

基于已知位姿的构图算法 (Grid-based)

主讲人 曾书格

越凡创新技术负责人
电子科技大学硕士





建图算法



1. 地图分类



2. 覆盖栅格建图算法



3. 计数(Count Model)建图算法



4. TSDF建图算法



建图算法



1. 地图分类



2. 覆盖栅格建图算法



3. 计数(Count Model)建图算法



4. TSDF建图算法



地图分类



概念

- 地图即为环境的空间模型
- 环境地图是机器人进行定位和规划的前提
- 地图主要分为三类：



尺度地图



拓扑地图



语义地图



建图算法



1. 地图分类



2. 覆盖栅格建图算法



3. 计数(Count Model)建图算法



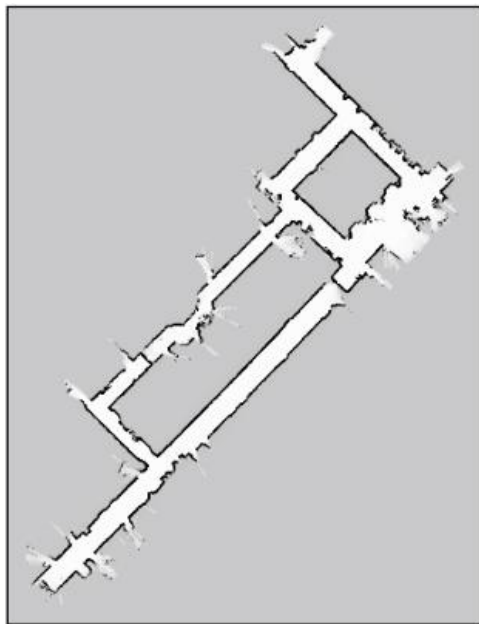
4. TSDF建图算法



覆盖栅格建图算法



栅格地图的特点



尺度地图

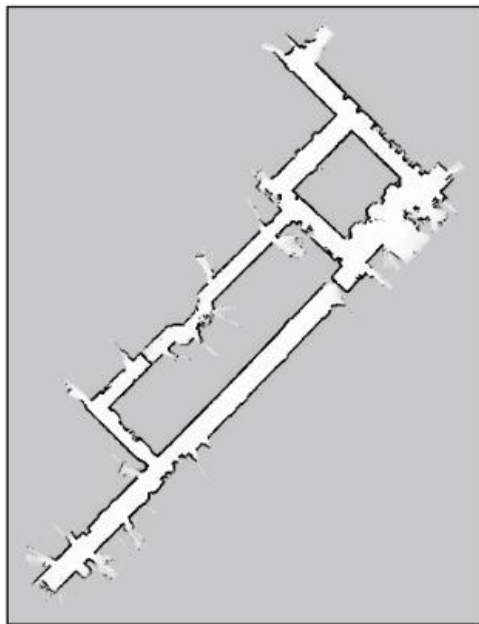
- 把环境分解成一个小一个小的栅格
- 每个栅格有两种状态：占用(Occupied)或者空闲(free)
- 非参模型
- 随着地图的增大，内存需求急剧增加
- 天然区分可通行区域，适合进行轨迹规划



覆盖栅格建图算法



构建栅格地图



尺度地图



数学描述

- 给定机器人的位姿和传感器的观测数据 (主要指激光雷达)。

$$data = \{x_1, z_1, x_2, z_2, \dots, x_t, z_t\}$$

- 估计出最可能的地图

$$m^* = \arg \max_m P(m|data)$$

↓

$$m^* = \arg \max_m P(m|x_{1:t}, z_{1:t})$$



覆盖栅格建图算法



假设

- 栅格地图中的栅格 m_i 是一个二值随机变量, 只能取两个值: 占用(Occupied)或者空闲(Free)

$p(m_i) = 1$ 表示被占用,

$p(m_i) = 0$ 表示空闲。

- 考虑建图的过程中环境不会发生改变的情况, 假设地图中的每一个栅格都是独立的, 因此数学表达式可以表示为:

$$p(m) = \prod p(m_i)$$

- 地图估计问题表示为:

$$p(m|x_{1:t}, z_{1:t}) = \prod p(m_i|x_{1:t}, z_{1:t}) \quad (1)$$

$x_{1:t}$ 表示第1到 t 时刻机器人的位姿;

$z_{1:t}$ 表示第1到 t 时刻的激光数据;

m_i 表示第 i 个栅格地图。

因此, 估计环境的地图只需要对每一个独立的栅格进行估计即可。



- 对于 $p(m_i|x_{1:t}, z_{1:t})$ ，根据贝叶斯公式可得：

$$p(m_i|x_{1:t}, z_{1:t}) = \frac{p(z_t|m_i, z_{1:t-1}, x_{1:t})p(m_i|z_{1:t-1}, x_{1:t})}{p(z_t|z_{1:t-1}, x_{1:t})} \quad (2)$$

由于每帧激光数据都是相互独立的，所以 $p(z_t|m_i, z_{1:t-1}, x_{1:t}) = p(z_t|m_i, x_t)$ ，此外每个栅格 m_i 与 x_t ， z_t 同时相关，单与其中某一项不相关。所以可化简得：

$$p(m_i|x_{1:t}, z_{1:t}) = \frac{p(z_t|m_i, x_t)p(m_i|z_{1:t-1}, x_{1:t-1})}{p(z_t|x_t)} \quad (3)$$

其中， $p(z_t|m_i, x_t)$ 可通过贝叶斯公式得：

$$p(z_t|m_i, x_t) = \frac{p(m_i|z_t, x_t)p(z_t|x_t)}{p(m_i|x_t)} \quad (4)$$



覆盖栅格建图算法



地图估计

- 根据公式(3)与公式(4)可得:

$$\begin{aligned} p(m_i | x_{1:t}, z_{1:t}) &= \frac{p(m_i | z_t, x_t) p(z_t | x_t)}{p(m_i | x_t)} \frac{p(m_i | z_{1:t-1}, x_{1:t-1})}{p(z_t | x_t)} \\ &= \frac{p(m_i | z_t, x_t) p(m_i | z_{1:t-1}, x_{1:t-1})}{p(m_i)} \end{aligned} \quad (5)$$

- 同理, 对于 $\neg m_i$:

$$\begin{aligned} p(\neg m_i | x_{1:t}, z_{1:t}) &= \frac{p(\neg m_i | z_t, x_t) p(z_t | x_t)}{p(\neg m_i | x_t)} \frac{p(\neg m_i | z_{1:t-1}, x_{1:t-1})}{p(z_t | x_t)} \\ &= \frac{p(\neg m_i | z_t, x_t) p(\neg m_i | z_{1:t-1}, x_{1:t-1})}{p(\neg m_i)} \end{aligned} \quad (6)$$



- 两者之比：

$$\begin{aligned}\frac{p(m_i|x_{1:t}, z_{1:t})}{p(\neg m_i|x_{1:t}, z_{1:t})} &= \frac{p(\neg m_i)}{p(m_i)} \frac{p(m_i|z_t, x_t)p(m_i|z_{1:t-1}, x_{1:t-1})}{p(\neg m_i|z_t, x_t)p(\neg m_i|z_{1:t-1}, x_{1:t-1})} \\ &= \frac{p(m_i|z_t, x_t)}{p(\neg m_i|z_t, x_t)} \frac{p(m_i|z_{1:t-1}, x_{1:t-1})}{p(\neg m_i|z_{1:t-1}, x_{1:t-1})} \frac{p(\neg m_i)}{p(m_i)}\end{aligned}$$

由于 $p(m_i)$ 非0即1，因此可得：

$$\begin{aligned}\frac{p(m_i|x_{1:t}, z_{1:t})}{p(\neg m_i|x_{1:t}, z_{1:t})} &= \frac{p(m_i|x_{1:t}, z_{1:t})}{1 - p(m_i|x_{1:t}, z_{1:t})} \\ &= \frac{p(m_i|z_t, x_t)}{1 - p(m_i|z_t, x_t)} \frac{p(m_i|z_{1:t-1}, x_{1:t-1})}{1 - p(m_i|z_{1:t-1}, x_{1:t-1})} \frac{1 - p(m_i)}{p(m_i)}\end{aligned}$$



覆盖栅格建图算法



地图估计

$$\frac{p(m_i|x_{1:t}, z_{1:t})}{p(\neg m_i|x_{1:t}, z_{1:t})} = \frac{p(m_i|z_t, x_t)}{1 - p(m_i|z_t, x_t)} \frac{p(m_i|z_{1:t-1}, x_{1:t-1})}{1 - p(m_i|z_{1:t-1}, x_{1:t-1})} \frac{1 - p(m_i)}{p(m_i)} \quad (7)$$

- 对于 $p(x)$, 定义对应的Log-Odd项: $l(x) = \log \frac{p(x)}{1 - p(x)}$

则: $p(x) = 1 - \frac{1}{1 + \exp(l(x))}$

- 将公式(7)两边同时取log, 可得:

$$\begin{aligned} & \log \frac{p(m_i|x_{1:t}, z_{1:t})}{p(\neg m_i|x_{1:t}, z_{1:t})} \\ &= \log \frac{p(m_i|z_t, x_t)}{1 - p(m_i|z_t, x_t)} + \log \frac{p(m_i|z_{1:t-1}, x_{1:t-1})}{1 - p(m_i|z_{1:t-1}, x_{1:t-1})} + \log \frac{1 - p(m_i)}{p(m_i)} \end{aligned}$$



覆盖栅格建图算法



地图估计

$$\log \frac{p(m_i|x_{1:t}, z_{1:t})}{p(\neg m_i|x_{1:t}, z_{1:t})} = \log \frac{p(m_i|z_t, x_t)}{1 - p(m_i|z_t, x_t)} + \log \frac{p(m_i|z_{1:t-1}, x_{1:t-1})}{1 - p(m_i|z_{1:t-1}, x_{1:t-1})} + \log \frac{1 - p(m_i)}{p(m_i)}$$

根据 $l(x)$ 的定义，上式可变成：

$$l(m_i|x_{1:t}, z_{1:t}) = l(m_i|z_t, x_t) + l(m_i|z_{1:t-1}, x_{1:t-1}) - l(m_i) \quad (8)$$

- $l(m_i|x_t, z_t)$ 表示激光雷达的逆观测模型(inverse measurement Model)
- $l(m_i|x_{1:t-1}, z_{1:t-1})$ 表示栅格 m_i 在 $t-1$ 时刻的状态
- $l(m_i)$ 表示栅格 m_i 的先验值，该值对所有栅格都相同



覆盖栅格建图算法



算法流程

occupancy_grid_mapping($\{l_{t-1,i}\}, x_t, z_t$):

```
1:   for all cells  $m_i$  do
2:       if  $m_i$  in perceptual field of  $z_t$  then
3:            $l_{t,i} = l_{t-1,i} + \text{inv\_sensor\_model}(m_i, x_t, z_t) - l_0$ 
4:       else
5:            $l_{t,i} = l_{t-1,i}$ 
6:       endif
7:   endfor
8:   return  $\{l_{t,i}\}$ 
```

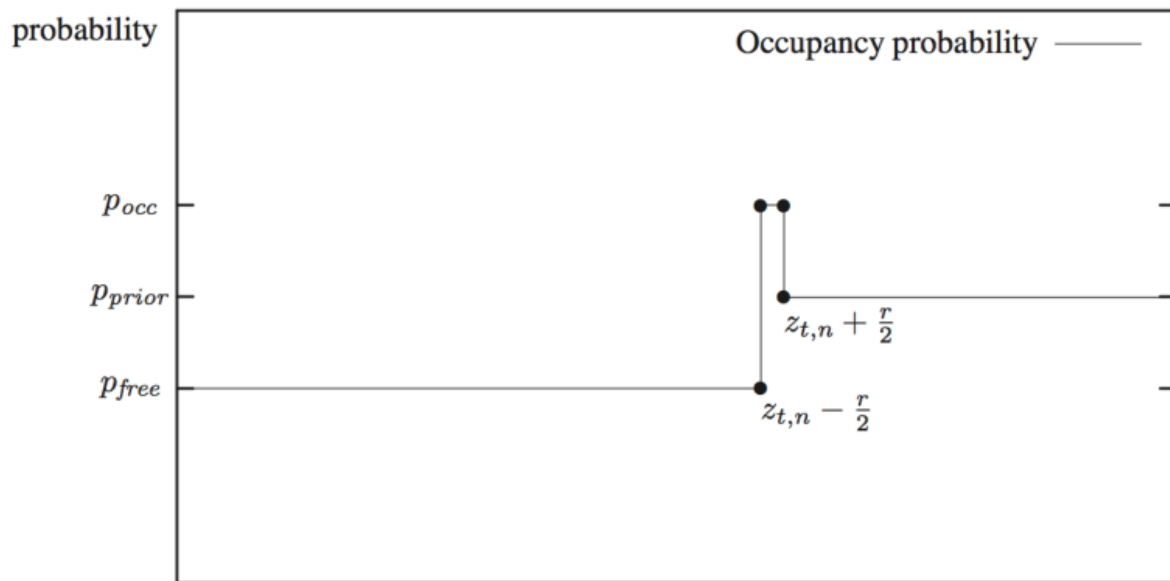
- 该算法对某一个栅格进行操作的时候，只有加法操作，因此具有非常快的更新速度。
- 更新的时候，需要知道传感器的逆测量模型。



覆盖栅格建图算法



激光雷达的逆观测模型



- 经过的栅格都为Free
- 击中的栅格为Occupied
- 其余栅格为Unknown



建图算法



1. 地图分类



2. 覆盖栅格建图算法



3. 计数(Count Model)建图算法



4. TSDF建图算法



计数建图算法



概念

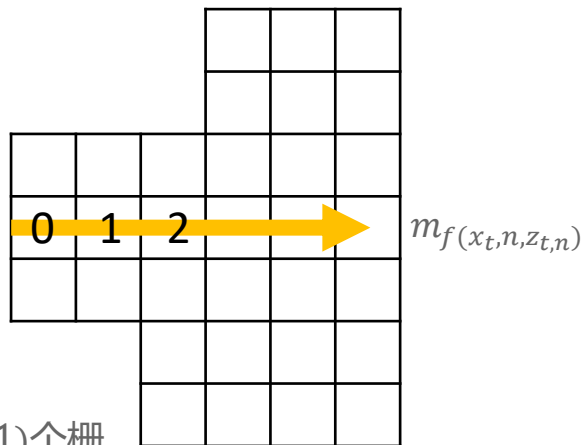
- 对于每一个栅格统计两个量: $misses(i)$ 和 $hits(i)$,
 $misses(i)$ 表示栅格 i 被激光束通过的次数, 即被标为free的次数;
 $hits(i)$ 表示栅格 i 被激光束击中的次数, 即被标为occupied的次数。
- 当 $hits(i) / (misses(i) + hits(i))$ 超过阈值则认为该栅格为Occupied, 否则认为栅格是Free的。
- $hits(i) / (misses(i) + hits(i))$ 表示栅格 i 的极大似然估计。



计数建图算法



观测模型



- t 时刻的机器人位姿为 x_t
- t 时刻的激光雷达数据为 z_t , 第 n 个激光束为 $z_{t,n}$ (经过第 $0 - (z_{t,n} - 1)$ 个栅格)
- $c_{t,n}$ 表示 t 时刻的第 n 个激光束是否为最大值(达到激光的测距极限, 比如说打在空气上或者光被吸收)。
 $c_{t,n} = 1$ 表示最大值, $c_{t,n} = 0$ 表示正常值
- $f(x_t, n, z_{t,n})$ 表示 t 时刻第 n 个激光束击中的栅格的下标, $m_{f(x_t, n, z_{t,n})}$ 表示对应的栅格的占用概率。

$$p(z_{t,n}|x_t, m) = \begin{cases} \prod_{k=0}^{z_{t,n}-1} (1 - m_{f(x_t, n, k)}) & c_{t,n} = 1 \\ m_{f(x_t, n, z_{t,n})} \prod_{k=0}^{z_{t,n}-1} (1 - m_{f(x_t, n, k)}) & c_{t,n} = 0 \end{cases}$$



计数建图算法



地图估计

- 地图估计的数学表达式:

$$m^* = \arg \max_m P(m|x_{1:t}, z_{1:t})$$

等价于:

$$\begin{aligned} m^* &= \arg \max_m P(\mathbf{z}_{1:t}|m, x_{1:t}) \\ &= \arg \max_m \prod P(z_t|m, x_t) \\ &\Leftrightarrow \arg \max_m \sum \ln P(z_t|m, x_t) \end{aligned}$$



计数建图算法



地图估计

$$m^* = \arg \max_m \sum \ln P(z_t | m, x_t)$$



$$m^* = \arg \max_m \sum_{j=0}^J \sum_{t=1}^T \sum_{n=1}^N \left(I(f(x_t, n, z_{t,n}) = j) \cdot (1 - c_{t,n}) \cdot \ln m_j + \sum_{k=0}^{z_{t,n}-1} I(f(x_t, n, z_{t,k}) = j) \cdot \ln(1 - m_j) \right)$$

$a_j = \sum_{t=1}^T \sum_{n=1}^N I(f(x_t, n, z_{t,n}) = j) \cdot (1 - c_{t,n})$ 表示栅格 j 被激光集中的次数, 即 hits(j)

$b_j = \sum_{t=1}^T \sum_{n=1}^N \sum_{k=0}^{z_{t,n}-1} I(f(x_t, n, z_{t,k}) = j)$ 表示栅格 j 被激光通过的次数, 即 misses(j)

则:

$$m^* = \arg \max_m \sum_{j=0}^J a_j \ln m_j + b_j \ln(1 - m_j)$$



计数建图算法



地图估计

- 目标函数：

$$m^* = \arg \max_m \sum_{j=0}^J a_j \ln m_j + b_j \ln(1 - m_j)$$

显然是关于 m_j 的函数，其极值可直接求对于 m_j 的导数，令其等于0即可：

$$\frac{\partial F(x)}{\partial m_j} = \frac{a_j}{m_j} - \frac{b_j}{1 - m_j} = 0$$

可得： $m_j = \frac{a_j}{a_j + b_j},$

其中， a_j 表示 $hits(j)$ ， b_j 表示 $misses(j)$



建图算法



1. 地图分类



2. 覆盖栅格建图算法



3. 计数(Count Model)建图算法



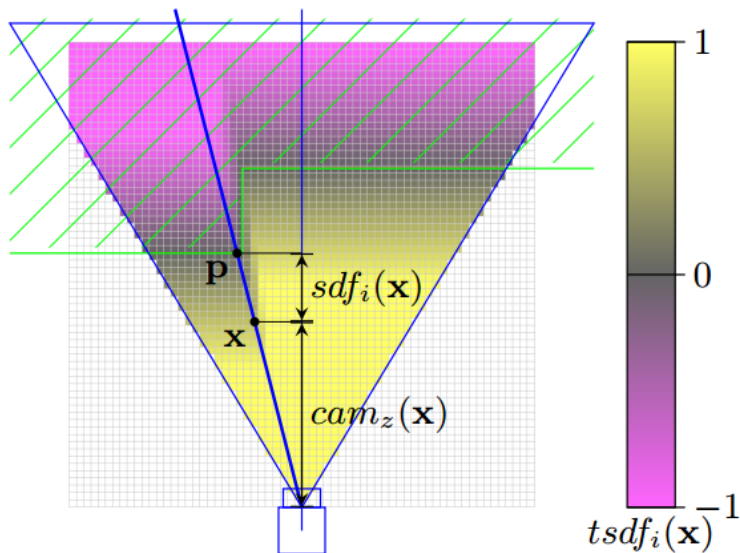
4. TSDF建图算法



TSDF建图算法



TSDF示意图



TSDF示意图



基本思想

- 充分考虑传感器测量的不确定性，利用多次测量数据来实现更精确的表面重构，从而得到更精确、更细、更薄的地图。
- 为障碍物表面建立Signed Distance Function。
- 距离表面较远的点忽略不计，因为不会影响到表面重构。



TSDf建图算法



数学描述

- $sdf(x)$ 的定义:

$$sdf_i(x) = laser_i(x) - dist_i(x)$$

$laser_i(x)$ 表示激光测量距离

$dist_i(x)$ 表示栅格离传感器原点的距离

- $tsdf(x)$ 的定义:

$$tsdf_i(\mathbf{x}) = \max(-1, \min(1, \frac{sdf_i(\mathbf{x})}{t}))$$

- 多次观测的融合更新方法:

$$TSDF_i(\mathbf{x}) = \frac{W_{i-1}(\mathbf{x})TSDF_{i-1}(\mathbf{x}) + w_i(\mathbf{x})tsdf_i(\mathbf{x})}{W_{i-1}(\mathbf{x}) + w_i(\mathbf{x})}$$

$$W_i(\mathbf{x}) = W_{i-1}(\mathbf{x}) + w_i(\mathbf{x})$$

- 不同的观测值不断按照上式进行融合，即可构造出整个地图的TSDf场，从地图的TSDf场中可以重构得到曲面。



TSDF建图算法



TSDF实例

- 假设机器人位置为(0,0)，障碍物的位置在(10,0)。
- 对障碍物进行了5次测量，测量值分别为9.9494 10.0178 9.9733 10.0068 9.9676。
- 分辨率为0.05，截断距离为0.1。
- 需要更新的栅格一共有5个，终点距离分别为：9.90,9.95,10.00,10.05,10.10。分别分cell1~cell5表示。

cell1	0.049376	0.117762	0.073328	0.106820	0.067608
cell2	0.00062	0.06776	0.02332	0.05682	0.017608
cell3	-0.0506	0.01776	-0.026675	0.00682	-0.03239
cell4	-0.100624	-0.032238	-0.076672	-0.043180	-0.082392
cell5	-0.150624	-0.082238	-0.126672	-0.093180	-0.132392



TSDF建图算法



TSDF实例

- 按照公式进行更新得：

cell1-9.90	cell2-9.95	cell3-10.00	cell4-10.05	cell5-10.10
0.082979	0.032979(b)	-0.010702(a)	-0.067021	-0.11702

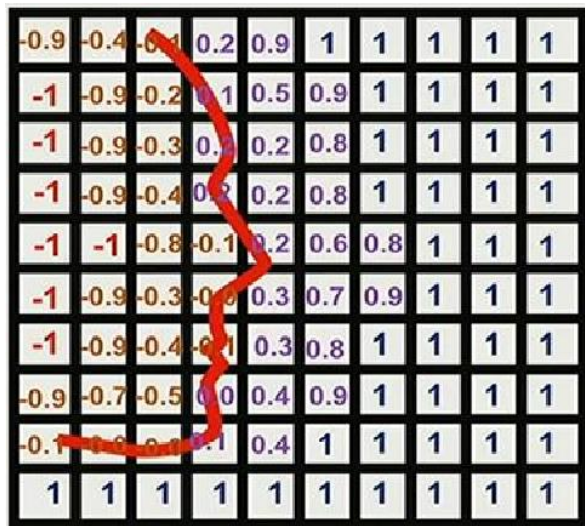
- 插值得到的表面位置： $x = 9.95 - 0.05*b/(a - b) = 9.9830$
- 原始数据直接进行平均得到位置为： $(9.9494+10.0178+9.9733+10.0068+9.9676)/5.0 = 9.9830$
- TSDF等价于加权最小二乘！！！！
- 因此如果传感器的噪声服从高斯分布，那么通过TSDF进行融合，等价于通过最小二乘来进行融合，能比较好的进行曲面重构。



TSDF建图算法



TSDF算法



TSDF场示意图

- 寻找TSDF场中，符号进行变化的栅格，符号进行变化的地方即是曲面的所在。
- 在两个栅格之间进行插值，插值得到值为0的坐标就是曲面的精确位置。
- 融合多帧观测，等价于用加权最小二乘方法来对多帧数据进行融合。
- 能插值出确切的曲面，构建的地图最多只有一个栅格的厚度。



参考资料

[1] Probabilistic Robotics

[2] Truncated Signed Distance Function-Experiments on Voxel Size



作业



详细见作业说明



结语

感谢聆听

Thanks for Listening

