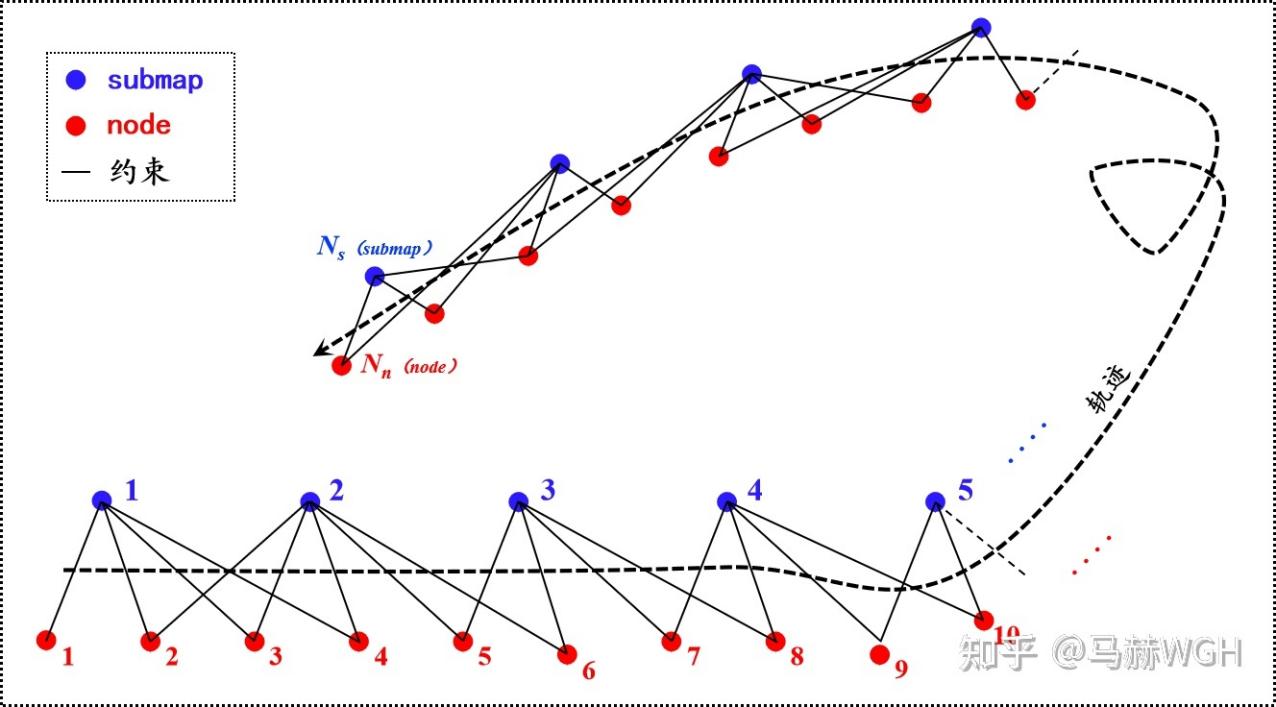
Pose\_graph\_2d.cc文件需要好好看看



每个子图有一个子图原点，这个是submap\_query中slice\_pose。而每个子图之间是有微弱旋转的，而控制旋转的中心也是这个slice\_pose（子图原点）。

每一个关键帧都会在**前端**中与**1个**子图进行 scan-to-submap 配准，得到一个相对位姿，关键帧按照此相对位姿插入到最新的**2个**子图中，并与这2个子图分别构建约束（称为 **intra constraint**）；

第一个约束, 根据SPA论文, 就是节点global坐标系下的相对坐标变换.

所以, 第二个约束怎么求呢?

cartographer中设置了如下5种第二个约束及残差, 共同进行优化.

第一种残差 将节点(tracking的位姿)本人认为这个是扫描匹配之后的位姿，与节点(子图原点位姿)本人认为这个是回环检测之后的位姿，这里是子图遇到回环之后的，结合初始位姿处节点（包括节点维护的图）得到的节点在global坐标系下的相对位姿 与

约束 的差值作为残差项

第二种残差 landmark数据 与 通过2个节点位姿插值出来的相对位姿 的差值作为残差项

第三种残差 节点与节点间在global坐标系下的相对坐标变换 与 通过里程计数据插值出的相

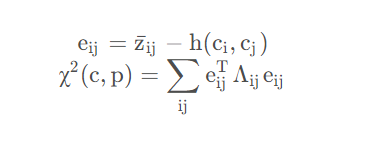
对坐标变换 的差值作为残差项

第四种残差 节点与节点间在global坐标系下的相对坐标变换 与 相邻2个节点在local坐标系

下的相对坐标变换 的差值作为残差项

第五种残差 节点与gps坐标系原点在global坐标系下的相对坐标变换 与 通过gps数据进行插

值得到的相对坐标变换 的差值作为残差项



回环检测的目的就是再回到初始位姿时预先让Z = 0（这个可以通过第二种方式算出来，即第二种约束）；计算Ci到Cj的位姿变换

再计算残差eij.最后在计算所有的残差，最小二乘求最小值。

前端在local坐标系下，后端在global坐标系下