

# 使用 robot\_pose\_ekf 轻舟机器人传感器信息融合

AI 航团队

robot\_pose\_ekf 是 ROS Navigation stack 中的一个包，用于评估机器人的 3D 位姿，使用了来自不同源的位姿测量信息，它使用带有 6D（3D position and 3D orientation）模型信息的扩展卡尔曼滤波器来整合来自轮子里程计，IMU 传感器和视觉里程计的数据信息。基本思路就是用松耦合方式融合不同传感器信息实现位姿估计。

## 一、如何使用 EKF

EKF node 的缺省启动文件位于 robot\_pose\_ekf 包中，文件中有许多配置参数：

- freq: 滤波器更新和发布频率，注意：频率高仅仅意味着一段时间可以获得更多机器人位姿信息，但是并不表示可以提高每次位姿评估的精度
- sensor\_timeout: 当某传感器停止给滤波器发送信息时，滤波器应该等多长时间方可以在没有该传感器信息状况下继续工作
- odom\_used, imu\_used, vo\_used: enable/disable 输入源

启动文件中配置参数设置可以被修改，看起来大致如下所示：

```
<launch>
  <node pkg="robot_pose_ekf" type="robot_pose_ekf" name="robot_pose_ekf">
    <param name="output_frame" value="odom"/>
    <param name="freq" value="30.0"/>
    <param name="sensor_timeout" value="1.0"/>
    <param name="odom_used" value="true"/>
    <param name="imu_used" value="true"/>
    <param name="vo_used" value="true"/>
    <param name="debug" value="false"/>
    <param name="self_diagnose" value="false"/>
  </node>
</launch>
```

## 二、Subscribed Topics

odom ([nav\\_msgs/Odometry](#))

2D 位姿(车轮测程法使用): 2D 位姿包含机器人在地平面上的位置和姿态，以及该位姿的协方差。发送这个 2D 姿态的消息实际上代表了一个 3D 姿态，但是 z、滚动和俯仰都被忽略了

imu\_data ([sensor\\_msgs/Imu](#))

3D 方向(由 IMU 使用): 3D 方向提供了关于机器人基础框架相对于世界参考框架的滚动、俯仰和偏航角的信息。横摇角和俯仰角被解释为绝对角度(因为 IMU 传感器有一个重力参考)，而偏航角被解释为相对角度。协方差矩阵指定方向测量的不确定度。当机器人只接收到关于这个主题的信息时，其姿态 ekf 不会启动；它还希望收到关于“vo”或“odom 主题的信息。

vo ([nav\\_msgs/Odometry](#))

3D 姿态(由视觉测距术使用): 3D 姿态表示机器人的完整位置和方向，以及该姿态的协方差。当传感器只测量 3D 姿态的一部分时(例如，车轮里程表只测量 2D 姿态)，只需在未实际测量的 3D 姿态部分指定一个大的协方差。

### 三、Published Topics

robot\_pose\_ekf/odom\_combined ([geometry\\_msgs/PoseWithCovarianceStamped](#))  
滤波器的输出(估计的 3D 机器人姿态)。

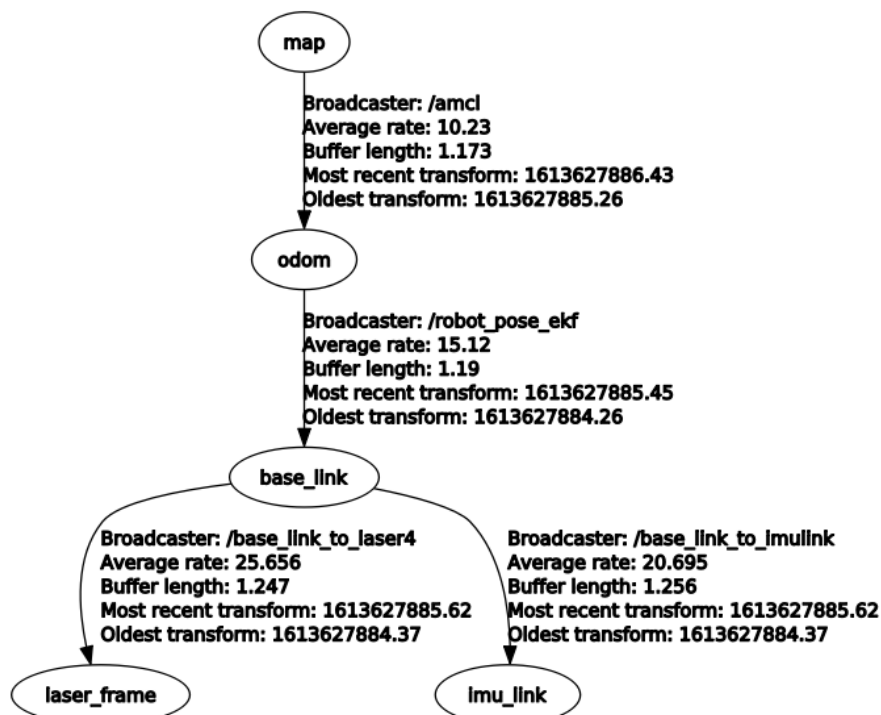
发布 tf: robot\_pose\_ekf 在输出 odom\_combined 信息同时还会发布相关的坐标变换:  
odom\_combined → base\_footprint

### 四、轻舟机器人传感器信息融合

在 qinzhou\_bringup.launch 文件中, 加入如下文件启动 robot\_pose\_ekf 节点。

```
<node pkg="robot_pose_ekf" type="robot_pose_ekf" name="robot_pose_ekf" output="screen" respawn="true">
  <param name="output_frame" value="odom"/>
  <param name="base_footprint_frame" value="base_link"/>
  <param name="freq" value="30.0"/>
  <param name="sensor_timeout" value="1.0"/>
  <param name="odom_used" value="true"/>
  <param name="imu_used" value="true"/>
  <param name="vo_used" value="false"/>
  <param name="gps_used" value="false"/>
  <param name="debug" value="false"/>
  <param name="self_diagnose" value="false"/>
</node>
```

这个节点将订阅/odom 话题和/imu\_data 话题。同时我们设置输出的 tf 变换为  
/odom→base\_link;



/imu\_data 话题我们上节已经讲过, 在 imu 的数据经过 imu\_filter\_madgwick 滤波器以后, 将会发布/imu\_data 话题, 另外 odom 话题已经在 qinzhou\_bringup 节点中发布, 代码如下:

```

nav_msgs::Odometry odom; //创建nav_msgs::Odome
odom.header.stamp = current_time;
odom.header.frame_id = "odom";
odom.child_frame_id = "base_link";

//set the position
odom.pose.pose.position.x = x;
odom.pose.pose.position.y = y;
odom.pose.pose.position.z = 0.0;
odom.pose.pose.orientation = odom_quat;

odom.twist.twist.linear.x = linearSpeed; //线速度
odom.twist.twist.linear.y = 0;
odom.twist.twist.linear.z = 0;
odom.twist.twist.angular.x = 0;
odom.twist.twist.angular.y = 0;
odom.twist.twist.angular.z = angularSpeed; //角速度

if(encoderLeft == 0 && encoderRight == 0){
    odom.pose.covariance = {1e-9, 0, 0, 0, 0, 0,
                           0, 1e-3, 1e-9, 0, 0, 0,
                           0, 0, 1e6, 0, 0, 0,
                           0, 0, 0, 1e6, 0, 0,
                           0, 0, 0, 0, 1e6, 0,
                           0, 0, 0, 0, 0, 1e-9};
    odom.twist.covariance = {1e-9, 0, 0, 0, 0, 0,
                             0, 1e-3, 1e-9, 0, 0, 0,
                             0, 0, 1e6, 0, 0, 0,
                             0, 0, 0, 1e6, 0, 0,
                             0, 0, 0, 0, 1e6, 0,
                             0, 0, 0, 0, 0, 1e-9};
}
else{
    odom.pose.covariance = {1e-3, 0, 0, 0, 0, 0,
                           0, 1e-3, 0, 0, 0, 0,
                           0, 0, 1e6, 0, 0, 0,
                           0, 0, 0, 1e6, 0, 0,
                           0, 0, 0, 0, 1e6, 0,
                           0, 0, 0, 0, 0, 1e3};
    odom.twist.covariance = {1e-3, 0, 0, 0, 0, 0,
                             0, 1e-3, 0, 0, 0, 0,
                             0, 0, 1e6, 0, 0, 0,
                             0, 0, 0, 1e6, 0, 0,
                             0, 0, 0, 0, 1e6, 0,
                             0, 0, 0, 0, 0, 1e3};
}
//publish the message
pub_odom.publish(odom);

```

robot\_pose\_ekf 包在订阅以上两个话题以后, 将发布 robot\_pose\_ekf/odom\_combined 话题, 但是需要注意的是/odom 和/robot\_pose\_ekf/odom\_combined 两个话题的消息类型不同, 前者是 nav\_msgs/Odometry, 后者是 geometry\_msgs/PoseWithCovarianceStamped, 两者的区别是后者的内容是前者的一部分。

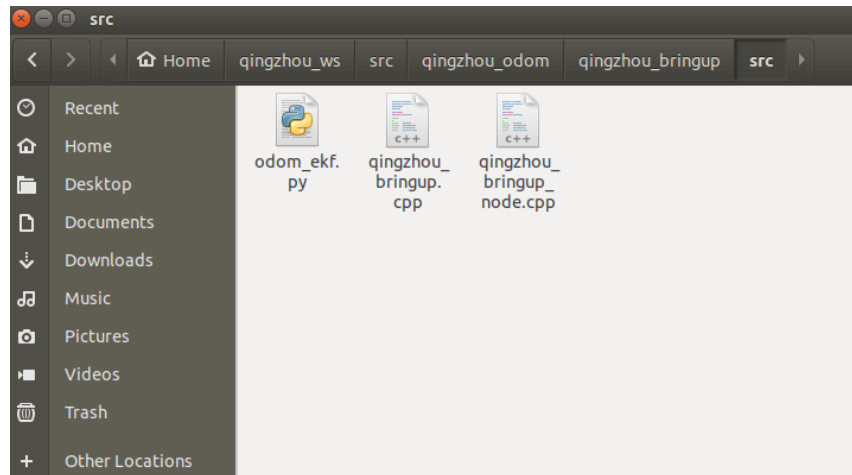
```

/raw
/roadLine
/roadLine angle
/robot_pose_ekf/odom_combined
/rosout
/rosout_agg
/startStopCommand
/stoppoint
/tf

```

现在我们得到了/robot\_pose\_ekf/odom\_combined 发布的消息。在 movebase 需要订阅

/odom 的话题，想让这个消息被 movebase 使用。这里使用 qingzhou\_bringup 包里提供的一个节点：odom\_ekf.py 实现格式转化。



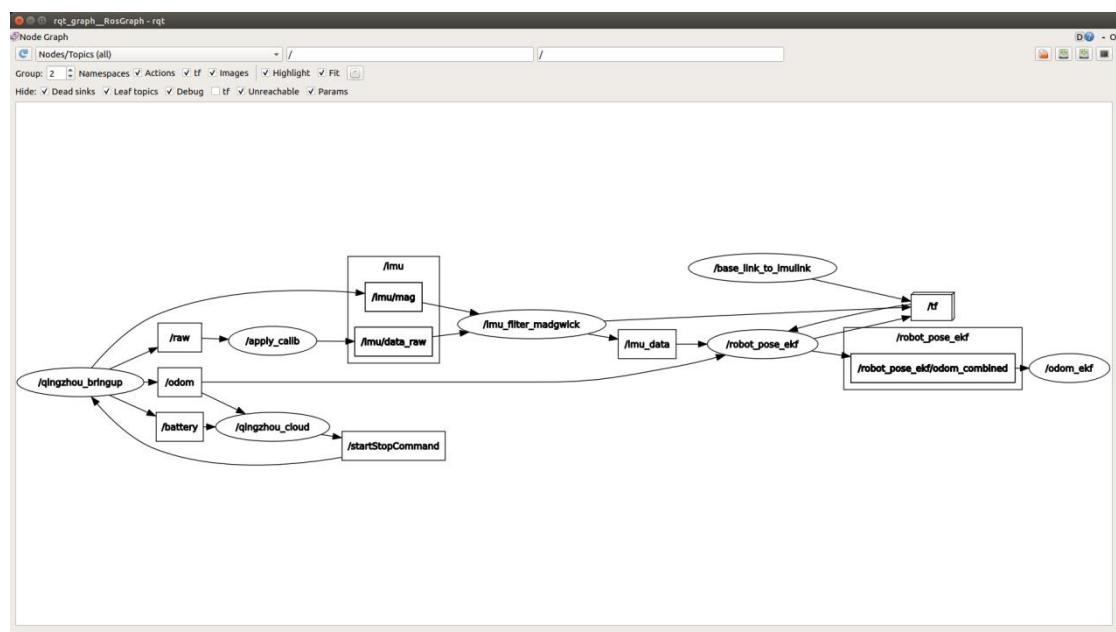
转化后将发布/odom\_ekf 话题，这个就可以给 movebase 使用了，我们需要在启动 move\_base 的 launch 文件中将其订阅的里程计话题 remap 成/odom\_ekf。

```
<node pkg="move_base" type="move_base" respawn="false" name="move_base" output="screen">
  <param name="base_global_planner" value="global_planner/GlobalPlanner" />
  <param name="planner_frequency" value="1.0" />
  <param name="planner_patience" value="5.0" />

  <param name="base_local_planner" value="dwa_local_planner/DWAPlannerROS" />
  <param name="controller_frequency" value="5.0" />
  <param name="controller_patience" value="5.0" />

  <rosparam file="$(find qingzhou_nav)/config/costmap_common_params.yaml" command="load" ns="global_costmap" />
  <rosparam file="$(find qingzhou_nav)/config/costmap_common_params.yaml" command="load" ns="local_costmap" />
  <rosparam file="$(find qingzhou_nav)/config/local_costmap_params.yaml" command="load" />
  <rosparam file="$(find qingzhou_nav)/config/global_costmap_params.yaml" command="load" />
  <rosparam file="$(find qingzhou_nav)/config/dwa_local_planner_params.yaml" command="load" />
  <rosparam file="$(find qingzhou_nav)/config/global_planner_params.yaml" command="load" />
  <remap from="/odom" to="/odom_ekf"/>
</node>
```

最后整个数据融合的网络图如下：



1. 同学们在使用过程中，如果发现内容有疏漏或者不严谨的地方，请与我们联系，将会有轻舟积分送上！QQ：270220858
2. 内容如有雷同，侵删！

2021 年 2 月