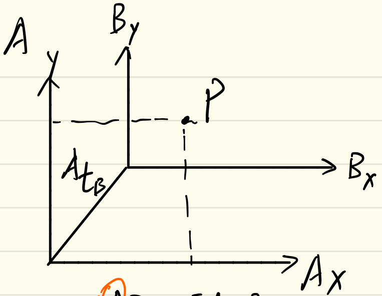
Algorithm for Sensor-Based Robotics



$${}^A t_B \in \mathbb{R}^2$$

$${}^A p = {}^A t_B + {}^B p$$

Frame

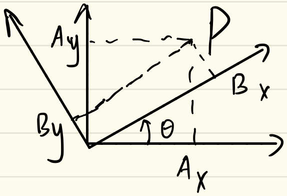
$${}^A p = \begin{bmatrix} A_x \\ A_y \end{bmatrix}$$

$${}^B p = \begin{bmatrix} B_x \\ B_y \end{bmatrix}$$

$${}^B p = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow {}^A p = {}^A t_B$$

Inverse: ${}^B t_A = -{}^A t_B$ $\qquad {}^B p = {}^B t_A + {}^A p$

Rotation. 2D



$${}^A p = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} {}^B p$$

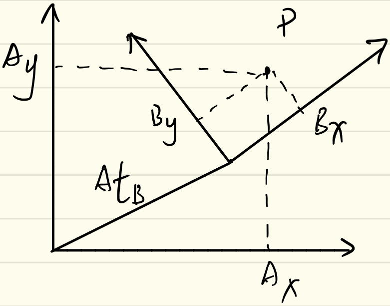
$$\begin{bmatrix} A_x \\ A_y \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} B_x \\ B_y \end{bmatrix}$$

$${}^A p = {}^A R_B \, {}^B p \qquad {}^A R_B \in SO(2)$$

Special Orthogonal 2D

$\det(R) = 1 \qquad R \cdot R^T = I$

Inverse: ${}^B R_A = {}^A R_B^{-1} = {}^A R_B^T$



$$^AP = {^AR_B}\,{^BP} + {^At_B}$$

Non-linear.

$$^AP = \begin{bmatrix} ^Ax \\ ^Ay \\ 1 \end{bmatrix} \qquad ^BP = \begin{bmatrix} ^Bx \\ ^By \\ 1 \end{bmatrix}$$

homogeneous coordinates.

$$^AP = {^AE_B}\,{^BP}$$

$$\begin{bmatrix} ^Ax \\ ^By \\ 1 \end{bmatrix} = \begin{bmatrix} \overset{2\times2}{^AR_B} & \overset{2\times1}{^At_B} \\ \underset{1\times2}{0} & 1 \end{bmatrix} \begin{bmatrix} ^Bx \\ ^By \\ 1 \end{bmatrix}$$

Inverse

$$^BE_A = {^AE_B}^{-1} = \begin{bmatrix} \overset{2\times2}{^AR_B^T} & \overset{2\times1}{-{^AR_B^T}\,{^At_B}} \\ \underset{1\times2}{0} & 1 \end{bmatrix}$$

homogeneous Transformation.

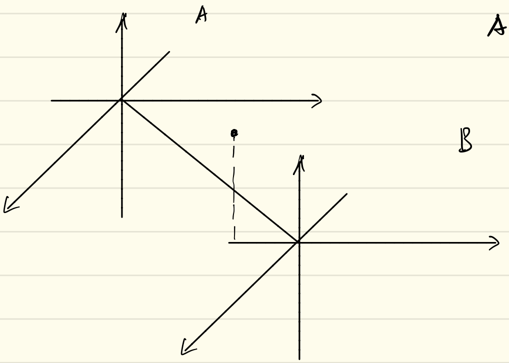
$$^AE_B \in SE(2)$$

Special Euclidean 2D

$${}^{A}t_{B} = \{ (t, R) : t \in R^2, R \in SO(2) \}$$

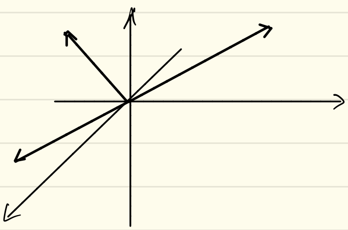
$$\mathbb{R}^2 \times SO(2)$$

3D translation.



$${}^{A}P = {}^{A}t_{B}\, {}^{B}P \qquad {}^{A}t_{B} \in \mathbb{R}^3$$

3D Rotation.



$${}^{A}P = {}^{A}R_{B}\, {}^{B}P \qquad R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{33} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

$$R = R_x R_y R_z$$

$${}^{A}P = (R_x R_y R_z)\, {}^{B}P$$

$$(R_z R_y R_x)\, {}^{B}P$$

$^AP = {}^AR_B\,{}^BP + {}^At_B$ affine equation.

$$\begin{bmatrix}{}^AP \\ 1\end{bmatrix} = {}^AE_B\,{}^BP = \begin{bmatrix} {}^AR_B & {}^At_B \\ 0\;0\;0 & 1 \end{bmatrix}\begin{bmatrix}{}^BP \\ 1\end{bmatrix}$$

From B to A

3X3 $SO(3)$

3x1 $\mathbb{R}^3$

$^AE_B \in SE(3)$

special Euclidean transformation

$\{(t, R),\ t \in \mathbb{R}^3,\ R \in SO(3)\}$

Scale or Shrink (Not Rigid)

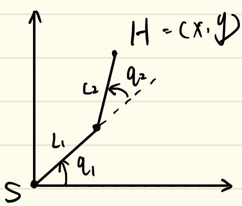
$$^AP = {}^AE_B\,{}^BP = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Not a legal R. , $\det(R) \neq 1$

$R \cdot R^T \neq I$

$$^AE_B^{-1} = {}^BE_A = \begin{bmatrix} {}^AR_B & {}^At_B \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} {}^AR_B^T & -{}^AR_B^T\,{}^At_B \\ 0 & 1 \end{bmatrix}$$

2D Rigid Motion.

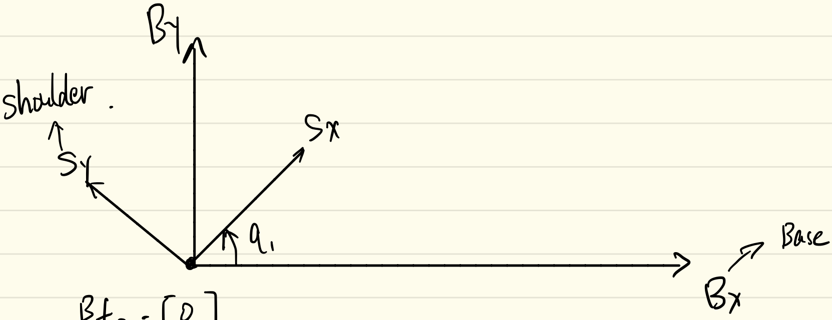


$E_x = L_1 \cos(q_1)$

$E_y = L_1 \sin(q_1)$

$H_x = E_x + L_2 \cos(q_1 + q_2) = L_1 \cos(q_1) + L_2 \cos(q_1 + q_2)$

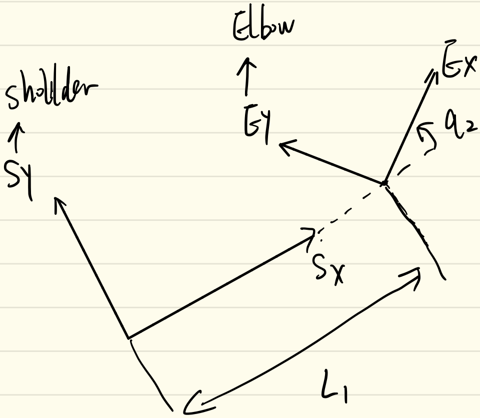
$H_y = E_y + L_2 \sin(q_1 + q_2) = L_1 \sin(q_1) + L_2 \sin(q_1 + q_2)$



$$^{B}t_S = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$^{B}R_S = \begin{bmatrix} \cos(q_1) & -\sin(q_1) \\ \sin(q_1) & \cos(q_1) \end{bmatrix}$$

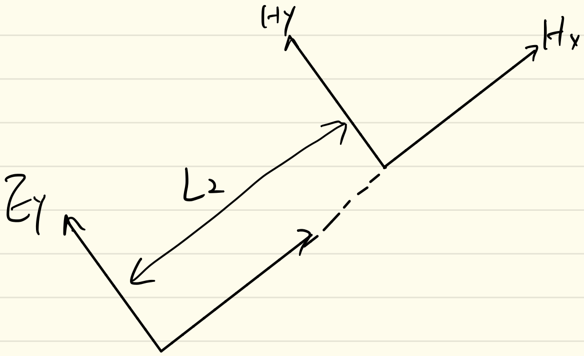
$$^{B}E_S = \begin{bmatrix} ^{B}R_S & ^{B}t_S \\ 0 & 1 \end{bmatrix}$$



$$^{S}t_E = \begin{bmatrix} L_1 \\ 0 \end{bmatrix}$$

$$^{S}R_E = \begin{bmatrix} \cos(q_2) & -\sin(q_2) \\ \sin(q_2) & \cos(q_2) \end{bmatrix}$$

$$^{S}E_E = \begin{bmatrix} ^{S}R_E & ^{S}t_E \\ 0 & 1 \end{bmatrix}$$



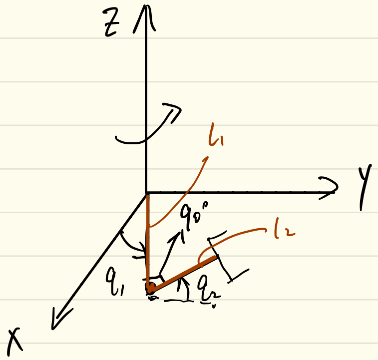
$${}^{E}t_{H} = \begin{bmatrix} L_2 \\ 0 \end{bmatrix}$$

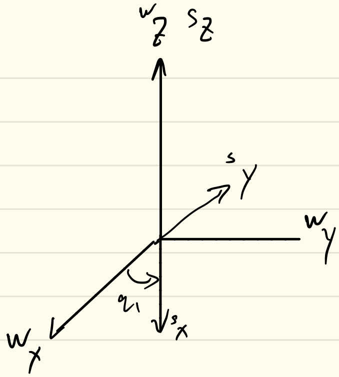
$${}^{E}R_{H} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$${}^{E}E_{H} = \begin{bmatrix} 1 & 0 & L_2 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} {}^{B}E_{H} &= {}^{B}E_{S} \cdot {}^{S}E_{E} \cdot {}^{E}E_{H} \\ &= {}^{B}E_{S}(q_1) \cdot {}^{S}E_{E}(q_2) \cdot {}^{E}E_{H} \\ &= \begin{bmatrix} {}^{B}R_{H}(q_1 q_2) & {}^{B}t_{H}(q_1, q_2) \\ 0 & 1 \end{bmatrix} \end{aligned}$$

Matlab : rot(q_1)



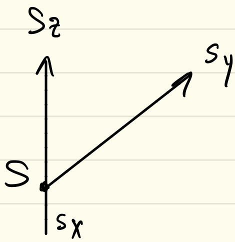


$${}^{w}t_s \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

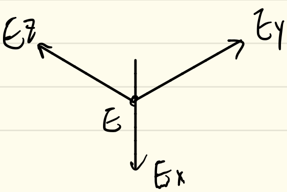
$${}^{w}R_s(q) = \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 \\ \sin(q_1) & \cos(q_1) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^{w}E_s = \begin{bmatrix} {}^{w}R_s & {}^{w}t_s \\ 0\,0\,0 & 1 \end{bmatrix} = \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 & 0 \\ \sin(q_1) & \cos(q_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

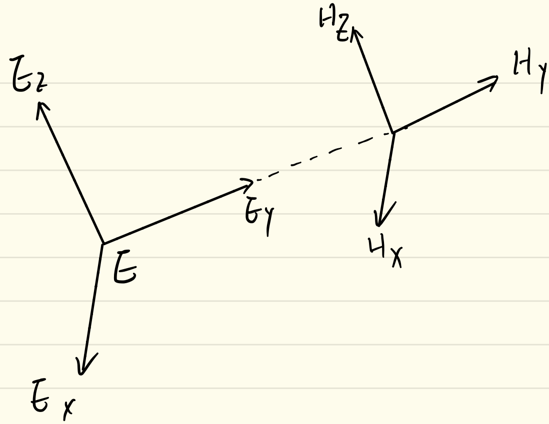
2)



$${}^{s}t_E = \begin{bmatrix} l_1 \\ 0 \\ 0 \end{bmatrix} \qquad {}^{s}R_E = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(q_2) & -\sin(q_2) \\ & \sin(q_2) & \cos(q_2) \end{bmatrix}$$



$${}^{s}E_E = \begin{bmatrix} {}^{s}R_E & {}^{s}t_E \\ 0\,0\,0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & l_1 \\ 0 & \cos(q_2) & -\sin(q_2) & 0 \\ 0 & \sin(q_2) & \cos(q_2) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



$$ {}^{E}t_H = \begin{bmatrix} 0 \\ l_2 \\ 0 \end{bmatrix} $$

$$ {}^{E}R_H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{E}E_H = \begin{bmatrix} {}^{E}R_H & {}^{E}t_H \\ 0\,0\,0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & l_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{W}E_H = {}^{W}E_S(q_1)\, {}^{S}E_E(q_2)\, {}^{E}E_H $$

$$ = \begin{bmatrix} {}^{W}R_H & {}^{W}t_H \\ 0\,0\,0 & 1 \end{bmatrix} $$

$H:(x, y, z)$

$$ = \begin{bmatrix} \cos(q_1) & -\cos(q_2)\sin(q_1) & \sin(q_1)\sin(q_2) & l_1\cos(q_1) - l_2\cos(q_2)\sin(q_1) \\ \sin(q_1) & \cos(q_1)\cos(q_2) & -\cos(q_1)\sin(q_2) & l_1\sin(q_1) + l_2\cos(q_1)\cos(q_2) \\ 0 & \sin(q_2) & \cos(q_2) & l_2\sin(q_2) \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$${}^{w}\vec{V}_n = \sum_{i=1}^{2} \left(\frac{\partial E_H(q)}{\partial q_1} \; {}^{w}E_H^{-1}(q_1) \right) \dot{q}_1$$

Matlab: diff(Ewh, 'q1') * inv (Ewh)

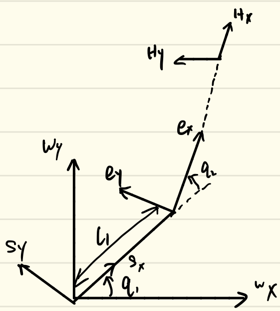
$$i=1 \quad \frac{\partial\, {}^{w}E_H(q)}{\partial q_1}\, {}^{w}E_H^{-1}(q) = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\hat{w} = \begin{bmatrix} 0 & -w_z & w_y \\ w_z & 0 & -w_x \\ w_y & w_x & 0 \end{bmatrix}$$

$$i=2 \quad \frac{\partial\, {}^{w}E_H(q_1)}{\partial q_2}\, {}^{w}E_H^{-1}(q) = \begin{bmatrix} 0 & 0 & \sin q_1 & 0 \\ 0 & 0 & -\cos q_1 & 0 \\ -\sin q_1 & \cos q_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \cos q_1 \\ \sin q_1 \\ 0 \end{bmatrix}$$

$$J(q_1) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \cos q_1 \\ 0 & \sin q_1 \\ 1 & 0 \end{bmatrix} \qquad \begin{bmatrix} {}^{w}V_H^{s} \\ {}^{w}\omega_H^{s} \end{bmatrix} = J(q_1) \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}$$

$$\begin{bmatrix} V_x \\ V_y \\ V_z \\ w_x \\ w_y \\ w_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \cos(q_1)\, \dot{q}_1 \\ \sin(q_1)\, \dot{q}_2 \\ \dot{q}_1 \end{bmatrix}$$



$${}^{w}E_H = \begin{bmatrix} {}^{w}R_H & {}^{w}t_H \\ 0\,0\,0 & 1 \end{bmatrix}$$

$${}^{w}\hat{V}_n^{s} = \sum_{i=1}^{2} \frac{\partial\, {}^{w}E_H}{\partial q_i}\, {}^{w}E_H^{-1}\, \dot{q}_i$$

$$J(q) = \Big[$$

$q_1 = 0,\ q_2 = 0 .\ \dot{q}_1 = 0,\ \dot{q}_2 = 1$

$$\begin{bmatrix} V_x \\ V_y \\ W \end{bmatrix} = J(q) \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}$$

Hand-Eye Calibration.

$A_1 = \Bigg[$

$$\log(R) = \hat{\omega}\theta = \frac{\theta}{2\sin\theta}(R - R^T), \quad \theta = \cos^{-1}\left(\frac{tr(R)-1}{2}\right)$$

Using $\alpha_1 = R_x\beta_i \rightarrow R_x\beta_i - \alpha_1 = 0$

$$R \in SO(3) \quad \sum_{i=1}^{N} \| R\beta_i - \alpha_i \|^2$$

$$= \sum_{i=1}^{N} (R\beta_i - \alpha_i)^T (R\beta_i - \alpha_i)$$

$$= \sum_{i=1}^{N} (\beta_i^T R^T R \beta_i - \beta_i^T R^T \alpha_i - \alpha_i^T R \beta_i + \alpha_i^T \alpha_i)$$

$$= \sum_{i=1}^{N} (\beta_i^T \beta_i - \beta_i^T R^T \alpha_i - \alpha_i^T R \beta_i + \alpha_i^T \alpha_i) \quad \text{No "R"}$$

$$= \sum_{i=1}^{N} (-\beta_i^T R^T \alpha_i - \alpha_i^T R \beta_i)$$

$$= \sum_{i=1}^{N} (-2\beta_i^T R^T \alpha_i)$$

$$= \underset{R \in SO(3)}{\operatorname{argmin}} -2\,\mathrm{trace}\left(R \sum_{i=1}^{N} \beta_i \alpha_i^T\right)$$

$$M = \sum_{i=1}^{N} \beta_i \alpha_i^T$$

$$\underset{R \in SO(3)}{\arg\max} \quad \mathrm{trace}(RM)$$

Polar Decomposition

$M = US$ $\rightarrow$ positive semi-definite matrix

$\hookrightarrow$ orthognal matrix

$S = (M^T M)^{1/2}$ $\qquad$ $U = M(M^T M)^{-1/2}$

Maximize $\quad tr(RUS) \rightarrow$ PSD

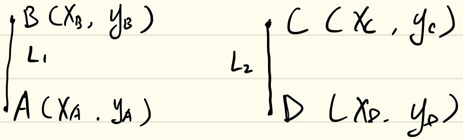
Rotation

$tr(A)$ = sum eignvalues

Maximum $tr(RUS)$ is attained, when $RU = I$

$R = U^T = (M(M^T M)^{-1/2})^T$

$R = (M^T M)^{-1/2} M^T$



Dimension of C-space.

1) Dimension of C-space.

2) $d(A, B) = L_1$. $\dim(C\text{-space}) = 7$

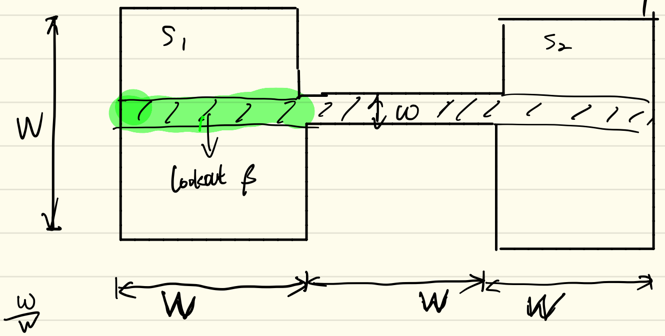
3) $d(C, D) = L_2$. $\dim(C\text{-space}) = 6$

4) $B = C$. $\dim(C\text{-space}) = 4$

5) $A = (0, 0)$. $\dim(C\text{-space}) = 2$.

$\epsilon \quad \alpha \quad \beta$

smallest reachability in the middle of corridor.



$w << W$

$\mu(Q_{free}) = (2W+w)W$

$$\epsilon = \frac{3wW}{(2W+w)W} \approx \frac{w}{W}$$

$\frac{w}{W}$

$$\approx \beta = \frac{2wW}{(W+w)W}$$

$2wW \longrightarrow$ Volume of subset reachable from S_1

$(W+w)W \longrightarrow$ Volume of $Q_{free} \longrightarrow$ Volume of S_1

$(2W+w)W - W^2$.

$$\alpha = \frac{wW}{W^2} = \frac{w}{W}$$

lookout

β of S_1 : Volume of free space that is outside the subset.

State Space

What's a state? Position, Velocity, efforts, maps.

$$\begin{array}{cccc} \text{often} & \text{sometimes} & \text{seldom} & \text{mapping} \\ \downarrow & \downarrow & \downarrow & \downarrow \\ \text{Position,} & \text{Velocity,} & \text{efforts,} & \text{maps.} \end{array}$$

$$X_t = \begin{bmatrix} x_t \\ y_t \\ z_t \\ \hline \dot{x}_t \\ \dot{y}_t \\ \dot{z}_t \\ \hline \vdots \end{bmatrix}$$

Our goal is to find the state of the robot.

Given 1) How the state transition from t to $t+1$ (an equation)

2) How the state is observed with a sensor (an equation)

3) Transitions and observations are subject to Gaussian noise

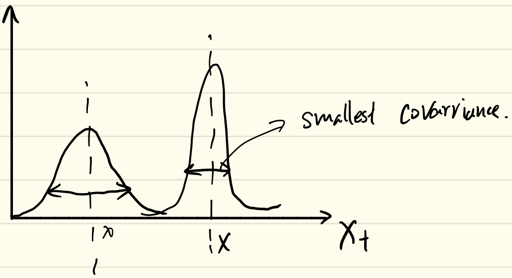
What's an observation?

$$Z_t = \begin{bmatrix} \text{long} \\ \text{lat} \\ \text{Alti} \\ \dot{x}_t \\ \vdots \\ \vdots \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix}a\\a\\a\end{matrix}}\right\} \text{GPS} \\ \left.\vphantom{\begin{matrix}a\\a\\a\end{matrix}}\right\} \text{techometer.} \end{matrix}$$

Kalman Filters

Given a sequence of transitions and observations.

KF will compute the postier state estimate with the smallest covarriance.



$X_t = $ 1-D position

KF Assumptions.

1) state transition

external input Command

$$\boxed{X_{t+1} = A\,X_t} + Bu_t + W_t \rightarrow \text{noise with a known covariance.}$$

$$W_t \sim N(0, Q)$$

linear equation for state transition.

If you know X_t and A, $\rightarrow$ know X_{t+1}

$$X_t = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \qquad X_{t+1} = \begin{bmatrix} X_1 & 0 \\ 0 & X_2 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} + w_t$$

$\times$ Not linear.

$$X_{t+1} = \begin{bmatrix} \cos & -\sin \\ \sin & \cos \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + w_t$$

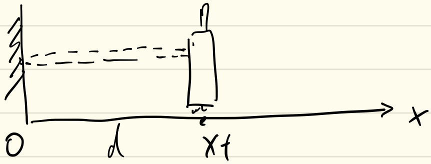
2) Observations

$$\boxed{Z_t = H X_t} + v_t \quad \text{noise with covariance } R \qquad v_t \sim N(0, R)$$

observation vector $\in \mathbb{R}^n$

Linear

if we know X_t and H, then we know ---



$$Z_t = \left[\frac{1}{2v}\right] x_t + v_t \sim N(0, R)$$

$$\frac{d}{2} = vt \Rightarrow t = \frac{d}{2v}$$

State Transition.

Know system covariance, you must figure out Q

$$X'_{t+1} = A X_t + B u_t + w_t \sim \boxed{N(0, Q)}$$

Given X_t, we can compute X_{t+1} subject to noise w_t.

If Q is large, we have noisy transition.

~ small, ~ accurate ~.

State Observation

$$Z_t = H X_t + u_t \rightarrow N(0, R)$$

If R large, noisy measurement

small, accurate.

If $Q \rightarrow 0$, perfect system. $X_{t+1} = X'_{t+1}$

If $R \rightarrow 0$, perfect sensor. $X_t = H^{-1} Z_t$

How much weight ("faith") should we put on state transitions. vs sensor?

Kalman Filter does that. not only that. it does it opitmally.

Given $x'_{t+1} = Ax_t + Bu_t + w_t$

x'_{t+1} = a prior state estimate (AKA Prediction)

KF computes a posteriori state estimate.

$$\hat{x}_{t+1} = x'_{t+1} + \underbrace{K_t \, (z_t - Hx'_t)}_{\text{correction to the a priori estimate}}$$

K_t: Kalman Gain

$z_t - Hx'_t$: error between measurement and predicted.

z_t: real measurement.

Hx'_t: predicted measurement.

$z_t - Hx'_t$ is known as "innovation" or "measurement residual"

if prediction is good. sensor is bad. $K_t = ? \Rightarrow 0$

if prediction is bad, sensor is good. $K_t = ? \neq 0$

Find K_t that is optimal trade off between sensor and Prediction

Covariance Matricies

$Q = E[w_t w_t^T] \Rightarrow w_t \sim N(0, Q)$

$R = E[v_t v_t^T] \rightarrow v_t \sim N(0, R)$

if X_t is the true state (we don't know X_t) and $\hat{X}_t$ is the a posteriori estimate

$$e = X_t - \hat{X}_t \Rightarrow \text{a posteriori estimate error}$$

$$e' = X_t - X_t' \rightarrow \text{a priori estimate error}$$

$$\Sigma' = E[e_t' e_t'^T] \Rightarrow \text{a priori estimate error covariance}$$

$$\Sigma = E[e_t e_t^T] \Rightarrow \text{a posteori estimate error covariance.}$$

$$\hat{X}_t = X_t' + K_t(HX_t + V_t - HX_t')$$

$$\Sigma_t = E\left[\left[(I - K_tH)(X_t - X_t') - K_tV_t\right]\left[(I - K_tH)(X_t - X_t') - K_tV_t\right]^T\right]$$

$$e_t = X_t - X_t'$$

$$\Sigma = (I - K_tH)\underbrace{E[(X_t - X_t')(X_t - X_t')^T]}(I - K_tH)^T + K_t\underbrace{E[V_tV_t^T]}K_t^T$$

$$= (I - K_tH)\,\Sigma_t\,(I - K_tH)^T + K_tRK_t^T$$

The diagnoal of Σ_t contains the mean squared errors.

$$\Sigma_t = E[e_t e_t^T] = \begin{bmatrix} E[e_1e_1] & E[e_1e_2] & \cdots & E[e_1e_N] \\ E[e_2e_1] & & & \vdots \\ \vdots & & & \\ & & & E[e_Ne_N] \end{bmatrix}$$

trace: $tr(\Sigma) = E[e_1 e_1] + \cdots E[e_N e_N]$

Minimize trace → use minimize the sum of squared errors.

$$\Sigma_t = \Sigma'_t - K_t H \Sigma'_t - \Sigma'_t H^T K_t^T + K_t (H \Sigma'_t H^T + R) K_t^T \quad (10)$$

$$tr(xy^T) = tr(x^T y)$$

$$tr(\Sigma_t) = tr(\Sigma'_t) - 2tr(K_t H \Sigma'_t) + tr(K_t (H \Sigma'_t H^T + R) K_t^T)$$

We want to find K_t that minimize $tr(\Sigma_t)$, so derive wrt K_t and set to zero.

$$\frac{d\, tr(\Sigma'_t)}{dK_t} = -2(H \Sigma'_t)^T + 2K_t (H \Sigma'_t H^T + R) = 0$$

solve for K_t

$$(H \Sigma_t)^T = K_t (H \Sigma_t H^T + R)$$

$$K_t = \Sigma'_t H^T (H \Sigma'_t H^T + R)^{-1} \quad (11)$$

Plug (11) into (10).

$$\Sigma_t = \Sigma'_t - \Sigma'_t H_t (H \Sigma'_t H^T + R)^{-1} H \Sigma'_t$$

$$= \Sigma'_t - K_t H \Sigma'_t$$

$$\Sigma_t = (I - K_t H) \Sigma'_t$$

$$e'_{t+1} = x_{t+1} - x'_{t+1}$$

Prediction based on posterior state estimate. at time t.

$$= (Ax_t + w_t) - A\hat{x}_t$$

$$\hat{x}_{t+1} = x'_{t+1} + k_t(z_t - Hx'_{t+1})$$

$$= Ae_t + w_t$$

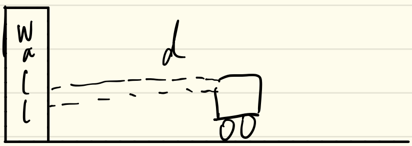
$$e_t = x_t - \hat{x}_t$$

$$\Sigma'_{t+1} = E[e'_{t+1} e'^T_{t+1}]$$

$$= E[(Ae_t + w_t)(Ae_t + w_t)^T]$$

$$= E[(Ae_t)(Ae_t)'] + E[w_t w_t^T]$$

$$\Sigma'_{t+1} = A\Sigma_t A^T + Q$$



HC-SR04

$$2d = vt \Rightarrow t = \frac{2d}{v} \Rightarrow z_t = Hx_t + v_t$$

$N(0, R)$

$$\boxed{z_t = \frac{2}{v} x_t + v_t}$$

$R = 0.05\ s^2$

340 m/s → speed of sound.

second.

noise of the sensor.

Robot

$x_{t+1} = A_{xt} + B_{ut} + w_t$

$x_{t+1} = x_t + u_t + w_t \rightarrow N(0, Q)$ $\quad Q = 0.0001\, m^2$

$x_0 = 1m$. $\Sigma_0 = 0.5 m^2$, $U_0 = 0.01 m$

Step1 :

$x_1' = x_0 + 0.01 = 1 + 0.01 = 1.01\ m$ (Prediction)

$\Sigma_1' = A \Sigma_0 A^T + Q = 1 \times 0.5 \times 1 + 0.0001 = 0.5001 m^2$ (Variance of Pred

Step2 :

$z_1 = 0.20445$

$K_1 = 0.5001/170 \left(\frac{1}{170} \cdot 0.5001 \cdot \frac{1}{170} + 0.05 \right)^{-1}$

$= 0.0588$

$\hat{x} = x_1' + K_1 (z_1 - H x_1')$

$= 1.01 + 0.0588 (0.2044 - 0.0059)$

$= 1.0216\ m$ (posterior state)

step 3 :

$\Sigma_1 = (1 - K_1 H) \Sigma_1' = 0.4999 m^2$ covariance of

$\Sigma_1 = E[\underbrace{(x_1 - \hat{x}_1)}_{e_t} \underbrace{(x_1 - \hat{x}_1)^T}_{e_t^T}]$

In 2D

noise $\sim N(0, Q)$

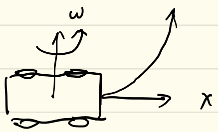
$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \\ \theta_{t+1} \end{bmatrix} = \begin{bmatrix} x_t + \delta t\, v_t \cos(\theta_t) + w_{x_t} \\ y_t + \delta t\, v_t \sin(\theta_t) + w_{y_t} \\ \theta_t + \delta t\, \omega_t + w_{\theta_t} \end{bmatrix}$$

$$ {}^{B}E'_{M+1} = {}^{B}E_{M_t}\, {}^{M}E_{M_{t+1}} $$

Predicted Pos/Ori of Robot at time $t+1$ — Current - - - $\cdots t$ — small incremental motion

moving in x-axis.

$$= \begin{bmatrix} \cos(\theta_t) & -\sin(\theta_t) & x_t \\ \sin(\theta_t) & \cos(\theta_t) & y_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\delta t\, \omega_t) & -\sin(\delta t\, \omega_t) & \delta t\, v_t \\ \sin(\delta t\, \omega_t) & \cos(\delta t\, \omega_t) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



$$= \begin{bmatrix} \cos(\theta_t + \delta t\, \omega_t) & -\sin(\theta_t + \delta t\, \omega_t) & x_t + \delta t\, v_t \cos(\theta_t) \\ \sin(\theta_t + \delta t\, \omega_t) & \cos(\theta_t + \delta t\, \omega_t) & y_t + \delta t\, v_t \sin(\theta_t) \\ 0 & 0 & 1 \end{bmatrix}$$

In 3D

$$ {}^{B}E'_{M_{t+1}} = {}^{B}E_{M_t}\,{}^{M_t}E_{M_{t+1}} = \begin{bmatrix} {}^{B}R_{M_t} & {}^{B}t_{M_t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{M_t}R_{M_{t+1}} & {}^{M_t}t_{M_{t+1}} \\ 0 & 1 \end{bmatrix} $$

wrt world

wrt itself (local)

$$ = \begin{bmatrix} R_z(y)\,R_y(P)\,R_x(r) & \begin{matrix} {}^{B}x_{M_t} \\ {}^{B}y_{M_t} \\ {}^{B}z_{M_t} \end{matrix} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_z(\Delta t\,\omega t) & \\ 0 & 1 \end{bmatrix} $$

Predicted α, β, γ buried in here

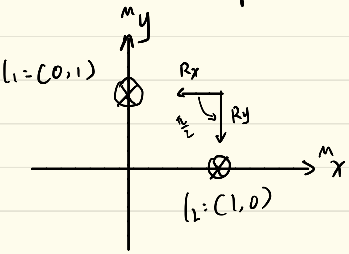
Predicted x, y, z.

$$ = \begin{bmatrix} {}^{B}R_{M_{t+1}} & {}^{B}t_{M_{t+1}} \\ 0 & 1 \end{bmatrix} $$

wrt world.

EKF Range-Bearing Localization in 2D

We have a map (according to landmarks)



$l_1 = (x_{l_1}, y_{l_1}) = (0, 1)$

$l_2 = (x_{l_2}, y_{l_2}) = (1, 0)$

$$ z_t = \begin{bmatrix} r_1 \\ b_1 \\ r_2 \\ b_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ \pi/2 \end{bmatrix} $$

Robot System Model

$S_t = [\, x_t \quad y_t \quad \theta_t \,]^T$

$u_t = [\, v_t \quad w_t \,]^T$

$$\left.\begin{aligned} x_{t+1} &= x_t + \Delta t\, v_t \boxed{\cos(\theta_t)} + w_{xt} \\ y_{t+1} &= y_t + \Delta t\, v_t \boxed{\sin(\theta_t)} + w_{yt} \\ \theta_{t+1} &= \theta_t + \Delta t\, \omega_t + w_{\theta t} \end{aligned}\right\}$$

not linear in S_t

$S_{t+1} = A s_t + B u_t + w_t$

so we can't use "plain" Kalman Filter.

Range - Bearing Sensor.

$$z_{l_1} = h_1(S_t, v_t) = \begin{bmatrix} \sqrt{(x_t - x_{l_1})^2 + (y_t - y_{l_1})^2} \\ \tan^{-1}\left(\dfrac{y_t - y_{l_1}}{x_t - y_{l_1}}\right) - \theta_t \end{bmatrix} + \begin{bmatrix} v_{Rt} \\ v_{Bt} \end{bmatrix}$$

distance between the robot and landmark # 1

angle between the robot and landmark #1 (aka "bearing")

$z_{l_2} = h_2(S_t, V_t) = \cdots\cdots$

not linear, can't write as $z_t = H S_t + v_t$

Jacobian of System Model

$$A(S_t) = \begin{bmatrix} \dfrac{\partial x_{t+1}}{\partial x_t} & \dfrac{\partial x_{t+1}}{\partial y_t} & \dfrac{\partial x_{t+1}}{\partial \theta_t} \\ \dfrac{\partial y_{t+1}}{\partial x_t} & \dfrac{\partial t_{t+1}}{\partial y_t} & \dfrac{\partial y_{t+1}}{\partial \theta_t} \\ \dfrac{\partial \theta_{t+1}}{\partial x_t} & \dfrac{\partial \theta_{t+1}}{\partial y_t} & \dfrac{\partial \theta_{t+1}}{\partial \theta_t} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\Delta t\, v_t \sin(\theta_t) \\ 0 & 1 & \Delta t\, v_t \cos(\theta_t) \\ 0 & 0 & 1 \end{bmatrix}$$

linear approximation of S_{t+1}

$$Z_t = \begin{bmatrix} h_1(S_t, V_t) \\ h_2(S_t, V_t) \end{bmatrix}$$

$$H(S_t) = \begin{bmatrix} \frac{\partial h_{r_1}}{\partial x_t} & \frac{\partial h_{r_1}}{\partial y_t} & \frac{\partial h_{r_1}}{\partial \theta_t} \\ \frac{\partial h_{b_1}}{\partial x_t} & \frac{\partial h_{b_1}}{\partial y_t} & \frac{\partial h_{b_1}}{\partial \theta_t} \\ \frac{\partial h_{r_2}}{\partial x_t} & \frac{\partial h_{r_2}}{\partial y_t} & \frac{\partial h_{r_2}}{\partial \theta_t} \\ \frac{\partial h_{b_2}}{\partial x_t} & \frac{\partial h_{b_2}}{\partial y_t} & \frac{\partial h_{b_2}}{\partial \theta_t} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{x_t - x_{l_1}}{\sqrt{(x_t - x_{l_1})^2 + (y_t - y_{l_1})^2}} & - - - & 0 \\ \frac{-(y_t - y_{l_1})}{\sqrt{(\quad)^2 + (\quad)^2}} & \cdot - - & -1 \\ \vdots & & 0 \\ \vdots & & -1 \end{bmatrix}$$

$P(Z \mid X)$ Probability of observing sensor data Z_t given state X_t

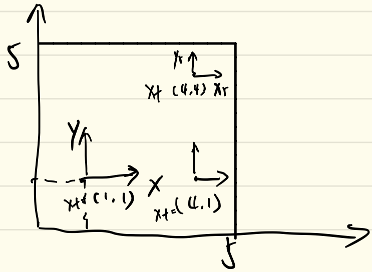
$r_1 = 1$

$r_2 = 1$

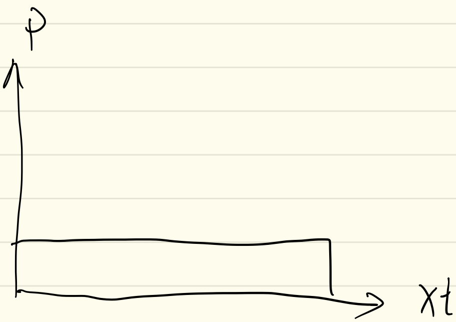
$r_3 = 1$

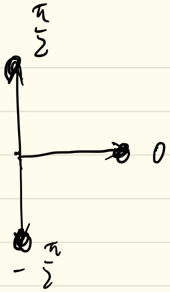
Robot only translate.

$f(x_t) = (1,1)$



$P(x_{t+1} \mid x_t, u_t)$





$$r = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

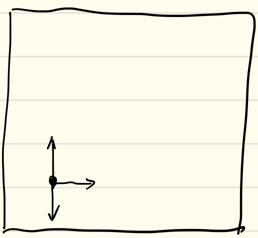
$$\sigma_{Hit} = 0.5$$

$$P_{Hit} = \eta_{Hit} \, N(d, \sigma_{Hit})$$

$$\eta_{Hit} = \left(\int_0^{\infty} N(d, \sigma_{Hit}) \right)^{-1}$$

$$\approx 0.4$$

$$s_1 = [1, 1]^T$$



$d_1 = 1$

$d_2 = 1$

$d_3 = 0$

$$P_{Hit_1} = \eta_{Hit} \, N(d_1, \sigma_{Hit}) = 0.054$$

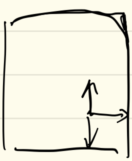
$$P_{Hit_2} = \eta_{Hit} \, N(d_2, \sigma_{Hit}) = 0.054$$

$$P_{Hit_3} = \eta_{Hit} \, N(d_3, \sigma_{Hit}) = 0.4$$

$$P = P_{Hit_1} + P_{Hit_2} + P_{Hit_3} = 0.001$$

0	0	0	0	0	0
0	1	1	1	1	0
0	1	2	2	1	0
0	1	2	2	1	0
0	1	1	1	1	0
0	0	0	0	0	0

$$s = [4, 1]^T$$



$d_1 = 1 \quad d_2 = 0 \quad d_3 = 0$

$P_{Hit_1} = 0.054 \quad P_{Hit_2} = 0.4 \quad P_{Hit} = 0.4$

$P = \cdots = 0.007$

$s = [4, 4]^T \qquad P = 0.007$