# AMCL包

## 一、说明

AMCL在gmapping成功构建地图的基础上进行的定位；

amcl是移动机器人二维环境下的概率定位系统，根据订阅到的地图数据配合激光扫描特征，使用粒子滤波获取最佳定位点。

## 二、订阅话题

1. Laser\_scan\_filter（激光数据）：sensor\_msgs::LaserScan，AmclNode::laserReceived，

回调函数laserReceived是粒子滤波主要过程,根据激光扫描数据更新粒子

1. tf：th/TfMessage，将激光数据转换到在odom\_frame\_id里程计坐标系下的数据
2. initialpose：gemetry\_msgs/PoseWithCovarianceSamped，用于重新初始化粒子滤波器的均值和协方差

## 三、发布的话题：

1、amcl\_pose：geometry\_msgs/PoseWithCovarianceStamped，机器人在地图中的估计位姿和协方差：后验位姿+6\*6的协方差矩阵（xyz+三个转角）

2、particlecloud：geometry\_msgs/PoseArray，粒子位姿的数组

3、tf：th/TfMessage，

3、一个15秒的定时器：AmclNode::checkLaserReceived，检测上一次收到的激光雷达数据至今是否超过15秒，如超过则报错。

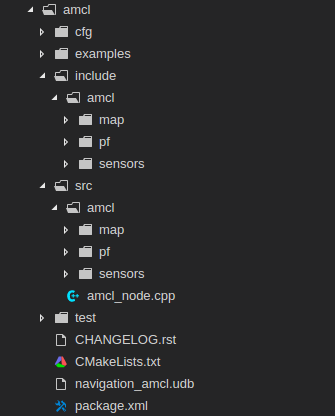
## 四、发布的服务：

1. global\_localization：&AmclNodee::globalLocalizationCallback，在没有给定初始位姿的情况下，在全局范围内初始化粒子位姿，该Callback调用pf\_init\_model，然后调用AmclNode::uniformPoseGenerator在地图的free点随机生成pf->max\_samples个粒子、
2. request\_nomotion\_update：std\_srvs/Empty，设置标志，使amcl在无运动时，手动执行更新和发布粒子的服务
3. set\_map：setMapCallback，用于手动设置新地图和姿势的服务，调用如下两个函数

handleMapMessage(req.map);//进行地图转换 ，记录free space ，以及初始化pf\_t 结构体，实例化运动模型(odom)和观测模型(laser)

handleInitialPoseMessage(req.initial\_pose);根据接收的初始位姿消息,在该位姿附近高斯采样重新生成粒子集

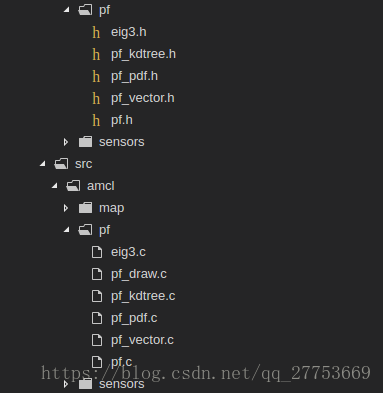
## 五、代码结构



AMCL最为核心的是include目录和src目录，一共包含三个库和一个节点：map、pf（particle filter）、sensors以及amcl\_node.cpp。

map定义地图相关功能；pf是算法的核心粒子滤波(particle filter)；sensors是处理激光和里程计传感器；amcl\_node.cpp定义了一个可运行amcl节点，定位过程主要在此文件中组织实现。

## 六、pf库



eig3：实现3x3对称矩阵的特征值与特征向量的计算。

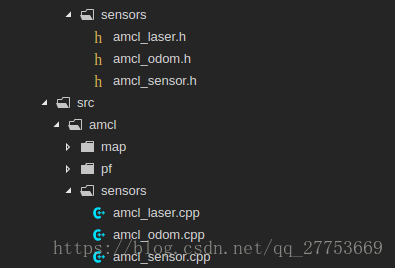
pf\_kdtree：定义了一个kdtree以及维护方法来管理所有粒子

pf\_pdf：定义从给定pdf中采样粒子的方法：高斯pdf取样

pf\_vector：三维列向量和三维矩阵和基本的运算方法

e).pf定义了粒子单元pf\_sample\_t,粒子集pf\_sample\_set\_t,粒子滤波pf\_t的数据类型,以及 pf\_cluster\_t表示粒子集的聚类信息,包含粒子滤波中的运动更新,观测更新,重采样三个过程。

## 七、sensor库

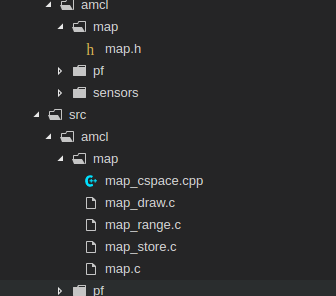


amcl\_sencor：定义了一个基类,其成员函数都为虚函数，为两个派生类amcl\_laser和amcl\_odom提供统一接口。

bamcl\_laser定义了激光数据类型，三种观测更新模型（整体过滤器，激光模型和odometery模型），具体实现了UpdateSensor，用于计算粒子权值

amcl\_odom具体实现了基类定义的UpdateAction函数，用于根据运动更新粒子,定义了两种运动模型，：差分和全向。

## 八、map库



map中主要定义了概率栅格地图，新建地图、删图、取call。

map\_cspace地图单元数据、缓存距离题图

map\_draw取地图

map\_range提取最大范围度数

map\_store加载占用栅格地图

## 九、amcl节点

输入：激光地图；初始化粒子滤波器的均值（初始位姿）协方差（所有粒子的后验位姿），用于初始化粒子（粒子集）；里程计信息；激光数据；tf坐标变化

输出：在地图中的估计姿势（粒子后验位姿以及协方差矩阵，xyz和三个转角）、粒子集

在源码中，amcl\_node.cpp文件实现了amcl节点功能，主要流程为：

1：对构造初始化，从参数服务器中获取数据初始化各类参数。包括：接收地图设置、gui显示发布频率、保存位姿到参数服务器频率、laser测距范围及其概率模型参数、odom概率模型参数、粒子滤波及kld重采样参数、从参数服务器获取初始位姿，然后初始化订阅者,发布者,服务)。在这里，若未设置参数，则初始过滤器状态将是以（0,0,0）为中心的中等大小的粒子云；

2：获取激光地图：两种方式(订阅/map话题；请求服务得到地图)；

3：获取用于初始化粒子滤波器的均值（初始位姿）协方差（所有粒子的后验位姿）pf、激光数据laser、tf（里程计信息）；

4：更新机器人位置坐标：回调函数LaserRecieved()

a：获取激光数据对应于里程计中的坐标

c：通过里程计的运动模型（差分型双轮机器人）更新pf\_t中现有的samples每一个粒子的位姿（得到当前时刻的先验位姿）

odom->updateAction()

d：根据当前激光数据更新各里程计对应的权值weights：laser\_[laser\_index]->updateSensor（）

e：获取权值最高的坐标点进行聚类（得到均指和方差，把相近的很多粒子融合成一个粒子），然后将新粒子发布到全局坐标系下（地图坐标系）。

f：遍历所有粒子簇，找出权重均指最大的簇，这个平均位姿即为当前机器人的位姿（后验位姿）

g：更新位姿、粒子集、协方差矩阵，即分别在话题/amcl\_pose和/particlecloud上发布位姿和粒子集。