



ALL UNANSWERED

Search or ask your question

ASK YOUR QUESTION

4

Localization problem with gmapping and RPLidar

gmapping

localization

hydro

rplidar

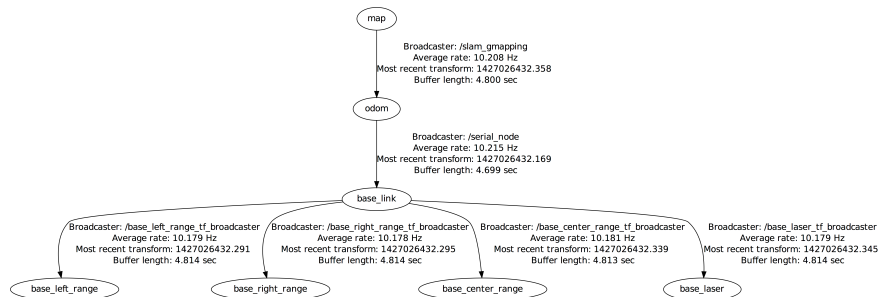
Hello ROS community,

I'm working on a project which goal is to create an [autonomous mobile robot](#) running SLAM algorithm using gmapping. I'm using odometry from motors encoder and laser scan data from RPLidar. ROS Hydro is running on a UDOO board.

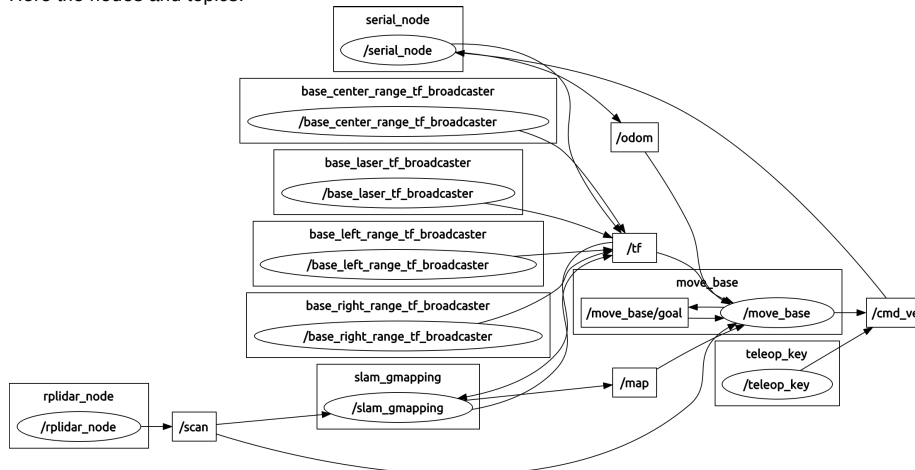
I complete the navigation stack setup and, as soon the robot is not moving, all seems to work fine (transformations published, map published, laser scan data published). When I move the robot using teleop the map->odom transformation broadcasted by gmapping seems to make no-sense. Better than other words is a [video of RViz](#).

Apart the localization problem I cannot understand why the odom and base_link frame are not overlapped after start. They don't should be?

Here the transformations tree:



Here the nodes and topics:



This the gmapping node configuration:

```

throttle_scans: 1
base_frame: base_link
map_frame: map
odom_frame: odom
map_update_interval: 10
maxUrange: 5.5
maxRange: 6
  
```

Question Tools

Follow

5 followers

[subscribe to rss feed](#)

Stats

Asked: Mar 23 '15

Seen: 4,342 times

Last updated: Mar 14 '16

Related questions

[occupancy_grid_utils with ros hydro](#)
[What is the best mapping/localization package for using pre-made maps?](#)
[rplidar a2 error time out](#)
[Easiest way to show travelled path on gmapping generated map](#)
[Mapping with kinect instead of Hokuyo laser](#)
[navfn planning through wall](#)
[Kinect No Image received](#)
[slam_gmapping Laser is mounted upwards](#)
[How to transform a laserscan message with TF?](#)
[AMCL Covariance : define a "good" covariance](#)

```
sigma: 0.05
kernelSize: 1
lstep: 0.05
astep: 0.05
iterations: 5
lsigma: 0.075
ogain: 3.0
lskip: 0
minimumScore: 0.0
srr: 0.1
srt: 0.2
str: 0.1
stt: 0.2
linearUpdate: 1.0
angularUpdate: 0.5
temporalUpdate: -1.0
resampleThreshold: 0.5
particles: 30
xmin: -10
xmax: 10
ymin: -10
ymax: 10
delta: 0.05
llsamplerange: 0.01
llsamplestep: 0.01
asamplerange: 0.005
lasamplestep: 0.005
transform_publish_period: 0.1
occ_thresh: 0.25
```

I will really appreciate any suggestion to fix my problem. I did not publish other configurations since the problem seems to be related to gmapping: if other informations are needed I will be happy to provide them.

Many thanks! Ale

UPDATE

As suggested by paulbovbel I follow the guide test odometer quality. The result is quite good for straight path, a little bit less for rotation.

Watching the video I think the problem could not be in odometry: in the video the first seconds (until time 0:08) after robot starts moving all seems to be fine. During this time the position is updated based on odometry only (at least... I guess!) and laser scan data (in red) match the map: this means that odometer data is quite good. After 0:08 the map->odom transformation (broadcasted by gmapping) changes: in theory to compensate odometry drift but, at the end, it spoils the estimate position of the robot. After position estimation is spoiled also laser scan data make no sense and this cause builded map to be... a no-sense! This make sense or I make some mistake in my think?

Just to give more info: the video show the robot just a minute after system starts. When the video starts the robot was stopped in its initial position: for this reason I expect base_link, odom and map frame overlap (drift must be zero and robot it's in center of map).

UPDATE

I'm still working in order to fix this problem. I performed some test to check the quality of my odometry. On the attached image from RViz you can ... [\(more\)](#)

[add a comment](#)

2 Answers

Sort by » [oldest](#) [newest](#) [most voted](#)

1

Hi, afranceson

I am building a SLAM system which is very similiar to yours (with RPLidar + gmapping).

I'm not familiar with the odometry used in such systems. So can you offer me some information about the odometry you used? Or would you please recommend me a good odometry which is well suited for ROS and my system. FYI, my vehicle is a four-wheel rear-drive car which is made by myself. I'd llike to

answered Mar 14 '16

Clack 
65  4  7  13
<http://www.shanghaitec...>

offer any detail about my vehicle if it is needed.

Thank you in advance!

Comments

[link](#)

Dear Clack,

on my robot I'm using MD25 motor control board and EMG30 motors. They provide encoders with 360 clicks per revolution and feedback. This works quite well on my robot.

On www.geduiuo.org you can find more info and sources. Feel free to contact me, I will be happy to help you.

Regards

 [afranceson](#) (Mar 14 '16)

Thank you so much! I think I'd better collect some information on those devices first :)

 [Clack](#) (Mar 14 '16)

Another option would be to use the laser_scan_matcher package to simulate odometry info.

 [Icehawk101](#) (Mar 23 '16)

Thanks to your advice, Icehawk. But we've decided to use a real odometry in our system :)

 [Clack](#) (Mar 23 '16)

Anyway, if you want to use the laser scan data only you should consider to use Hector Slam instead of Gmapping.

 [afranceson](#) (Mar 23 '16)

Thanks, I'll see to that :)

 [Clack](#) (Mar 23 '16)

[add a comment](#)

2

The transformation between odom and base_link represents your robot's best estimate of odometry using wheel odometry (thought you may potentially fuse other sources using robot_pose_ekf or robot_localization), so the two frames should not overlap.

answered Mar 23 '15

 [paulbovbel](#) 
4236 ● 6 ● 41 ● 69
<http://www.bovbel.com/>

updated Mar 28 '15

In the perfect-odometry case, odom and map would overlap. The transform represents gmapping's localization correction. It looks like your odometry drifts quite a bit based on the video. Have you tried tuning it based on the nav stack guide (<http://wiki.ros.org/navigation/Tutori...>

If you get a lot of natural drift in odometry, you could try increasing srr, srt, str, stt parameters to pass that information to gmapping.

Finally, increasing the particle count is always a good bet, although it makes gmapping more processor intensive.

EDIT That laserscan overlay DOES look fairly consistent. Maybe your issue is more with sensor noise then? I've never benchmarked an RPLidar, but I know they're on the cheap side. I'm not sure if there's anything you can tweak in gmapping to help account for that - maybe using a larger grid size, or the increasing minimumScore parameter?


Comments

[link](#)


Thanks for your suggestion: I follow the guide and sensor data and odometry result quite good for forward/back direction, for rotation can be improved. Any way on my test I only move my robot forward and the drift in rotation does not explain the big error I see on RViz.

 [afranceson](#) (Mar 23 '15)


paulbovbel, is it possible to get localization information from the GMapping node itself? I need a good pose estimate that comes from the SLAM algorithm itself not from the odometry data. I need to use it in an exploration algorithm.

 [RND](#) (Apr 18 '15)

Also, can you briefly explain to me what are the meanings of srr, srt, str and stt with respect to the gmapping SLAM algorithm? What are we doing if we increase these parameters? thanks

 [RND](#) (Apr 18 '15)

@RND, read the gmapping ROS wiki first.

 [paulbovbel](#) (Apr 22 '15)

[add a comment](#)

Your Answer

Please start posting anonymously - your entry will be published after you log in or create a new account.

Add Answer

[about](#) | [faq](#) | [help](#) | [privacy policy](#) | [terms of service](#)
Powered by Askbot version 0.10.2

ROS Answers is licensed under Creative Commons Attribution 3.0 Content
on this site is licensed under a [Creative Commons Attribution Share Alike](#)
3.0 license.

