar
```

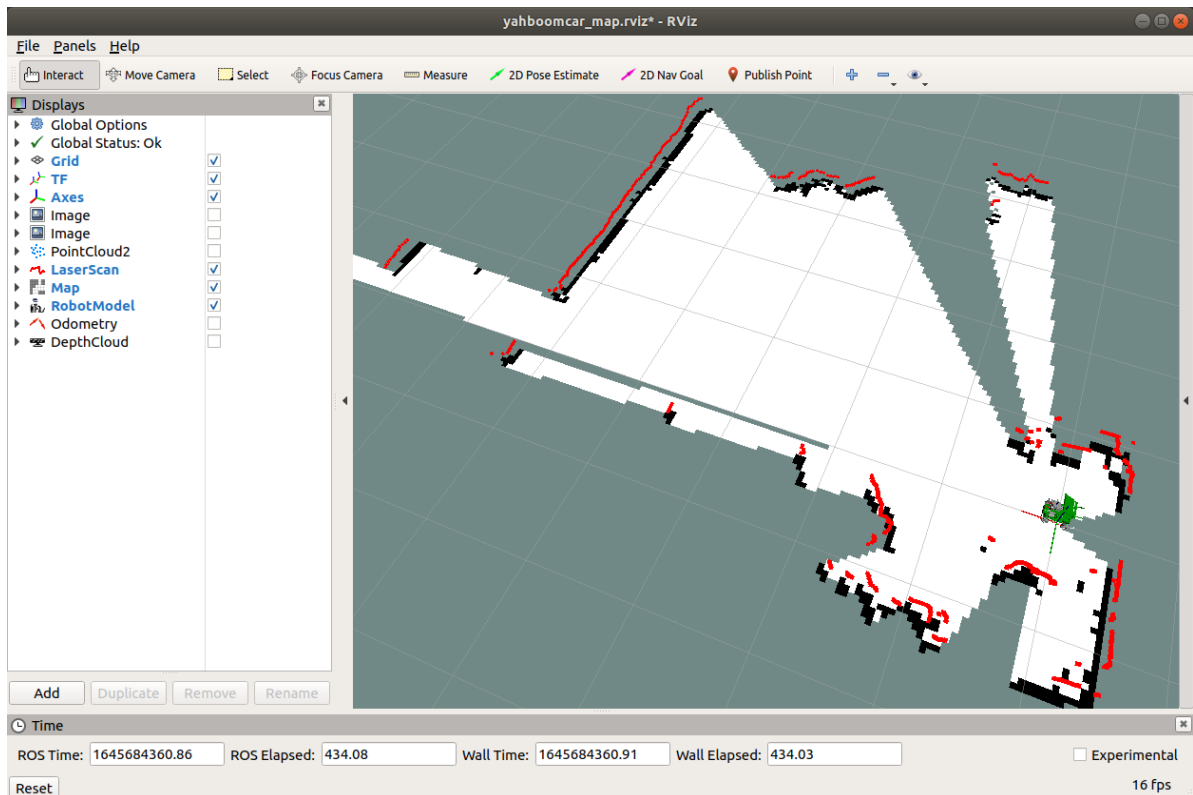
Mapping command (robot side)

```
roslaunch yahboomcar_nav yahboomcar_map.launch use_rviz:=false
map_type:=gmapping
```

- [use_rviz] parameter: whether to enable rviz visualization.
- [map_type] parameter: set the mapping algorithm [gmapping].

Open the visual interface (virtual machine side)

```
roslaunch yahboomcar_nav view_map.launch
```



The gap at the back of the robot is due to the occlusion of the installation position of the display screen, so a certain range of Lidar data is shielded. The shielding range can be adjusted or not shielded according to the actual situation. For details, see [01. Lidar Basic Course].

6.2.2 Control the robot

- The keyboard controls the movement of the robot

```
roslaunch yahboomcar_ctrl yahboom_keyboard.launch # system integration
roslaunch yahboomcar_ctrl yahboom_keyboard.launch # custom
```

- The handle controls the movement of the robot

Make the robot walk all over the area to be built, and the map is as closed as possible.

There may be some scattered points during the mapping process. If the mapping environment is well closed and regular, the movement is slow, and the scattering phenomenon is much smaller.

6.2.3 Map save

```
roslaunch map_server map_saver -f ~/yahboomcar_ws/src/yahboomcar_nav/maps/my_map
# The first way
bash ~/yahboomcar_ws/src/yahboomcar_nav/maps/map.sh
# second way
```

The map will be saved to ~/yahboomcar_ws/src/yahboomcar_nav/maps/ folder, a pgm image, a yaml file.

map.yaml

```

image : map.pgm
resolution : 0.05
origin : [-15.4,-12.2,0.0]
negate : 0
occupied_thresh : 0.65
free_thresh: 0.196

```

Parameter parsing:

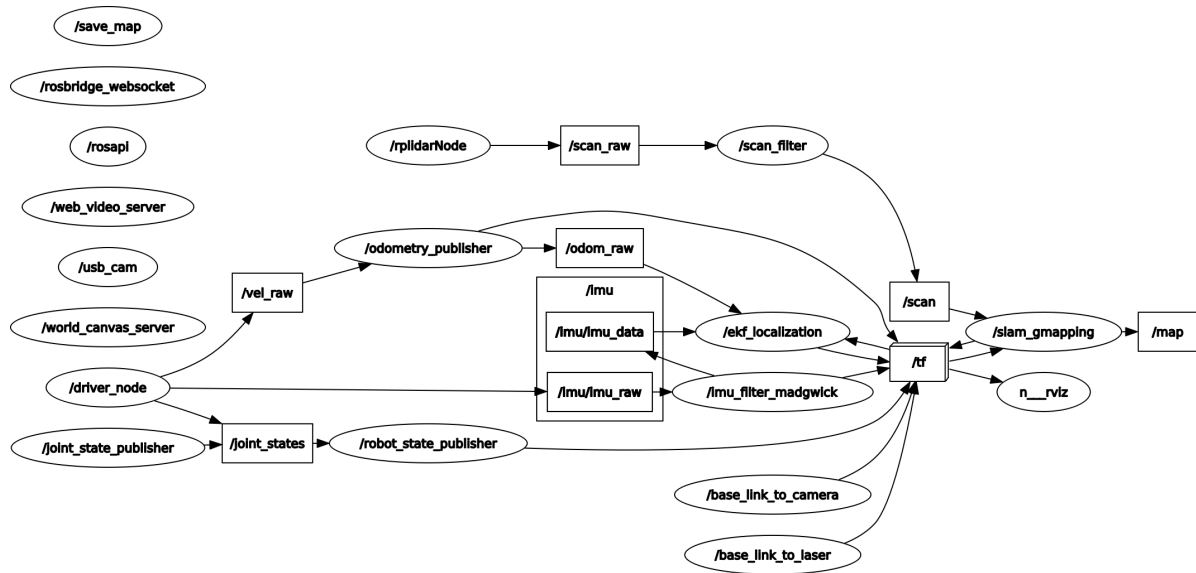
- image: The path of the map file, which can be an absolute path or a relative path
- resolution: the resolution of the map, m/pixel
- origin: The 2D pose(x, y, yaw) of the lower left corner of the map, where yaw is rotated counterclockwise(yaw=0 means no rotation). Many parts of the system currently ignore the yaw value.
- negate: whether to reverse the meaning of white/black, free/occupy(the interpretation of the threshold is not affected)
- occupied_thresh: Pixels with an occupancy probability greater than this threshold will be considered fully occupied.
- free_thresh: Pixels with an occupancy probability less than this threshold will be considered completely free.

6.2 Topics and Services

| Subscribe to topics | type | describe |
|---------------------|------------------------|---|
| tf | tf/tfMessage | Used to convert between lidar coordinate system, base coordinate system, and odometer coordinate system |
| scan | sensor_msgs/LaserScan | Lidar scan data |
| Post a topic | type | describe |
| map_metadata | nav_msgs/MapMetaData | Publish map Meta data |
| map | nav_msgs/OccupancyGrid | Publish map raster data |
| ~entropy | std_msgs/Float64 | Publish an estimate of the robot pose distribution entropy |
| Serve | type | describe |
| dynamic_map | nav_msgs/GetMap | Get map data |

Node view

```
rqt_graph
```



6.3 configuration parameters

| parameter | type | Defaults | describe |
|----------------------|--------|-------------|---|
| ~throttle_scans | int | 1 | After receiving the laser data of this number of frames, only one frame of data is processed. By default, it is processed every time a frame of data is received. |
| ~base_frame | string | "base_link" | Robot base coordinate system |
| ~map_frame | string | "map" | map coordinate system |
| ~odom_frame | string | "odom" | Odometer Coordinate System |
| ~map_update_interval | float | 5.0 | The map update frequency, the lower the value, the greater the computational load |
| ~maxUrange | float | 80.0 | The maximum range that the laser can detect |
| ~sigma | float | 0.05 | Standard Deviation of Endpoint Match |
| ~kernelSize | int | 1 | Find in the corresponding kernel |
| ~lstep | float | 0.05 | Optimization step size during translation |
| ~astep | float | 0.05 | Optimization step size during rotation |
| ~iterations | int | 5 | The number of iterations to scan for matches |
| ~lsigma | float | 0.075 | Laser Standard Deviation for Likelihood Calculations |
| ~ogain | float | 3.0 | Used to smooth the resampling effect when calculating the likelihood |
| ~lskip | int | 0 | Number of beams skipped per scan |
| ~minimumScore | float | 0 | Scan the lowest value for matching results. Avoid jumping pose estimation in large open spaces when using laser scanners with limited range(e.g. 5m) |
| ~srr | float | 0.1 | translation function(ρ/ρ), odometer error while translation |
| ~srt | float | 0.2 | Rotation function(ρ/θ), odometry error in translation |

| parameter | type | Defaults | describe |
|---------------------------|-------|----------|---|
| ~str | float | 0.1 | Translation function(θ/ρ), odometry error while rotating |
| ~stt | float | 0.2 | Rotation function(θ/θ), odometer error while rotating |
| ~linearUpdate | float | 1.0 | After the robot translates this distance, the laser scan data is processed once |
| ~angularUpdate | float | 0.5 | The laser scan data is processed every time the robot rotates this distance |
| ~temporalUpdate | float | -1.0 | If the latest scan is being processed slower than the update, then a scan is processed. A negative value turns off time-based updates |
| ~resampleThreshold | float | 0.5 | Resampling threshold based on Neff |
| ~particles | int | 30 | number of particles in the filter |
| ~xmin | float | -100.0 | map x to initial minimum size |
| ~ymin | float | -100.0 | Map y to initial minimum size |
| ~xmax | float | 100.0 | map x to initial maximum size |
| ~ymax | float | 100.0 | The initial maximum size of the map y direction |
| ~delta | float | 0.05 | map resolution |
| ~llsamplerange | float | 0.01 | Shifted sampling distance for likelihood calculation |
| ~llsamplestep | float | 0.01 | Shifted sampling step size for likelihood calculation |
| ~lasamplerange | float | 0.005 | Rotation sampling distance for likelihood calculation |
| ~lasamplestep | float | 0.005 | Rotation sampling step size for likelihood calculation |
| ~transform_publish_period | float | 0.05 | The time interval at which the TF transform is published |
| ~occ_threh | float | 0.25 | Threshold for raster map occupancy |
| ~maxRange(float) | float | - | maximum range of the sensor |

6.4 TF transformation

| Required TF Transform | describe |
|------------------------|---|
| laser-->base_link | The transformation between the lidar coordinate system and the base coordinate system, generally published by robot_state_publisher or static_transform_publisher |
| base_link-->odom | The transformation between the map coordinate system and the robot's odometer coordinate system to estimate the robot's pose in the map |
| Published TF Transform | describe |
| map-->odom | The transformation between the map coordinate system and the robot's odometer coordinate system to estimate the robot's pose in the map |

View tf tree

```
rosrun rqt_tf_tree rqt_tf_tree
```

