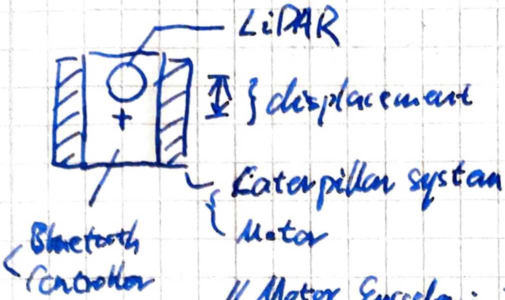
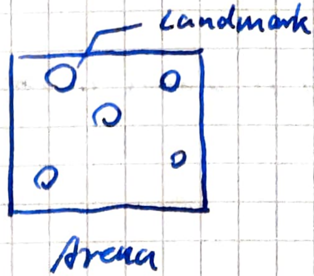
SLAM - Unit A Motion-Model

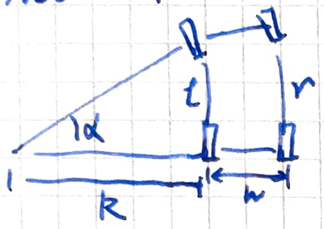
1. 小车与环境 Arena.





// Motor Encoder: 1 Tick = 0.349mm ，通过 Log-file 读取电机数据.

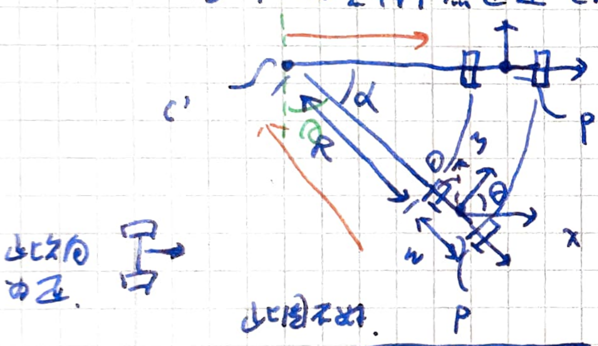
2. Model of Car



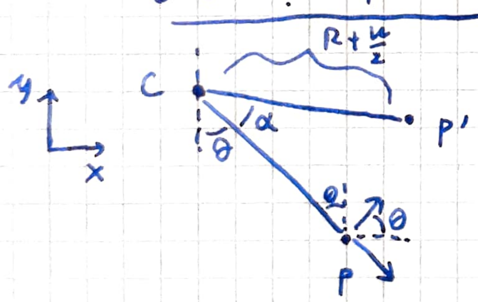
① 已知 l, r, w, 求 α, R

$$r = \alpha \cdot (R + w)$$
$$l = \alpha \cdot R$$
$$\alpha = \frac{r - l}{w}, \quad R = \frac{l}{\alpha} = \frac{l \cdot w}{r - l}$$

② 旧的坐标P点通过C点得到新的坐标P'



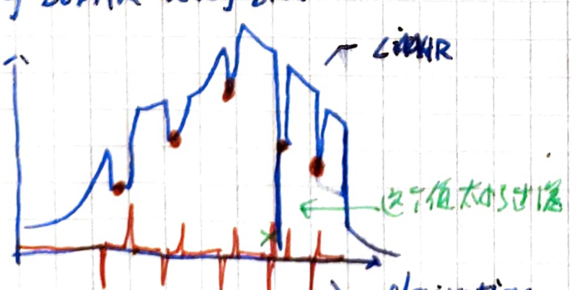
$$\begin{cases} \vec{c} = \vec{p} - (R + \frac{w}{2}) \cdot \begin{bmatrix} \sin\theta \\ -\cos\theta \end{bmatrix} \\ \vec{p'} = \vec{c} + (R + \frac{w}{2}) \cdot \begin{bmatrix} \sin(\theta + \alpha) \\ -\cos(\theta + \alpha) \end{bmatrix} \\ \vec{p'} = \vec{p} - (R + \frac{w}{2}) \cdot \begin{bmatrix} \sin\theta \\ -\cos\theta \end{bmatrix} + (R + \frac{w}{2}) \cdot \begin{bmatrix} \sin(\theta + \alpha) \\ -\cos(\theta + \alpha) \end{bmatrix} \end{cases}$$



$$\begin{cases} \vec{c} = \vec{p} - (R + \frac{w}{2}) \cdot \begin{bmatrix} \sin\theta \\ -\cos\theta \end{bmatrix} \\ \vec{p'} = \vec{c} + (R + \frac{w}{2}) \cdot \begin{bmatrix} \sin(\theta + \alpha) \\ -\cos(\theta + \alpha) \end{bmatrix} \\ \vec{p'} = \vec{p} + (R + \frac{w}{2}) \begin{bmatrix} \sin(\theta + \alpha) - \sin\theta \\ -\cos(\theta + \alpha) + \cos\theta \end{bmatrix} \end{cases}$$

Code: 1. 考虑 displacement
2. 改变 robot_width 修正轨迹.

3. 基于 LiDAR 探测 LM



Derivation: $f'(i) = \frac{f(i+1) - f(i-1)}{2}$

① 对 LiDAR 小于 20 的值过滤

② 对左 edge 和右 edge 计数，得到 LM 数量.

③ 左 edge 与右 edge 之间的数值求平均值 得到 LM 的位置，即光线的角度和深度.
angle range

④ 将 angle 与 range 转换
Cartesian_coordinates

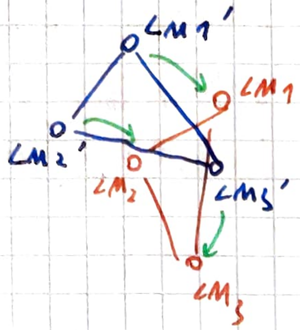
Q1 : Number of LM : 6

2 : log-tile : LegoLogfile

3: In Motion Model, θ is always in range of [0, 2pi]

Unit B Correktion, Featurebased / Featureless Approach, ICP

1. 通过 LM 估计与实际位置的差距, 建立相似变换公式



① 计算 LM cartesian koordinate

② match LM

③ 建立相似变换. $\lambda \cdot \vec{R} \cdot \vec{l_i} + \vec{t} = \vec{r_i}$

λ: Scale; $\vec{R}$: Rotation-Matrix $\underbrace{\begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix}}_{\text{nonlinear}}$; $\vec{t}$: translation $\begin{bmatrix} t_x \\ t_y \end{bmatrix}$

2. Least-Square - nonlinear 求解相似变换.

$$\min \sum_i \| \underbrace{\lambda \cdot \vec{R}}_{\text{nonlinear}} \cdot \vec{l_i} + \vec{t} - \vec{r_i} \|^2$$

// 不能用 pseudo-inverse 解决.

综: ① 计算 $\vec{l_m}$, $\vec{r_m}$ 2个点云中心
② 每个点 减去点云中心
③ 代入 Least-Square cost-function.
④. 化简得到 LS 方程最小的项.

// 点云中心

// $\vec{l_i'}$ $\vec{r_i'}$

① $\begin{cases} \vec{l_m} = \frac{1}{m} \sum_i \vec{l_i} \\ \vec{r_m} = \frac{1}{m} \sum_i \vec{l_i} \end{cases}$

② $\begin{cases} \vec{l_i'} = \vec{l_i} - \vec{l_m} \\ \vec{r_i'} = \vec{r_i} - \vec{r_m} \end{cases} \Rightarrow \begin{cases} \vec{l_i} = \vec{l_i'} + \vec{l_m} \\ \vec{r_i} = \vec{r_i'} + \vec{r_m} \end{cases}$

☆ 且 $\begin{cases} \sum_i \vec{l_i'} = 0 \\ \sum_i \vec{r_i'} = 0 \end{cases}$

// 利用的就是中心化之后点云之和为 0 的特性.

③ $\lambda \cdot \vec{R} \cdot \vec{l_i} + \vec{t} - \vec{r_i}$

$= \lambda \cdot \vec{R} \cdot (\vec{l_i'} + \vec{l_m}) + \vec{t} - (\vec{r_i'} + \vec{r_m})$

// 代入原 LS 方程 $\|\lambda \cdot \vec{R} \cdot \vec{l_i} + t - \vec{r_i}\|$

$= \lambda \cdot \vec{R}\, \vec{l_i'} - \vec{r_i'} + \underbrace{\lambda \cdot \vec{R} \cdot \vec{l_m} - \vec{r_m} + \vec{t}}_{\vec{t}'}$

$= \lambda \cdot \vec{R}\, \vec{l_i'} - \vec{r_i'} + \vec{t}'$

$\Rightarrow$ cost function: $\sum_i \| \lambda \cdot \vec{R}\, \vec{l_i} + \vec{t} - \vec{r_i} \|^2$

$= \sum_i \| \lambda \cdot \vec{R}\, \vec{l_i'} - \vec{r_i'} + \vec{t}' \|^2 = \sum_i \| \lambda \cdot \vec{R} \cdot \vec{l_i'} - \vec{r_i'} \|^2 + 2\vec{t}'^T \overbrace{\sum_i (\lambda \vec{R} \vec{l_i'} - \vec{r_i'})}^{0}$

$+ \sum_i \| \vec{t}' \|^2$ $\quad m \cdot \|t'\|^2$

≥ 0

$= \sum_i \| \lambda \cdot \vec{R} \cdot \vec{l_i'} - \vec{r_i'} \|^2 + \sum_i \| \vec{t}' \|^2$

≥ 0 // 消掉, 一共 m 个

// 得到 $\sum_i \| \lambda \cdot \vec{R} \cdot \vec{l_i'} - \vec{r_i'} \|^2 + \vec{t}'$

$$= \underbrace{\sum_i \| \lambda \cdot \vec{R} \cdot \vec{l_i'} - \vec{r_i'} \|^2}_{\geq 0 \;\text{ⓐ}} + \underbrace{m \cdot \| \vec{t}' \|^2}_{\geq 0 \;\text{ⓑ}} \rightarrow \min$$

$\vec{t}' = \lambda R \vec{l_m} - \vec{r_m} + \vec{t}$

$$\vec{t}' = \lambda R \vec{l}_m - \vec{r}_m + \vec{t}$$

$$\sum_i \| \lambda R \vec{l}_i' - \vec{r}_i' + \vec{t}' \|^2$$

$$= \underbrace{\sum_i \| \lambda R \vec{l}_i' - \vec{r}_i' \|^2}_{\text{ⓐ 求最小 } \geq 0} + \underbrace{m \cdot \| \vec{t}' \|^2}_{\text{ⓑ 求零点 } \geq 0} \to \min$$

// 得到 ⓐ、ⓑ 两项.

$\begin{cases} \text{ⓐ 求最小, 得到 } \lambda, R \\ \text{ⓑ 求零点, 得到 } \vec{t} \end{cases}$

ⓐ $\sum_i \| \lambda \vec{R} \vec{l}_i' - \vec{r}_i' \|^2 \to \min$

等效于 $\Rightarrow \sum_i \| \sqrt{\lambda} \vec{R} \vec{l}_i' - \frac{\vec{r}_i'}{\sqrt{\lambda}} \|^2 \to \min$

$$= \lambda \sum_i \| \vec{R} \vec{l}_i' \|^2 - \cancel{2 \sum_i \vec{R} \vec{l}_i' \vec{r}_i'}$$

$$- 2 \sum_i \vec{r}_i'^T \cdot \vec{R} \vec{l}_i' + \frac{1}{\lambda} \sum_i \| \vec{r}_i' \|^2$$

// 旋转不改变长度 $\Downarrow$

$\lambda \sum_i \| \vec{l}_i' \|^2$

等效于 $\Rightarrow \lambda a + b + \frac{1}{\lambda} c \xrightarrow{\min}$ $\quad \lambda a = \frac{1}{\lambda} c \Rightarrow \lambda^2 = \frac{c}{a}$

$$\Rightarrow \lambda^2 = \frac{\sum_i \| \vec{r}_i' \|^2}{\sum_i \| \vec{l}_i' \|^2}, \quad ☆ \ \lambda = \sqrt{\frac{\sum_i \| \vec{r}_i' \|^2 \quad // c}{\sum_i \| \vec{l}_i' \|^2 \quad // a}}$$

// 还剩下 b: $-2 \sum_i \vec{r}_i'^T \cdot \vec{R} \cdot \vec{l}_i'$ maximum.

$$\Rightarrow [r_x' \ \ r_y'] \cdot \begin{bmatrix} c\alpha & -s\alpha \\ s\alpha & c\alpha \end{bmatrix} \cdot \begin{bmatrix} l_x' \\ l_y' \end{bmatrix}$$

$$= \underbrace{[\cos\alpha \ \ \sin\alpha]}_{\text{向量1}} \cdot \underbrace{\begin{bmatrix} \sum_i (r_x' l_x' + r_y' l_y') \\ \sum_i (-r_x' l_y' + r_y' l_x') \end{bmatrix}}_{\text{向量2}} \to \max$$

// 当2个向量方向相同时,乘积最大.

$$☆ \Rightarrow \begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} = \frac{\begin{bmatrix} \sum_i (r_x' l_x' + r_y' l_y') \\ \sum_i (-r_x' l_y' + r_y' l_x') \end{bmatrix}}{\left\| \begin{bmatrix} \sum_i (r_x' l_x' + r_y' l_y') \\ \sum_i (-r_x' l_y' + r_y' l_x') \end{bmatrix} \right\|}$$

ⓑ $m \cdot \| \vec{t}' \|^2 \to \min$, 直接求零点.

$$\vec{t}' = \lambda \vec{R} \vec{l}_m - \vec{r}_m + \vec{t} = 0$$

$$☆ \ \vec{t} = \vec{r}_m - \lambda \cdot \vec{R} \cdot \vec{l}_m$$

精简版

已知 $\vec{l}_i$, $\vec{r}_i$, 求 λ, $\vec{R}$, $\vec{t}$

使得 $\| \lambda \vec{R} \vec{l}_i + \vec{t} - \vec{r}_i \|$ 最小.

① 去中心 $\begin{cases} \vec{l}_m, \\ \vec{r}_m \end{cases}$ ② $\vec{l}_i' = \vec{l}_i - \vec{l}_m$, $\vec{r}_i' = \vec{r}_i - \vec{r}_m$

③ $\lambda = \sqrt{\frac{\sum \| \vec{r}_i' \|^2}{\sum \| \vec{l}_i' \|^2}}$ ④ $\begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} = \frac{\begin{bmatrix} \sum (r_x' l_x' + r_y' l_y') \\ \sum (-r_x' l_y' + r_y' l_x') \end{bmatrix}}{\left\| \begin{bmatrix} \sum (r_x' l_x' + r_y' l_y') \\ \sum (-r_x' l_y' + r_y' l_x') \end{bmatrix} \right\|}$

⑤ $\vec{t} = \vec{r}_m - \lambda \cdot \vec{R} \cdot \vec{l}_m$

3. 通过相似变换公式　矫正姿势.
Similar transformation

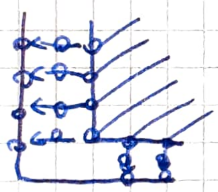
$$\begin{cases} \begin{bmatrix} x' \\ y' \end{bmatrix} = \lambda \cdot \begin{bmatrix} c\alpha & -s\alpha \\ s\alpha & c\alpha \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \\ \alpha = \mathrm{atan2}(\sin\alpha, \cos\alpha) \end{cases}$$

~~// 特征法求解 Featurebased Approach~~

4. Featurebased Approach.
- 基于 LM 的位置
- 当 LM 数量较少时, 轨迹不够平滑, 有突变.
  Trajectory (Smooth)

Featureless Approach
- 基于 Wall 信息, 数量多, 没有Feature
- 更顺滑
- every point need to ~~be~~ adjusted. There are boundaries between each other

解决方法: ICP: Iterative Closest Points Approach



每次优化后更新 Trafo, 迭代更新, 直到最优化

伪代码:
① Init Overall_trafo
② for j in xrange(Iteration)
　更新 point, 利用 trafo
　找到 wall 上匹配的 point
　找出 变换 trafo
　连接旧的 trafo 与 新的 trafo
③ 利用最终的 trafo 进行矫正.

Gegeben: $\vec{l_i}$, $\vec{r_i}$

Gesucht: λ, $\vec{R}$, $\vec{t}$

~~优化~~ 满足 $\min \sum_i \| \lambda \vec{R} \vec{l_i} + \vec{t} - \vec{r_i} \|^2$

① $\begin{cases} \vec{l_m} = \frac{1}{m} \sum_i \vec{l_i} \\ \vec{r_m} = \frac{1}{m} \sum_i \vec{r_i} \end{cases}$　② $\begin{cases} \vec{l_i'} = \vec{l_i} - \vec{l_m} \\ \vec{r_i'} = \vec{r_i} - \vec{r_m} \end{cases}$

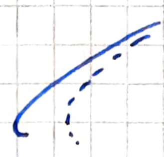
③ $\lambda = \sqrt{\dfrac{\sum_i \|\vec{r_i'}\|^2}{\sum_i \|\vec{l_i'}\|^2}}$　④ $\begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} = \dfrac{\begin{bmatrix} \sum_i (r_x' l_x' + r_y' l_y') \\ \sum_i (-r_x' l_y' + r_y' l_x') \end{bmatrix}}{\left\| \begin{bmatrix} \sum_i (r_x' l_x' + r_y' l_y') \\ \sum_i (-r_x' l_y' + r_y' l_x') \end{bmatrix} \right\|}$

⑤ $\vec{t} = \vec{r_m} - \lambda \vec{R} \vec{l_m}$

Unit C Bayes Filter, Normal distribution, Kalman Filter

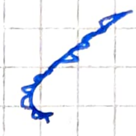
1. Error Type

① Systematic Error



+ use/add parameters, calibrations
− can't capture all real properties in the model

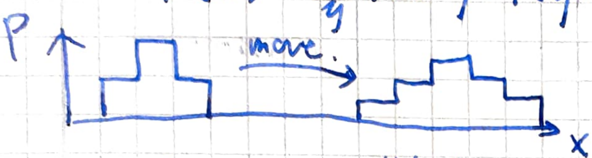
② random Error.



2.

① Motion − Convolution. // Prediction.

$$P(X) = \sum_y P(X|y)\, P(y)$$



$P(y) \to \frac{1}{4} \quad \frac{1}{2} \quad \frac{1}{4}$ (概率)

$y \to 99 \quad 100 \quad 101$ (3个位置)

$p(x)$: $\frac{1}{4} \quad \frac{1}{2} \quad \frac{1}{4} \quad \frac{1}{2} \quad \frac{1}{4} \quad \frac{1}{4} \quad \frac{1}{2} \quad \frac{1}{4}$

$x \to$ ○ ○ ○ ○ ○ (5个位置)

$\frac{1}{16} \quad \frac{1}{8}+\frac{1}{8} \quad \frac{1}{16}+\frac{1}{4}+\frac{1}{16} \quad \frac{1}{8}+\frac{1}{8} \quad \frac{1}{16}$ 解个数: 3+3−1=5

$= \frac{1}{16} \quad = \frac{4}{16} \quad = \frac{6}{16} \quad = \frac{4}{16} \quad = \frac{1}{16}$

$\Sigma = 1$

$\alpha = \frac{1}{2} = 1$

(normalization constant)

② Measurement − Multiplication // Correction

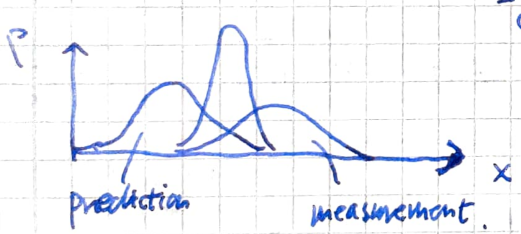
$$P(X|Z) = \alpha \cdot P(Z|X) \cdot P(X)$$

↑ posterior ↑ normalization constant ↑ likelihood. ↑ prior

from Bayes:

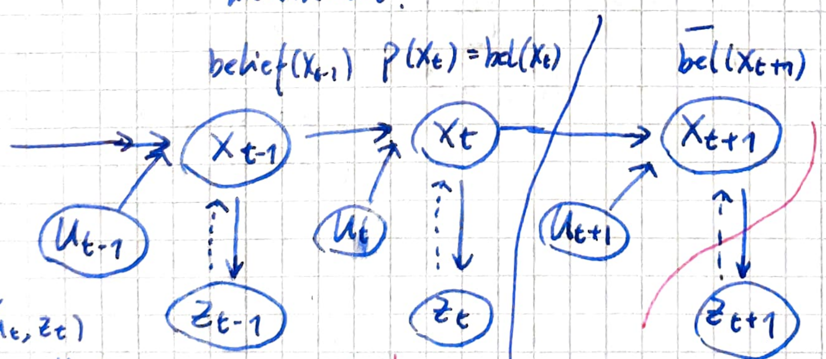
$$\underbrace{P(X|Z)}_{\text{posterior}} = \frac{\overset{\text{likelihood}}{P(Z|X)} \cdot \overset{\text{prior}}{P(X)}}{\underset{\text{(evidence)}}{P(Z)}} = \frac{P(Z|X) \cdot P(X)}{\sum_i P(Z|X_i)\, P(X_i)} \leftarrow \alpha.$$

$$= \alpha \cdot P(Z|X)\, P(X)$$



3.

belief(X_{t-1}) $\quad P(X_t) = \overline{bel}(X_t)$ $\quad \overline{bel}(X_{t+1})$ //当还没有观测到 z_{t+1} 时.

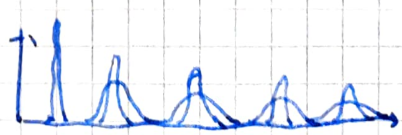


Bayes Filter $(bel(x_{t-1}), u_t, z_t)$

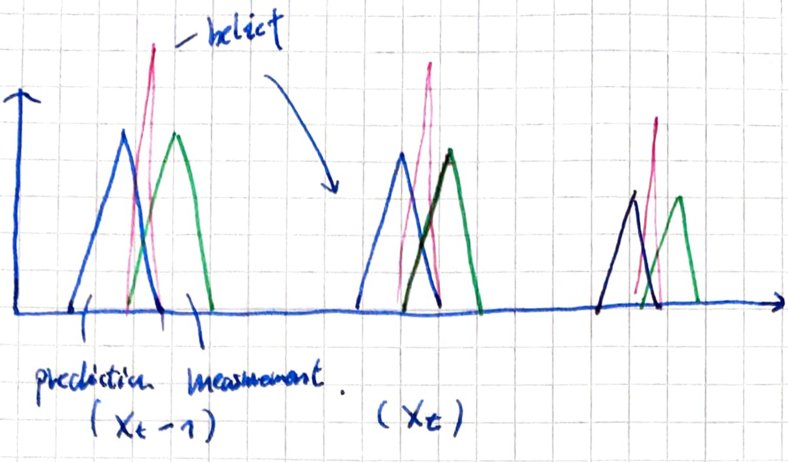
For all x_t:

$$\begin{cases} \overline{bel}(x_t) = \overset{\text{convolution}}{\sum_{x_{t-1}}} P(x_t \mid x_{t-1}, u_t) \cdot bel(x_{t-1}) & \text{// Prediction, Model (convolution) 卷积} \\ bel(x_t) = \alpha \cdot P(z_t \mid x_t) \overset{\text{multi}}{\cdot} \overline{bel}(x_t) & \text{// Correction, Measurement (multiplication) 相乘} \end{cases}$$

return $bel(x_t)$



4.



$$
\begin{cases}
\overline{bel}(x_t) = \sum_{x_{t-1}} p(x_t \mid x_{t-1}, u_t) \cdot bel(x_{t-1}) & \text{// prediction} \\
bel(x_t) = \alpha \cdot p(z_t \mid x_t) \cdot \overline{bel}(x_t) & \text{// correction}
\end{cases}
$$

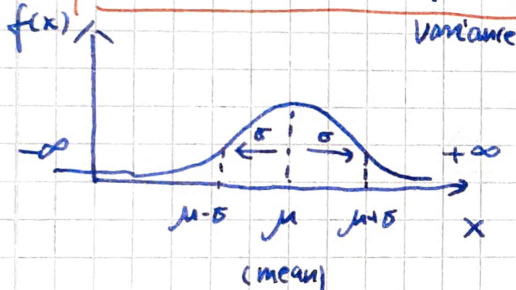
5. Normal Distribution

$$\boxed{f(x) = e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} > 0}$$

mean (μ), variance (σ)

$f(x=\mu) = e^0 = 1$

$f(x=\mu \pm \sigma) = e^{-\frac{1}{2}}$



// 正态分布积分与σ有关.

// 正态分布的积分 $\int_{-\infty}^{+\infty} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} dx \neq 1 = \sqrt{\frac{\pi}{a}}, \quad a = \frac{1}{2\sigma^2}$

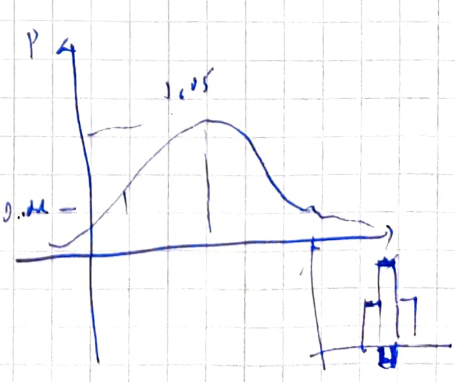
normal distribution density: $\boxed{f(x) = \frac{1}{\sqrt{2\pi} \cdot \sigma} \cdot e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}}$

(正态分布密度)

(Gaussian distribution)

$N(x, \mu, \sigma^2)$

$$
\begin{cases}
\sigma: \text{standard deviation} & \text{标准差} \\
\sigma^2: \text{variance} & \text{方差}
\end{cases}
$$



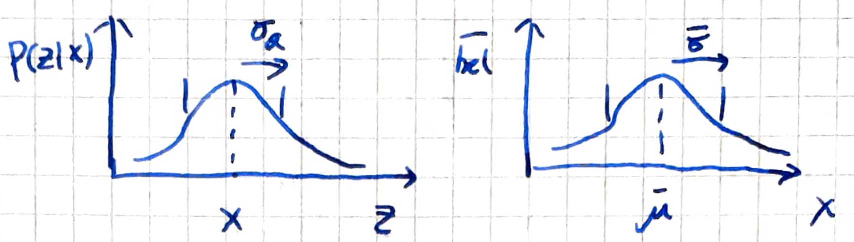
6. correction step with normal distribution // 矫正步骤中使用正态分布.

① $\underbrace{bel(x)}_{N(x;\mu,\sigma^2)} = \alpha \cdot \underbrace{P(z|x)}_{N(z;cx,\sigma_Q^2)} \cdot \underbrace{\overline{bel}(x)}_{N(x;\bar{\mu},\bar{\sigma}^2)}$

☆ $N(x;\mu,\sigma^2)$ 需要确定的.

$N(z; cx, \sigma_Q^2)$: c scale factor(1); cx mean; σ_Q^2 variance of measurement

$N(x;\bar{\mu},\bar{\sigma}^2)$: $\bar{\mu}$ mean; $\bar{\sigma}^2$ variance of belief (prediction)



求 $\begin{matrix} \mu = \\ \sigma^2 = \end{matrix} F(\bar{\mu}, \bar{\sigma}^2, z, \sigma_Q^2, c)$

②. $bel(x) = \alpha \cdot P(z|x) \cdot \overline{bel}(x)$

$$= \alpha' \cdot e^{-\frac{1}{2}\left(\frac{z-cx}{\sigma_Q}\right)^2} \cdot e^{-\frac{1}{2}\left(\frac{x-\bar{\mu}}{\bar{\sigma}}\right)^2}$$

$$= \alpha' \cdot e^{-\frac{1}{2}\underbrace{\left(\left(\frac{z-cx}{\sigma_Q}\right)^2 + \left(\frac{x-\bar{\mu}}{\bar{\sigma}}\right)^2\right)}_{g(x)}}$$ // quadratic in x

$$= \alpha'' \cdot e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$ // $bel(x)$ is normal distributed.

$\sim N(x;\mu,\sigma^2)$

③. 确定 $g(x) = \frac{1}{2}\left(\left(\frac{z-cx}{\sigma_Q}\right)^2 + \left(\frac{x-\bar{\mu}}{\bar{\sigma}}\right)^2\right)$ 作为2次函数的 A, B, C

ⓐ 若 $f(x) = \frac{1}{2}A(x-B)^2 + C$

☆ $\begin{cases} \frac{\partial f(x)}{\partial x} = A(x-B) \stackrel{!}{=} 0 \Rightarrow x = B \\ \frac{\partial^2 f(x)}{\partial x^2} = A \end{cases}$

ⓑ. $g(x) = \frac{1}{2} \cdot \frac{1}{\sigma_Q^2}(z-cx)^2 + \frac{1}{2}\frac{1}{\bar{\sigma}^2}(x-\bar{\mu})^2$

$\frac{\partial g(x)}{\partial x} = \frac{1}{\sigma_Q^2}(z-cx)\cdot(-c) + \frac{1}{\bar{\sigma}^2}(x-\bar{\mu}) \stackrel{!}{=} 0$

$x \cdot \left(\frac{c^2}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2}\right) - \frac{cz}{\sigma_Q^2} - \frac{\bar{\mu}}{\bar{\sigma}^2} = 0$

☆. $B = x = \dfrac{\frac{cz}{\sigma_Q^2} + \frac{\bar{\mu}}{\bar{\sigma}^2}}{\frac{c^2}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2}} = B$

☆ $\frac{\partial^2 g(x)}{\partial x^2} = \frac{c^2}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2} = A$

④. 将 A, B 代入 $-\frac{1}{2}A(X-B)^2+C$ 中, 则

$$\boxed{\alpha \cdot e^{-\frac{1}{2}A(X-B)^2+C} = \alpha' \cdot e^{-\frac{1}{2}\left(\frac{X-\mu}{\sigma}\right)^2}}$$

$$☆\begin{cases} A = \frac{1}{\sigma^2} & \sim \left(\frac{C^2}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2}\right) \\ B = \mu & \sim \left(\frac{CZ}{\sigma_Q^2} + \frac{\bar{\mu}}{\bar{\sigma}^2}\right) \Big/ \left(\frac{C^2}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2}\right) \sim \sigma^2\left(\frac{CZ}{\sigma_Q^2} + \frac{\bar{\mu}}{\bar{\sigma}^2}\right) \end{cases}$$

$$\Rightarrow ☆.\begin{cases} \sigma^2 = \frac{1}{A} = \frac{1}{\frac{C^2}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2}} \\ \mu = \sigma^2 \cdot \left(\frac{CZ}{\sigma_Q^2} + \frac{\bar{\mu}}{\bar{\sigma}^2}\right) \end{cases}$$

$\Rightarrow bel(X) = \mathcal{N}(X; \mu, \sigma^2)$ // 2个正态分布相乘仍是正态分布

⑤. Kalman gain,

//对 μ 重构.

$$\mu = \sigma^2\left(\frac{CZ}{\sigma_Q^2} + \frac{\bar{\mu}}{\bar{\sigma}^2}\right)$$

$$= \sigma^2\left(\frac{C}{\sigma_Q^2}(Z \overbrace{- C\bar{\mu} + C\bar{\mu}}^{=0}) + \frac{\bar{\mu}}{\bar{\sigma}^2}\right)$$

$$= \sigma^2\left(\frac{C}{\sigma_Q^2}(Z - C\bar{\mu}) + \underbrace{\frac{C^2}{\sigma_Q^2}\cdot\bar{\mu} + \frac{1}{\bar{\sigma}^2}\cdot\bar{\mu}}\right)$$

$$= \frac{\sigma^2}{\sigma_Q^2}\cdot C(Z - C\bar{\mu}) + \bar{\mu} \qquad \underbrace{\left(\frac{C^2}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2}\right)\cdot\bar{\mu}} = \frac{1}{\sigma^2}\cdot\bar{\mu}$$

$= \dfrac{\text{variance of prediction belief}}{\text{variance of measurement}} \cdot$ scalevector $\cdot$ (actual measurement − scalevector · predicted state)

+ predicted state

$$☆☆☆ \Rightarrow \underbrace{\mu}_{\text{corrected state}} = \underbrace{\bar{\mu}}_{\text{predicted state}} + \underbrace{K}_{\substack{\text{kalman gain} \\ \frac{\sigma^2}{\sigma_Q^2}\cdot C}} \cdot \overbrace{(\underbrace{Z}_{\text{actual measurement}} - \underbrace{\underbrace{C}_{\substack{\text{scale vector} \\ (1)}} \cdot \underbrace{\bar{\mu}}_{\text{predicted state}}}_{\text{predicted measurement}})}^{\text{innovation}}$$

其中, $\sigma^2 = \dfrac{1}{\frac{C^2}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2}}$

$$\mu = \bar{\mu} + K \cdot (z - \underbrace{C \cdot \bar{\mu}}_{\text{predicted measurement}})$$

μ: corrected state; $\bar{\mu}$: predicted state; K: Kalman gain $= \frac{\sigma^2}{\sigma_Q^2} \cdot C$; z: actual measurement; $C \cdot \bar{\mu}$: predicted measurement.

当 $C=1$ 时, $\begin{cases} K = 0 \Rightarrow \mu = \bar{\mu}, \text{ just predicted} \\ K = 1 \Rightarrow \mu = z, \text{ just measurement} \end{cases}$

// 比较 Kalman gain 不同大小时, 可以对 prediction 或 measurement 施加不同的 weight.

☆. $K = \frac{\sigma^2}{\sigma_Q^2} \cdot C = \frac{C}{\sigma_Q^2 \cdot \frac{1}{\sigma^2}} = \frac{C}{\sigma_Q^2 \cdot \left(\frac{C^2}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2}\right)}$

$= \frac{C \cdot \bar{\sigma}^2}{\left(C^2 + \frac{\sigma_Q^2}{\bar{\sigma}^2}\right) \cdot \bar{\sigma}^2} = \frac{C \cdot \bar{\sigma}^2}{C^2 \cdot \bar{\sigma}^2 + \underbrace{\sigma_Q^2}_{\text{variance of measurement, measurement noise}}}$

$\sigma_Q \uparrow, K \downarrow$

$\Rightarrow K \in (0, 1)$

$\begin{cases} \text{当 measurement noise} \uparrow, \sigma_Q^2 \uparrow, K \downarrow, \text{越接近 prediction } \bar{\mu} \\ \text{当 measurement noise} \downarrow, \sigma_Q^2 \downarrow, K \uparrow, \text{越接近 measurement } z \end{cases}$

// 利用 K 表达 σ^2

$$\sigma^2 = \frac{1}{\frac{C^2}{\sigma_Q^2} + \frac{1}{\bar{\sigma}^2}} = \frac{\sigma_Q^2 \bar{\sigma}^2}{C^2 \bar{\sigma}^2 + \sigma_Q^2} = \left(1 - \frac{C^2 \bar{\sigma}^2}{C^2 \bar{\sigma}^2 + \sigma_Q^2}\right) \cdot \bar{\sigma}^2$$

$$= (1 - K) \cdot \bar{\sigma}^2 \qquad \begin{cases} K = 0 \Rightarrow \sigma^2 = \bar{\sigma}^2 \\ K > 0 \Rightarrow \sigma^2 < \bar{\sigma} \end{cases}$$

// 有 K, 则 variance 会变小.

☆☆☆. ⑥. 总结. Correction Step.

$$\underbrace{bel(x)}_{N(x;\, \mu, \sigma^2)} = \alpha \cdot \underbrace{P(z \mid x)}_{N(z;\, Cx, \sigma_Q^2)} \cdot \underbrace{\overline{bel}(x)}_{N(x;\, \bar{\mu}, \bar{\sigma}^2)}$$

σ_Q^2: variance of measurement; $\bar{\sigma}^2$: variance of prediction.

$\begin{cases} ① \ K = \frac{C \bar{\sigma}^2}{C^2 \bar{\sigma}^2 + \sigma_Q^2} \\ ② \begin{cases} \mu = \bar{\mu} + K(z - C \cdot \bar{\mu}) \\ \sigma^2 = (1 - K \cdot C) \cdot \bar{\sigma}^2 \end{cases} \end{cases}$

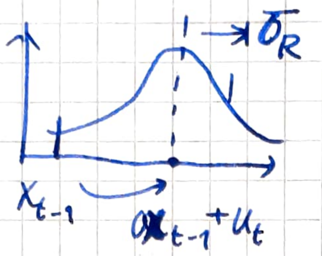
$\begin{cases} \sigma_Q^2 \uparrow, K \downarrow, \rightarrow \bar{\mu} \\ \sigma_Q^2 \downarrow, K \uparrow, \rightarrow z \end{cases}$

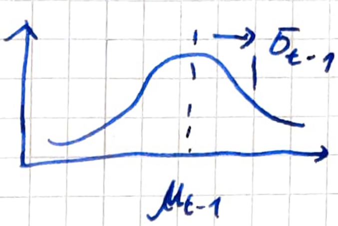
7. Prediction Step using normal distribution

$$\overline{bel}(X_t) = \int \overbrace{P(X_t \mid \underset{\text{Control}}{X_{t-1}, u_t})}^{\text{control Part}} \cdot \overbrace{bel(\underset{\text{last State}}{X_{t-1}})}^{\text{last belief}} \, dX_{t-1}$$

$$N(X_t;\ \bar{\mu}_t,\ \bar{\sigma}_t^2) \qquad N(X_t;\ aX_{t-1}+u_t,\ \sigma_R^2) \qquad N(X_{t-1};\ \mu_{t-1},\ \sigma_{t-1}^2)$$

Variance of movement (system noise)





$$\Rightarrow \int e^{-\frac{1}{2}\left(\frac{x_t-(ax_{t-1}+u_t)}{\sigma_R}\right)^2} \cdot e^{-\frac{1}{2}\left(\frac{x_{t-1}-\mu_{t-1}}{\sigma_{t-1}}\right)^2} \, dX_{t-1}$$

// still normal distribution. 证明略.

$$\Rightarrow \overline{bel}(X_t) \sim N(X_t;\ \bar{\mu}_t,\ \bar{\sigma}_t^2)$$

☆☆. $\begin{cases} \bar{\mu}_t = a \cdot \mu_{t-1} + u_t & \text{— control} \\ \bar{\sigma}_t^2 = a^2 \cdot \sigma_{t-1}^2 + \sigma_R^2 \end{cases}$

movement noise (system noise)

☆☆☆☆. 8. Kalman Filter ($(\mu_{t-1}, \sigma_{t-1}^2)$, // $N(X_{t-1};\ \mu_{t-1},\ \sigma_{t-1}^2)$ Blief of State X_{t-1}

(parametric)

(u_t, σ_R^2), // $N(X_t;\ aX_{t-1}+u_t,\ \sigma_R^2)$ Control, movement, prediction.

(z, σ_Q^2), // $N(z;\ cX,\ \sigma_Q^2)$ measurement, correction.

)

① Prediction Step:

Convolution

$$\overline{bel}(X_t) = \int P(X_t \mid X_{t-1}, u_t) \cdot bel(X_{t-1}) \, dX_{t-1}$$

$\hookrightarrow N(X_t;\ \bar{\mu}_t,\ \bar{\sigma}_t^2)$

☆. $\begin{cases} \bar{\mu}_t = a\mu_{t-1} + u_t \\ \sigma_t^2 = a^2\sigma_{t-1}^2 + \sigma_R^2 \end{cases}$

② Correction Step:

multiplication

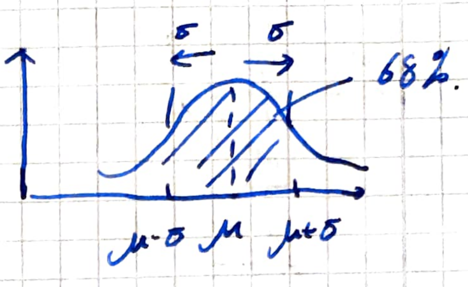
$$bel(X_t) = \alpha \cdot P(z; X_t) \cdot \overline{bel}(X_t) \sim N(X_t;\ \mu_t,\ \sigma_t^2)$$

☆. $K_t = \frac{c_t \bar{\sigma}_t^2}{c_t^2 \bar{\sigma}_t^2 + \sigma_Q^2}$, $\mu_t = \bar{\mu}_t + K_t(z_t - c_t\bar{\mu}_t)$, $\sigma_t^2 = (1 - K_t \cdot c_t) \cdot \bar{\sigma}_t^2$

Unit D

1. $\boxed{\text{1D Kalman Filter}}$

$$P = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

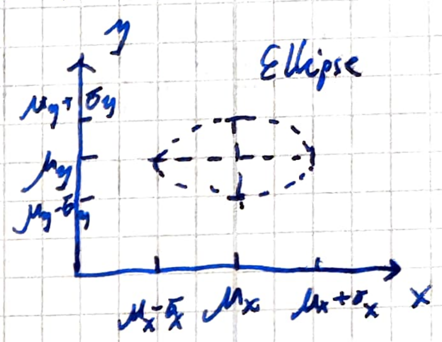


$\boxed{\text{2D Kalman Filter}}$

$$\left(\frac{x-\mu}{\sigma}\right)^2 = 1 \rightsquigarrow \frac{(x-\mu_x)^2}{\sigma_x^2} + \frac{(y-\mu_y)^2}{\sigma_y^2} = 1$$

$$[x-\mu_x,\ y-\mu_y] \cdot \underbrace{\begin{bmatrix} \frac{1}{\sigma_x^2} & 0 \\ 0 & \frac{1}{\sigma_y^2} \end{bmatrix}}_{\Sigma^{-1}} \cdot \begin{bmatrix} x-\mu_x \\ x-\mu_y \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix}$$

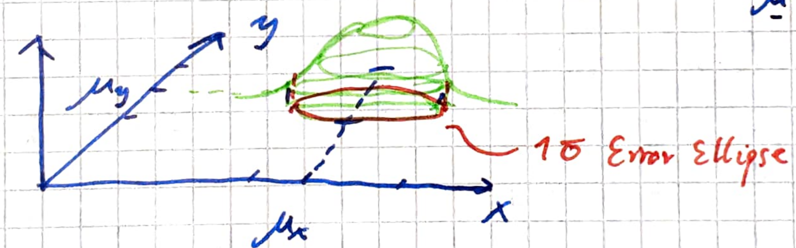


Covariance Matrix

$\boxed{\text{PDF}}$ (Probability Density Function)

$$PDF = \alpha \cdot e^{-\frac{1}{2}(\underline{x}-\underline{\mu})^T \cdot \underline{\Sigma}^{-1} \cdot (\underline{x}-\underline{\mu})}, \quad \underline{x} = \begin{bmatrix} x \\ y \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix}$$

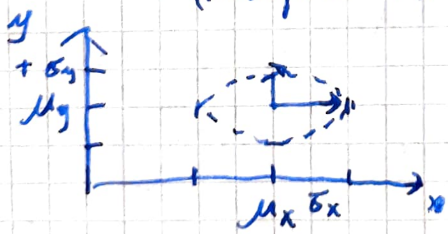
$$\underline{\mu} = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}$$

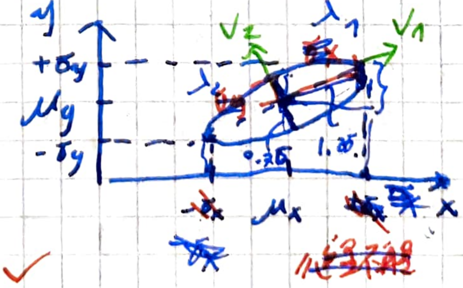


// 2 Type of Covariance.

$$\Sigma = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{xy} & \sigma_y^2 \end{bmatrix}$$

x, y uncorrelated (independent) — x, y correlated (dependent)





$$\Sigma = \underline{V} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \underline{V}^T$$

$$= [\underline{V_1}\ \underline{V_2}] \cdot \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \cdot \begin{bmatrix} \underline{V_1}^T \\ \underline{V_2}^T \end{bmatrix}$$

EV — EW

$$\lambda_1 = \sqrt{\sigma_x^2 + \sigma_y^2}$$

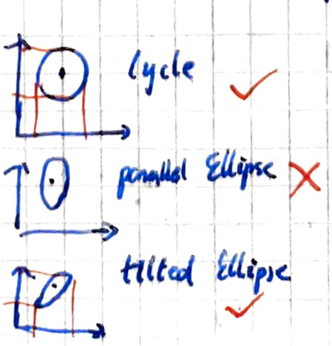
// question.

$X \sim N(\mu, \sigma^2)$

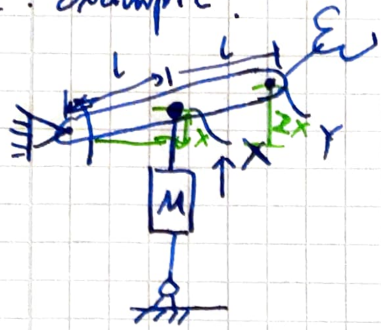
$Y \sim N(3\mu, \sigma^2)$

$Z \sim N(\mu, \Sigma)$

$$\begin{bmatrix} \mu \\ 3\mu \end{bmatrix} \begin{bmatrix} \sigma^2 & ? \\ ? & \sigma^2 \end{bmatrix}$$



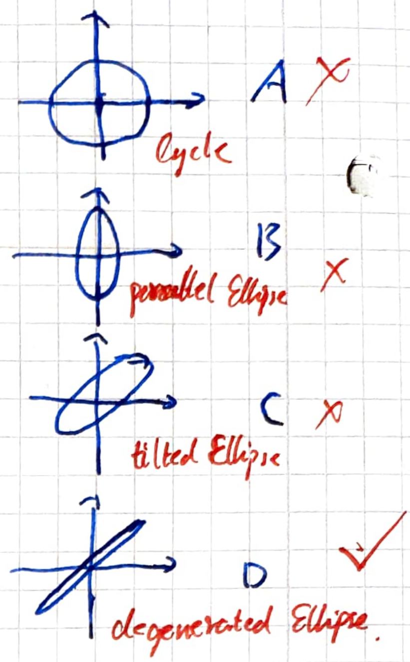
2. Example



$$X \sim N(0, \sigma_x^2)$$

$$Y \sim N(0, \sigma_y^2)$$

$$Z = \begin{bmatrix} X \\ Y \end{bmatrix}, \quad \Sigma = ?$$



$$Z = \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} X \\ 2X \end{bmatrix}$$

$$☆ \cdot \Sigma = E\left((Z - \underbrace{E(Z)}_{0}) \cdot (Z - E(Z))^T \right)$$

$$X \sim N(0, \sigma_x^2)$$

$$E(X) = 0 \Rightarrow E(Z) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$☆ . \Sigma = E(Z \cdot Z^T)$$

$$= E\left(\begin{bmatrix} X \\ 2X \end{bmatrix} [X \; 2X] \right)$$

$$= E\left(\begin{bmatrix} X^2 & 2X^2 \\ 2X^2 & 4X^2 \end{bmatrix} \right)$$

$$= \sigma_x^2 \cdot \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}$$

$$(\lambda E - A) \cdot x = 0$$

$$\det(\lambda E - A) \overset{!}{=} 0.$$

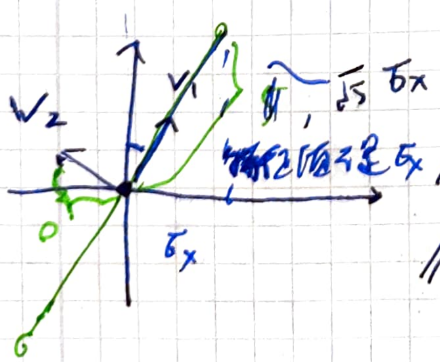
$$\det \begin{bmatrix} \lambda - 1 & -2 \\ -2 & \lambda - 4 \end{bmatrix} = 0.$$

$$(\lambda - 1)(\lambda - 4) - 4 = 0$$

$$\Rightarrow \begin{cases} \lambda_1 = 0 \\ \lambda_2 = 5 \end{cases}$$

EV of $\begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}$

$$\begin{cases} v_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \\ v_2 = \begin{bmatrix} -2 \\ 1 \end{bmatrix} \end{cases}$$



$\sqrt{5}\,\sigma_x$ // degenerated Ellipse.

$// \; \sigma_y^2 = 4\sigma_x^2$

$$☆ \begin{bmatrix} \lambda_1 = 0 \\ \sigma^* \; \lambda_2 = 5\sigma_x^2 \end{bmatrix}$$

$\|$

长轴 $= \sqrt{\lambda_2} = \sqrt{5}\,\sigma_x$

$\sigma_x =$

$\sigma_y = 2\sigma_x$

$\lambda_1 \sqrt{\sigma_x^2 + (2\sigma_x)^2}$

$= 5\sigma_x^2$

3. ND Kalman Filter

① PDF (Probability Density Function)

(1D) $$P(x) = \frac{1}{\sqrt{2\pi}\,\sigma} \cdot e^{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}}$$

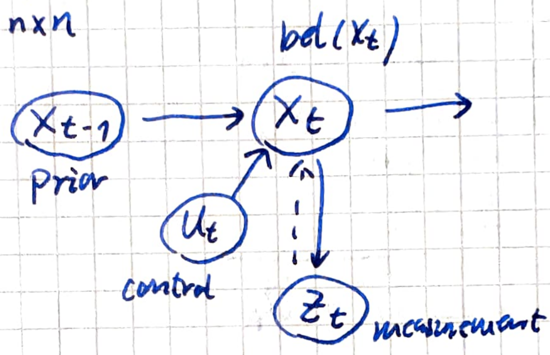
$$= (2\pi\sigma^2)^{-\frac{1}{2}} \cdot e^{-\frac{1}{2}(x-\mu)\sigma^{-2}(x-\mu)}$$

(nD) $$P(x) = \det(2\pi\Sigma)^{-\frac{1}{2}} \cdot e^{-\frac{1}{2}(\underline{x}-\underline{\mu})\Sigma^{-1}(\underline{x}-\underline{\mu})}$$

$$\Sigma = \begin{bmatrix} \sigma_x^2 & \\ & \sigma_y^2 \\ & & \ddots \end{bmatrix}$$

$$\mathcal{N}(\underbrace{X}_{nD};\ \underbrace{\mu}_{nD},\ \underbrace{\Sigma}_{n\times n})$$

② Kalman Filter (linear)



(1D): $\boxed{\text{Prediction:}}$ $x_t = a_t \cdot x_{t-1} + b_t \cdot u_t + \varepsilon_R$ // movement

$\overline{\text{bel}}(x_t)$, $\mathcal{N}(x_t; \bar{\mu}_t, \bar{\sigma}_t^2)$:
$$\begin{cases} \underset{\text{(mean)}}{\bar{\mu}_t} = a_t \cdot \mu_{t-1} + b_t \cdot u_t \\ \underset{\text{(variance)}}{\bar{\sigma}_t^2} = a_t^2 \cdot \sigma_{t-1}^2 + \sigma_R^2 \end{cases}$$

↑ movement noise (ε_R). ↑ movement variance (σ_R^2).

$\boxed{\text{Correction:}}$ $z_t = c_t \cdot \bar{x}_t + \varepsilon_Q$ // measurement

↑ measurement noise (ε_Q).

$$\underset{\text{(Kalman gain)}}{k_t} = \frac{c_t \bar{\sigma}_t^2}{c_t^2 \bar{\sigma}_t^2 + \sigma_Q^2}$$

↑ measurement variance (σ_Q^2).

bel(x_t), $\mathcal{N}(x_t; \mu_t, \sigma_t^2)$:
$$\begin{cases} \mu_t = \bar{\mu}_t + K_t(z_t - c_t \cdot \bar{\mu}_t) \\ \sigma_t^2 = (1 - K_t \cdot c_t) \cdot \bar{\sigma}_t^2 \end{cases}$$

(nD). $\boxed{\text{Prediction.}}$

$$x_t = A_t x_{t-1} + B_t u_t + \varepsilon_R$$

$$\begin{cases} \bar{\mu}_t = A_t \mu_{t-1} + B_t u_t \\ \bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t \end{cases}$$

$$R_t \downarrow \cancel{V_t \Sigma_{control} V_t^T}$$

$\boxed{\text{Correction}}$

$$z_t = C_t \cdot \bar{x}_t + \varepsilon_Q$$

$$\begin{cases} K_t = \dfrac{\bar{\Sigma}_t C_t^T}{C_t \bar{\Sigma}_t C_t^T + Q_t} \\ \mu_t = \bar{\mu}_t + K_t(z_t - C_t \cdot \bar{\mu}_t) \\ \Sigma_t = (I - K_t C_t)\bar{\Sigma}_t \end{cases}$$

☆☆ ③ Example for Lego Car.

$$X = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad 3\times1 \text{ // System State}$$

$$U = \begin{bmatrix} l \\ r \end{bmatrix} \quad 2\times1 \text{ // Controll} \qquad Z = \begin{bmatrix} \text{range} \\ \text{angle} \end{bmatrix} \quad 2\times1 \text{ // measurement}$$

$\boxed{\text{Prediction}}$

$$\overset{3\times1}{X_t} = \overset{3\times3}{A_t} \cdot \overset{3\times1}{X_{t-1}} + \overset{3\times2}{B_t} \cdot \overset{2\times1}{U_t} + \overset{3\times1}{\varepsilon_R}$$

$$\begin{cases} \overset{3\times1}{\bar{\mu}_t} = \overset{3\times3}{A_t} \cdot \overset{3\times1}{\mu_{t-1}} + \overset{3\times2}{B_t} \cdot \overset{2\times1}{U_t} & \text{// mean} \\ \overset{3\times3}{\bar{\Sigma}_t} = \overset{3\times3}{A_t} \cdot \overset{3\times3}{\Sigma_{t-1}} \cdot \overset{3\times3}{A_t^T} + \overset{3\times3}{R_t} & \text{// variance} \end{cases}$$

$\boxed{\text{Correction}}$

$$\overset{2\times1}{Z_t} = \overset{2\times3}{C_t} \cdot \overset{3\times1}{\bar{X}_t} + \overset{2\times1}{\varepsilon_Q}$$

$$\begin{cases} \overset{3\times2}{K_t} = \dfrac{\overset{3\times3}{\bar{\Sigma}_t} \cdot \overset{3\times2}{C_t^T}}{\overset{2\times3}{C_t} \cdot \overset{3\times3}{\bar{\Sigma}_t} \cdot \overset{3\times2}{C_t^T} + \overset{2\times2}{Q_t}} \\ \mu_t = \bar{\mu}_t + K_t (Z_t - C_t \cdot \bar{\mu}_t) \\ \Sigma_t = (I - K_t C_t) \bar{\Sigma}_t \end{cases}$$

④ State of Lego Car.

$$\begin{cases} \alpha = \dfrac{r-l}{w} \\ R = l/\alpha \quad r \neq l \end{cases}$$

$\underline{r \neq l}$

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} (R+\frac{w}{2})(\sin(\theta+\alpha) - \sin\theta) \\ (R+\frac{w}{2})(-\cos(\theta+\alpha) + \sin\theta) \\ \alpha \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = g(\underbrace{x, y, \theta}_{\text{State } \underline{x}}, \underbrace{l, r}_{\text{Control } \underline{u}})$$

$$\underline{x}' = g(\underline{x}, \underline{u})$$

$\underline{r = l}$

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} l \cdot \cos\theta \\ l \cdot \sin\theta \\ 0 \end{bmatrix}$$

⑤ Extended Kalman Filter - Prediction Step.

$\boxed{\text{Kalman Filter}}$ linear

$x_t = A_t x_{t-1} + B_t u_t + \varepsilon_t$ // linear

$$\begin{cases} \bar{\mu}_t = A_t \mu_{t-1} + B_t u_t \\ \bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t \end{cases}$$

☆ $\boxed{\text{EKF}}$ non linear.

$x_t = g(x_{t-1}, u_t)$ // non-linear

$$\begin{cases} \bar{\mu}_t = g(\mu_{t-1}, u_t) \\ \bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t \end{cases} \qquad V_t \cdot \Sigma_{control} \cdot V_t^T, \quad \frac{\partial g}{\partial control}$$

// G_t : Jacobian matrix of g

$$G_t = \frac{\partial g_t}{\partial x_t} = \frac{\partial g_t}{\partial State} = \begin{bmatrix} \frac{\partial g_1}{\partial x} & \frac{\partial g_1}{\partial y} & \frac{\partial g_1}{\partial \theta} \\ \vdots & \vdots & \vdots \\ \frac{\partial g_n}{\partial x} & \frac{\partial g_n}{\partial y} & \frac{\partial g_n}{\partial \theta} \end{bmatrix}$$

(analytical) 分析法.

☆☆. // 求解 G : Jacobian Matrix of g

$$g(\underbrace{x, y, \theta}_{x_{t-1}}, \underbrace{l, r}_{u_t}) = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} (R+\frac{w}{2})(\sin(\theta+\alpha) - \sin\theta) \\ (R+\frac{w}{2})(-\cos(\theta+\alpha) + \cos\theta) \\ \alpha \end{bmatrix} \begin{matrix} // g_1 \\ // g_2 \\ // g_3 \end{matrix}$$

$\boxed{r \neq l}$

$$G = \begin{bmatrix} 1 & 0 & (R+\frac{w}{2})(\cos(\theta+\alpha) - \cos\theta) \\ 0 & 1 & (R+\frac{w}{2})(\sin(\theta+\alpha) - \sin\theta) \\ 0 & 0 & 1 \end{bmatrix}$$

$\boxed{r = l}$

$$G = \begin{bmatrix} 1 & 0 & -l\sin\theta \\ 0 & 1 & l\cos\theta \\ 0 & 0 & 0 \end{bmatrix}$$

// 求解 G, numerical. 数值法.

$$\frac{\partial g_1}{\partial x} = \lim_{\Delta \to 0} \frac{g_1(x+\Delta, y, \theta) - g_1(x, y, \theta)}{\Delta} \quad // \Delta \approx 10^{-x}$$

$$\frac{\partial g_1}{\partial y} = \lim_{\Delta \to 0} \frac{g_1(x, y+\Delta, \theta) - g_1(x, y, \theta)}{\Delta}$$

$\vdots$

// by now, $\bar{\mu}_t = g(\mu_{t-1}, u_t)$

$$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + \underbrace{V_t \cdot \Sigma_{control} \cdot V_t^T}_{R_t}, \quad V_t = \frac{\partial g}{\partial control}.$$

$$\begin{cases} V_t = \frac{\partial g}{\partial control} = \begin{bmatrix} \frac{\partial g_1}{\partial l} & \frac{\partial g_1}{\partial r} \\ \vdots & \\ \frac{\partial g_3}{\partial l} & \frac{\partial g_3}{\partial r} \end{bmatrix} 3\times 2 \\ R_t = \underset{3\times 2}{V_t} \cdot \underset{2\times 2}{\begin{bmatrix} \sigma_l^2 & 0 \\ 0 & \sigma_r^2 \end{bmatrix}} \underset{2\times 3}{V_t^T} \end{cases}$$

, V_t 还需未求?

// 求解 V_t. $R = \frac{l}{\alpha}$, $\alpha = \frac{r-l}{w}$

$$V = \begin{bmatrix} \frac{\partial g_1}{\partial l} & \frac{\partial g_1}{\partial r} \\ \vdots & \vdots \\ \frac{\partial g_3}{\partial l} & \frac{\partial g_3}{\partial r} \end{bmatrix}, \quad \underset{3\times3}{R_t} = \underset{3\times2}{V_t} \cdot \underset{2\times2}{\Sigma_{control}} \cdot \underset{2\times3}{V_t^T}$$

$$= V_t \cdot \begin{bmatrix} \sigma_l^2 & 0 \\ 0 & \sigma_r^2 \end{bmatrix} \cdot V_t^T$$

$\boxed{r \neq l}$

$$\begin{cases} \frac{\partial g_1}{\partial l} = \frac{wr}{(r-l)^2}(\sin(\theta+\alpha) - \sin\theta) - \frac{r+l}{2(r-l)}\cos(\theta+\alpha) \\ \frac{\partial g_2}{\partial l} = \frac{wr}{(r-l)^2}(-\cos\theta' + \cos\theta) - \frac{r+l}{2(r-l)}\sin(\theta+\alpha) \\ \frac{\partial g_3}{\partial l} = -\frac{1}{w} \\ \frac{\partial g_1}{\partial r} = \frac{-wr}{(r-l)^2}(\sin(\theta+\alpha) - \sin\theta) + \frac{r+l}{2(r-l)}\cos(\theta+\alpha) \\ \frac{\partial g_2}{\partial r} = \frac{-wr}{(r-l)^2}(-\cos(\theta+\alpha) + \cos\theta) + \frac{r+l}{2(r-l)}\sin(\theta+\alpha) \\ \frac{\partial g_3}{\partial r} = \frac{1}{w} \end{cases}$$

$\boxed{r = l}$

$$\begin{cases} \frac{\partial g_1}{\partial l} = \frac{1}{2}(\cos\theta + \frac{l}{w}\sin\theta) & \frac{\partial g_3}{\partial l} = -\frac{1}{w} \\ \frac{\partial g_2}{\partial l} = \frac{1}{2}(\sin\theta - \frac{l}{w}\cos\theta) & \\ \frac{\partial g_1}{\partial r} = \frac{1}{2}(-\frac{l}{w}\sin\theta + \cos\theta) & \frac{\partial g_3}{\partial r} = \frac{1}{w} \\ \frac{\partial g_2}{\partial r} = \frac{1}{2}(\frac{l}{w}\cos\theta + \sin\theta) & \end{cases}$$

☆ **Summary**:

prediction step.

$\boxed{\text{nDEKF}}$

$$\begin{cases} \bar{\mu}_t = g(\mu_{t-1}, u_t) \\ \bar{\Sigma}_t = \underset{\frac{\partial g}{\partial state}}{G_t} \Sigma_{t-1} G_t^T + \overbrace{\underset{\frac{\partial g}{\partial control}}{V_t} \underset{\begin{bmatrix} \sigma_l^2 & 0 \\ 0 & \sigma_r^2 \end{bmatrix}}{\Sigma_{control}} V_t^T}^{R_t} \end{cases}$$

$$\begin{cases} \sigma_l^2 = (\alpha_1 \cdot l)^2 + (\alpha_2 (l-r))^2 \\ \sigma_r^2 = (\alpha_1 \cdot r)^2 + (\alpha_2 (l-r))^2 \end{cases}$$

// movement variance

//straight // turn

⑥. Extended Kalman Filter - Correction Step.

$$\underset{2\times1}{z_t} = \underset{2\times3}{C_t} \cdot \underset{3\times1}{\bar{x}_t} + \underset{2\times1}{\varepsilon_Q} \quad \sim \mathcal{N}(z; Cx, \sigma_Q^2)$$

z_t: measurement. C_t: scale vector. $\bar{x}_t$: prediction state. ε_Q: measurement noise.

$$\begin{cases} \underset{3\times2}{K_t} = \underset{3\times3}{\bar{\Sigma}_t} \underset{3\times2}{C_t^T} \left(\underset{2\times3}{C_t} \underset{3\times3}{\bar{\Sigma}_t} \underset{3\times2}{C_t^T} + \underset{2\times2}{Q_t} \right)^{-1} & \text{// Kalman gain} \\ \mu_t = \underbrace{\bar{\mu}_t}_{\text{mean of prediction}} + \underbrace{K_t}_{\text{Kalman gain}} (\underbrace{z_t}_{\text{measurement (actual)}} - \underbrace{C_t \cdot \bar{\mu}_t}_{\text{predicted measurement}}) & \text{// mean of correction} \\ \Sigma_t = (I - K_t C_t) \cdot \underbrace{\bar{\Sigma}_t}_{\text{covariance of prediction}} & \text{// covariance of correction} \end{cases}$$

// ~~mean~~ covariance of correction

---

linear

$$z_t = C_t \bar{x}_t + \varepsilon_Q$$

$$\begin{cases} K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + Q_t)^{-1} \\ \mu_t = \bar{\mu}_t + K_t (z_t - C_t \cdot \bar{\mu}_t) \\ \Sigma_t = (I - K_t C_t) \bar{\Sigma}_t \end{cases}$$

★. **non-linear**

$$z_t = h(\bar{x}_t) \qquad \frac{\partial h}{\partial \text{state}} \qquad \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix}$$

$$\underset{3\times2}{K_t} = \underset{3\times3}{\bar{\Sigma}_t} \cdot \underset{3\times2}{H_t^T} (\underset{2\times3}{H_t} \underset{3\times3}{\bar{\Sigma}_t} \underset{3\times2}{H_t^T} + Q_t)^{-1}$$

$$\mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t))$$

$$\Sigma_t = (I - \underset{3\times2}{K_t} \underset{2\times3}{H_t^{*}}) \cdot \bar{\Sigma}_t$$

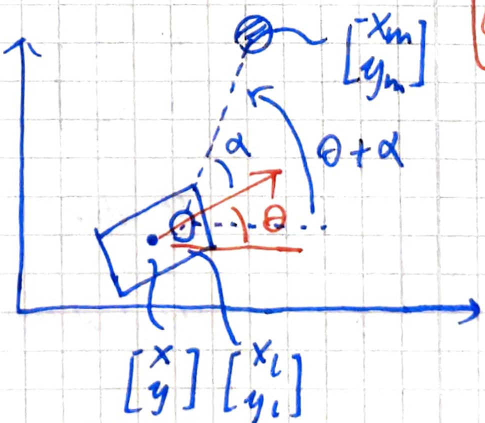
这个值要满足 $(-\pi, \pi)$

(landmark α + π) % 2π − π

要求 $h(x_t)$

$$\begin{bmatrix} x_l \\ y_l \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + d \cdot \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$$

$$\begin{cases} r = \sqrt{(x_m - x_l)^2 + (y_m - y_l)^2} \\ \alpha = \operatorname{atan}\left(\frac{y_m - y_l}{x_m - x_l}\right) - \theta \end{cases}$$



$$\Rightarrow h(\underbrace{x, y, \theta}_{\text{state } x_t}) = z_t = \underbrace{\begin{bmatrix} r \\ \alpha \end{bmatrix}}_{\text{predicted measurement}} = \begin{bmatrix} \sqrt{(x_m - x_l)^2 + (y_m - y_l)^2} \\ \operatorname{atan}\left(\frac{y_m - y_l}{x_m - x_l}\right) - \theta \end{bmatrix}$$

// measurement model

// 求解 $H = \frac{\partial h}{\partial state} = \frac{\partial \begin{bmatrix} r \\ \alpha \end{bmatrix}}{\partial (x, y, \theta)}$

$$\begin{cases} x_l = x + d\cos\theta \\ y_l = y + d\sin\theta \\ q = (x_m - x_l)^2 + (y_m - y_l)^2 \\ \Delta x = x_m - x_l \\ \Delta y = y_m - y_l \end{cases}$$

$$\frac{\partial r}{\partial x} = \frac{\partial}{\partial x}\sqrt{(x_m - x_l)^2 + (y_m - y_l)^2}$$

$$= \frac{-2(x_m - x_l)}{2\sqrt{(x_m - x_l)^2 + (y_m - y_l)^2}}$$

$$= \frac{-\Delta x}{\sqrt{q}}$$

$$\frac{\partial r}{\partial y} = \frac{-\Delta y}{\sqrt{q}}$$

$$\frac{\partial r}{\partial \theta} = \frac{d}{\sqrt{q}}(\Delta x \sin\theta - \Delta y \cos\theta)$$

$$\frac{\partial \alpha}{\partial x} = \frac{\Delta y}{q} \qquad // \frac{\partial}{\partial \alpha} \operatorname{atan}(\alpha) = \frac{1}{1+\alpha^2}$$

$$\frac{\partial \alpha}{\partial y} = \frac{-\Delta x}{q}$$

$$\frac{\partial \alpha}{\partial \theta} = -\frac{d}{q}(\cos\theta\,\Delta x + \sin\theta \cdot \Delta y) - 1$$

// Q_t 矩阵.

$$Q = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\alpha \end{bmatrix}$$

σ_r – standard deviation of distance

σ_α – standard deviation of angle.

// measurement variance

// Kalman Filter is a recursive least-square estimation

7. Summary of EKF.

Prediction Step. $x_t = g(x_{t-1}, u_t)$ // movement model

$$\overline{bel}(x_t) = \sum_{x_{t-1}} P(x_t \mid x_{t-1}, u_t) \cdot bel(x_{t-1})$$ // Bayes Filter, convolution.

$$\mathcal{N}(x_t \mid \bar{\mu}_t, \bar{\Sigma}_t) \qquad \mathcal{N}(x_t \mid g(x_{t-1}, u_t), R_t) \qquad \mathcal{N}(x_{t-1} \mid \mu_{t-1}, \Sigma_{t-1})$$ // normal distribution

$$\begin{cases} \bar{\mu}_t = g(\mu_{t-1}, u_t) \\ \bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t \end{cases}$$

G_t: $\frac{\partial g}{\partial state}$, $R_t = V_t \Sigma_{control} V_t^T$, V_t: $\frac{\partial g}{\partial control}$

$$\Sigma_{control} = \begin{bmatrix} \sigma_l^2 & 0 \\ 0 & \sigma_R^2 \end{bmatrix}$$

$$\begin{cases} \sigma_l^2 = (\alpha_1 l^2) + (\alpha_2 (l-r))^2 \\ \sigma_r^2 = (\alpha_1 r^2) + (\alpha_2 (l-r))^2 \end{cases}$$

Correction Step $z_t = h(x_t)$ // measurement model

$$bel(x_t) = \alpha \cdot P(z_t \mid x_t) \cdot \overline{bel}(x_t)$$ // Bayes Filter, multiplication.

$$\mathcal{N}(x_t \mid \mu_t, \Sigma_t) \qquad \mathcal{N}(z_t \mid h(x_t), Q_t) \qquad \mathcal{N}(x_t \mid \bar{\mu}_t, \bar{\Sigma}_t)$$ // normal distribution.

$$\begin{cases} K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + Q_t)^{-1} \\ \mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t)) \\ \Sigma_t = (I - K_t H_t) \cdot \bar{\Sigma}_t \end{cases}$$

// Kalman gain

$$Q_t = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix}$$

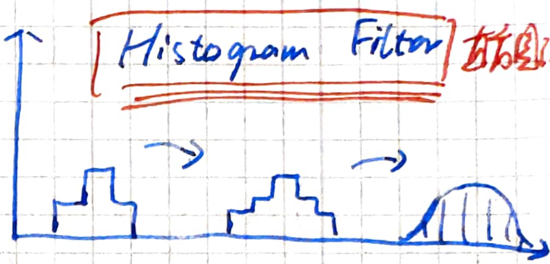
$$\boxed{H_t = \underset{2\times 3}{\frac{\partial h}{\partial state}} = \begin{matrix} \frac{\partial h_1}{\partial x} & \frac{\partial h_1}{\partial y} & \frac{\partial h_1}{\partial \theta} \\ \frac{\partial h_2}{\partial x} & \frac{\partial h_2}{\partial y} & \frac{\partial h_2}{\partial \theta} \end{matrix}}$$

$z_t - h(\bar{\mu}_t)$: Innovation mod 2π

Unit E Particle Filter

(单一模式)

1. Bayes Filter → Kalman Filter → unimodal beliefs



① Bayes Filter ($bel(x_{t-1})$, u_t, z_t)

↑ last belief, ↑ control, ↑ measurement

for all x_t :

convolution Ⓟ Prediction: $\overline{bel}(x_t) = \int P(x_t \mid x_{t-1}, u_t) \cdot bel(x_{t-1})\, dx_{t-1}$

multiplication Ⓒ Correction: $bel(x_t) = \alpha \cdot P(z_t \mid x_t)\, \overline{bel}(x_t)$

return $bel(x_t)$

---

② Kalman Filter, using normal distribution | Parametrize | 参数化.

Ⓟ $\underbrace{\overline{bel}(x_t)}_{N(x_t \mid \bar{\mu}_t, \bar{\Sigma}_t)} = \int \underbrace{P(x_t \mid x_{t-1}, u_t)}_{N(x_t \mid g(x_{t-1}, u_t), R_t)} \cdot \underbrace{bel(x_{t-1})}_{N(x_t \mid \mu_{t-1}, \Sigma_{t-1})} dx_{t-1}$ // Bayes. convolution

// normal distribution

$g(x_{t-1}, u_t)$ // motion model $\quad R_t = V_t \Sigma_{control} V_t^T$

// mean $\bar{\mu}_t = g(\mu_{t-1}, u_t)$

// covariance $\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + V_t \Sigma_{control} V_t^T$

$G_t = \frac{\partial g}{\partial state}$, $V_t = \frac{\partial g}{\partial control}$, $\Sigma_{control} = \begin{bmatrix} \sigma_l^2 & 0 \\ 0 & \sigma_r^2 \end{bmatrix}$ $\begin{cases} \sigma_l^2 = (\alpha_1 l)^2 + (\alpha_2 (l-r))^2 \\ \sigma_r^2 = (\alpha_1 r)^2 + (\alpha_2 (l-r))^2 \end{cases}$

Ⓒ $\underbrace{bel(x_t)}_{N(x_t \mid \mu_t, \Sigma_t)} = \alpha \cdot \underbrace{P(z_t \mid x_t)}_{N(z_t \mid h(x_t), Q_t)} \underbrace{\overline{bel}(x_t)}_{N(x_t \mid \bar{\mu}_t, \bar{\Sigma}_t)}$ // Bayes, multiplication

// measurement model $h(x_t)$ // normal distribution.

$K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + Q_t)$ // Kalman Gain

$H_t = \frac{\partial h}{\partial state}$, $Q_t = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix}$ // Covariance of measurement

$\mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t))$ // mean

$\Sigma_t = (I - K_t \cdot H_t) \cdot \bar{\Sigma}_t$ // covariance.

☆☆. ③. Distribution – discrete approx. (Histogram) VS Parametric ~ $N(\mu, \sigma^2)$ (normal distribution)

Distribution:
- ⊕ any distribution is possible, especially for multiple modes (peaks)

- ⊖ discret Raster cells, approximation, (computational) cost.

Parametric:
- ⊕ very efficient, only handle μ, Σ, exact for Gaussian distribution.
- ⊖ uni model (单一模式)

④ Localization Problem

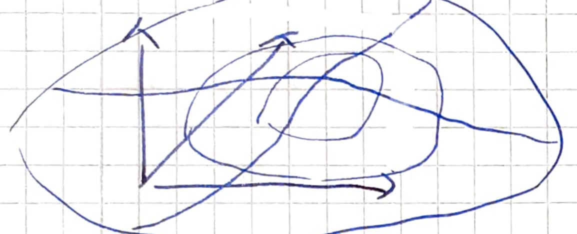
① Position tracking 位置追踪
- known initial pose 已知初始位置
- unimodel distribution (Gaussian)

//已知初始 未知位置

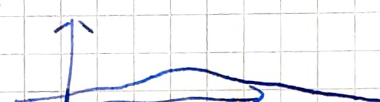
② Global localization
对于不知初始位置的机器人，可以用非常wide，flat的Gaussian Distribution来描述.
- unknown initial pose 未知初始位置
- unimodel distribution not helpful (in general)

③ kidnapped roboter
//被动移动 未知位置
- global localization plus 更精细的位置
- recovery from failures

//global localization



① 不知初始位置，可以用非常wide，flat的gaussian distribution描述

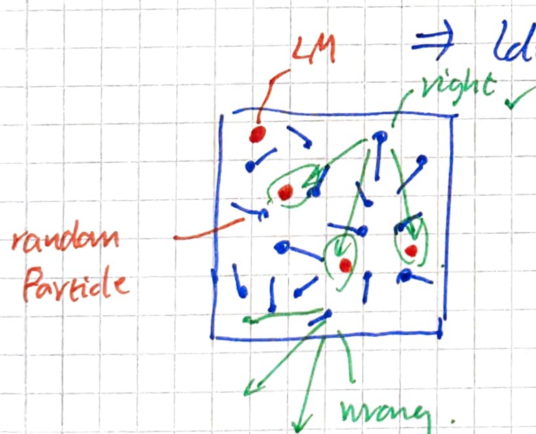


② Kalman Filter 也可能不起作用 due to LM assignment problems

LM 不一定找准. ~~want~~ wrong match

⑤ Particle Filter

⇒ Idea for Particle Filter

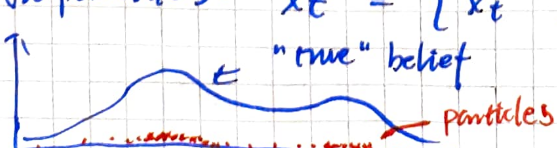


//初始化许多随机粒子
找到匹配LM程度最高的粒子.
解决LM assignment 问题.

特点:
- Represent $bel(x_t)$ by a set of random samples $\{x_t^{(i)}\}$
- Approximate
- Non parametric
- ~~Abl~~ Able to represent distributions with multiple modes.

Particle: $x_t^{[i]}$ ... is a hypothetical state

All particles: $X_t = \{x_t^{[1]}, x_t^{[2]}, \dots, x_t^{[M]}\}$, e.g. $M=1000$



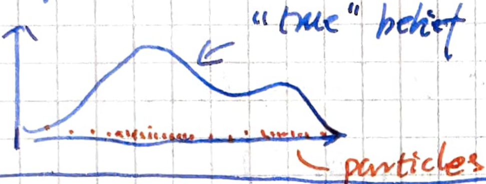
2. Particle Filters

① Eigenschaft:

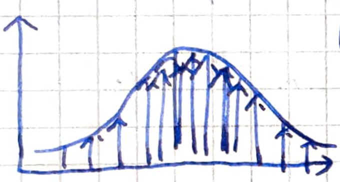
- Represent $bel(x_t)$ by a set of random samples $\{x_t^{[i]}\}$
- Approximate
- Nonparametric
- Able to represent distributions with multiple modes.

particle: $x_t^{[i]}$ ... is a hypothetical state

Allparticle: $\mathcal{X}_t = \{x_t^{[1]}, x_t^{[2]}, \dots, x_t^{[M]}\}$, e.g. $M = 1000$



② $\boxed{\text{Sample}}$ from normal distribution.



$\boxed{\mathcal{N}(\mu, \sigma^2)}$

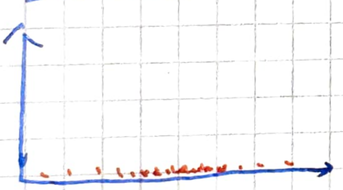
对已知分布
随机采样.

for $i = 1$ to M

  sample $x^{[i]} \sim \mathcal{N}(\mu, \sigma^2)$

return $\{x^{[1]}, x^{[2]}, \dots x^{[M]}\}$

$\boxed{\text{re build}}$ 重建



利用粒子分布
重新算出N

$$\hat{\mu} = \frac{1}{M}\sum_{i=1}^{M} x^{[i]}$$

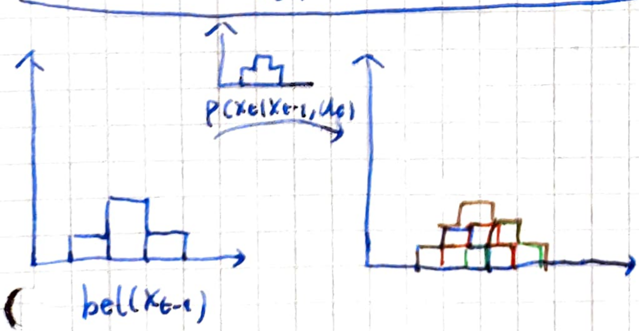
$$\hat{\sigma}^2 = \frac{1}{M-1}\sum_{i=1}^{M} [x^{[i]} - \hat{\mu})^2$$

③ particle Filter – Prediction.

$\boxed{\text{discrete Bayes Filter}}$

for all x_t:

$\overline{bel}(x_t) = \sum_{x_{t-1}} p(x_t | x_{t-1}, u_t) \cdot bel(x_{t-1})$



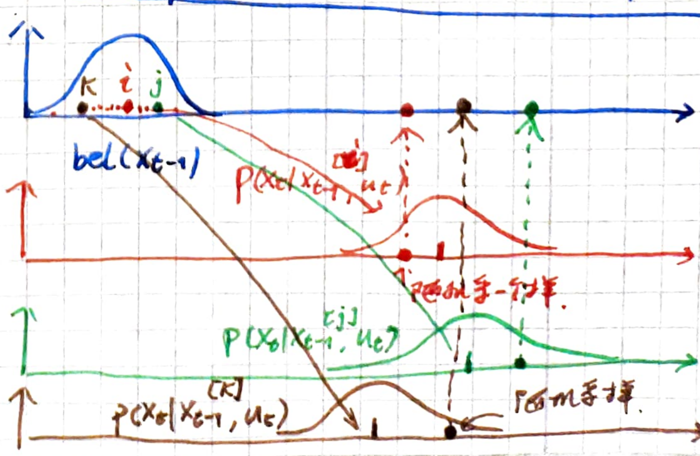
// smoothing by convolution

$\boxed{\text{Particle Filter}}$

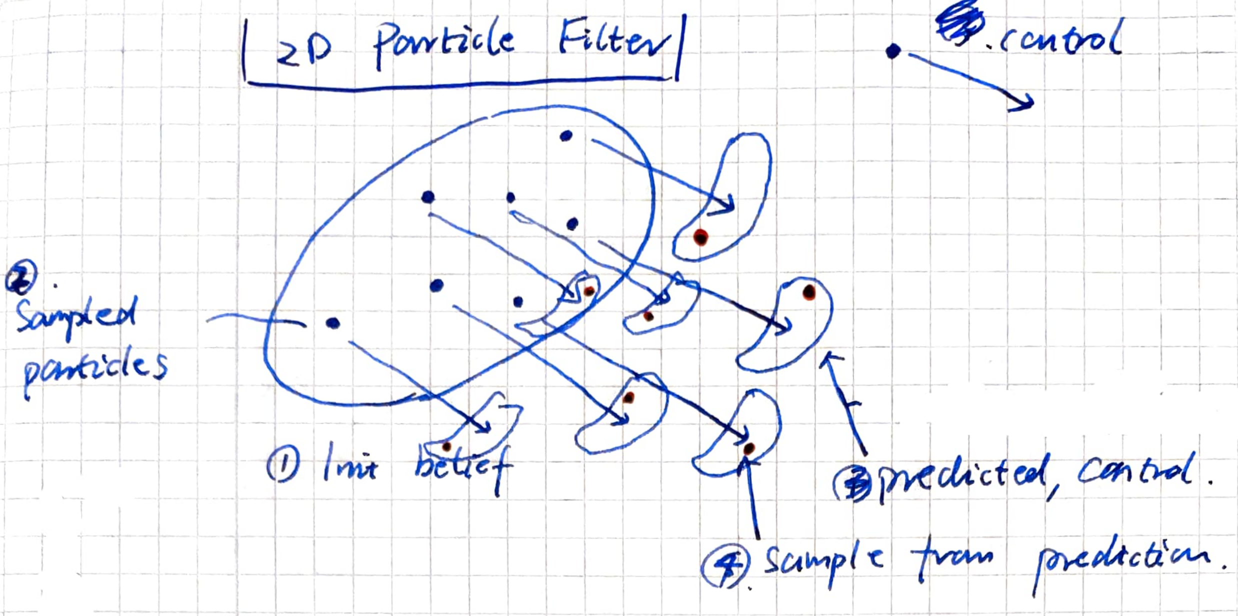
$\mathcal{X}_t = \{x_t^{[1]}, \dots, x_t^{[M]}\}$

for $m = 1$ to M

  sample $\bar{x}_t^{[m]} \sim p(x_t | x_{t-1}^{[m]}, u_t)$



2D Particle Filter



//初始分布 ① . Init belief $bel(x_{t-1}) \sim N(\mu_{t-1}, \sigma_{t-1}^2)$

//对初始分布进行采样得到粒子. ② Sample $x_{t-1}^{[m]} \sim N(\mu_{t-1}, \sigma_{t-1}^2)$

$$\{x_{t-1}\} = \{x_{t-1}^{[1]}, x_{t-1}^{[2]}, \dots, x_{t-1}^{[M]}\}$$

// 粒子移动后得到预测概率 ③ . control prediction, $p(x_t \mid x_{t-1}^{[m]}, u_t)$

// 对预测概率进行采样. ④ . Sample from prediction

for $m = 1$ to M, $\bar{x}_t^{[m]} \sim p(x_t \mid x_{t-1}^{[m]}, u_t)$

// 这里对预测概率进行采样, 其实就是对 control 进行采样.

$$\underset{\bar{x}_t^{[m]}}{x'} = g(\underset{x_{t-1}^{[m]}}{x}, \underset{\begin{bmatrix} l_t \\ r_t \end{bmatrix}}{u})$$

// 通过对 control 进行采样来得到新的粒子位置. l_t, r_t

// random.gauss(mu, sigma) for the sampling

$$\begin{cases} \text{Sample } l_t' \sim N(l_t, \sigma_{l_t}^2) \\ \text{Sample } r_t' \sim N(r_t, \sigma_{r_t}^2) \end{cases}$$

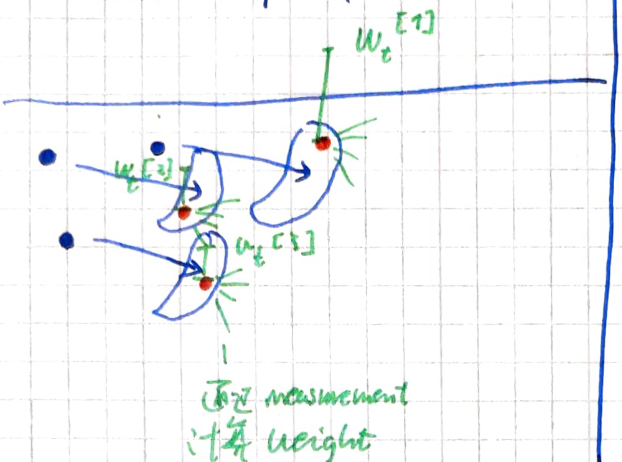
$$\begin{cases} \sigma_{l_t}^2 = (\alpha_1 l)^2 + (\alpha_2 (l-r))^2 \\ \sigma_{r_t}^2 = (\alpha_1 r)^2 + (\alpha_2 (l-r))^2 \end{cases}$$

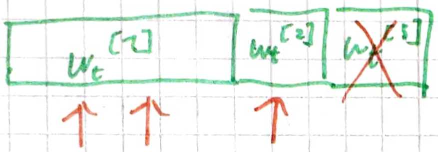
☆. $\bar{x}_t^{[m]} = g(x_{t-1}^{[m]}, \underline{\underline{\begin{bmatrix} l_t' \\ r_t' \end{bmatrix}}})$

④. Particle Filter – Correction.

Bayes Filter

$bel(x_t) = \alpha \cdot p(z_t|x_t) \cdot \overline{bel}(x_t)$





Sample.

Particle Filter

// 每个粒子分配一个权重.

for $m = 1$ to M:

$\quad w_t^{[m]} = p(z_t | \bar{x}_t^{[m]})$ // weight. Importance factor

importance sampling
(根据权重~~来~~对粒子选择)

$X_t = \{\ \}$ // init

for $m = 1$ to M

$\quad$ draw i with probability $\propto w_t^{[i]}$

$\quad X_t = X_t \cup \{\bar{x}_t^{[i]}\}$

⇒ Importance Sampling

- Draws M particles with replacement from set $\{\bar{x}_t^{[i]}\}$ with probability propotional to the weights $w_t^{[i]}$
- Before sampling, particles approximate $\overline{bel}(x_t)$ // predicted
- After sampling, particles approximate $bel(x_t)$ // posterio
- Result will contain duplicates
- Particles with a low weights will likely disappear. // 低权重的粒子消失.
- Importance sampling: general technique:

$$w \propto \frac{\text{target distribution}}{\text{proposal distribution}} = \frac{bel(x_t)}{\overline{bel}(x_t)} = \frac{\alpha\, p(z|x)\, \cancel{\overline{bel}(x)}}{\cancel{\overline{bel}(x)}}$$

$$\Rightarrow \quad w = p(z|x)$$

$$p(z|x) = p(d-d') \cdot p(\alpha-\alpha')$$

$$= \underbrace{N(d-d', 0, \sigma_d^2)}_{\text{PDF}} \cdot N(\alpha-\alpha', 0, \sigma_\alpha^2)$$

$$N(x, \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\,\sigma} \cdot e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

for every $\overset{\text{measurement}}{LM_i}$:

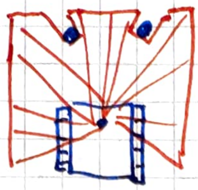
$P(z_i | x)$

$$\boxed{p(z|x) = \prod_i \underset{\uparrow \text{detected LM}}{P(z_i|x)}}$$

// to computer the probability of overall measurement for a fixed pose, loop the detected ~~LM~~, compute the probability of the measurement,

Algorithmus :

① //得到观测值.



get_cylinders_from_scan(scan)

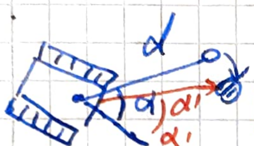
↓

cylinders.

---

②

//观测值与 ~~[struck]~~ 真实值比较 ~~逼~~ 差2, 即 测量值与预测的测量值之间的比较.



$\begin{bmatrix} x_m \\ y_m \end{bmatrix}$

compute_weights(cylinders, landmarks)

for every particles

assign_cylinders(cylinders, pose, landmarks)

↳ [((d, α), (x, y)), ( ), ( )]

LM1 LM2 LM3 .. LM6

//最多6个.

h(state, landmarks)

↳ (d′, α′)

predicted measurement

$\frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$

//PDF = f(x , μ, σ²,), 得到此点的概率密度.

actual measurement $\begin{bmatrix} d \\ \alpha \end{bmatrix}$

$N(d-d', 0, \sigma_d^2) \cdot N(\alpha-\alpha', 0, \sigma_\alpha^2)$

predicted measurement $\begin{bmatrix} d' \\ \alpha' \end{bmatrix}$

③ probability_of_measurement ( )

④ $w^{[m]} = P(Z|X) = \prod_i P(z_i|X) = \prod_i N(d_i - d_i', 0, \sigma_{d_i}^2) \cdot N(\alpha_i - \alpha_i', 0, \sigma_{\alpha_i}^2)$

weight for particle m.　　one measurement.

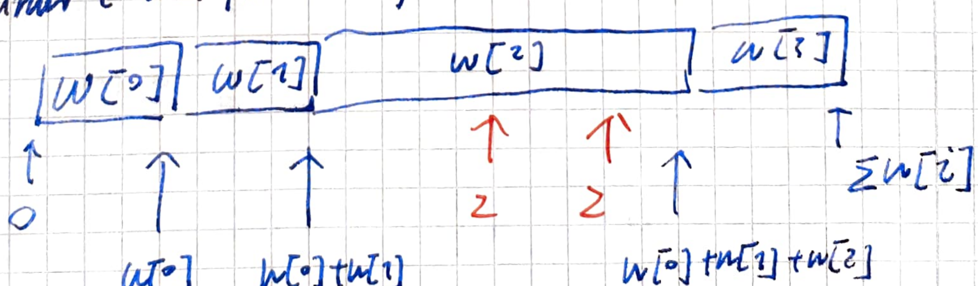
//每个粒子一个 weight

每个 weight 由其匹配的 LM 计算 PDF 得来.

---

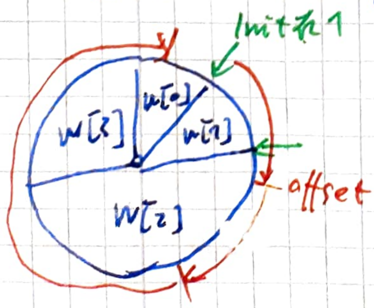
⑤. Resampling.

方法一: draw i with probability $\propto w_t^{[i]}$



$O[M \log M]$

方法二: Resampling wheel 轮盘赌.



offset ~ uniform (0, 2 · ~~max~~ weighted)

均值的分布　　0 − 2倍最大权重.

→ pick 2 , Init 到 2

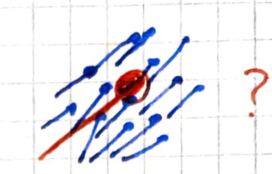
→ pick 2

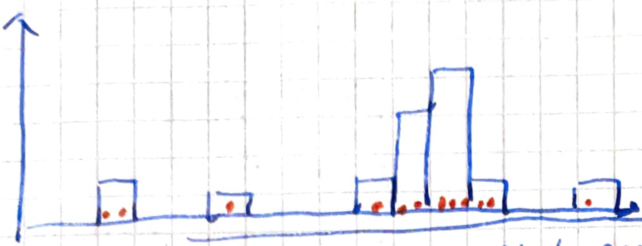
→ pick 0

Hint:

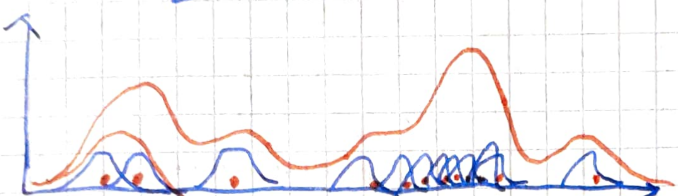
```
max_weight = max(weights)
index = random.randint(0, M-1)
for i in xrange (M)
    offset += random.uniform(0, 2·max
    while ...:
    new particle. append (particle[index
```

⑤ Density Estimation.

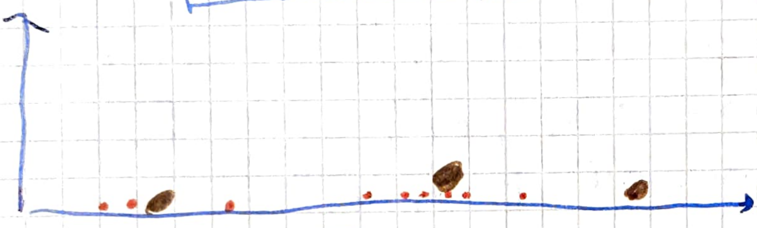




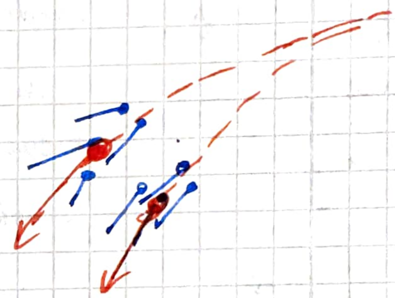
Histogram over state space

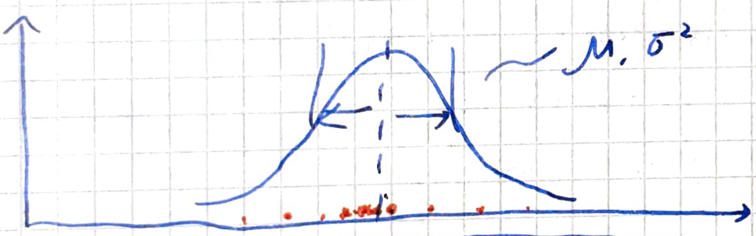


Kernel density ~~ex~~ estimation



K-Means Clustering





☆. Gaussian approximation.
only if unimodal

easy to implement.

$\hat{x} = \frac{1}{M} \sum_i x_i$ // mean position

$$\begin{cases} V_x = \frac{1}{M} \sum_i \cos \theta_i \\ V_y = \frac{1}{M} \sum_i \sin \theta_i \end{cases}$$ // mean heading vector

$\hat{\theta} = \operatorname{atan} \frac{V_y}{V_x}$ // mean heading angle

$= \operatorname{atan2}(V_y, V_x)$

// 求粒子的密度中心.

① ~~mean 直接求平均值~~.

②

① position 直接求平均值.

② heading 求 heading vector 的平均值.

3. Conclusion.

①
- Particle Filter — Monte Carlo Localization (MCL)
- — Density-estimation.
  — Particle-depreciation.
- PF very easy to implement
- PF very popular, in robotics

$\boxed{\text{Resampling}}$.

① too often, lose diversity
② too seldom, waste particles in low probability region
③ ☆ no resampling if robot is static
④ multiple measurement /如果太近说明粒子要重采样.
⑤ use variance of weight to determine if resampling is necessary.

② 思路.

$\boxed{\text{Prediction:}}$

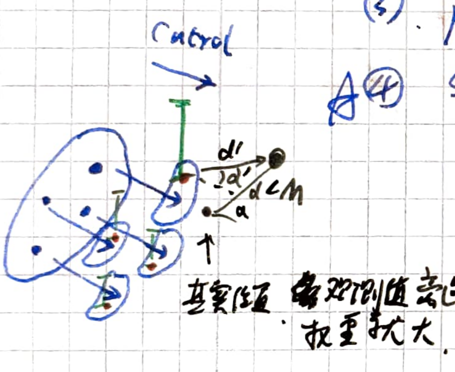
① Init belief distribution. $bel(x_{t-1}) = N(\mu_{t-1}, \sigma_{t-1}^2)$

//sample 使用
random.gauss(μ,σ)

☆② . Sample from Init distribution, $x_{t-1}^{[m]} \sim N(\mu_{t-1}, \sigma_{t-1}^2)$

m particles : $\{x_{t-1}^{[m]}\} = \{x_{t-1}^{[1]}, x_{t-1}^{[2]}, \dots, x_{t-1}^{[M]}\}$

③ . prediction with control, $P(x_t^{[m]} | x_{t-1}^{[m]}, u_t)$



☆④ sample from prediction,

for $m = 1$ to M, $\bar{x}_t^{[m]} \sim P(x_t^{[m]} | x_{t-1}^{[m]}, u_t)$

//具体操作: 对 Control 进行采样.

$$x' = g(x, u)$$

$$\bar{x}_t^{[m]} \quad x_{t-1}^{[m]} \quad \begin{bmatrix} l_t \\ r_t \end{bmatrix}$$

☆ $\begin{cases} \text{sample } l_t' \sim N(l_t, \sigma_{l_t}^2) \\ \text{sample } r_t' \sim N(r_t, \sigma_{r_t}^2) \end{cases}$, $\begin{cases} \sigma_{l_t}^2 = (\alpha_1 l)^2 + (\alpha_2 (l-r))^2 \\ \sigma_{r_t} = (\alpha_1 r)^2 + (\alpha_2 (l-r))^2 \end{cases}$

$$\bar{x}_t^{[m]} = g(x_{t-1}^{[m]}, \begin{bmatrix} l_t' \\ r_t' \end{bmatrix})$$

$\boxed{\text{Correction}}$

① get measurement, 比如 get_cylinder_scan(scan)
//得到观测值.
↳ cylinders $\begin{bmatrix} d \\ \alpha \end{bmatrix}$

② . 对每一个粒子的观测值, 与LM进行匹配, 并得到LM的预测观测值.
assign_cylinder( )
↳ $[[(d, a), (d', a')], (\ ), \dots, (\ )]$
LM① LM② //一共可以有6个

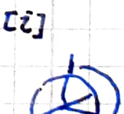
③ . for every particle.

☆☆ $\cdot\ w^{[m]} = P(Z|x) = \prod_i P(z_i|x) = \prod_i N(d_i - d_i', 0, \sigma_d^2) \cdot N(\alpha_i - \alpha_i', 0, \sigma_\alpha^2)$

$\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$

//每个观测值. //PDF, 得到观测与预测的差值对应的概率密度.

④ Resampling
{ ① Draw 抽签 draw i with probability $\propto w_t^{[i]}$
{ ② Resampling wheel, offset $[0, 2 \cdot \text{max\_weight}]$
uniform



⑤ density Estimation.
{ ① Histogram
② kernel density estimation
③ K-mean
④ gaussian-estimation

$$\begin{cases} \hat{x} = \frac{1}{m} \sum_i x_i \\ \begin{cases} V_x = \frac{1}{m} \sum_i \cos\theta_i \\ V_y = \frac{1}{m} \sum_i \sin\theta_i \end{cases} \\ \hat{\theta} = \operatorname{atan2}(V_y, V_x) \end{cases}$$

Unit F SLAM, EKF-SLAM (Simultaneous localization and mapping)

☆ //无 LM 的信息，需要后期确定.

System State $\boxed{x \mid y \mid \theta}$

1. ① EKF
//已知初始位置，已知LM

$$\begin{cases} \text{prediction} & \overline{bel}(x_t) = \int P(x_t \mid x_{t-1}, u_t)\, bel(x_{t-1}) \\ \text{correction} & bel(x_t) = \alpha\, P(z_t \mid x_t)\, \overline{bel}(x_t) \end{cases}$$

② Particle
//未知初始位置，已知LM

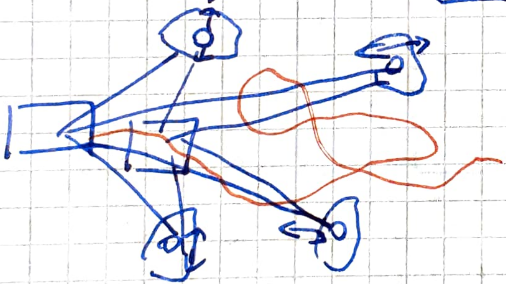
- prediction – sample $m = 100$ $\{x_{t-1}^{[m]}\} \sim N(\mu_{t-1}, \Sigma_{t-1})$
  $\Longrightarrow \bar{x}_t^{[m]} = g(x_{t-1}^{[m]}, \begin{bmatrix} l_t' \\ r_t' \end{bmatrix})$, Sample $l_t' \sim N(l_t, \sigma_{l_t}^2)$, $r_t' \sim N(r_t, \sigma_{r_t}^2)$
- correction
  ① LM assignment
  ② for every particles
  $$w^{[m]} = P(z_t \mid x_t) = \prod_i P(z_{i_t} \mid x_t) = \prod_i N(d - d';\, 0, \sigma_d^2)\, N(\alpha - \alpha';\, 0, \sigma_\alpha^2)$$
  //通过观测的数据与绘制的数据之差计算 weight.
- ↓ resampling (根据 weight 重选择)
- ↓ density estimation. $\hat{x} = \frac{1}{m} \sum_i x_i$
  $$\begin{cases} v_x = \frac{1}{m} \sum_i \cos\theta_i \\ v_y = \frac{1}{m} \sum_i \sin\theta_i \end{cases}, \quad \hat{\theta}_i = \operatorname{atan2}(v_y, v_x)$$

EKF SLAM
$\boxed{x \mid y \mid \theta} \boxed{x_1\ y_1} \boxed{x_2\ y_2} \cdots$

③ SLAM – Simultaneous Localization And Mapping 即同时定位与地图构建.
//未知初始位置，未知LM.

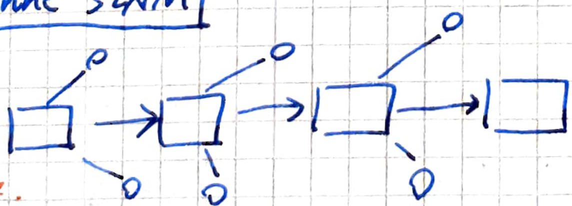
①
$$\begin{cases} \text{EKF System State} & \boxed{x \mid y \mid \theta} \\ \text{SLAM System State} & \boxed{x \mid y \mid \theta} \boxed{x_1\ y_1} \boxed{x_2\ y_2} \cdots \end{cases}$$

②



//通过无穷的路段观测，LM之间的相对误差(relative error)会减小直至零，但其绝对误差无法修正.

③ $\boxed{\text{online SLAM}}$ $\quad P(x_t, m \mid z_{1:t}, u_{1:t})$



SLAM Problem
- Continuous component $x, y, \theta, x_1 y_1 \ldots x_n y_n$ too large.
- Discrete component correspondence of objects to previous detected objects.
- calculation infeasible.
  - high dimension
  - large number of correspondence.

State variable: $x, y, \theta, \begin{bmatrix} x_{m_1} \\ y_{m_1} \end{bmatrix}, \cdots \begin{bmatrix} x_{m_{50}} \\ y_{m_{50}} \end{bmatrix}$, $3 + 2 \times 50 = 103$

$\boxed{\text{Full SLAM}}$ $\quad P(x_{1:t}, m \mid z_{1:t}, u_{1:t})$, $t = 200$, $m = 50$.

$$\begin{bmatrix} x_1 \\ y_1 \\ \theta_1 \end{bmatrix}, \begin{bmatrix} x_2 \\ y_2 \\ \theta_2 \end{bmatrix}, \cdots \begin{bmatrix} x_{200} \\ y_{200} \\ \theta_{200} \end{bmatrix}, \begin{bmatrix} x_{m_1} \\ y_{m_1} \end{bmatrix}, \cdots, \begin{bmatrix} x_{m_{50}} \\ y_{m_{50}} \end{bmatrix}$$

$3 \times 200 + 2 \times 50 = 700$

2. EFF-SLAM $\boxed{\text{Prediction}}$

$R = \frac{l}{\alpha}$

$\alpha = \frac{r-l}{w}$

① $\bar{\mu}_t = g(\mu_{t-1}, u_t)$

$$\bar{\Sigma}_t = \underbrace{G_t}_{\frac{\partial g}{\partial \text{state}}} \underbrace{\Sigma_{t-1}}_{\text{covariance}} G_t^T + \underbrace{R_t}_{V_t \Sigma_{control} V_t^T}$$

$$V_t = \frac{\partial g}{\partial \text{control}}, \quad \Sigma_{control} = \begin{bmatrix} \sigma_l^2 & 0 \\ 0 & \sigma_r^2 \end{bmatrix}$$

$$g(\underbrace{x, y, \theta}_{\substack{\text{State} \\ \mu_{t-1}}}, \underbrace{l, r}_{\substack{\text{control} \\ u_t}})$$

$$= \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} (R+\frac{w}{2})(\sin(\theta+\alpha) - \sin\theta) \\ (R+\frac{w}{2})(-\cos(\theta+\alpha) + \cos\theta) \end{bmatrix}$$

$$G = \frac{\partial g}{\partial \text{state}} = \begin{bmatrix} 1 & 0 & * \\ 0 & 1 & * \\ 0 & 0 & 1 \end{bmatrix} \begin{cases} l \neq r \\ l = r \end{cases}$$

$$V = \frac{\partial g}{\partial \text{control}} = \begin{bmatrix} * & * \\ * & * \\ * & * \end{bmatrix}$$

② // 把LM放入state中考虑

//motion model

$$g(\underbrace{x, y, \theta, \underbrace{x_1, y_1, x_2, y_2, \ldots,}_{\text{LM}}}_{\text{State}} \underbrace{l, r}_{\text{control}})$$

$$= \begin{bmatrix} x \\ y \\ \theta \\ x_1 \\ y_1 \\ x_2 \\ y_2 \end{bmatrix} + \begin{bmatrix} (R+\frac{w}{2})(\sin(\theta+\alpha) - \sin\theta) \\ (R+\frac{w}{2})(-\cos(\theta+\alpha) + \cos\theta) \\ \alpha \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{cases} R = \frac{l}{\alpha} \\ \alpha = \frac{r-l}{w} \end{cases}$$

$\boxed{\text{state}}$

$$\begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \xrightarrow{\text{add LM}} \begin{bmatrix} x \\ y \\ \theta \\ x_1 \\ y_1 \end{bmatrix} \xrightarrow{\text{add LM}} \begin{bmatrix} x \\ y \\ \theta \\ x_1 \\ y_1 \\ x_2 \\ y_2 \end{bmatrix}$$

//原矩阵

$$G = \frac{\partial g}{\partial \text{state}} = \begin{bmatrix} \begin{bmatrix} 1 & 0 & * \\ 0 & 1 & * \\ 0 & 0 & 1 \end{bmatrix} & \begin{bmatrix} 0 \end{bmatrix} \\ \begin{bmatrix} 0 \end{bmatrix} & \begin{bmatrix} 1 & & 0 \\ & 1 & \\ & & 1 \\ 0 & & & 1 \end{bmatrix} \end{bmatrix}, \quad \underbrace{\quad}_{3} \underbrace{\quad}_{2N}$$

$(3+2N) \times (3+2N)$

左上角原矩阵

右下角单位矩阵.

$\frac{\partial g}{\partial \text{control}}$.

$$R_t = V_t \cdot \Sigma_{control} \cdot V_t^T$$

//原矩阵

$$= \begin{bmatrix} * & * \\ * & * \\ * & * \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \sigma_l^2 & 0 \\ 0 & \sigma_r^2 \end{bmatrix} \cdot \begin{bmatrix} * & * & * & 0 & 0 & 0 & 0 \\ * & * & * & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix} & \begin{bmatrix} 0 \end{bmatrix} \\ \begin{bmatrix} 0 \end{bmatrix} & \begin{bmatrix} 0 & & & \\ & 0 & & \\ & & 0 & \\ & & & 0 \end{bmatrix} \end{bmatrix} (3+2N) \times (3+2N)$$

$\underbrace{\quad}_{3} \underbrace{\quad}_{2N}$

//covariance

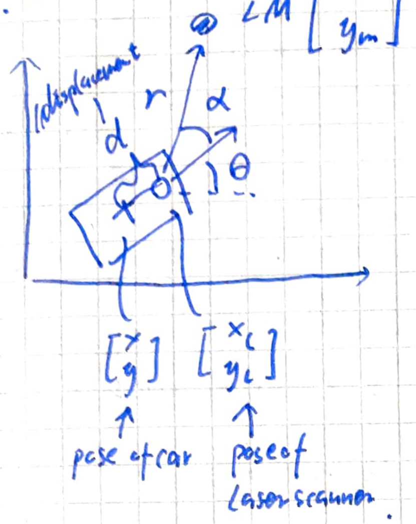
$$\Sigma_t \rightarrow \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{x\theta} \\ \sigma_{xy} & \sigma_y^2 & \sigma_{y\theta} \\ \sigma_{x\theta} & \sigma_{y\theta} & \sigma_\theta^2 \end{bmatrix} \xrightarrow{\text{add one LM}} \underbrace{\begin{bmatrix} \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{x\theta} \\ \sigma_{xy} & \sigma_y^2 & \sigma_{y\theta} \\ \sigma_{x\theta} & \sigma_{y\theta} & \sigma_\theta^2 \end{bmatrix} & \begin{bmatrix} 0 \end{bmatrix} \\ \begin{bmatrix} 0 \end{bmatrix} & \underbrace{\begin{matrix} \sigma_{x_1}^2 & 0 & & \\ 0 & \sigma_{y_1}^2 & & \\ & & \sigma_{x_2}^2 & \\ & & & \sigma_{y_2}^2 \end{matrix}}_{\Sigma_{LM}} \end{bmatrix}}_{3+2N} (3+2N) \times (3+2N)$$

// $\sigma_{x_1}^2 = \sigma_{y_1}^2 = \infty$

初始设为∞, 10^{10}

3. EKF - Correction.

① Observation



Laserscanner

$$\begin{bmatrix} x_L \\ y_L \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + d \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$$

$\begin{bmatrix} x_L \\ y_L \end{bmatrix}$: position of Laserscanner; $\begin{bmatrix} x \\ y \end{bmatrix}$: position of car; d: displacement; θ: heading

Landmark

$$\begin{bmatrix} r \\ \alpha \end{bmatrix} = \begin{bmatrix} \sqrt{(x_m - x_L)^2 + (y_m - y_L)^2} \\ \operatorname{atan}\left(\frac{y_m - y_L}{x_m - x_L}\right) - \theta \end{bmatrix}$$

$$\Rightarrow \star \begin{bmatrix} r \\ \alpha \end{bmatrix} = h(x, y, \theta) = \begin{bmatrix} \sqrt{(x_m - x_L)^2 + (y_m - y_L)^2} \\ \operatorname{atan}\left(\frac{y_m - y_L}{x_m - x_L}\right) - \theta \end{bmatrix}$$

$$H = \frac{\partial h}{\partial \text{state}} = \begin{bmatrix} \frac{\partial r}{\partial x} & \frac{\partial r}{\partial y} & \frac{\partial r}{\partial \theta} \\ \frac{\partial \alpha}{\partial x} & \frac{\partial \alpha}{\partial y} & \frac{\partial \alpha}{\partial \theta} \end{bmatrix}$$

---

② // now LM is unknown, so $h(x, y, \theta) \Longrightarrow h(x, y, \theta, x_m, y_m)$

$$\Rightarrow H = \frac{\partial h(x, y, \theta, x_m, y_m)}{\partial \text{State}(x, y, \theta, x_m, y_m)} = \begin{bmatrix} \frac{\partial r}{\partial x} & \frac{\partial r}{\partial y} & \frac{\partial r}{\partial \theta} & \frac{\partial r}{\partial x_m} & \frac{\partial r}{\partial y_m} \\ \frac{\partial \alpha}{\partial x} & \frac{\partial \alpha}{\partial y} & \frac{\partial \alpha}{\partial \theta} & \frac{\partial \alpha}{\partial x_m} & \frac{\partial \alpha}{\partial y_m} \end{bmatrix}$$

$\frac{\partial h}{\partial LM}$

i.e. $\begin{cases} \delta x = x_m - x_L \\ \delta y = y_m - y_L \\ q = \delta x^2 + \delta y^2 \end{cases}$

EKF — SLAM, LM

$\underline{\underline{H}}$: $\times(3+2N)$

	$\frac{\partial h}{\partial x}$	$\frac{\partial h}{\partial y}$	$\frac{\partial h}{\partial \theta}$	$\frac{\partial h}{\partial x_m}$	$\frac{\partial h}{\partial y_m}$
r	$-\frac{\delta x}{\sqrt{q}}$	$-\frac{\delta y}{\sqrt{q}}$	$\frac{d}{\sqrt{q}}(\delta x \sin\theta - \delta y \cos\theta)$	$\frac{\delta x}{\sqrt{q}}$	$\frac{\delta y}{\sqrt{q}}$
α	$\frac{\delta y}{q}$	$-\frac{\delta x}{q}$	$-\frac{d}{q}(\delta x \cos\theta + \delta y \sin\theta) - 1$	$-\frac{\delta y}{q}$	$\frac{\delta x}{q}$

$\times -1$

$\underline{\underline{H}}$: 223 LM_j

$\times -1$

	$\frac{\partial h}{\partial x}$	$\frac{\partial h}{\partial y}$	$\frac{\partial h}{\partial \theta}$	$\frac{\partial h}{\partial x_1}$	$\frac{\partial h}{\partial y_1}$	…	$\frac{\partial h}{\partial x_j}$	$\frac{\partial h}{\partial y_j}$	…		
r				0	0	…			…	0	0
α				0	0	…			…	0	0

~~$\mu_t = k_t$~~ $K_t = \Sigma_t H_t^T (H_t \Sigma_t H_t^T + Q_t)^{-1}$

$\mu_t = \bar{\mu}_t + K_t(z_t + h(\bar{\mu}))$

$\Sigma_t = (I + K_t \cdot H_t) \cdot \bar{\Sigma}_t$

4. Conclusion.

Number of landmarks

	Landmark assignment	localization
Only a few	(+)	(−)
many	(−)	(+)

⇒ assignment Problem ↓ provisional list.

- Full SLAM: posterior over all states & assignments of LM
  - computationally infeasible
- online SLAM: deterministic computation of assignments
  - brittle with respect to LM computation.
  - update complexity

---

5. Summary. EKF-SLAM

$$\mu_t = \begin{bmatrix} x \\ y \\ \theta \\ x_1 \\ y_1 \\ x_2 \\ y_2 \\ \vdots \end{bmatrix} \quad (3+2N)\times 1$$

(P). $\bar{\mu}_t = g(\mu_{t-1}, u_t)$

$\bar{\mu}_t$: $(3+2N)\times 1$.

$$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t = G_t \Sigma_{t-1} G_t^T + \overbrace{V_t \Sigma_{control} V_t^T}^{(3+2N)\times(3+2N)}$$

$\bar{\Sigma}_t$: $(3+2N)(3+2N)$; G_t: $(3+2N)(3+2N)$; Σ_{t-1}: $(3+2N)(3+2N)$; G_t^T: $(3+2N)(3+2N)$; V_t: $(3+2N)\times 2$; $\Sigma_{control}$: 2×2; V_t^T: $(2\times 3+2N)$

$$G_t = \frac{\partial g}{\partial state} = \begin{bmatrix} \begin{bmatrix} 1 & 0 & * \\ 0 & 1 & * \\ 0 & 0 & 1 \end{bmatrix} & [0] \\ [0] & \begin{bmatrix} 1 & \\ & 1_{1} \end{bmatrix} \end{bmatrix} \quad (3+2N)\times(3+2N)$$

$$V_t = \frac{\partial g}{\partial control} = \begin{bmatrix} * & * \\ * & * \\ * & * \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (3+2N)\times 2$$

$$\Sigma_{control} = \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_w^2 \end{bmatrix} \quad 2\times 2.$$

$$R_t = V_t \Sigma_{control} V_t^T = \begin{bmatrix} \begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix} & [0] \\ [0] & \begin{bmatrix} 0 & \\ & 0_0 \end{bmatrix} \end{bmatrix} \quad (3+2N)\times(3+2N)$$

(C). $K_t = \underbrace{\bar{\Sigma}_t H_t^T}_{(3+2N)\times 2} (\underbrace{H_t \bar{\Sigma}_t H_t^T}_{2\times 2} + Q_t)^{-1}$

K_t: $(3+2N)\times 2$; $\bar{\Sigma}_t$: $(3+2N)(3+2N)$; H_t^T: $(3+2N)\times 2$; H_t: $2\times(3+2N)$; $\bar{\Sigma}_t$: $(3+2N)(3+2N)$; H_t^T: $(3+2N)\times 2$; $Q_t = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix}$

$$\mu_t = \bar{\mu}_t + \underbrace{K_t(z_t - h(\bar{\mu}_t))}_{(3+2N)\times 1}$$

μ_t: $(3+2N)\times 1$; $\bar{\mu}_t$: $(3+2N)\times 1$; K_t: $(3+2N)\times 2$; z_t: 2×1; $h(\bar{\mu}_t)$: 2×1

$$\Sigma_t = (I - K_t \cdot H_t)\bar{\Sigma}_t$$

Σ_t: $(3+2N)(3+2N)$; I: $(3+2N)(3+2N)$; K_t: $(3+2N)\times 2$; H_t: $2\times(3+2N)$; $\bar{\Sigma}_t$: $(3+2N)(3+2N)$

$$H_t = \begin{array}{c|ccc|cc|cc|c} & \frac{\partial h}{\partial x} & \frac{\partial h}{\partial y} & \frac{\partial h}{\partial \theta} & \frac{\partial h}{\partial x_1} & \frac{\partial h}{\partial y_1} & \frac{\partial h}{\partial x_j} & \frac{\partial h}{\partial y_j} & \cdots \\ \hline r & * & * & * & 0 & 0 & * & * & 0\ 0 \\ \alpha & * & * & * & 0 & 0 & * & * & 0\ 0 \end{array}$$

$\times(-1)$

$2\times(3+2N)$

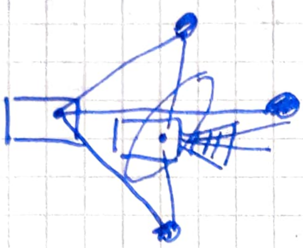
$$\frac{\partial h}{\partial LM} = \begin{bmatrix} \frac{\partial h}{\partial x_m} & \frac{\partial h}{\partial y_m} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\partial h}{\partial x} & \frac{\partial h}{\partial y} \end{bmatrix} \cdot (-1)$$

Unit 6 Particle Filter SLAM (Fast SLAM)

1. 回顾.

① EKF

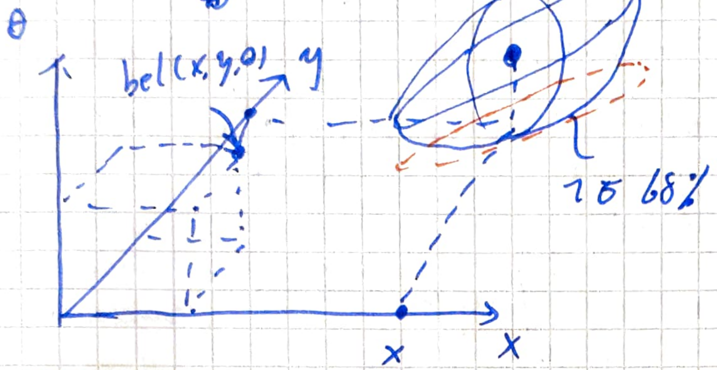


3D state $\mu = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$

covariance $\Sigma = \begin{bmatrix} \boxed{\begin{matrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{xy} & \sigma_y^2 \end{matrix}} & \begin{matrix} \sigma_{x\theta} \\ \sigma_{y\theta} \end{matrix} \\ \begin{matrix} \sigma_{x\theta} & \sigma_{y\theta} \end{matrix} & \boxed{\sigma_\theta^2} \end{bmatrix}$

Error Ellipse

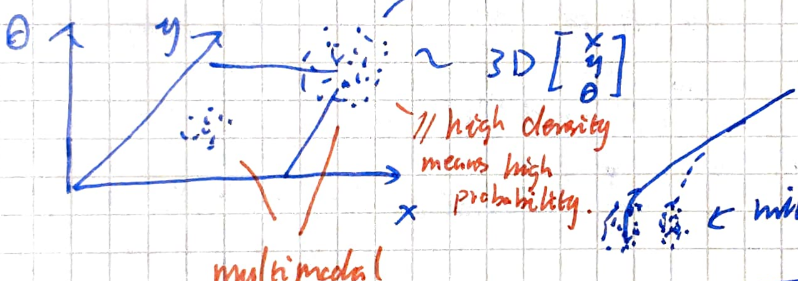
heading disk.



// LM-fixed

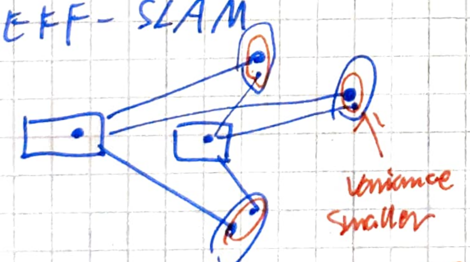
② Particle Filter

hypothetical state of particles



// LM-fixed

③ EKF-SLAM



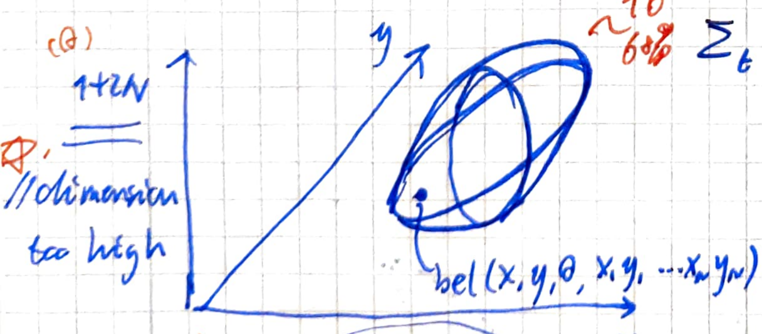
$\mu = \begin{bmatrix} \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \\ \begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ \vdots \end{bmatrix} \end{bmatrix}$ } state, } LM

$G_t = \frac{\partial g}{\partial state} = \begin{bmatrix} \begin{matrix} 1 & 0 & * \\ 0 & 1 & * \\ 0 & 0 & 1 \end{matrix} & [0] \\ [0] & I_{2N} \end{bmatrix}$ $\overbrace{\quad}^{3+2N}$ } 3+2N

$R_t = V_t \cdot \Sigma_{control} \cdot V_t^T$

$= \begin{bmatrix} * & * \\ * & * \\ * & * \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_w^2 \end{bmatrix} \cdot \begin{bmatrix} * & * & * & 0 & 0 & \cdots & 0 \\ * & * & * & 0 & 0 & \cdots & 0 \end{bmatrix} = \begin{bmatrix} [***] & [0] \\ [0] & [0] \end{bmatrix}_{(3+2N)}$

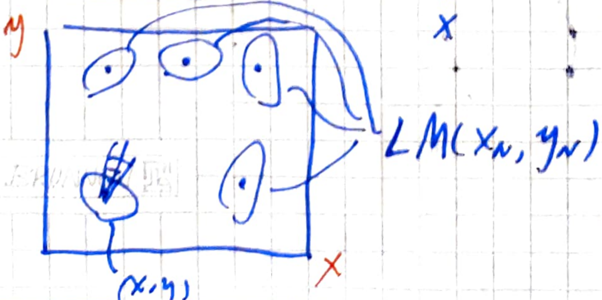
$(3+2N) \times (3+2N)$



$\Sigma_t = \begin{bmatrix} \begin{bmatrix} \sigma_x^2 & * & * \\ * & \sigma_y^2 & * \\ * & * & \sigma_\theta^2 \end{bmatrix} & [\quad] \\ [\quad] & \begin{bmatrix} \sigma_{x_1}^2 & \\ & \sigma_{y_1}^2 \\ & & \ddots \end{bmatrix} \end{bmatrix}$ } 3, } 2N

$\underbrace{\quad}_{3}$ $\underbrace{\quad}_{2N}$

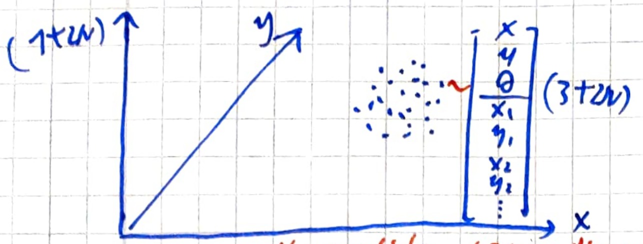
$\sigma_{x_1}^2 = \sigma_{y_1}^2 = \infty \approx 10^{10}$ (它刚开始很看不到)



$H = \begin{bmatrix} \frac{\partial h}{\partial x} & \frac{\partial h}{\partial y} & \frac{\partial h}{\partial \theta} & 0 & 0 & \frac{\partial h}{\partial x_m} & \frac{\partial h}{\partial y_m} & 0 & 0 \\ & & & 0 & 0 & & & 0 & 0 \end{bmatrix}$

$\underbrace{\quad\quad\quad}_{2N.}$

④. EKF-SLAM → PF SLAM



// 1 particle (3+2N) dimension.

- Curse of dimensionality
- particle filter scale exponentially with the number of dimensions.
- simple approach not feasible!

---

2. Particle Filter

① Factorization of the posterior 后验量的因式分解.

// posterior of full SLAM Problem. //particles //LM

$$P(x_{1:t}, m \mid z_{1:t}, u_{1:t}) = \underbrace{P(x_{1:t} \mid z_{1:t}, u_{1:t})}_{\text{represented using particles (path)}} \cdot \underbrace{\prod_{i=1}^{N} P(m_i \mid x_{1:t}, z_{1:t})}_{\text{Conditional (EKF) independence}}$$

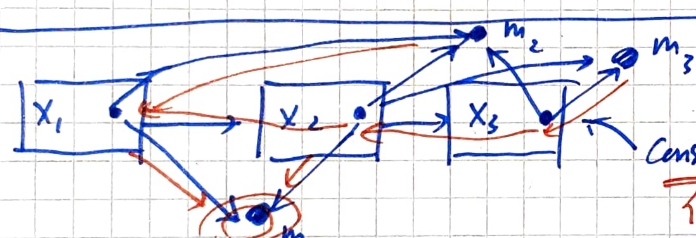
all path / landmark / all measurement / all control

// Rao-Blackwellized Particle Filter

represented using independent EKFs (one EKF per LM) ← // if the path $x_{1:t}$ is known → locations of the LMs are independent

---

②.



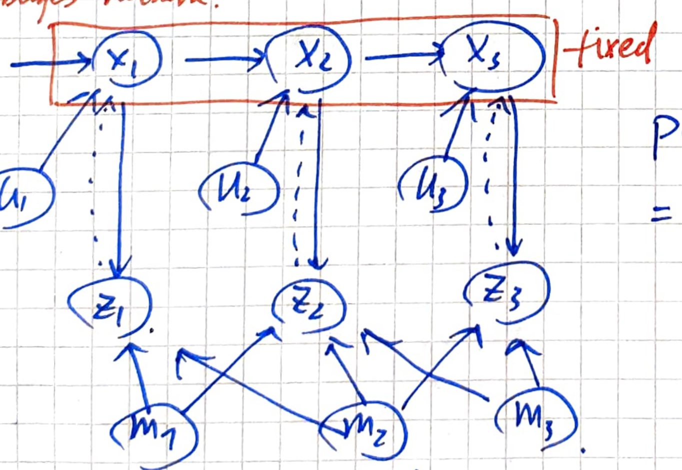
原本 m_3 与 m_2 受 x_1 影响.

Consider the x_1, x_2, x_3 are fixed

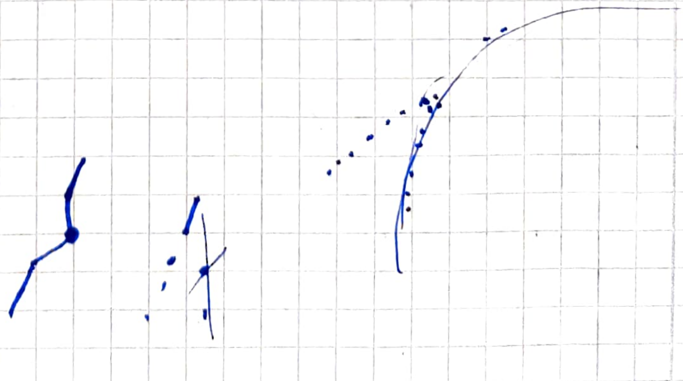
假设 x_1, x_2, x_3 已经被确认

则 m_1, m_2, m_3 independent

//dynamic Bayes network.



$$P(m \mid x_{1:t}, z_{1:t}) = \prod_{i=1}^{N} P(m_i \mid x_{1:t}, z_{1:t})$$

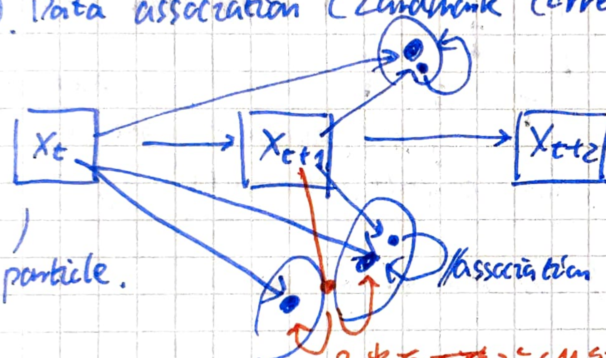


③. $\underbrace{P(x_{1:t} \mid z_{1:t}, u_{1:t})}_{\text{Particle Filter for the path}} \cdot \underbrace{\prod_{i=1}^{N} P(m_i \mid x_{1:t}, z_{1:t})}_{LM}$

PF for the path	LM 1	LM 2	…	LM N
particle 1: $x_{1:t}^{[1]}$	$\mu_1^{[1]}, \Sigma_1^{[1]}$	$\mu_2^{[1]}, \Sigma_2^{[1]}$	….	$\mu_N^{[1]}, \Sigma_N^{[1]}$
particle 2: $x_{1:t}^{[2]}$	$\mu_1^{[2]}, \Sigma_1^{[2]}$	$\mu_2^{[2]}, \Sigma_2^{[2]}$	….	$\mu_N^{[2]}, \Sigma_N^{[2]}$
⋮				
particle M: $x_{1:t}^{[M]}$	$\mu_1^{[M]}, \Sigma_1^{[M]}$	$\mu_2^{[M]}, \Sigma_2^{[M]}$	….	$\mu_N^{[M]}, \Sigma_N^{[M]}$

PF for the path — Individual EKF (one for each LM)

④. Data association (Landmark correspondences) LM 匹配.



? 当有~~…~~2个LM组合时, 下一时刻观测到的LM 很难判断是前2个中的哪一个.

⇒ each particle has its own data associations.

⇒ (likelihood) $L = \frac{1}{\sqrt{\det(2\pi Q)}} e^{-\frac{1}{2}\Delta z^T Q_t^{-1} \Delta z}$

☆ $P(\underset{\text{Particle state}}{x_{1:t}}, \underset{LM}{m} \mid \underset{\text{Measurement}}{z_{1:t}}, \underset{\text{Control}}{u_{1:t}}, \underset{\text{correspondences}}{c_{1:t}}) = \underbrace{P(x_{1:t} \mid z_{1:t}, u_{1:t}, c_{1:t})}_{\text{Particles, all the path}} \cdot \underbrace{\prod_{i=1}^{N} P(m_i \mid x_{1:t}, z_{1:t}, c_{1:t})}_{\text{LM, EKF (one for each LM)}}$

- Each Particle maintains individual data associations
  (EKF-SLAM only represents one particular sequence of data associations)
- Fast SLAM maintains posterior over multiple data associations

⑤. - Fast SLAM solves Full SLAM Problem.

particle i: | $x_{1:t}^{[i]}$ | $\mu_1^{[i]}, \Sigma_1^{[i]}$ | … | $\mu_N^{[i]} . \Sigma_N^{[i]}$ |

☆ all path

- Fast SLAM solves also online SLAM Problem

particle i: | $x_t^{[i]}$ | $\mu_1^{[i]}, \Sigma_1^{[i]}$ | … | $\mu_N^{[i]} . \Sigma_N^{[i]}$ |

☆ only last pose

⑥. Algorithmus.

$\boxed{\text{Prediction Step.}}$ // Same as normal particle. Filter

①. Init particles.

// 通过初始的分布采样 m 个粒子.

Init belief distribution. $bel(x_{t-1}) = N(\mu_{t-1}, \Sigma_{t-1})$

sample from Init distribution $x_{t-1}^{[m]} \sim N(\mu_{t-1}, \Sigma_{t-1})$

M particles: $\{x_{t-1}\} = \{x_{t-1}^{[1]}, \dots x_{t-1}^{[M]}\}$

②. predict with Control $P(x_t^{[m]} | x_{t-1}^{[m]}, u_t)$. $\bar{x}_t^{[m]} \sim P(x_t^{[m]} | x_{t-1}^{[m]}, u$

// 通过采样 Control 得到新的 m 个粒子.

sample from control ⇒ sample from prediction.

对 control 采样.

$$\text{sample} \begin{cases} l_t' \sim N(l_t, \sigma_{l_t}^2), & \sigma_{l_t}^2 = (\alpha_1 l)^2 + (\alpha_2 (l-r))^2 \\ r_t' \sim N(r_t, \sigma_{r_t}^2), & \sigma_{r_t}^2 = (\alpha_1 r)^2 + (\alpha_2 (l-r))^2 \end{cases}$$

$$\bar{x}_t^{[m]} = g(x_{t-1}^{[m]}, \begin{bmatrix} l_t' \\ r_t' \end{bmatrix})$$ ← 采样后的 control.

$\boxed{\text{Correction Step.}}$

//In class ~~class~~ Fast SLAM

correct(cylinder):

//更新并计算 weights
重新采样 weights, 得到particle.

weights = update_and_compute_weights (cylinders) // 每个 particle 一个 weight ← measurement

resample (weights) // resampling - wheel

//In class Fast SLAM

update_and_computer_weights(cylinders):

weights = [ ], for p in particles //对每一个粒子

weight = 1.0

//对每一个粒子, 经一对到的所有测量值, 更新测量值对粒子, 并返回该测量值对应的权重 weight
计算所有~~测量~~测量值的权重, 得到该粒子此时 ~~时~~ 对应的权重.

for m in cylinders //对每一个测量到的 LM

w = p. update_particle (m) //更新测量到的 LM 到每一个粒子. 并返回 weight 对应

weight = weight * w

weights. append(weight)

return weights.

//In class particle.

update_particle (m) ← measurement. $\begin{bmatrix} d \\ \alpha \end{bmatrix}$ - range - bearing angle.

particle:

x	y	θ

x_1 y_1	x_2 y_2	...	Σ_1	Σ_2	...

LMs position   LMs covariance.

//检查测量值是新 LM 还是旧 LM, 并返回一个权重.

Ⓐ. compute the likelihood of correspondence for any existing LM
//计算输入测量值与已有的 LM 的相似度.

Depending on result:

Ⓑ. initialize new LM // 没有一个新的 LM

Ⓒ update LM (likelihood is above a threshold) (EKF) // 根据相似度若超过一定阈值, 更新 LM (使用 EKF)

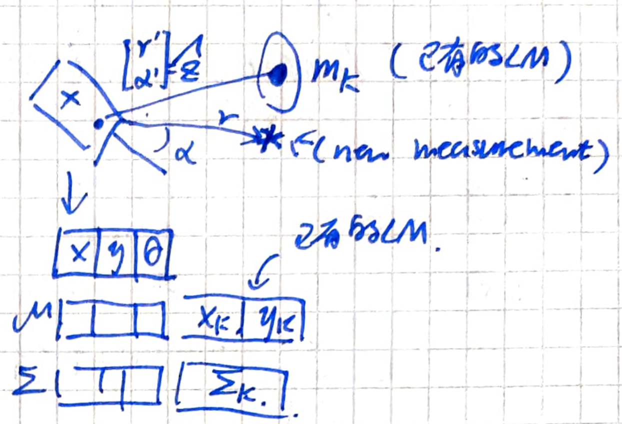
// In Class particle (cylinders)

update_particle (measurement)

Ⓐ. compute likelihood of existing LMs

⇒ { Ⓑ Initialize new LM (likelihood below threshold)
-or-
Ⓒ update LM using EKF
(likelihood above a threshold) }

Ⓐ. - Compute Likelihoods.



Expected Measurement

① . $\hat{z} = h(\underbrace{(x, y, \theta)}_{\text{LM的状态}}, m_k)$

$\hat{z}$ ← predicted measurement $\begin{bmatrix} d \\ \alpha \end{bmatrix}$ range, bearing angle.

m_k ← 已有的LM

② . $H = \left.\dfrac{\partial h}{\partial \text{Landmark}}\right|_{(x, y, \theta, m_k)}$ (2x2)

之前 EKF-SLAM 算过，就是 $\dfrac{\partial h}{\partial x}\ \dfrac{\partial h}{\partial y} \times (-1)$

③ . $Q_L = \underbrace{H \cdot \Sigma_k H^T}_{\text{measurement variance due to the Landmark variance.}} + \underbrace{Q_t}_{\text{variance due to the actual measurement.}}$

Q_L: covariance of the LM measurement. LM测量误差.

Σ_k: Covariance of LM_k

$Q_t = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix}$ 测量误差.

已存在的LM误差.

④ . $\Delta z = z - \hat{z}$

z: measured z; $\hat{z}$: expected $\hat{z}$

$\boxed{\text{Likelihood}}$ 相似度，越大越相似。PDF

★. $L = \dfrac{1}{2\pi\sqrt{\det(Q_L)}} \cdot e^{-\frac{1}{2}\Delta z^T Q_L^{-1} \Delta z}$

Ⓑ Initialize new LM.

$\hat{z} = h((x, y, \theta), m_k)$

$\hat{z}$ ← predicted measurement. $(\hat{r}, \hat{\alpha})$

m_k ← 现有的LM

Inverse.

$m = h^{-1}((x, y, \theta), z)$

m ← measured LM; z: (r, α), actual measurement.

$H = \left.\dfrac{\partial h}{\partial \text{Landmark}}\right|_{((x, y, \theta), m)}$

$\Sigma = H^{-1} \cdot Q_t \cdot (H^{-1})^T$, $Q_t = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix}$.

$\Rightarrow m, \Sigma$ $\quad m = \begin{bmatrix} x_m \\ y_m \end{bmatrix}$

Ⓒ. Landmark Update.

$K = \Sigma_{old} H^T (H \Sigma_{old} H^T + Q_t)^{-1}$

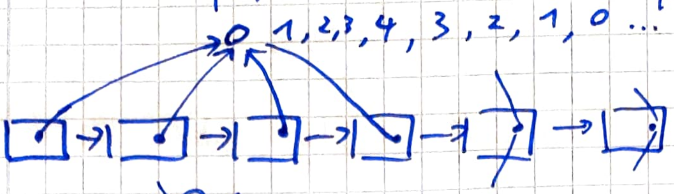
$= \Sigma_{old} H^T \cdot Q_L^{-1}$ 已存在的LM. 注意.

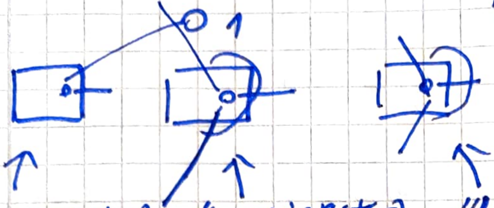
☆ $\begin{cases} \mu_{new} = \mu_{old} + K(z - h((x, y, \theta), \mu_{old})) \\ \Sigma_{new} = (I - KH) \cdot \Sigma_{old}. \end{cases}$

{ H, Q_L ... H_QL_jacobian_and_measurement_covariance_for_landmark()

$h()$ ... h_expected_measurement_for_landmark() }

⑦. Removal of previous landmarks





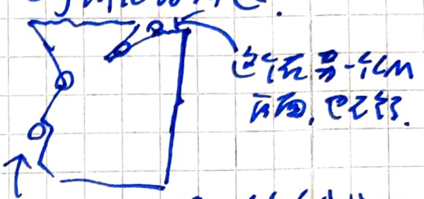
// LM在视野范围且看到，+1

// LM在视野范围但没有看到，-1

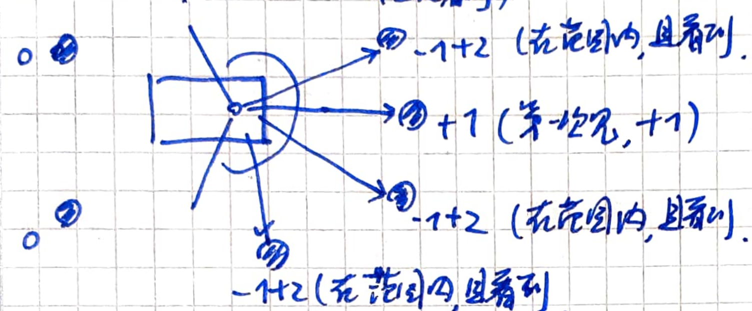
// LM并不在视野范围，因没有看到，并不 -1

// 当机器人看不到LM时，就将其丢掉，但这策略并不适合我们的Arena.

// 有个oversimplified 过于简化的问题.



⇒ 最终方案.



① 在范围内的LM 通通 -1

② 有匹配的observation +2.

③. 第一次见的LM +1.

⇒ ④ counter < 0 ⇒ delete.

---

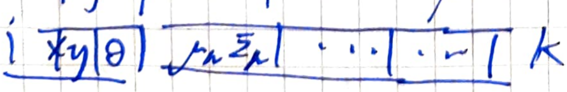
⑧ Efficiency of Fast SLAM.

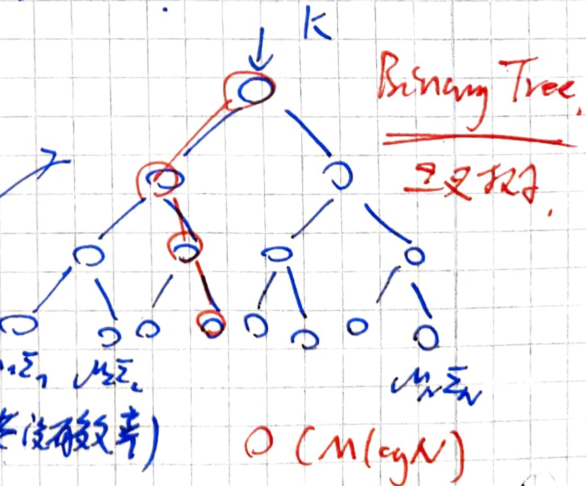
- Proposal distribution. (based on given control, which is very noisy)
  (建议分布) inefficiency
  "Fast SLAM 1.0"

  → Fast SLAM 2.0 modified proposal distribution.

- Map management

  M particles
  N map features (LM)
  time complexity $O(M \cdot N)$
  copy of the particles (整个复制太浪费效率)





3. Conclusion. FastSLAM

①
- Particle filters SLAM.
- Each particle is one path plus one map. (LMs). Full SLAM.
- Map features are independent (given the path).
  - → one (independent) EKF per feature
  - → In contrast to EKF SLAM: no correlations ⊕.
  - → maintains dependences only implicity ⊖
- Each particle uses its own data associations
  - → in contract to EKF SLAM
- Solves both: offline and online SLAM
  (Full) all path, (last pose)

☆.

② Fast SLAM 精简版.

$\boxed{\text{Predition Step}}$ //same as normal Particle Filter

①. sample from init distribution. $x_{t-1}^{[m]} \sim N(\mu_{t-1}, \Sigma_{t-1})$

②. sample from control $\overline{x_t}^{[m]} = g(x_{t-1}^{[m]}, \begin{bmatrix} l_t' \\ r_t' \end{bmatrix})$

sample $\begin{cases} l_t' \sim N(l_t, \sigma_{l_t}^2) \\ r_t' \sim N(r_t, \sigma_{r_t}^2) \end{cases}$

$\boxed{\text{Correction Step}}$

① update_and_compute_weights(cylinders)
measurement

⇒ 每次更新完所有粒子的权重后，重新采样粒子.

resample(weights).

//对每个粒子的每个测量值计算判断，是否是新的LM，并给出权重。返回该粒子总的权重

```
weights = []
for p in particles
    weight = 1.0
    for m in cylinders.
        w = p.update_particle(m)
        weight = weight * w
    weights.append(weight)
return weights.
```

② update_particle(m).

Ⓐ. likelihood.

//predicted measurement ① $\hat{z} = h((x,y,\theta), m_k)$
已有的LM_k

② $H = \frac{\partial h}{\partial LM}\Big|_{((x,y,\theta), m_k)}$, //等于 $\frac{\partial h}{\partial(x,y,\theta)} \times (-1)$

//covariance of LM measurement ③ $Q_L = H \Sigma_k H^T + Q_t$
Σ_k: covariance of LM_k; Q_t: covariance of measurement.

$\underbrace{H \Sigma_k H^T}$: measurement covariance due to LM covariance.

④ $\Delta z = \underset{\text{measured}}{z} - \underset{\text{predicted}}{\hat{z}}$

⑤ $L = \frac{1}{2\pi\sqrt{\det(Q_L)}} e^{-\frac{1}{2}\Delta z^T Q_L^{-1} \Delta z}$ ⇒ threshold.

Ⓑ. init new LM, $L <$ threshold measurement.

predicted LM $\begin{cases} m = h^{-1}((x,y,\theta), z) \\ H = \frac{\partial h}{\partial LM}\Big|_{((x,y,\theta), m)} \quad \frac{\partial h}{\partial LM} = \frac{\partial h}{\partial(x,y)} \times (-1) \\ \Sigma = H^{-1} \cdot Q_t \cdot (H^{-1})^T \end{cases}$

⇒ return m, Σ for the new LM

Ⓒ. update old LM. $L >$ threshold.

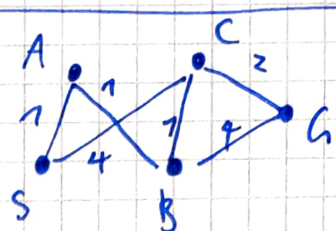
$\begin{cases} K = \Sigma_{old} H^T (H \Sigma_{old} H^T + Q_t)^{-1} \\ \quad = \Sigma_{old} H^T Q_L^{-1} \\ \mu_{new} = \mu_{old} + K(z - h((x,y,\theta), \mu_{old})) \\ \Sigma_{new} = (I - K \cdot H) \Sigma_{old} \end{cases}$

Unit PP.

- Dijkstra // 找到最短路径，避开障碍物.
- A^* // 结合 Greedy 贪心算法，减少寻找路径.
  // potential-function, keep away the path from obstacles.
- "car" planner // with minimum turn radio

---

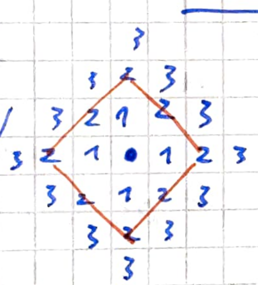
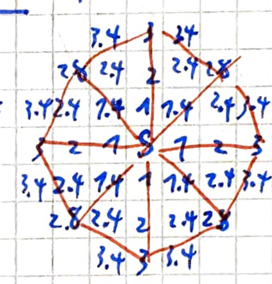
1. ~~Dijkstra~~-Algorithms.



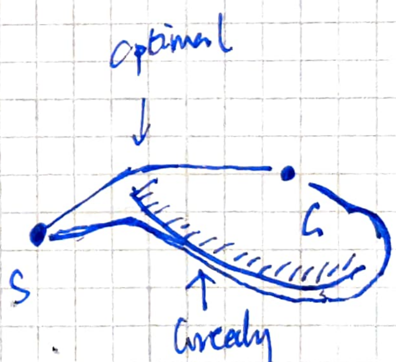
front		visited
0 S	✓	S
1 A	✓	A
4 C	skip	B
2 B	✓	C
3 C	✓	G
6 G	skip	
5 G	✓	

① front = start_node
while front not empty. (pop node n with min cost from HEAP(front))
  get node n with minimum cost to start.
  ☆. skip, if n has been visited already
  mark n as visited
  for any direct neighbor m of n, and m not visited
    add m to front. (push m on to HEAP(front))

②.

4N  8N: 

③. // efficiency problem, too slow?
- ① search min O(K)
- ② sort + pick min
  O(K log K) + O(1)
- ③ keep sorted O(K) + O(1)
- ④ HEAP O(log K)



④. path
/ remember the previous node of a node in front
VISITED[A] = cost
CAME-FROM [A] = D
记住每个 Front 中的 node 的 previous_node

Dijkstra — expands many nodes, optimal (最优)
Greedy — expands fewer nodes, suboptimal (次最优) 不一定最优.

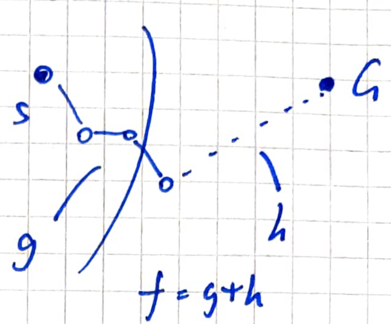
⑤. Dijkstra - ~~Cost~~ Greedy 贪心算法.
// 不是离 S 最近，而是离 Goal 最近. { min_cost to start / min_cost to Goal ✓

⑥. A* $f = g + h$

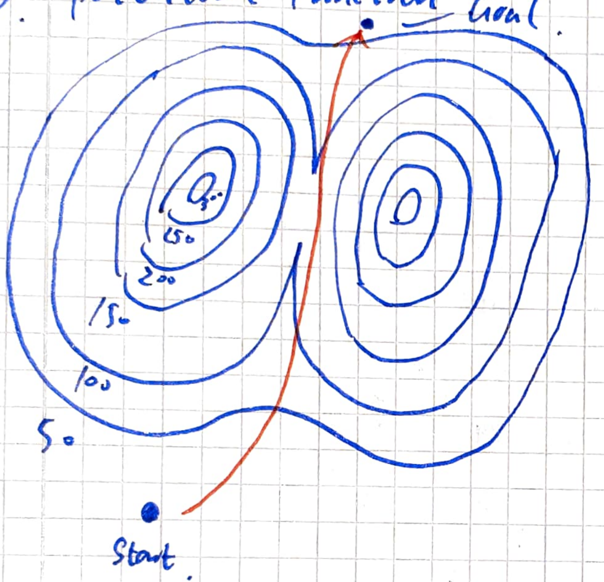
g: Dijkstra // cost from s to n

h: Greedy // cost from n to G.

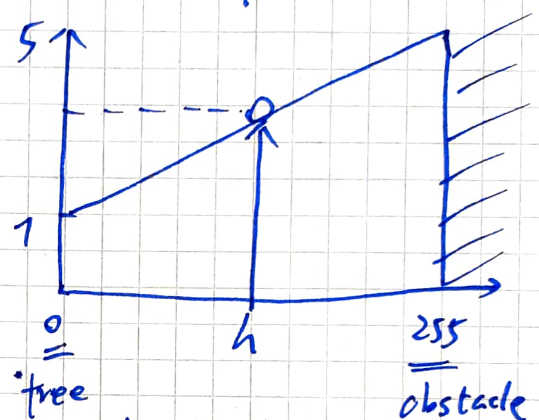
- usually expand less nodes than Dijkstra
- Generated to find optimal solution



⑦. potential Function

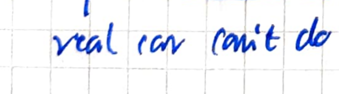


cost of see level

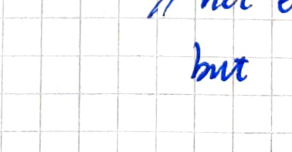


$\frac{\text{Obstacles}}{64}$ as potential_cost.

⑧. real car situation.



real car can't do that.

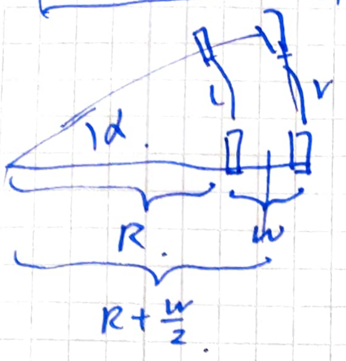


→ $l = 5$ // unit length

→ $c = \begin{cases} +1/10 \\ 0 \\ -1/10 \end{cases}$ // unit curve

// not exactly the goal but the tolerence of the goal

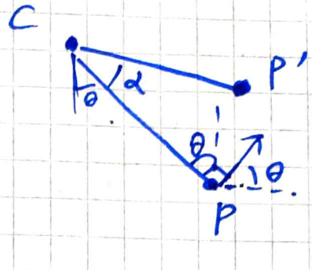
Motion Model



$$\begin{cases} R\alpha = l \\ (R+w)\alpha = r \end{cases}$$

$$w \cdot \alpha = r - l$$

$$\begin{cases} \alpha = \frac{r-l}{w} \\ R = \frac{l}{\alpha} \end{cases}$$



$$\begin{cases} C = P + (R+\frac{w}{2})\begin{pmatrix} -\sin\theta \\ +\cos\theta \end{pmatrix} \\ P' = C + (R+\frac{w}{2})\begin{pmatrix} \sin(\theta+\alpha) \\ -\cos(\theta+\alpha) \end{pmatrix} \\ P' = P + (R+\frac{w}{2})\begin{pmatrix} \sin(\theta+\alpha) - \sin\theta \\ -\cos(\theta+\alpha) + \cos\theta \end{pmatrix} \end{cases}$$

Similar Transformation

$$\min_i \| \lambda \vec{R}\, \vec{l_i} + \vec{t} - \vec{r_i} \|^2$$

① $\begin{cases} l_m = \frac{1}{m}\sum_i \vec{l_i} \\ r_m = \frac{1}{m}\sum_i \vec{r_i} \end{cases}$ ② $\begin{cases} \vec{l_i'} = \vec{l_i} - \vec{l_m} \\ \vec{r_i'} = \vec{r_i} - \vec{r_m} \end{cases}$

③. $\lambda = \sqrt{\dfrac{\sum_i \|\vec{r_i'}\|^2}{\sum_i \|\vec{l_i'}\|^2}}$ ④ $\begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} = \dfrac{\begin{bmatrix} \sum_i (r_x' l_x' + r_y' l_y') \\ \sum_i (-r_x' l_y' + r_y' l_x') \end{bmatrix}}{\left\| \begin{bmatrix} \sum_i (r_x' l_x' + r_y' l_y') \\ \sum_i (-r_x' l_y' + r_y' l_x') \end{bmatrix} \right\|}$

⑤. $\vec{t} = \vec{r_m} - \lambda \vec{R}\, \vec{l_m}$

Bayes - EKF

(P). $\overline{bel}(X_t) = \sum_{X_{t-1}} P(X_t \mid X_{t-1}, u_t)\, bel(X_{t-1})$ // $g(X_{t-1}, u_t)$ motion model

$N(X_t \mid \bar{\mu}_t, \bar{\Sigma}_t)$ $N(X_t \mid g(\mu_{t-1}, u_t))$ $N(X_{t-1} \mid \mu_{t-1}, \Sigma_{t-1})$

$$\begin{cases} \bar{\mu}_t = g(\mu_{t-1}, u_t) \\ \bar{\Sigma}_t = \underbrace{G_t}_{\frac{\partial g}{\partial state}} \Sigma_{t-1} G_t^T + R_t = G_t \Sigma_{t-1} G_t^T + \underbrace{V_t}_{\frac{\partial g}{\partial control}} \underbrace{\Sigma_{control}}_{\begin{bmatrix} \sigma_l^2 & 0 \\ 0 & \sigma_r^2 \end{bmatrix}} V_t^T \end{cases}$$

$$\begin{cases} \sigma_l^2 = (\alpha_1 l)^2 + (\alpha_2 (l-r))^2 \\ \sigma_r^2 = (\alpha_1 r)^2 + (\alpha_2 (l-r))^2 \end{cases}$$

(C). $bel(X_t) = \alpha\, P(z_t \mid X_t)\, \overline{bel}(X_t)$ // $h(X_t)$ measurement model

$N(\mu_t, \Sigma_t)$ $N(z_t \mid h(X_t), Q_t)$ $N(X_t \mid \bar{\mu}_t, \bar{\Sigma}_t)$

$$\begin{cases} K_t = \bar{\Sigma}_t \underbrace{H_t}_{\frac{\partial h}{\partial state}\ (2\times 3)}{}^T (H_t \bar{\Sigma}_t H_t^T + \underbrace{Q_t}_{\begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix}})^{-1} \quad \text{// kalman gain} \\ \mu_t = \bar{\mu}_t + K_t (\underbrace{z_t - h(\bar{\mu}_t)}_{\text{Innovation}}) \\ \Sigma_t = (I - K_t H_t) \cdot \bar{\Sigma}_t \end{cases}$$

Bayes Filter

$$\overline{bel}(x_t) = \sum_{x_{t-1}} P(x_t \mid x_{t-1}, u_t) \cdot bel(x_{t-1})$$ // prediction, convolution (movement)

$$bel(x_t) = \alpha \cdot P(z_t \mid x_t) \cdot \overline{bel}(x_t)$$ // correction, multiplication (measurement)

Kalman Filter

Bayes Filter with normal distribution. continuous.

prediction

$$\overline{bel}(x_t) = \int P(x_t \mid x_{t-1}, u_t) \cdot bel(x_{t-1})$$ // Bayes, convolution.

$$\mathcal{N}(x_t \mid \bar{\mu}_t, \bar{\sigma}_t^2) \qquad \mathcal{N}(x_t \mid \underbrace{a_t x_{t-1} + u_t}_{\text{motion model}}, \underbrace{\sigma_R^2}_{\text{movement noise}}) \quad \mathcal{N}(x_{t-1} \mid \mu_{t-1}, \sigma_{t-1}^2)$$ // normal distribution

$$\begin{cases} \bar{\mu}_t = a_t \mu_{t-1} + u_t & \text{// mean of } \overline{bel} \\ \bar{\sigma}_t^2 = a_t^2 \sigma_{t-1}^2 + \sigma_R^2 & \text{// variance of } \overline{bel} \end{cases}$$

Correction.

$$bel(x_t) = \alpha P(z_t \mid x_t) \overline{bel}(x_t)$$ // Bayes, multiplication

$$\mathcal{N}(x_t \mid \mu_t, \sigma_t^2) \qquad \mathcal{N}(z_t \mid c_t x_t, \sigma_Q^2) \quad \mathcal{N}(x_t \mid \bar{\mu}_t, \bar{\sigma}_t^2)$$ // normal distribution.

$$\begin{cases} K_t = \dfrac{c_t \bar{\sigma}_t^2}{c_t^2 \bar{\sigma}_t^2 + \sigma_Q^2} & \text{// kalman gain.} \\ \mu_t = \bar{\mu}_t + K_t \underbrace{(z_t - c_t \bar{\mu}_t)}_{\text{Innovation.}} & \text{// mean of bel} \\ \sigma_t^2 = (1 - K_t c_t) \bar{\sigma}_t^2 & \text{// variance of bel} \end{cases}$$

Kalman Filter.

1D, nD, discrete, linear, non linear (EKF)

1D linear.

Prediction: $x_t = a_t x_{t-1} + b_t u_t + \varepsilon_R$ (σ_R)

$$\overline{bel}(x_t) = \mathcal{N}(x_t \mid \bar{\mu}_t, \bar{\sigma}_t^2)$$

$$\begin{cases} \bar{\mu}_t = a_t \mu_{t-1} + b_t u_t \\ \bar{\sigma}_t^2 = a_t^2 \sigma_{t-1}^2 + \sigma_R^2 \end{cases}$$

correction: $z_t = c_t x_t + \varepsilon_Q$ (σ_Q)

$$bel(x_t) = \mathcal{N}(x_t \mid \mu_t, \sigma_t^2)$$

$$\begin{cases} K_t = \dfrac{c_t \bar{\sigma}_t^2}{c_t^2 \bar{\sigma}_t^2 + \sigma_Q^2} \\ \mu_t = \bar{\mu}_t + K_t (z_t - c_t \bar{\mu}_t) \\ \sigma_t^2 = (1 - K_t c_t) \bar{\sigma}_t^2 \end{cases}$$

nD linear.

Prediction: $x_t = A_t x_{t-1} + B_t \cdot u_t + \varepsilon_R$ (R_t)

$$\overline{bel}(x_t) = \mathcal{N}(x_t \mid \bar{\mu}_t, \bar{\sigma}_t^2)$$

$$\begin{cases} \bar{\mu}_t = A_t \cdot \bar{\mu}_{t-1} + B_t \cdot u_t \\ \bar{\Sigma}_t = A_t \cdot \Sigma_{t-1} \cdot A_t^T + R_t \end{cases}$$

Correction: $z_t = C_t \cdot x_t + \varepsilon_Q$ (Q_t)

$$bel(x_t) = \mathcal{N}(x_t \mid \mu_t, \sigma_t^2)$$

$$\begin{cases} K_t = \bar{\Sigma}_t \cdot C_t^T (C_t \cdot \bar{\Sigma}_t \cdot C_t^T + Q_t)^{-1} \\ \mu_t = \bar{\mu}_t + K_t (z_t - C_t \cdot \bar{\mu}_t) \\ \Sigma_t = (I - K_t \cdot C_t) \bar{\Sigma}_t \end{cases}$$

nD nonlinear, EKF

Prediction: $x_t = g(x_{t-1}, u_t)$

$$\overline{bel}(x_t) = \mathcal{N}(x_t \mid \bar{\mu}_t, \bar{\sigma}_t^2)$$

$$\begin{cases} \bar{\mu}_t = g(\mu_{t-1}, u_t) \\ \bar{\Sigma}_t = G_t \cdot \Sigma_{t-1} \cdot G_t^T + R_t \end{cases}$$

$R_t \rightarrow V_t \cdot \Sigma_{\text{control}} \cdot V_t^T$

Correction $z_t = h(x_t)$

$$bel(x_t) = \mathcal{N}(x_t \mid \mu_t, \sigma_t^2)$$

$$\begin{cases} K_t = \bar{\Sigma}_t \cdot H_t^T (H_t \cdot \bar{\Sigma}_t H_t^T + Q_t)^{-1} \\ \mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t)) \\ \Sigma_t = (I - K_t \cdot H_t) \cdot \bar{\Sigma}_t \end{cases}$$

$$Q_t = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\phi^2 \end{bmatrix}$$

$$G_t = \frac{\partial g}{\partial \text{state}}, \quad V_t = \frac{\partial g}{\partial \text{control}}$$

$$H_t = \frac{\partial h}{\partial \text{state}}$$

[Particle Filter]

Ⓟ.
① Sample from initial distribution.
m Particles. $X_{t-1}^{[m]} \sim \mathcal{N}(\mu_{t-1}, \sigma_{t-1}^2)$ // 采样 粒子.
$\{X_{t-1}\} = \{X_{t-1}^{[1]}, \cdots, X_{t-1}^{[M]}\}$
②. sample from control. $\begin{cases} l_t' \sim \mathcal{N}(l_t, \sigma_{l_t}^2) \\ r_t' \sim \mathcal{N}(r_t, \sigma_{r_t}^2) \end{cases}$ // 采样 Control
③. predict. $\bar{X}_t^{[m]} = g(\bar{X}_{t-1}^{[m]}, \begin{bmatrix} l_t' \\ r_t' \end{bmatrix})$ // prediction.

Ⓒ.
① ~~get_observation~~,
get_cylinders_from_scan
//assignment. ②. assign_cylinders (with LMs) ⇒ predicted measurement.
③. for every particles

$$\omega = P(z_t|X_t) = \prod_i P(z_{ti}|X_t)$$

weight for every particles

$$= \cancel{\prod_i P(z_i}$$

$$= \prod_i \mathcal{N}(\underbrace{d_i - d_i'}_{\text{(actual measurement - predicted measurement)}}, 0, \sigma_{d_i}^2)\, \mathcal{N}(\alpha_i - \alpha_i', 0, \sigma_{\alpha_i}^2)$$

for every observations.

// 计算权重.
$\omega = P(z_t|X_t)$
$= \prod_i P(z_{ti}|X_t)$
$= \prod_i \mathcal{N}(d - d', 0, \sigma_{d_i}^2) \cdot \mathcal{N}(\alpha - \alpha', 0, \sigma_{\alpha_i}^2)$

④ Resampling.
- draw i with probability $\propto w_t^{[i]}$
- Resampling wheel. offset ~ uniform(0, 2·max_weight)

// resampling.

[Density Estimation]

gaussian approximation
$$\hat{x} = \frac{1}{m}\sum_i^M X_i \quad \text{// mean position.}$$
$$\begin{cases} V_x = \frac{1}{m}\sum_i^M \cos\theta_i \\ V_y = \frac{1}{m}\sum_i^M \sin\theta_i \end{cases} \quad \hat{\theta} = \mathrm{atan2}(V_y, V_x) \quad \text{// mean heading angle.}$$

① Histogram
②. Kernel density estimation. (multi modal)
③ K-means
④. gaussian approximation. (uni modal)

EKF-SLAM.

(P). $\underbrace{\bar{\mu}_t}_{(3+2N)\times 1} = g(\mu_{t-1}, u_t)$

$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + \underbrace{R_t}_{(3+2N)\times(3+2N)}$ $\quad 2\times 2$: $\begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}$

$\underbrace{\quad}_{(3+2N)(3+2N)} = \underbrace{\underbrace{G_t}_{(3+2N)\times(3+2N)} \underbrace{\Sigma_{t-1}}_{(3+2N)\times(3+2N)} G_t^T}_{(3+2N)\times(3+2N)} + \underbrace{V_t}_{(3+2N)\times 2} \Sigma_{control} \underbrace{V_t^T}_{2\times(3+2N)}$

(C). $\underbrace{K_t}_{(3+2N)\times 2} = \underbrace{\underbrace{\bar{\Sigma}_t}_{(3+2N)\times(3+2N)} \underbrace{H_t^T}_{(3+2N)\times 2}}_{(3+2N)\times 2} \underbrace{(\overbrace{\underbrace{H_t}_{2\times(3+2N)} \underbrace{\bar{\Sigma}_t}_{(3+2N)\times(3+2N)} \underbrace{H_t^T}_{(3+2N)\times 2}}^{2\times 2} + \overbrace{Q_t}^{2\times 2})^{-1}}_{2\times 2}$

$(3+2N)\times 2$.

$\underbrace{\mu_t}_{(3+2N)\times 1} = \underbrace{\bar{\mu}_t}_{(3+2N)\times 1} + \underbrace{\underbrace{K_t}_{(3+2N)\times 2} (\underbrace{z_t}_{2\times 1} - \underbrace{h(\bar{\mu}_t)}_{2\times 1})}_{(3+2N)\times 1}$

$\underbrace{\Sigma_t}_{(3+2N)\times(3+2N)} = \underbrace{(\underbrace{I}_{(3+2N)\times(3+2N)} - \underbrace{K_t}_{(3+2N)\times 2} \underbrace{H_t}_{2\times(3+2N)})}_{(3+2N)\times(3+2N)} \cdot \underbrace{\bar{\Sigma}_t}_{(3+2N)\times(3+2N)}$

// 任一时刻点，对观测到的 LM 进行校正，若 LM 是新的，则添加到 state 中。

$\mu_t = \begin{bmatrix} x \\ y \\ \theta \\ x_1 \\ y_1 \\ x_2 \\ y_2 \\ \vdots \end{bmatrix} (3+2N)$

$G_t = \begin{bmatrix} \begin{bmatrix} 1 & 0 & * \\ 0 & 1 & * \\ 0 & 0 & 1 \end{bmatrix} & 0 \\ 0 & \begin{matrix} 1 & & \\ & 1 & \\ & & 1 \end{matrix} \end{bmatrix} 3+2N.$

$\Sigma_t = \underbrace{\begin{bmatrix} \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{x\theta} \\ \sigma_{xy} & \sigma_y^2 & \sigma_{y\theta} \\ \sigma_{x\theta} & \sigma_{y\theta} & \sigma_\theta^2 \end{bmatrix} & [\,0\,] \\ [\,0\,] & \begin{bmatrix} \sigma_{x_1}^2 & & & \\ & \sigma_{x_2}^2 & & \\ & & \sigma_{y_1}^2 & \\ & & & \sigma_{y_2}^2 \end{bmatrix} \end{bmatrix}}_{3+2N} \Big\} 3+2N$

$V_t = \underbrace{\begin{bmatrix} * & * \\ * & * \\ * & * \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}}_{2} (3+2N)$

$R_t = \underbrace{\begin{bmatrix} \begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix} & [\,0\,] \\ [\,0\,] & \begin{bmatrix} 0 & \\ & 0 \end{bmatrix} \end{bmatrix}}_{(3+2N)} \Big\} (3+2N)$

$H_t = \begin{array}{c|ccc|cc|cc|c} & \frac{\partial h}{\partial x} & \frac{\partial h}{\partial y} & \frac{\partial h}{\partial \theta} & \frac{\partial h}{\partial x_1} & \frac{\partial h}{\partial y_1} & \frac{\partial h}{\partial x_j} & \frac{\partial h}{\partial y_j} & \\ \hline r & * & * & * & 0 & 0 & * & * & \cdots \\ \alpha & * & * & * & 0 & 0 & * & * & \end{array}$

$\times(-1)$

$2\times(3+2N)$

SLAM - EKF

$\mu_t = \begin{bmatrix} x \\ y \\ \theta \\ x_1 \\ y_1 \\ x_2 \\ y_2 \\ \vdots \end{bmatrix} (3+2N)$

Ⓟ. $\overline{bel}(x_t) = \sum_{x_{t-1}} P(x_t \mid x_{t-1}, u_t)\, bel(x_{t-1})$

$N(x_t \mid \bar{\mu}_t, \bar{\Sigma}_t)$ $\quad N(x_t \mid \underbrace{g(x_{t-1}, u_t)}_{\text{motion model}}, \underbrace{R_t}_{\text{motion covariance}})$ $\quad N(x_{t-1} \mid \mu_{t-1}, \Sigma_{t-1})$

$$\begin{cases} \bar{\mu}_t = g(\mu_{t-1}, u_t) \\ \bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t = G_t \Sigma_{t-1} G_t^T + V_t \Sigma_{control} V_t^T \end{cases}$$

$G_t = \frac{\partial g}{\partial state}$ $(3+2N)\times(3+2N)$, Σ_{t-1}: $(3+2N)\times(3+2N)$, $\underbrace{G_t \Sigma_{t-1} G_t^T}_{(3+2N)\times(3+2N)}$

$V_t = \frac{\partial g}{\partial control}$ $\cancel{2\times(3+2N)}$ $(3+2N)\times 2$, $\Sigma_{control} = \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_w^2 \end{bmatrix}$ 2×2, V_t^T: $2\times(3+2N)$, $\underbrace{V_t \Sigma_{control} V_t^T}_{(3+2N)\times(3+2N)}$

$G_t = \frac{\partial g}{\partial state} = \underbrace{\begin{bmatrix} \begin{bmatrix} 1 & 0 & * \\ 0 & 1 & * \\ 0 & 0 & 1 \end{bmatrix} & [0] \\ [0] & \begin{bmatrix} 1 & & \\ & 1 & \\ & & 1 \end{bmatrix} \end{bmatrix}}_{(3+2N)\times(3+2N)}$

$V_t = \begin{bmatrix} * & * \\ * & * \\ * & * \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$ $\cancel{2\times(3+2N)}$ $(3+2N)\times 2$.

Ⓒ. $bel(x_t) = \alpha P(z_t \mid x_t)\, \overline{bel}(x_t)$

$N(x_t \mid \mu_t, \Sigma_t)$ $\quad N(z_t \mid \underbrace{h(x_t)}_{\text{measurement model}}, \underbrace{Q_t}_{\text{measurement covariance}})$ $\quad N(x_t \mid \bar{\mu}_t, \bar{\Sigma}_t)$

$$\begin{cases} K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + Q_t)^{-1} \\ \mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t)) \\ \Sigma_t = (I - K_t H_t) \cdot \bar{\Sigma}_t \end{cases}$$

★. $R_t = \begin{bmatrix} \begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix} & [0] \\ [0] & \begin{bmatrix} 0 & & \\ & 0 & \\ & & 0 \end{bmatrix} \end{bmatrix}$ $(3+2N)\times(3+2N)$

$H_t = \frac{\partial g}{\partial control} = \begin{array}{c} \\ r \\ \alpha \end{array} \begin{array}{ccc} \frac{\partial h}{\partial x} & \frac{\partial h}{\partial y} & \frac{\partial h}{\partial \theta} \\ \hline * & * & * \\ * & * & * \end{array}$

$\begin{array}{c} \\ r \\ \alpha \end{array} \begin{array}{cc} \frac{\partial h}{\partial x_j} & \frac{\partial h}{\partial y_j} \\ \hline * & * \\ * & * \end{array}$ $\times(-1)$

★. $H_t = \begin{bmatrix} * & * & * & 0 & 0 & \cdots & * & * & \cdots \\ * & * & * & 0 & 0 & & * & * & \end{bmatrix}$ $\times(-1)$

$2\times(3+2N)$

Particle - ~~EKF~~ SLAM

all the path, all measurement, all control, LM

$$P(X_{1:t}, m \mid Z_{1:t}, U_{1:t}, C_{1:t}) = P(X_{1:t} \mid Z_{1:t}, U_{1:t}, C_{1:t}) \cdot \prod_{i=1}^{N} P(m_i \mid X_{1:t}, Z_{1:t}, C_{1:t})$$

correspondence.

particle 1 $X_{1:t}^{[1]}$ $\boxed{\mu_1^{[1]}, \Sigma_1^{[1]}}, \ldots, \boxed{\mu_N^{[1]}, \Sigma_N^{[1]}}$

particle 2 $X_{1:t}^{[2]}$ $\boxed{\mu_1^{[2]}, \Sigma_1^{[2]}}, \ldots, \boxed{\mu_N^{[2]}, \Sigma_N^{[2]}}$

⋮

particle M $X_{1:t}^{[M]}$ $\boxed{\mu_1^{[M]}, \Sigma_1^{[M]}}, \ldots, \boxed{\mu_N^{[M]}, \Sigma_N^{[M]}}$

PF — individual EKF.

☆☆. $$P(X_{1:t}, m \mid Z_{1:t}, U_{1:t}, C_{1:t}) = \underbrace{P(X_{1:t} \mid Z_{1:t}, U_{1:t}, C_{1:t})}_{\text{PF prediction sampling}} \underbrace{\prod_{i}^{N} P(m_i \mid X_{1:t}, Z_{1:t}, C_{1:t})}_{\text{individual EKF correction. weight.}}$$

(P). //same as particle filter.

① sample m particles from init distribution.

$$\{X_{t-1}\} = \{X_{t-1}^{[1]}, \ldots, X_{t-1}^{[m]}\} \sim N(\mu_{t-1}, \Sigma_{t-1})$$

②. sample $\begin{cases} l_t' \sim N(l_t, \sigma_{l_t}^2) \\ r_t' \sim N(r_t, \sigma_{r_t}^2) \end{cases}$ ③ prediction $\bar{X}_t^{[m]} = g(X_{t-1}^{[m]}, \begin{bmatrix} l_t' \\ r_t' \end{bmatrix})$

(C). ① compute weights.

```
for every particle,
    for m in cylinders.
        ω = p.update_particle(m)
        weight = weight · ω.
    weights.append(weight)
return weights.
```

Update_particle:

Ⓐ likelihood

$$\hat{z} = h((x, y, \theta), m_k) \quad \begin{cases} x_k \\ y_k \\ \Sigma_k \end{cases} \text{ existing } LM_k$$

$$H = \frac{\partial h}{\partial LM} = \left[\frac{\partial h}{\partial x}\ \frac{\partial h}{\partial y}\right] x{-}1.$$

$$Q_L = H \Sigma_k H^T + Q_t \quad \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix}$$

$$\Delta z = z - \hat{z}$$

$$\mathcal{L} = \frac{1}{2\pi\sqrt{\det(Q_L)}} \cdot e^{-\frac{1}{2}\Delta z^T Q_L^{-1} \Delta z}$$

Ⓑ. likelihood < threshold

Init new LM

$$\begin{cases} m = h^{-1}(x, y, \theta, z) \\ H = \frac{\partial h}{\partial LM} = \left[\frac{\partial h}{\partial x}, \frac{\partial h}{\partial y}\right] x{-}1 \\ \Sigma = H^{-1} \cdot Q_t \cdot (H^{-1})^T \end{cases}$$

☆

$\Rightarrow m, \Sigma$

Ⓒ likelihood > threshold

$$\begin{cases} K = \Sigma_{old} H^T (H \Sigma_{old} H^T + Q_t)^{-1} = \Sigma_{old} H^T Q_L^{-1} \\ \mu_{new} = \mu_{old} + K(z - h((x, y, \theta), \mu_{old})) \\ \Sigma_{new} = (I - KH) \cdot \Sigma_{old} \end{cases}$$

② sampling based on weights.

Dijkstra

front = Start_node

while front not empty.

①
- ~~get~~ get node_n with min_cost from Start. (normal Dijkstra)
- get node_n with min_cost from ~~Start~~ Start. And to Goal. (A^*, Greedy)
- get node_n with min_cost from Start, to Goal, and potential cost. (potential cost)
- pop node_n ~~from~~ with min_cost from HEAP (front) (HEAP)

② . skip, if node_n already visited

③ . mark node_n as visited

④ . for any direct neighbor m of n, and ~~m~~ not visited
add m to front (push m onto HEAP (front))

Ⓐ . likelihood.

$\hat{z} = h((x, y, \theta), m_k)$ $\quad m_k = \begin{bmatrix} x_k \\ y_k \\ \Sigma_k \end{bmatrix}$

predicted measurement.

$H = \frac{\partial h}{\partial LM} \Big|_{(x,y,\theta), m_k} = \left[\frac{\partial h}{\partial x} \; \frac{\partial h}{\partial y} \right] \times (-1)$

$Q_L = H \Sigma_k H^T + Q_t$ $\quad Q_t = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix}$

$\Delta t = z - \hat{z}$

$\Rightarrow L = \cancel{\frac{1}{\det(\dots)}} \frac{1}{2\pi \sqrt{\det(Q_L)}} e^{-\frac{1}{2} \Delta z^T Q_L^{-1} \Delta z}$

Ⓑ . Likelihood < threshold.

Init new LM.

$m_k = h^{-1}((x, y, \theta), z_k)$

$H = \frac{\partial h}{\partial LM} = \left[\frac{\partial h}{\partial x}, \frac{\partial h}{\partial y} \right] \times (-1)$

$\Sigma_k = H^{-1} \cdot Q_t \cdot (H^{-1})^T$

$\rightarrow (m_k, \Sigma_k)$

Ⓒ . Likelihood > Threshold

use EKF to update LM_k

$K_k = \Sigma_{old} H^T (H \Sigma_{old} H^T + Q_t)^{-1}$

$= \Sigma_{old} H^T Q_L^{-1}$

$\mu_{new} = \mu_{old} + K_k (z_k - h((x,y,\theta), \mu_{old}))$

$\Sigma_{new} = (I - KH) \Sigma_{old}$.

$w = P(z|x) = \prod_i P(z_i|x)$

$= \prod_i N(d - d', 0, \sigma_d^2) N(\alpha - \alpha', 0, \sigma_\alpha^2)$