

# 移动机器人导论考前黄金押题卷

## 1-2

第一章第二章为概论，不做考察。

## 3

移动机器人自由度分析

移动机器人运动学模型

## 4

极坐标控制器设计，为什么要这样。

## 5 移动机器人传感

### 5.1 里程计误差模型推导

光电传感器 怎么推导里程计模型，怎么进行传感器融合。

从以下开始，考察的内容均为思路，不需要完全了解。

### 5.2 激光定位原理

- TOF测距原理：路程初一速度，为了获得3mm的测量精度，计算机系统需要具备的时间精度为
$$t = \frac{3 \times 10^{-3}}{3 \times 10^8} = 0.01 ns$$

### 最小二乘法

最小二乘法的核心思想也是基于优化的，希望找到一条最优的曲线，满足该曲线和真实曲线的误差平方和最小。假设对于直线

$$y = ax + b$$

最小二乘的必要条件是函数对参数的偏导为0。

未知点的坐标可以通过三角化的方法来确定。对于已知的一组观测 $(x_i, y_i)$ 和当前位置 $(x, y)$

$$\begin{aligned} \sqrt{(x_i - x)^2 + (y_i - y)^2} &= d_i \\ \implies x^2 + y^2 - 2x_i x - 2y_i y &= d_i^2 - (x_i^2 + y_i^2) \end{aligned}$$

我们如何最小二乘法来求解定位问题呢？首先整理得

$$2(x_n - x_i)x + 2(y_n - y_i)y = d_i^2 - d_n^2 + (x_n^2 + y_n^2) - (x_i^2 + y_i^2)$$

令向量 $\bar{x} = [x, y]^T$ ，上式可以写成

$$A\bar{x} = B$$

然后利用矩阵形式的最小二乘法求解即可。

## 提取线和面特征的方法

### 提取线特征

- Split-and-merge: 通过迭代测试点的终点，计算点个数来拟合一个曲线。
- line-regression: 线回归的放啊，类似滑动窗口来框取一些感兴趣点，对这些点进行线回归。
- RANSAC: 随机抽样一致，假设数据由局内点组成，数据分布可以由某种模型来解释。噪声即使那些不能适应该模型的点。RANSAC的输入是1) 一组观测数据，一个可以解释或者应用于2) 观测数据的模型，以及一些3) 可信的参数。随后RANSAC通过反复选择数据中的一组随机自己来计算出适合该模型的参数以及对应的损失。当由足够多的点满足模型的假设，就认为这个模型是合理的。最后通过错误率评估，投票出一个最好的模型。
- 霍夫变换: 类似数字图像处理，将图像空间投影到霍夫参数空间，可以直观的用点表示各种直线。

	Complexity	Speed (Hz)	False positives	Precision
Split-and-Merge	$N \log N$	1500	10%	+++
Incremental	$SN$	600	6%	+++
Line-Regression	$NN_f$	400	10%	+++
RANSAC	$SNk$	30	30%	++++
Hough-Transform	$SNN_C + SN_R N_C$	10	30%	++++

- 最小二乘拟合

## SVD方法

基于SVD的机器人定位算法。SVD全称为奇异值分解。

### 问题描述

对于两组对应的点集  $P \in \mathbb{R}^{2 \times n}$ ，基于世界坐标系。以及  $Q \in \mathbb{R}^{2 \times n}$  基于激光坐标系

$$P = \{p_1, p_2, \dots, p_n\}$$

$$Q = \{q_1, q_2, \dots, q_n\}$$

为了计算出他们之间的刚体变换  $R, t$ ，可以将其建模为如下的数学形式

$$(R, t) = \arg \min \sum_{i=1}^n w_i \|(Rp_i + t) - q_i\|^2$$

### 利用最小二乘法求转移矩阵 $t$

优化目标可以变换为

$$\begin{aligned} \sum_{i=1}^n w_i \|(Rp_i + t) - q_i\|^2 &= \sum_{i=1}^n w_i \|Rp_i + \hat{q} - R\hat{p} - q_i\|^2 \\ &= \sum_{i=1}^n w_i \|R(p_i - \hat{p}) - (q_i - \hat{q})\|^2 \end{aligned} \quad (5)$$

我们通过令  $x_i := p_i - \hat{p}$ ,  $y_i := q_i - \hat{q}$  来进行去中心化，更新后的优化目标等于

$$R = \arg \min \sum_{i=1}^n w_i \|Rx_i - y_i\|^2$$

经过一系列的推导可以转化为一个奇异值分解的问题，最后通过奇异值分解得到旋转矩阵和平移矩阵

给定两个在 2 维空间中对应的点集合  $P = \{p_1, p_2, \dots, p_n\}$  和  $Q = \{q_1, q_2, \dots, q_n\}$ :

- 加权平均:

$$\hat{p} = \frac{\sum_{i=1}^n w_i p_i}{\sum_{i=1}^n w_i} \quad \hat{q} = \frac{\sum_{i=1}^n w_i q_i}{\sum_{i=1}^n w_i}$$

其中  $w_i$  为权重参数, 可根据激光测量距离取  $w_i = 1/\sigma_i(\rho_i)$ 。

- 去中心化:

$$x_i := p_i - \hat{p} \quad y_i := q_i - \hat{q}$$

构造矩阵  $W = \text{diag}(w_1, \dots, w_n)$ ,  $X = [x_1, x_2, \dots, x_n]$ ,  $Y = [y_1, y_2, \dots, y_n] \in R^{2 \times n}$ .

- 构造 SVD 矩阵: 令  $S = XWY^T = U\Sigma V^T$ , 由 SVD 原理, 求取  $U, V$  单位正交阵。
- 求取旋转矩阵  $R$ : 根据求取的  $U$  和  $V$  得:

$$R = VU^T$$

- 求取平移矩阵  $t$ :

$$t = \hat{q} - R\hat{p}$$

## ICP方法

ICP方法称为最近迭代点方法, 步骤为



### 1) 计算最近点集

- 选取全部采样点
- 均匀采样
- 随机采样
- 基于特征的采样
- 法向量空间采样

### 2) 最近点匹配算法

- 最近点匹配
- Normal Shooting Matching
- 点到面匹配
- 投影匹配
- KD树或者八叉树匹配

# 卡尔曼滤波

## Q: 什么是卡尔曼滤波?

卡尔曼滤波是一种利用线性系统的状态方程, 通过系统输入输出观测数据, 对系统状态进行最优估计的算法。由于观测数据中包括系统中的噪声和干扰的影响, 所以最优估计也可看作是滤波过程。

## 基于概率思路

### Q: 为什么采用概率的方法来解决移动机器人问题?

- 由于来自机器人传感器的数据受测量误差的影响, 我们只能计算当前机器人状态的可能性。
- 概率机器人技术的关键思想是使用概率论来表示不确定性, 而不是给出机器人位于当前位置的最佳估计。概率机器人技术中每一个模块都给出一个分布来表示置信度, 方便后续算法进行分析和融合。

基于概率思路的卡尔曼滤波核心是利用贝叶斯公式和概率求和与乘积的思路, 对多次独立测量进行加权和。权重称为卡尔曼增益。令  $q \in \mathbb{R}^n$  表示系统状态,  $p_1(q), p_2(q)$  分别表示两次独立测量的结果和型人都, 假设这两次测量服从正态分布

$$p_1(q) = N(\hat{q}_1, P) \quad p_2(q) = N(\hat{q}_2, Q)$$

其中  $Q, P$  分别表示两次测量的协方差矩阵。有全概率定理可以知道, 两次测量之后最终分布也服从正态分布。同时由标量融合的算法可以得到总体的二分部  $p(q) = N(\hat{q}, R)$

$$\begin{aligned} \hat{q} &= \hat{q}_1 + P(P + Q)^{-1}(\hat{q}_2 - \hat{q}_1) \iff \hat{q} = \hat{q}_1 + K(\hat{q}_2 - \hat{q}_1) \\ R &= P - P(P + Q)^{-1}P \iff R = P - K \cdot \Sigma_{IN} \cdot K^T \end{aligned}$$

其中

$$K = P(P + Q)^{-1}$$

为卡尔曼增益的更新公式, 可以看到他与两次测量的协方差都有关系, 而总协方差的更新公式为

$$\Sigma_{IN} = P + Q$$

## 基于优化思路

### 问题描述:

一线性离线系统可用如下方程描述:

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad (1)$$

$$z_k = Hx_k + v_k \quad (2)$$

其中  $x_k \in \mathbb{R}^n$  为系统状态,  $z_k \in \mathbb{R}^m$  为测量输出,  $u_k \in \mathbb{R}^l$  为系统输入。  $w_k \in \mathbb{R}^n$  为过程噪声 (Process Noise),  $v_k \in \mathbb{R}^m$  为测量噪声 (Measurement Noise) 为白噪音:

$$p(w) = N(0, Q) \quad p(v) = N(0, R) \quad (3)$$

其中,  $Q$  为过程噪声协方差阵,  $R$  为测量噪声协方差阵

所有优化问题我们都需要定义损失, 我们先定义先验和后验以及对应的估计

- 先验
  - 先验估计  $\hat{x}_k^- \in \mathbb{R}^n$  是指通过状态方程推测出来的次态。
  - 先验估计的误差表示为  $e_k^- = x_k - \hat{x}_k^-$

- 先验估计的误差协方差定义为  $P_k^- = E[e_k^- e_k^{-T}]$
- 后验
  - 后验表示结合先验和观测数据得到的结果，定义后验为  $\hat{x}$
  - 后验误差  $e_k = x_k - \hat{x}$
  - 后验误差协方差矩阵  $P_k = E[e_k e_k^T]$

后验和先验的关系为

$$\hat{x}_k = \hat{x}_k^- + K(z_k - H\hat{x}_k^-)$$

其中  $K$  表示增益矩阵，第二项括号内部分表示测量残差。站在优化角度下，我们的卡尔曼滤波也是优化问题

$$\arg \min_K P_k$$

即通过优化卡尔曼增益矩阵  $K$  来最小化后验误差

经过一系列推导后得到

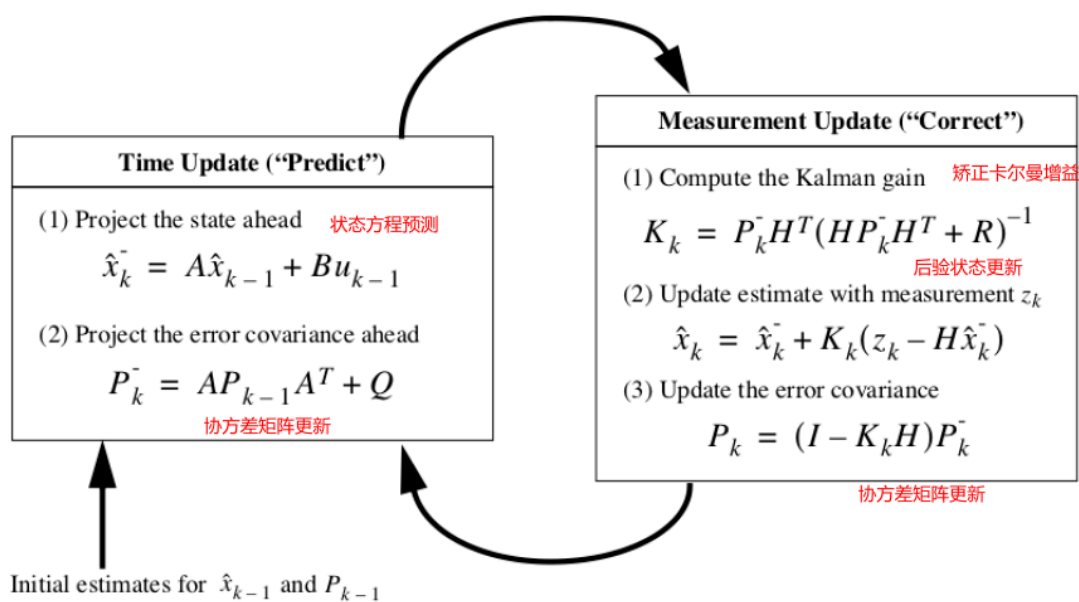
目标函数：

$$\arg \min_K P_k = (I - KH)P_k^-(I - KH)^T + K R K^T$$

可得最优的卡尔曼增益如下：

$$\frac{\partial P_k}{\partial K} = -2P_k^- H^T + 2K H P_k^- H^T + K R = 0 \Rightarrow K = P_k^- H^T (H P_k^- H^T + R)^{-1}$$

最终整个卡尔曼滤波框架可以分成“预测”和“矫正”两部分



Q:对卡尔曼滤波的核心思想理解，为什么可以将两个垃圾传感器融合更好

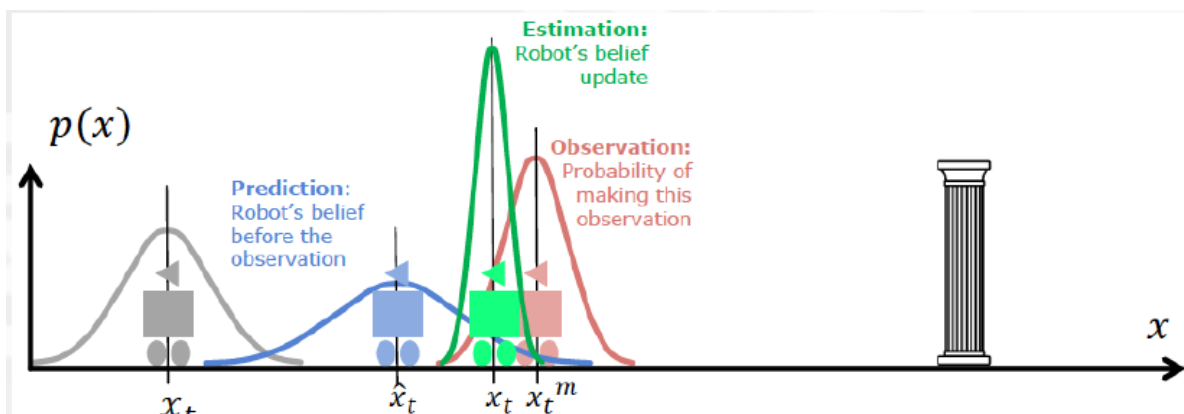
这个我没找到答案，但是我猜测的是不同传感器的噪声形式（体现在协方差矩阵上）不同，多个传感器经过卡尔曼滤波器可以起到互补的作用。

# 卡尔曼滤波定位

简短版：

【】

- 只能处理位置跟踪问题
- 严重依赖匹配精度，如果出现匹配错误，则机器人定位失败。
- 无法判断是否定位失败
- 午饭从定位失败中恢复
- 收敛速度收到初始状态误差以及协方差矩阵精度影响大。
- 无法进行全局定位，无法处理机器人绑架问题。



这里给出的一个例子可以看成是对里程计数据（蓝色）和雷达数据（观测，红色）进行加权平均，得到一个当前位姿的估计。

## 1 预测模型（先验）

：里程计模型（世界坐标系下）

$$\begin{cases} x_{k+1} = x_k + \Delta s \cos(\theta_k + \frac{\Delta \theta}{2}) \\ y_{k+1} = y_k + \Delta s \sin(\theta_k + \frac{\Delta \theta}{2}) \\ \theta_{k+1} = \theta_k + \Delta \theta \end{cases}$$

## 2 先验估计的误差模型（非线性的里程计模型）

② 先验估计误差模型：里程计误差传导（非线性）

$$\sum_{p'} = \nabla_p f \cdot \sum_p \cdot \nabla_p f^T + \nabla_{rl} f \cdot \sum_{\Delta} \cdot \nabla_{rl} f^T$$

其中：

$$F_p = \nabla_p f = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} & \frac{\partial f}{\partial \theta} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\Delta s \sin(\theta_k + \frac{\Delta \theta}{2}) \\ 0 & 1 & \Delta s \cos(\theta_k + \frac{\Delta \theta}{2}) \\ 0 & 0 & 1 \end{bmatrix}$$

$$F_{\Delta} = \nabla_{rl} f = \begin{bmatrix} \frac{\partial f}{\partial \Delta s} & \frac{\partial f}{\partial \Delta \theta} \end{bmatrix} = \begin{bmatrix} \frac{r}{2} \cos(\theta_k + \frac{\Delta \theta}{2}) - \frac{r}{4d} \Delta s \sin(\theta_k + \frac{\Delta \theta}{2}) & \frac{r}{2} \cos(\theta_k + \frac{\Delta \theta}{2}) + \frac{r}{4d} \Delta s \sin(\theta_k + \frac{\Delta \theta}{2}) \\ \frac{r}{2} \sin(\theta_k + \frac{\Delta \theta}{2}) + \frac{r}{4d} \Delta s \cos(\theta_k + \frac{\Delta \theta}{2}) & \frac{r}{2} \sin(\theta_k + \frac{\Delta \theta}{2}) - \frac{r}{4d} \Delta s \cos(\theta_k + \frac{\Delta \theta}{2}) \end{bmatrix}$$

## 3 观测模型：激光ICP方法确定匹配关系，SVD方法确定机器位姿

③ 观测模型：激光定位算法（ICP：确定匹配关系，SVD：确定机器人位姿）

给定两个在2维空间中对应的点集  $P = \{p_1, p_2, \dots, p_n\}$  和  $Q = \{q_1, q_2, \dots, q_n\}$ ：

• 加权平均：

$$\hat{p} = \frac{\sum_{i=1}^n w_i p_i}{\sum_{i=1}^n w_i} \quad \hat{q} = \frac{\sum_{i=1}^n w_i q_i}{\sum_{i=1}^n w_i}$$

其中  $w_i$  为权重参数，可根据激光测量距离取  $w_i = 1/\sigma_i(\rho_i)$ 。

• 去中心化：

$$x_i := p_i - \hat{p} \quad y_i := q_i - \hat{q}$$

构造矩阵  $W = \text{diag}(w_1, \dots, w_n)$ ,  $X = [x_1, x_2, \dots, x_n]$ ,  $Y = [y_1, y_2, \dots, y_n] \in R^{2 \times n}$ 。

• 构造SVD矩阵：令  $S = XWY^T = U\Sigma V^T$ ，由SVD原理，求取  $U$ ,  $V$  单位正交阵。

• 求取旋转矩阵  $R$ ： $R = VU^T$

• 求取平移矩阵  $t$ ： $t = \hat{q} - R\hat{p}$

## 4 观测误差模型：SVD定位误差模型

④ 观测误差模型：SVD 定位误差模型

最小的匹配误差为：

$$\arg \min \sum_{i=1}^n w_i \| (Rp_i + t) - q_i \|^2 = \sum_{i=1}^n w_i x_i^T x_i + \sum_{i=1}^n w_i y_i^T y_i - 2 \sum_{i=1}^n \sigma_i$$

其中，

- $w_i$  为权重参数，可根据激光测量距离取  $w_i = 1/\sigma_i(\rho_i)$ ，
- $\rho_i$  为激光测距方差，可对激光雷达离线标定获得。
- $x_i := p_i - \hat{p}$  为地图特征去中心化后的点。
- $y_i := q_i - \hat{q}$  为激光雷达坐标系下识别的特征去中心化后的点。

## 5 基于卡尔曼滤波定位算法流程

### 基于卡尔曼滤波定位算法流程 (5)——后验估计

⑤ 基于观测的后验估计（对先验估计的修正）：

• 卡尔曼增益  $K$ ：

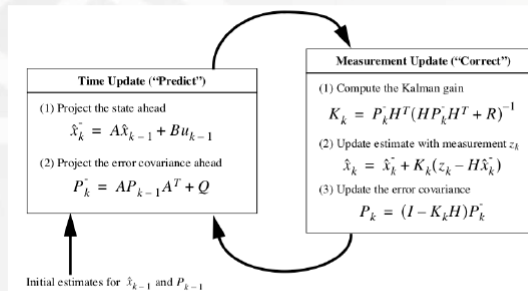
$$K = P_k^- H^T (H P_k^- H^T + R)^{-1}$$

• 后验估计  $\bar{x}_k$ ：

$$\bar{x}_k = \bar{x}_k^- + K (\bar{z}_k - H \bar{x}_k^-)$$

• 后验估计误差  $P_k$ ：

$$P_k = (I - KH) P_{k-1}^-$$



- 激光测量频率小于里程计测量频率，如何处理？
- 激光测量有可能仅能得到机器人部分状态，如何处理？
- 该方法也可融合诸如 IMU 等其他传感器信息，如何融合？

## 蒙特卡洛定位算法MCL

局部定位中使用ICP方法，容易陷入局部最优或存在匹配错误

收敛性收到卡尔曼滤波的制约：对初始状态的精度要求高。

相比而言，蒙特卡洛定位的原理为：不断抽样，逐渐逼近。

- 优点：非常通用，可以解决十分复杂的问题。
- 缺点：收敛速度慢，不够精确。

任意概率分布都可以通过样本（粒子）分布来秒速，某一区样本越多，则对应发生概率约高。可以有效处理非高斯分布等各种复杂分布。



## 蒙特卡洛定位

蒙特卡洛定位分为4个步骤

初始化粒子群；模拟粒子运动；计算粒子评分；粒子群重采样

- 初始化粒子群
  - 如果你有初始定位信息，你就像卡尔曼滤波一样初始化在机器人周围，这样用很少的粒子就能表达机器人的当前位姿。
- 模拟粒子运动
  - 类似卡尔曼滤波从的先验过程，通过里程计模型进行推演。
- 计算粒子评分
  - 类似卡尔曼滤波的观测过程，通过当前帧的观测数据对已有的粒子进行评分和筛选
- 粒子群重采样
  - 利用观测数据和筛选后的粒子，进行一次重新撒点（重采样）。

## 自适应蒙特卡洛定位

### ■ 自适应体现在：

- ✓ 解决了机器人绑架问题，它会在发现粒子们的平均分数突然降低了（意味着正确的粒子在某次迭代中被抛弃了）的时候，在全局再重新撒一些粒子。
- ✓ 解决了粒子数固定的问题，因为有时候当机器人定位差不多得到了时候，比如这些粒子都集中在一块了，还要维持这么多的粒子没必要，这个时候粒子数可以少一点了。

### ■ 解决方法：

- ✓ 采用KLD(Kullback-Leibler Divergence)采样：库尔贝克-莱布勒散度，它是一种计算两个概率分布之间差异的方法
- ✓ KLD采样核心思想就是根据基于采样近似质量的统计界限来确定粒子数量。在粒子滤波器的每次迭代过程中，KLD采样以概率 $1 - \sigma$ 来确定样本数量，真实的后验概率与基于采样的近似分布之间的差异小于 $\epsilon$ 。对应于实际问题，就是在栅格地图中，看粒子占了多少栅格。占得多，说明粒子很分散，在每次迭代重采样的时候，允许粒子数量的上限高一些。占得少，说明粒子都已经集中了，那就将上限设低，采样到这个数就行了。

## 移动机器人地图的基本概念

常用的地图表示方法有

- 点云地图
  - 维护难度大，数据量大，不具有通用性，需要处理后才有地面信息，存在检测盲区和空洞
- 栅格地图
  - 对数几率模型更新栅格的占有状态。
  - 有点：可以详细描述环境信息，易于用于机器人定位和路径规划
  - 缺点：存储要求高
- 多分辨率地图
  - 八叉树等等，存储存在稀疏性，计算复杂度高，但是存储效率高。
- 特征地图
  - 连续先行地图，连续多边形地图，空间站有效率高，定位精度高
- 拓扑地图
  - 例如地铁公交地图，可以方便的进行导航和路径规划
  - 信息比较抽象，难以用于移动机器人精确可靠定位。



# 同步定位见图算法SLAM

- 定位
  - 根据环境地图和机器人运动序列、观测序列来确定机器人在地图中的位置和姿态。
  - 误差来源：全局定位的初始误差，局部定位的观测噪声
  - 前提是地图已知。
- 建图
  - 根据观测序列和机器人位姿序列来确定环境星系
  - 误差来源是观测噪声
  - 前提是机器人的位姿已知
- SLAM
  - 根据机器人的运动序列和观测序列，同时估计机器人的位姿和环境地图。
  - 数据关联是SLAM实现收敛的关键，闭环检测可以有效降低估计的不确定性。
  - SLAM分类
    - 基于粒子滤波：FastSLAM
    - 基于卡尔曼滤波：EKFSLAM
    - 基于优化：g2O

## 基于滤波方法的SLAM

### EKF-SLAM

- 所有路标之间为全连接
- 计算复杂度高，无法在大场景下使用
- 要求高斯分布，因此只能处理单峰假设
- 一旦数据关联错误就会大致发散
- 仅适用于人工路标情况
- 从机器人坐标系转到世界坐标系
- 只能用点云地图或特征地图，无法用栅格地图。

### Fast-SLAM

Fast-SLAM是粒子滤波器和EKF集成的，思想

用改进的粒子滤波器估计机器人路径的后验分布，机器人的位姿是用粒子分布来表示。粒子滤波器的每个粒子代表的一条可能运动路径，利用观测信息计算每个粒子的权重，以评价每条路机器人径的好坏。

- 路标用EKF来估计，即每个粒子拥有N个EKF<sub>s</sub>（2维），用于路径估计条件下的N个路标位置估计。
- 条件独立性：在机器人路径已知的情况下，路标位置的估计相对独立。

- 利用RBPL将SLAM后验估计问题进行降维, 提升算法的效率
- 利用粒子采样进行机器人位姿和路径估计
- 利用给定的粒子路径计算特征位置
- 基于随机采样的数据关联具有更高的鲁棒性
- 利用多假设分析可以在数据关联时忽略将机器人位姿误差的影响
- 粒子数量的选取
  - 内存利用率
  - 粒子枯竭——定位与建图错误

## 基于图优化的SLAM【看ppt吧太多了】

图优化问题中

- 图中的每一个节点均代表一个机器人的位姿。
- 图中的两个节点间的边代表了节点间的空间约束。
- 基于构建图和图中节点之间的约束关系, 修正机器人位姿以最大程度满足约束。

基于图优化的算法就是前端(构建图)和后端(图优化)不断迭代的结果。一致性地图描述有助于通过减少搜索空间来确定新的约束。可以在线SLAM, 也可以做离线优化。

假设一个机器人初始起点在0处( $x_0$ ), 然后机器人向前移动, 通过编码器测得它向前移动了0.8m( $x_1$ ), 此时与机器人初始状态时观测到了相同环境信息, 根据机器人观测的周围环境信息, 可得从机器人状态 $x_0$ 处观测机器人状态 $x_1$ 的虚拟测量为1, 该观测的方差为0.4, 请问, 机器人位姿的最优状态

□ 节点:  $x_0 = 0, x_1 = 0.8$ , 边:  $z_{12} = 1$  边信息矩阵 $\Omega_{12} = 1 \div 0.4 = 2.5$

□ 计算误差:

$$e_{12} = z_{12} - (x_1 - x_0) = 0.2$$

□ 计算雅可比:

$$J_{12} = (1 \quad -1)$$

□ 更新对应的系数向量 $\mathbf{b}_{ij}$ :

$$\mathbf{b}_{12}^T = e_{12}^T \Omega_{12} J_{12} = (0.5 \quad -0.5)$$

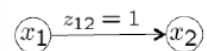
□ 更新对应的系数矩阵 $\mathbf{H}_{ij}$ :

$$\mathbf{H}_{12} = J_{12}^T \Omega_{12} J_{12} = \begin{pmatrix} 2.5 & -2.5 \\ -2.5 & 2.5 \end{pmatrix}$$

$$\det(\mathbf{H}_{12}) = 0?$$

□ 更新节点x状态:

$$\Delta x = -\mathbf{H}_{12}^{-1} \mathbf{b}_{12} \Rightarrow x_{k+1} = x_k + \Delta x = -\mathbf{H}_{12}^{-1} \mathbf{b}_{12}$$



## 6 运动规划

运动规划是一个过程, 根据所给定的地图, 寻找从起始状态到目标状态的运动序列。运动规划常常需要在运动受到约束的条件下找到最优解。

分层规划思想

