



激光SLAM第三次作业讲解



主讲人 张智辉



第一题

```
//TODO
tf::Vector3 trans_b, trans_e, trans_n;
tf::Quaternion rot_b, rot_e, rot_n;
trans_b = frame_start_pose.getOrigin();
trans_e = frame_end_pose.getOrigin();
rot_b = frame_start_pose.getRotation();
rot_e = frame_end_pose.getRotation();
for(int i = startIndex + 1, n = startIndex + beam_number - 1; i <= n; ++i){
    // 取出该点在未被校正的机器坐标系下的坐标
    tf::Vector3 p;
    p.setX(ranges[i] * cos(angles[i]));
    p.setY(ranges[i] * sin(angles[i]));
    p.setZ(0);
    // 得到被校正的机器坐标系
    double scale = (i - startIndex) / (beam_number - 1);
    trans_n = trans_b.lerp(trans_e, scale);
    rot_n = rot_b.slerp(rot_e, scale);
    tf::Pose rob_n(rot_n, trans_n);
    // 得到在新机器坐标系下的坐标
    tf::Vector3 pn = frame_base_pose.inverse() * rob_n * p;
    // 修正 ranges 和 angles
    ranges[i] = pn.length();
    angles[i] = atan2(pn.y(), pn.x());
}
//end of TODO
```

第一题

```
• //TODO
• tf::Vector3 trans_b, trans_e, trans_n;
• tf::Quaternion rot_b, rot_e, rot_n;
• trans_b = frame_start_pose.getOrigin();
• trans_e = frame_end_pose.getOrigin();
• rot_b = frame_start_pose.getRotation();
• rot_e = frame_end_pose.getRotation();
• for(int i = startIndex + 1, n = startIndex + beam_number - 1; i <= n; ++i){
•     // 取出该点在未被校正的机器坐标系下的坐标
•     tf::Vector3 p;
•     p.setX(ranges[i] * cos(angles[i]));
•     p.setY(ranges[i] * sin(angles[i]));
•     p.setZ(0);
```

第一题

```
• // 得到被校正的机器坐标系
• double scale = (i - startIndex) / (beam_number - 1);
• trans_n = trans_b.lerp(trans_e, scale);
• rot_n = rot_b.slerp(rot_e, scale);
• tf::Pose rob_n(rot_n, trans_n);
• // 得到在新机器坐标系下的坐标
• tf::Vector3 pn = frame_base_pose.inverse() * rob_n * p;
• // 修正ranges和angels
• ranges[i] = pn.length();
• angles[i] = atan2(pn.y(), pn.x());
• }
• //end of TODO
```

第一题

- frame_base_pose: 第一个激光点对应的机器位姿
- frame_end_pose: 最后一个激光点对应机器位姿
- 第一步: 分别取出两个位姿中的旋转和平移部分, 方便进行插值
- 第二步: 得到激光点在机器坐标系下的坐标
- 第三步: 通过插值得到校正后的机器位姿
- 第四步: 通过校正后的机器位姿来修正激光的测量值

第二题

- 题目要求：推导并证明已知对应点的ICP求解方法。
- 答案：参考第四章课程视频。

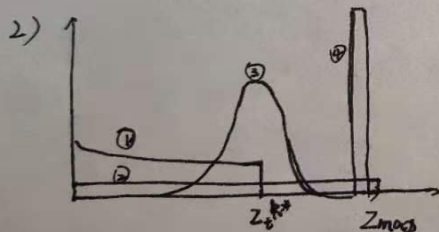
第三题

3. 1) 激光雷达测距原理:

(1). time-of-flight: 飞行时间法, 通过测量激光脉冲从发射, 到返回接收的总时间 Δt , 来计算到障碍物的距离:
$$r = \frac{c \cdot \Delta t}{n \cdot 2}$$
 其中 c 为真空光速, n 为空气折射率.

(2). phase difference: 相位差, 通过测量发射出的激光光波与返回的光波的相位差 $\Delta \varphi$, 来计算两者距离:

$$\Delta t = \frac{\Delta \varphi \cdot T_m}{2\pi}, \quad r = \frac{c \cdot \Delta t}{n \cdot 2} = \frac{c \cdot \Delta \varphi \cdot T_m}{4n\pi}$$
 其中 T_m 为1周期时间.



左侧图是四种激光测距分布图.

- ①. 在障碍物前的指数分布
- ②. 从最小到最大的均匀分布
- ③. 在障碍物附近的高斯分布
- ④. 在最大距离外的一个集中分布.

第四题

4. 1) 仅用 IMU 去除运动畸变可能会有那些不足之处? ↵

答: 去除运动畸变要求我们得到短时间内机器精确的旋转量与平移量, 并将这些旋转量与平移量与一帧中的每一条激光束联系起来。对于 IMU 来说, 虽然它的测量频率快, 角速度测量的比较准确, 但是对于大部分 IMU 来说, 它的线加速度测量的误差较大, 这就让它积分得到的平移量会有一个比较大的误差, 这就是 IMU 进行运动畸变去除的一个主要问题。↵

↵

2) 在仅有 IMU 和激光雷达传感器的情况下, 你会如何设计运动畸变去除方案 (平移加旋转), 达到较好的畸变去除效果? ↵

答: 考虑到用 IMU 积分得到的平移量误差较大, 所以我们先仅仅使用 IMU 的旋转量来进行畸变去除, 去除完毕之后进行 ICP, 利用这时 ICP 的结果, 加上匀速运动假设, 插值得到每一条激光光线的平移量, 平移量加上 IMU 的旋转量, 得到 2 次纠正的激光数据。利用二次纠正的激光数据再进行 ICP, 若两次 ICP 得到的平移量很接近则视为收敛, 否则不停的使用上一次的 ICP 计算结果来对激光数据进行纠正。↵

感谢各位聆听 !
Thanks for Listening

