NWPU 最优估计课程设计

刘振博

学号：2019201920

目录

1 贝叶斯滤波

1.1 贝叶斯滤波原理

$$bel\left(x_t\right)=p\left(x_t|z_{1:t},u_{1:t}\right) \tag{1}$$

贝叶斯公式:

$$=\frac{p\left(\boldsymbol{z}_t|\boldsymbol{x}_t,\boldsymbol{z}_{1:t-1},\boldsymbol{u}_{1:t}\right)p\left(\boldsymbol{x}_t|\boldsymbol{z}_{1:t-1},\boldsymbol{u}_{1:t}\right)}{p\left(\boldsymbol{z}_t|\boldsymbol{z}_{1:t-1},\boldsymbol{u}_{1:t}\right)} \tag{2}$$

$$=\eta p\left(z_t|x_t,z_{1:t-1},u_{1:t}\right)p\left(x_t|z_{1:t-1},u_{1:t}\right) \tag{3}$$

马尔可夫假设:

$$=\eta p\left(z_t|x_t\right)p\left(x_t|z_{1:t-1},u_{1:t}\right) \tag{4}$$

边缘概率公式:

$$=\eta p\left(z_t|x_t\right)\int_{x_{t-1}}p\left(x_t,x_{t-1}|z_{1:t-1},u_{1:t}\right)dx_{t-1} \tag{5}$$

全概率公式:

$$=\eta p\left(z_t|x_t\right)\int_{x_{t-1}}p\left(x_t|x_{t-1},z_{1:t-1},u_{1:t}\right)p\left(x_{t-1}|z_{1:t-1},u_{1:t}\right)dx_{t-1} \tag{6}$$

马尔可夫假设:

$$=\eta p\left(z_t|x_t\right)\int_{x_{t-1}}p\left(x_t|x_{t-1},u_t\right)p\left(x_{t-1}|z_{1:t-1},u_{1:t}\right)dx_{t-1} \tag{7}$$

马尔可夫假设:

$$=\eta p\left(z_t|x_t\right)\int_{x_{t-1}}p\left(x_t|x_{t-1},u_t\right)p\left(x_{t-1}|z_{1:t-1},u_{1:t-1}\right)dx_{t-1} \tag{8}$$

$$=\eta p\left(z_t|x_t\right)\int_{x_{t-1}}p\left(x_t|x_{t-1},u_t\right)\mathrm{bel}\left(x_{t-1}\right)dx_{t-1} \tag{9}$$

贝叶斯滤波器算法流程:

Algorithm Bayes filter(bel(x_{t-1}), u_t, z_t)

for all x_t do

$$\overline{bel}\left(x_t\right)=\int p\left(x_t|u_t,x_{t-1}\right)\ \mathrm{bel}(\ x_{t-1}\Big)\,dx_{t-1}$$

$$\text{bel }(x_t) = \eta p(z_t|x_t)\,\overline{\text{bel}}(x_t)$$

endfor

return bel (x_t)

1.2 卡尔曼滤波原理

状态方程与观测方程:

$$\boldsymbol{x}_t = \boldsymbol{A}_t\boldsymbol{x}_{t-1} + \boldsymbol{B}_t\boldsymbol{u}_t + \epsilon_t \tag{10}$$

$$\boldsymbol{z}_t = \boldsymbol{C}_t\boldsymbol{x}_t + \delta_t \tag{11}$$

$$\varepsilon_t \sim \mathcal{N}(0, Q_t) \tag{12}$$

$$\delta_t \sim \mathcal{N}(0, R_t) \tag{13}$$

根据上述线性方程得:

$$p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{u}_t) = \frac{1}{\sqrt{|2\pi\boldsymbol{R}_t|}} e^{-\frac{1}{2}(\boldsymbol{x}_t-\boldsymbol{A}_t\boldsymbol{x}_{t-1}-\boldsymbol{B}_t\boldsymbol{u}_t)^T\boldsymbol{R}_t^{-1}(\boldsymbol{x}_t-\boldsymbol{A}_t\boldsymbol{x}_{t-1}-\boldsymbol{B}_t\boldsymbol{u}_t)} \tag{14}$$

$$p(\boldsymbol{z}_t|\boldsymbol{x}_t) = \frac{1}{\sqrt{|2\pi\boldsymbol{Q}_t|}} e^{-\frac{1}{2}(z_t-\boldsymbol{C}_t\boldsymbol{x}_t)^T\boldsymbol{Q}_t^{-1}(z_t-\boldsymbol{C}_t\boldsymbol{x}_t)} \tag{15}$$

求 $\overline{bel}(\boldsymbol{x}_t)$**:**

$$\overline{bel}(\boldsymbol{x}_t) = \int p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)\ \text{bel }(\boldsymbol{x}_{t-1})\,d\boldsymbol{x}_{t-1} \tag{16}$$

$$= \int \frac{1}{\sqrt{|2\pi R_t|}} e^{-\frac{1}{2}(x_t-A_tx_{t-1}-B_tu_t)^TR_t^{-1}(x_t-A_tx_{t-1}-B_tu_t)} \frac{1}{\sqrt{|2\pi\Sigma_{t-1}|}} e^{-\frac{1}{2}(x_{t-1}-\mu_{t-1})^T\Sigma_{t-1}^{-1}(x_{t-1}-\mu_{t-1})}dx_{t-1} \tag{17}$$

$$= \eta e^{-\frac{1}{2}\left(\boldsymbol{x}_t-\boldsymbol{B}_t\boldsymbol{u}_t-\boldsymbol{A}_t\boldsymbol{\mu}_{t-1}\right)^T\left(\boldsymbol{A}_t\boldsymbol{\Sigma}_{t-1}^{-1}\boldsymbol{A}_t^T+\boldsymbol{R}_t\right)^{-1}\left(\boldsymbol{x}_t-\boldsymbol{B}_t\boldsymbol{u}_t-\boldsymbol{A}_t\boldsymbol{\mu}_{t-1}\right)} \tag{18}$$

$\overline{bel}(\boldsymbol{x}_t)$ **满足如下高斯分布:**

$$\overline{\boldsymbol{\mu}}_t = \boldsymbol{A}_t\boldsymbol{\mu}_{t-1} + \boldsymbol{B}_t\boldsymbol{u}_t \tag{19}$$

$$\overline{\boldsymbol{\Sigma}}_t=\boldsymbol{A}_t\boldsymbol{\Sigma}_{t-1}^{-1}\boldsymbol{A}_t^T+\boldsymbol{R}_t \tag{20}$$

求 $bel\left(\boldsymbol{x}_t\right)$:

$$\operatorname{bel}\left(\boldsymbol{x}_t\right)=\eta\frac{1}{\sqrt{\left|2\pi\boldsymbol{Q}_t\right|}}e^{-\frac{1}{2}\left(z_t-C_t\boldsymbol{x}_t\right)^TQ_t^{-1}\left(z_t-C_tx_t\right)}e^{-\frac{1}{2}\left(x_t-\boldsymbol{B}_tu_t-\boldsymbol{A}_t\mu_{t-1}\right)^T\left(\boldsymbol{A}_t\Sigma_{t-1}^{-1}\boldsymbol{A}_t^T+\boldsymbol{R}_t\right)^{-1}\left(x_t-B_tu_t-A_t\mu_{t-1}\right)} \tag{21}$$

经过推导可以证明（21）也是高斯分布：

$$bel\left(\boldsymbol{x}_t\right)=\eta e^{-\frac{1}{2}\left(x_t-\mu_t\right)^T\Sigma_t^{-1}\left(x_t-\mu_t\right)} \tag{22}$$

$$\mu_t=\overline{\mu}_t+\boldsymbol{K_t}(z_t-\boldsymbol{C}_t\overline{\mu}_t) \tag{23}$$

$$\Sigma_t=\left(I-K_tC_t\right)\overline{\Sigma}_t \tag{24}$$

$$K_t=\overline{\Sigma}_tC_t^T\left(C_t\overline{\Sigma}_tC_t^T+Q_t\right)^{-1} \tag{25}$$

卡尔曼滤波只需要每次计算均值与方差即可!!

1.3 扩展卡尔曼滤波原理

当运动方程与观测方程不是线性方程时，可以使用泰勒展开，对其近似，这就是扩展卡尔曼滤波（EKF）的基本思想。

$$X_{t+1}=f_t\left(X_t,u_t\right)+\varepsilon_t\quad\varepsilon_t\sim\mathcal{N}\left(0,Q_t\right) \tag{26}$$

$$Z_t=h_t\left(X_t\right)+\delta_t\quad\delta_t\sim\mathcal{N}\left(0,R_t\right) \tag{27}$$

以一阶泰勒展开为例，对上述非线性模型线性化。

运动方程：当 x_t 取值为 μ_t 的一个邻域内时，可以得到：

$$f_t\left(x_t,u_t\right)\approx f_t\left(\mu_t,u_t\right)+\frac{\partial f_t\left(\mu_t,u_t\right)}{\partial x_t}\left(x_t-\mu_t\right) \tag{28}$$

$$=f_t\left(\mu_t,u_t\right)+F_t\left(x_t-\mu_t\right) \tag{29}$$

观测方程：当 x_t 取值为 μ_t 的一个邻域内时，可以得到：

$$h_t\left(x_t\right) \approx h_t\left(\mu_t\right)+\frac{\partial h_t\left(\mu_t\right)}{\partial x_t}\left(x_t-\mu_t\right) \tag{30}$$

$$=h_t\left(\mu_t\right)+H_t\left(x_t-\mu_t\right) \tag{31}$$

之后，就按照线性 KF 滤波的方法更新其均值和协方差。

2 EKF-SLAM 设计思路

2.1 SLAM 问题定义

同时定位与建图（SLAM）的本质是一个估计问题，它要求移动机器人利用传感器信息实时地对外界环境结构进行估计，并且估算出自己在这个环境中的位置，Smith 和 Cheeseman 在上个世纪首次将 EKF 估计方法应用到 SLAM。

以滤波为主的 SLAM 模型主要包括三个方程：

1）运动方程：它会增加机器人定位的不确定性

2）根据观测对路标初始化的方程：它根据观测信息，对新的状态量初始化。

3）路标状态对观测的投影方程：根据观测信息，对状态更新，纠正，减小不确定度。

2.2 EKF-SLAM 维护的数据地图

系统状态 x 是一个很大的向量，它包括机器人的状态和路标点的状态，

$$\mathbf{x} = \begin{bmatrix} \mathcal{R} \\ \mathcal{M} \end{bmatrix} = \begin{bmatrix} \mathcal{R} \\ \mathcal{L}_1 \\ \vdots \\ \mathcal{L}_n \end{bmatrix} \tag{32}$$

其中 $\mathcal{R}$ 是机器人状态，$\mathcal{M} = (\mathcal{L}_1 \ \ldots \ \mathcal{L}_n)$ 是 n 个当前已经观测过的路标点状态集合。

在 EKF 中，x 被认为服从高斯分布，所以，EKF-SLAM 的地图被建模为 x 的均值 $\overline{x}$ 与协方差 $\mathbf{P}$，

$$\overline{\mathbf{x}} = \begin{bmatrix} \overline{\mathcal{R}} \\ \overline{\mathcal{M}} \end{bmatrix} = \begin{bmatrix} \overline{\mathcal{R}} \\ \overline{\mathcal{L}}_1 \\ \vdots \\ \overline{\mathcal{L}}_n \end{bmatrix} \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_{\mathcal{R}\mathcal{R}} & \mathbf{P}_{\mathcal{R}\mathcal{M}} \\ \mathbf{P}_{\mathcal{M}\mathcal{R}} & \mathbf{P}_{\mathcal{M}\mathcal{M}} \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{\mathcal{R}\mathcal{R}} & \mathbf{P}_{\mathcal{R}\mathcal{L}_1} & \cdots & \mathbf{P}_{\mathcal{R}\mathcal{L}_n} \\ \mathbf{P}_{\mathcal{L}_1\mathcal{R}} & \mathbf{P}_{\mathcal{L}_1\mathcal{L}_1} & \cdots & \mathbf{P}_{\mathcal{L}_n}\mathcal{L}_n \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{\mathcal{L}_n\mathcal{R}} & \mathbf{P}_{\mathcal{L}_n\mathcal{L}_1} & \cdots & \mathbf{P}_{\mathcal{L}_n\mathcal{L}_n} \end{bmatrix} \tag{33}$$

因此 EKF-SLAM 的目标就是根据运动模型和观测模型及时更新地图量 $\{\overline{\mathbf{x}}, \mathbf{P}\}$

2.3 EKF-SLAM 算法实施

2.3.1 地图初始化

显而易见，在机器人开始之前是没有任何路标点的信息的，因此此时地图中只有机器人自己的状态信息，因此 $n = 0, x = \mathcal{R}$。SLAM 中经常把机器人初始位姿认为是地图的原点，其初

始协方差可以按实际情况设定，比如：

$$\overline{\mathbf{x}} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{P} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{34}$$

2.3.2 运动模型

在 EKF 中如果 x 是状态量，u 是控制输入，n 是噪声变量，那么我们可以得到一般的状态更新函数：

$$\mathbf{x} \leftarrow f(\mathbf{x}, \mathbf{u}, \mathbf{n}) \tag{35}$$

EKF 的预测步骤为：

$$\begin{aligned} \overline{\mathbf{x}} &\leftarrow f(\overline{\mathbf{x}}, \mathbf{u}, 0) \\ \mathbf{P} &\leftarrow \mathbf{F_x} \mathbf{P} \mathbf{F_x}^\top + \mathbf{F_n} \mathbf{N} \mathbf{F_n}^\top \end{aligned} \tag{36}$$

其中雅克比矩阵 $\mathbf{F_x} = \frac{\partial f(\overline{\mathbf{x}}, \mathbf{u})}{\partial \mathbf{x}}$，$\mathbf{F_x} = \frac{\partial f(\overline{\mathbf{x}}, \mathbf{u})}{\partial \mathbf{x}}$，$\mathbf{N}$ 是随机变量 n 的协方差。

但是在 EKF-SLAM 中，只有一部分状态 $\mathcal{R}$ 是随运动而改变的，其余所有路标状态不改变，所以 SLAM 的运动方程为：

$$\begin{aligned} \mathcal{R} &\leftarrow f_{\mathcal{R}}(\mathcal{R}, \mathbf{u}, \mathbf{n}) \\ \mathcal{M} &\leftarrow \mathcal{M} \end{aligned} \tag{37}$$

因此我们可以得到稀疏的雅克比矩阵：

$$\mathbf{F_x} = \begin{bmatrix} \frac{\partial f_{\mathcal{R}}}{\partial \mathcal{R}} & 0 \\ 0 & \mathbf{I} \end{bmatrix} \quad \mathbf{F_n} = \begin{bmatrix} \frac{\partial f_{\mathcal{R}}}{\partial \mathbf{n}} \\ 0 \end{bmatrix} \tag{38}$$

最终我们得到了用于运动模型的 EKF 稀疏预测公式

$$\overline{\mathcal{R}} \leftarrow f_{\mathcal{R}}(\overline{\mathcal{R}}, \mathbf{u}, 0) \tag{39}$$

$$\mathbf{P}_{\mathcal{R}\mathcal{R}} \leftarrow \frac{\partial f_{\mathcal{R}}^T}{\partial \mathcal{R}} \mathbf{P}_{\mathcal{R}\mathcal{R}} \frac{\partial f_{\mathcal{R}}}{\partial \mathcal{R}} + \frac{\partial f_{\mathcal{R}}}{\partial \mathbf{n}} \mathbf{N} \frac{\partial f_{\mathcal{R}}^T}{\partial \mathbf{n}} \tag{40}$$

$$\mathbf{P}_{\mathcal{R}\mathcal{M}} \leftarrow \frac{\partial f_{\mathcal{R}}}{\partial \mathcal{R}} \mathbf{P}_{\mathcal{R}\mathcal{M}} \tag{41}$$

$$\mathbf{P}_{\mathcal{M}\mathcal{R}} \leftarrow \mathbf{P}_{\mathcal{R}\mathcal{M}}^\top \tag{42}$$

2.3.3 已经加入地图的状态量观测更新

在 EKF 中，我们有以下一般的观测方程

$$\mathbf{y} = h(\mathbf{x}) + \mathbf{v} \tag{43}$$

其中 $\mathbf{y}$ 是测量噪声，$\mathbf{x}$ 是全状态，$h()$ 是观测函数，v 是测量噪声。

典型的 EKF 观测更新为：

$$\overline{\mathbf{z}} = \mathbf{y} - h(\overline{\mathbf{x}}) \tag{44}$$

$$\mathbf{Z} = \mathbf{H}_{\mathbf{x}}\mathbf{P}\mathbf{H}_{\mathbf{x}}^{\top} + \mathbf{R} \tag{45}$$

$$\mathbf{K} = \mathbf{P}\mathbf{H}_{\mathbf{x}}^{\top}\mathbf{Z}^{-1} \tag{46}$$

$$\overline{\mathbf{x}} \leftarrow \overline{\mathbf{x}} + \mathbf{K}\overline{\mathbf{z}} \tag{47}$$

$$\mathbf{P} \leftarrow \mathbf{P} - \mathbf{K}\mathbf{Z}\mathbf{K}^{\top} \tag{48}$$

雅克比矩阵 $\mathbf{H}_{\mathbf{x}} = \frac{\partial h(\overline{\mathbf{x}})}{\partial \mathbf{x}}$，$\mathbf{R}$ 是测量噪声的协方差矩阵。

在 SLAM 中，观测指的是机器人上的传感器观测到某些路标点，并对路标点进行参数化的输出。每次可能对好多个路标点都有观测，这里依次对一个路标的观测结果进行状态更新。

观测的结果依赖于机器人的状态 $\mathcal{R}$, 传感器的状态 $\mathcal{S}$ 和路标点的状态 $\mathcal{L}_i$，并且这里假设，传感器的状态与机器人的状态差了一个固定的坐标变化。当观测到路标点 i 时，可以得到单一的观测函数：

$$\mathbf{y}_i = h_i\left(\mathcal{R}, \mathcal{S}, \mathcal{L}_i\right) + \mathbf{v} \tag{49}$$

这就是观测方程，它不依赖于除了 $\mathcal{L}_i$ 外的任何路标点状态。因此 EKF-SLAM 的雅克比 $\mathbf{H}_{\mathbf{x}}$ 也是稀疏的：

$$\mathbf{H}_{\mathbf{x}} = \left[\begin{array}{llllllll} \mathbf{H}_{\mathcal{R}} & 0 & \cdots & 0 & \mathbf{H}_{\mathcal{L}_i} & 0 & \cdots & 0 \end{array}\right] \tag{50}$$

其中 $\mathbf{H}_{\mathcal{R}} = \frac{\partial h_i\left(\overline{\mathcal{R}}, \mathcal{S}, \overline{\mathcal{L}}_i\right)}{\partial \mathcal{R}}$，$\mathbf{H}_{\mathcal{L}_i} = \frac{\partial h_i\left(\overline{\mathcal{R}}, \mathcal{S}, \overline{\mathcal{L}}_i\right)}{\partial \mathcal{L}_i}$。由于这里的稀疏性，EKF-SLAM 的观测更新变成：

$$\overline{\mathbf{z}} = \mathbf{y}_i - h_i\left(\overline{\mathcal{R}}, \mathcal{S}, \overline{\mathcal{L}}_i\right) \tag{51}$$

$$\mathbf{Z}=[\mathbf{H}_{\mathcal{R}}\mathbf{H}_{\mathcal{L}_i}]\begin{bmatrix}\mathbf{P}_{\mathcal{R}\mathcal{R}} & \mathbf{P}_{\mathcal{R}\mathcal{L}_i}\\ \mathbf{P}_{\mathcal{L}_i\mathcal{R}} & \mathbf{P}_{\mathcal{L}_i\mathcal{L}_i}\end{bmatrix}\begin{bmatrix}\mathbf{H}_{\mathcal{R}}^{\top}\\ \mathbf{H}_{\mathcal{L}_i}^{\top}\end{bmatrix}+\mathbf{R} \tag{52}$$

$$\mathbf{K}=\begin{bmatrix}\mathbf{P}_{\mathcal{R}\mathcal{R}} & \mathbf{P}_{\mathcal{R}\mathcal{L}_i}\\ \mathbf{P}_{\mathcal{M}\mathcal{R}} & \mathbf{P}_{\mathcal{M}\mathcal{L}_i}\end{bmatrix}\begin{bmatrix}\mathbf{H}_{\mathcal{R}}^{\top}\\ \mathbf{H}_{\mathcal{L}_i}^{\top}\end{bmatrix}\mathbf{Z}^{-1} \tag{53}$$

$$\overline{\mathbf{x}}\leftarrow\overline{\mathbf{x}}+\mathbf{K}\overline{\mathbf{z}} \tag{54}$$

$$\mathbf{P}\leftarrow\mathbf{P}-\mathbf{K}\mathbf{Z}\mathbf{K}^{\top} \tag{55}$$

2.3.4 观测方程可逆时的状态增广

这里的状态增广指的是新发现的路标点的初始化。当机器人发现了曾经未观测到的路标点时，会利用观测方程将新的路标状态加入地图，这一步操作会增大总状态向量的大小。可以看到 EKF-SLAM 中的滤波器大小动态变化的。

当传感器信息可以提供新发现路标点的所有自由度，也就是观测方程是双射时，只需要根据观测方程 $h()$ 的逆运算，即可以得到机器人状态 $\mathcal{R}$，传感器状态 $\mathcal{S}$，观测量 $\mathbf{y}_{n+1}$ 与新路标点状态的关系：

$$\mathcal{L}_{n+1}=g\left(\mathcal{R},\mathcal{S},\mathbf{y}_{n+1}\right) \tag{56}$$

上式是单个路标点的逆观测模型。

路标点的均值和雅克比：

$$\overline{\mathcal{L}}_{n+1}=g\left(\overline{\mathcal{R}},\mathcal{S},\mathbf{y}_{n+1}\right) \tag{57}$$

$$\mathbf{G}_{\mathcal{R}}=\frac{\partial g\left(\overline{\mathcal{R}},\mathcal{S},\mathbf{y}_{n+1}\right)}{\partial\mathcal{R}} \tag{58}$$

$$\mathbf{G}_{\mathbf{y}_{n+1}}=\frac{\partial g\left(\overline{\mathcal{R}},\mathcal{S},\mathbf{y}_{n+1}\right)}{\partial\mathbf{y}_{n+1}} \tag{59}$$

显然，新加路标点状态的协方差 $\mathbf{P}_{\mathcal{L}\mathcal{L}}$，以及该状态与地图其它状态的互协方差为：

$$\mathbf{P}_{\mathcal{L}\mathcal{L}}=\mathbf{G}_{\mathcal{R}}\mathbf{P}_{\mathcal{R}\mathcal{R}}\mathbf{G}_{\mathcal{R}}^{\top}+\mathbf{G}_{\mathbf{y}_{n+1}}\mathbf{R}\mathbf{G}_{\mathbf{y}_{n+1}}^{\top} \tag{60}$$

$$\mathbf{P}_{\mathcal{L}\mathbf{x}}=\mathbf{G}_{\mathcal{R}}\mathbf{P}_{\mathcal{R}\mathbf{x}}=\mathbf{G}_{\mathcal{R}}\left[\mathbf{P}_{\mathcal{R}\mathcal{R}}\mathbf{P}_{\mathcal{R}\mathcal{M}}\right] \tag{61}$$

然后将这些结果加入到地图中，可以得到总状态的均值与协方差矩阵：

$$\overline{\mathbf{x}} \leftarrow \begin{bmatrix} \overline{\mathbf{x}} \\ \overline{\mathcal{L}}_{n+1} \end{bmatrix} \tag{62}$$

$$\mathbf{P} \leftarrow \begin{bmatrix} \mathbf{P} & \mathbf{P}_{\mathcal{L}\mathbf{x}}^{\top} \\ \mathbf{P}_{\mathcal{L}\mathbf{x}} & \mathbf{P}_{\mathcal{L}\mathcal{L}} \end{bmatrix} \tag{63}$$

3 仿真实验

3.1 实验说明

3.1.1 传感器模型

传感器是一个 360 度的雷达，可以探测发现周围一定范围内的路标点及其类型，其探测半径为 r，其探测结果为 (ξ，s)。ξ 为路标点向量与当前雷达坐标系的夹角，s 为路标点与雷达局部坐标系原点的距离。

3.1.2 运动模型

将当前时刻雷达局部坐标系中的（u_1,0）点作为下一时刻雷达的位置：

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} u_1 \\ 0 \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} n_1 \\ 0 \end{bmatrix} \tag{64}$$

其方位:

$$\theta = \theta + u_2 + n_2 \tag{65}$$

3.1.3 观测模型

设观测到的路标点 i 的状态为 (${m_1}^i$，${m_2}^i$)，则其在当前雷达坐标系的坐标为:

$$\begin{bmatrix} {ladar_1}^i \\ {ladar_2}^i \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix}^{-1} \begin{bmatrix} {m_1}^i \\ {m_2}^i \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix} \tag{66}$$

则观测量为:

$$\xi = atan2\left({ladar_2}^i, {ladar_1}^i\right) + v_1 \tag{67}$$

$$s = \sqrt{\left({ladar_1}^i\right)^2 + \left({ladar_2}^i\right)^2} + v_2 \tag{68}$$

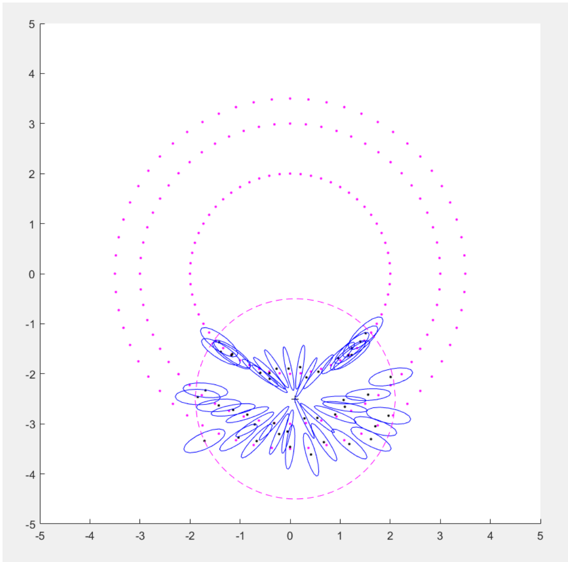
3.2 实验结果

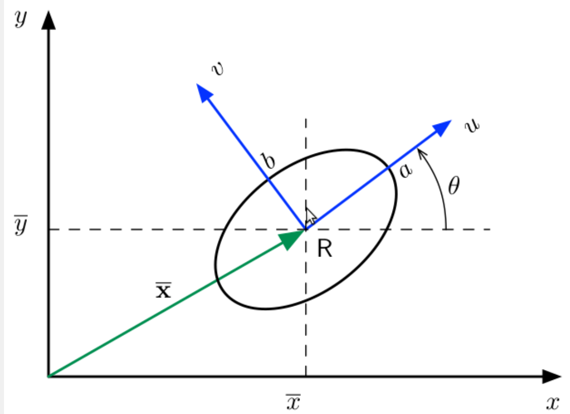
使用 MATLAB 编写程序进行仿真。

源代码 1：https://github.com/liuzhenboo/Optimal-Estimate-BigHomework

源代码 2：https://github.com/liuzhenboo/EKF-2D-SLAM

3.2.1 传感器模型，第一次状态增广





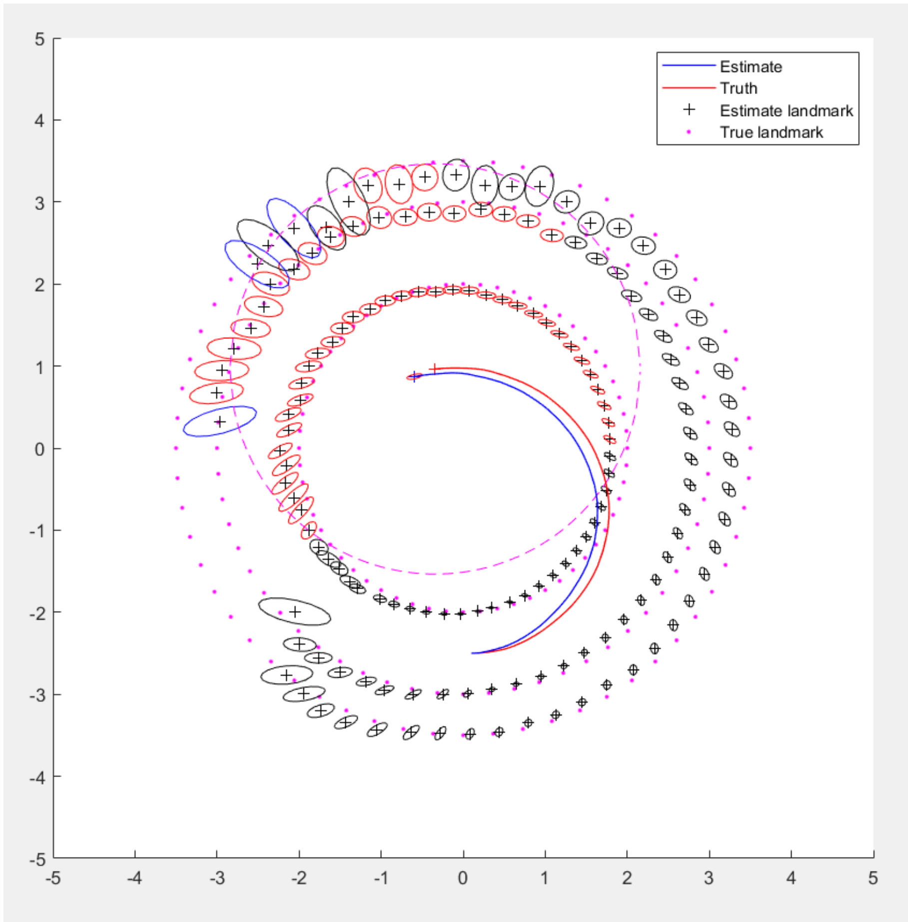
其中左图黑色的点表示滤波器新加入的路标点状态均值，绿色椭圆表征路标状态的协方差，其椭圆方程为：

$$(\mathbf{x}-\overline{\mathbf{x}})^{\top}\mathbf{P}^{-1}(\mathbf{x}-\overline{\mathbf{x}})= \text{ const} \tag{69}$$

其中 x 为路标状态,P 为路标的协方差。

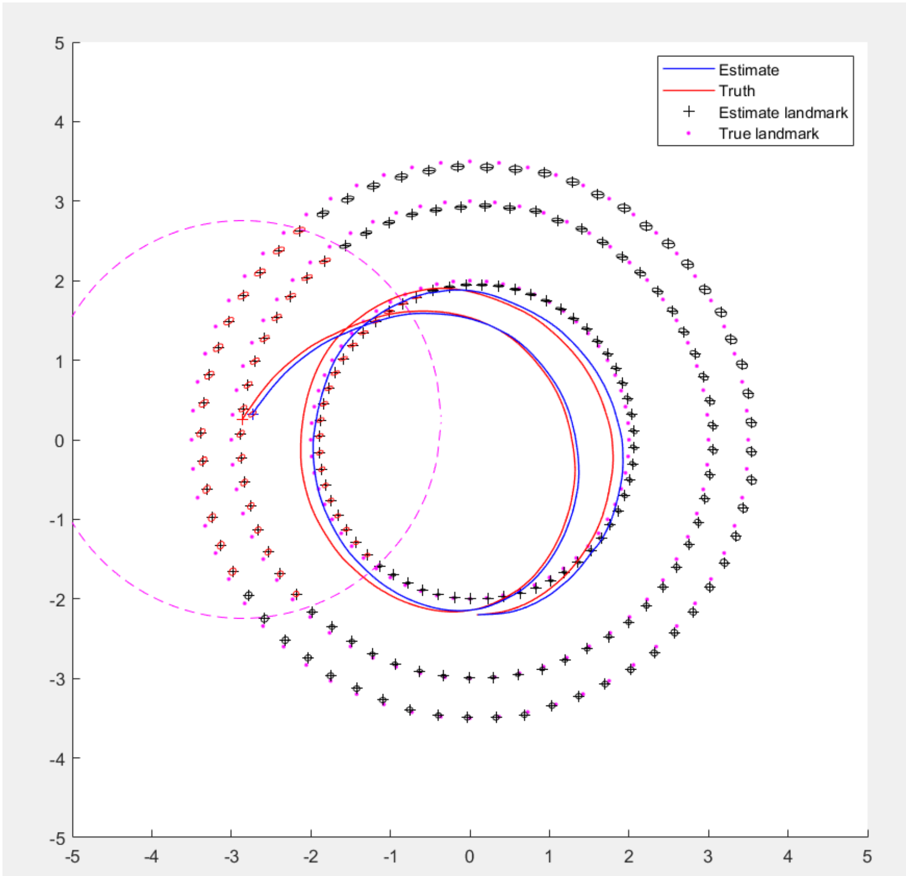
程序中是对 P 进行 SVD 分解，得到椭圆的方向 R 以及半轴到的长度，进行绘图。

3.2.2 状态增广，观测更新



如上图，黑色的协方差椭圆代表的路标点表示雷达曾经观测过它，但是当前没有观测到；红色的协方差椭圆代表的路标点表示雷达曾经观测过它，并且当前也观测到；蓝色代表当前刚刚初始化的新路标点。

3.2.3 状态不再增广



如上图，不再存在蓝色的协方差椭圆，代表状态增广停止，滤波器的大小不再改变。

4 附录

全概率公式：

$$p(x)=\sum_y p(x|y)p(y) \tag{70}$$

$$p(x)=\int p(x|y)p(y)\mathrm{d}y \tag{71}$$

条件概率公式：

$$p(x|y)=p(x,y)/p(y) \tag{72}$$

贝叶斯公式：

$$P(x|z)=\frac{P(z|x)P(x)}{P(z)}=\frac{\text{likelihood}\ \cdot\ \text{prior}}{\text{evidence}} \tag{73}$$

4.1 条件概率和联合概率

首先需要明确的是随机变量有随机性，其中随机性可以用一个概率分布来描述。$P(x|z)$ 指的是在随机变量 z 已经确定的情况下，继而推算出的 x 的概率，也即是 x 的概率分布。显然，$P(x|z)$ 是一个关于 x 和 z 的函数。

简单例子：

现有随机变量 x，y，z，其中 z，y 均服从 (0,1) 的高斯分布，且 z，y 两者不相互影响（即一方的取值不影响另一方）。x 满足如下关系：x = z + y，求 $P(x)$ 和 $P(x|z)$。

$P(x)$：此时根据 z，y 的概率分布可以求得 $P(x)$ 为 (0,2) 的高斯分布，也就是说除了 x 之外的随机变量会提供一个概率分布，我们通过随机变量的关系，由此来推出 x 的新的概率分布，这是一个仅仅关于 x 的函数。

$P(x|z)$：表示的是在随机变量 z 确定的情况下，随机变量 x 服从的概率分布 x = z(确定) + y。这时，随机变量 z 是一个确定的值，只有随机变量 y 为 x 传递概率分布信息。z 的确定代表 x 的不确定度下降，比如求 $P(x|z,y)$，当 z，y 确定时，x 也是定值，x 也就没有概率分布。所以 $P(x|z)$ 服从 (z,1) 的高斯分布，$P(x|z)$ 是关于 x，z 的函数。

4.2 联合概率

$P(y,z)$ 表示的是 y，z 的概率分布，定义为：$P(y,z)=P(y|z)P(z)$

4.3 边缘概率

当我们知道 P(y,z) 时，就可以利用 $\int P(y,z)dy$ 来求 P(z)