

实验五: EKF-SLAM

1 实验介绍

本次实验在实验四的基础上继续扩展。实验四中实现的机器人定位依赖于一个先验的已知地图，而本次实验则是要机器人在探索环境的同时实现自动的地图构建。

2 EKF-SLAM (90 分)

EKF-SLAM 的思路是把地图以概率的方式添加到状态中,也就是说将地图 M 上的地标 (landmark) m_i 的参数 α_i 和 r_i 添加到 EKF 状态, 原本的状态 $\mathbf{x} = [\mathbf{x}, \mathbf{y}, \theta]^\top$ 便扩增为:

$$\mathbf{x} = [\mathbf{x}, \mathbf{y}, \theta, \alpha^1, \mathbf{r}^1, \dots, \alpha^i, \mathbf{r}^i]^\top \quad (1)$$

假设地标在世界坐标系中是静止的。通过将地标添加到状态向量中, 可以获得地标的确定性及其之间的相关性。在 SLAM 实现的具体设计上, 为了简化了实现的细节, 通常假设地标的数量已知, 并在启动时对其进行粗略的初始化估计。

2.1 状态预测 (40 分)

EKF 状态扩展为公式 (1) 的形式后, 机器人的状态转移模型也相应随之改变。地标被认为是静止的, 即:

$$\dot{\alpha} = 0 \quad (2)$$

$$\dot{r} = 0 \quad (3)$$

任务: 以实验四的状态转移函数为基础, 重写公式 (1) 的状态转移函数, 及新的状态转移函数相对于状态和控制输入的雅可比函数 F_x 和 F_u 。并在 `transitionFunction.m` 中编程实现。

验证: 在 MATLAB 中运行 `test/validateTransitionFunction.m`。通过函数打印的文本判断是否编写正确。

2.2 测量更新 (40 分)

本次实验中的主要改进是在这一步骤。为了简化代码的实现和调试, 假设所有的地标都是可见的, 并且地标的数量固定。后续的测量关联和估计与实验四类似, 只是测量模型的雅可比矩阵 H^i 变化较大, 需要特别注意。

与实验四不同的是，本实验中地标位置的 Ground Truth 是未知的。因此，在全局固定参考系中，无法判断机器人或地标的估计位置是否错误，以及是否应该进行调整。举个例子，如果机器人和地标同时移动或旋转相同的量，那么将无法仅使用机器人相对于墙壁或车轮编码器信息的相对距离或方位测量来检测这种运动。为了消除这些自由度，采取的措施是在状态向量中固定前两个地标（假设两堵墙不平行），以便完全约束不可观测的状态。

任务：以实验四为基础，实现一个测量函数 $h(\hat{x}_t, i)$ ，该函数的输入为新的状态向量（包括机器人状态和地标）和地标索引，输出为预测测量值。并推导其雅可比矩阵 $\hat{H}_{x,i}$ 。要求在 `measurementFunction.m` 中编程实现。（提示：地标位置现在是状态向量的一部分，在雅可比矩阵的计算中需要考虑这一点。）

验证：在 MATLAB 中运行 `test/validateMeasurementFunction.m`。通过函数打印的文本判断是否编写正确。

2.3 完整 EKF-SLAM 系统的验证（10 分）

运行 `test/validateFilter.m`，对实现的 EKF-SLAM 进行评价，与 baseline 和 ground Truth 轨迹做比较。

运行 `test/validatecompletekloop.m`，使用随机生成的输入值对系统进行评价，输出应尽量接近 Ground Truth 轨迹。

3 Coppeliasim 仿真（10 分）

任务：在仿真环境中对上面实现的 EKF-SLAM 进行测试。

启动仿真平台，加载场景，在 MATLAB 中运行 `vrep/vrepSimulation.m` 开始模拟。机器人平台应开始沿圆形路径移动。在靠近实际机器人的地方，你应该会看到一个黄色的“幽灵”，它可以根据你的定位估计出姿势。

4 需提交内容

1. 实验报告

- (1). 题目
- (2). 姓名-学号-班级
- (3). 实验内容分析：比如对实验内容的理解、关键点、思路
- (4). 实验过程分析：比如每一步的解析
- (5). 实验结果分析
- (6). 遇到的问题和心得

2. 实验源代码