סדנא מפגש שני

טרנספורמ לב— ליאושכ כין —.

חלק I : (12 40)

- ליצור env חדש בשם robotics
  עם פייתון 3.9
- לעשות clone ל— PythonRobotics
- להתקין dep עם requirements.text
- לדבר קצת על התחום הרובוטיקה:

  קיל מפל את 3 דבר

  control
  localization
  pathplanning → pathtracking
- להראות קצת את בנים בסימולציה.

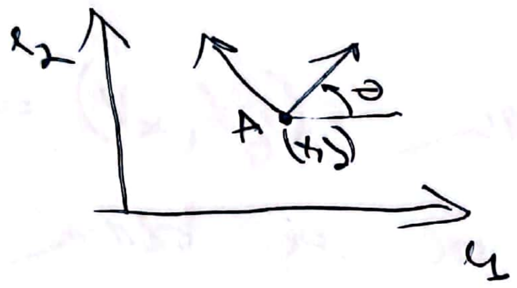
חלק II : טרנספורמציה (שי 3)

מאמר ללות :

$SE\text{-}2 \Rightarrow$ special euclidian 2

$$A \in SE2$$

$$A = (x, y, \theta)$$



אם היה משתנה רק בכיוון e_1

$$\dot{x} = V_x \Rightarrow x = x_0 + V_x t$$

ציר קרטזי : $A(t) = (x_0 + V_x t,\ y_0 + V_y t,\ \theta_0)$

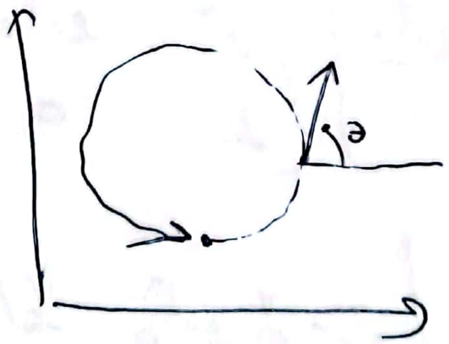
ציר סיבובי : $A(t) = (x_0,\ y_0,\ \theta_0 + \omega t)$

אבל אי אפשר ישיר. —

$$A \neq (x_0 + v_x t,\ y_0 + v_y t,\ \theta_0 + \omega t)$$

$$\underline{v} = (v_x, v_y)$$ נקודה המהירות הקווית

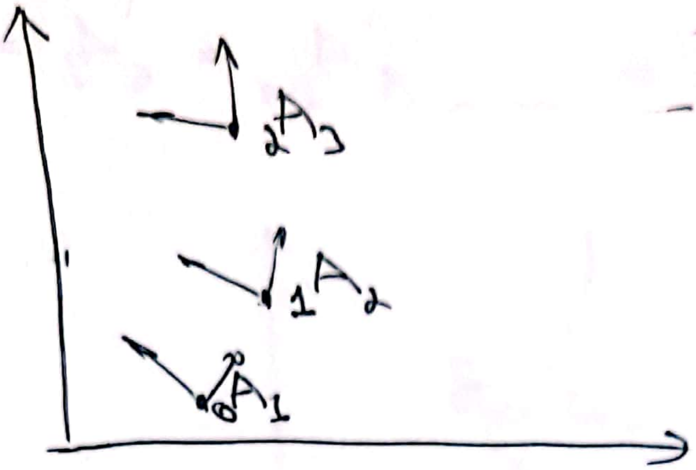
צריך להשתנות יחד עם כיוונית של הרובוט



$$\underline{v} = (v\cos\theta,\ v\sin\theta)$$

$$A = (x_0 + v\cos\theta,\ y_0 + v\sin\theta,\ \underbrace{\theta_0 + \omega t}_{\theta})$$

מאוד נפלל את-
המשגל לחלקים..



"Compose"

$${}_0A_2 = {}_0A_1 \cdot {}_0A_2 = ({}_0x_1, {}_0y_1, {}_0\theta_1)({}_1x_2, {}_1y_2, {}_1\theta_2) =$$

$$= ({}_0x_1 + c_{{}_0\theta_1}\,{}_1x_2 - s_{{}_1\theta_2}\,y_2,\ {}_0y_1 + s_{{}_0\theta_1}\,{}_1y_2 + c_{{}_0\theta_1}\,{}_1y_2,\ {}_0\theta_1 + {}_1\theta_2$$

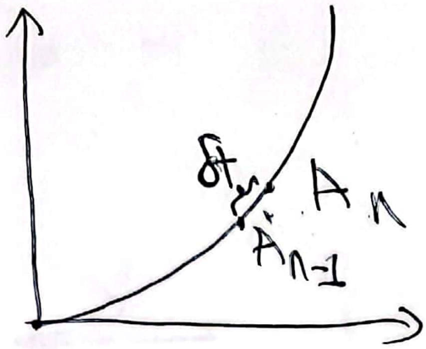
$${}_0A'_1\ {}_1A'_2 = \begin{bmatrix} {}_0R_1 & t_1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} {}_1R_2 & {}_1t_2 \\ 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} {}_0R_1\,{}_1R_2 & {}_0R_1\,{}_1t_2 + {}_0t_1 \\ 0 & 1 \end{bmatrix}$$

$${}_0A'_1 = \begin{bmatrix} {}_0R_1 & t^0_{0\to1} \\ 0 & 1 \end{bmatrix} \qquad {}_0t_1 = \begin{bmatrix} {}_0x_1 \\ {}_0y_1 \end{bmatrix}$$

$${}_0R_1 = \begin{bmatrix} c_{{}_0\theta_1} & s_{{}_0\theta_1} \\ -s_{{}_0\theta_1} & c_{{}_0\theta_1} \end{bmatrix}$$

$${}_0A'_1 = {}_1A'_2 = {}_{n-1}A'_n \ldots = \delta A$$



$${}_0A'_n = {}_0A'_1\,{}_1A_2 \cdots {}_{n-1}A'_n = (\delta A)^n$$

$$\delta A = \begin{bmatrix} \cos(\omega\delta t) & -\sin(\omega\delta t) & V_x\delta \\ \sin(\omega\delta t) & \cos(\omega\delta t) & V_y\delta \\ 0 & 0 & 1 \end{bmatrix} \approx \begin{bmatrix} 1 & -\omega\delta & V_x\delta \\ \omega\delta & 1 & V_y\delta \\ 0 & 0 & 1 \end{bmatrix}$$

$$= I + \delta t \underbrace{\begin{bmatrix} 0 & -\omega & V_x \\ \omega & 0 & V_y \\ 0 & 0 & 1 \end{bmatrix}}_{\xi}$$

$$A'_n = (\delta A)^n \approx (I + \delta t\,\xi)^n$$

$$\delta t = \frac{t}{n}$$

$$A'(t) \approx \left(I + \frac{t}{n}\hat{\xi}\right)^n$$

$$A'(t) = \lim_{n\to\infty}\left(I + \frac{t}{n}\hat{\xi}\right)^n = e^{t\hat{\xi}}$$

$$\xi = \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} \xleftarrow{\vee} \hat{\xi} = \begin{bmatrix} \cdot & -\omega & v_x \\ \omega & \cdot & v_y \\ \cdot & \cdot & 1 \end{bmatrix}$$

$\xrightarrow{\wedge}$

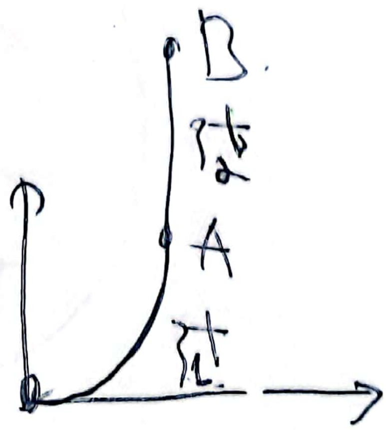
vector

Lie Algebra

log $\uparrow$ $\downarrow$ $\exp(\hat{\xi}t)$

$$A = \begin{bmatrix} c\theta & -s\theta & x \\ s\theta & c\theta & y \\ \cdot & \cdot & 1 \end{bmatrix}$$

Lie Group



$$\tau = \xi t$$

$$\tau = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \underset{\wedge}{\overset{\vee}{\rightleftarrows}} \hat{\tau} = \begin{bmatrix} \cdot & -\theta & x \\ \theta & \cdot & y \\ \cdot & \cdot & 1 \end{bmatrix}$$

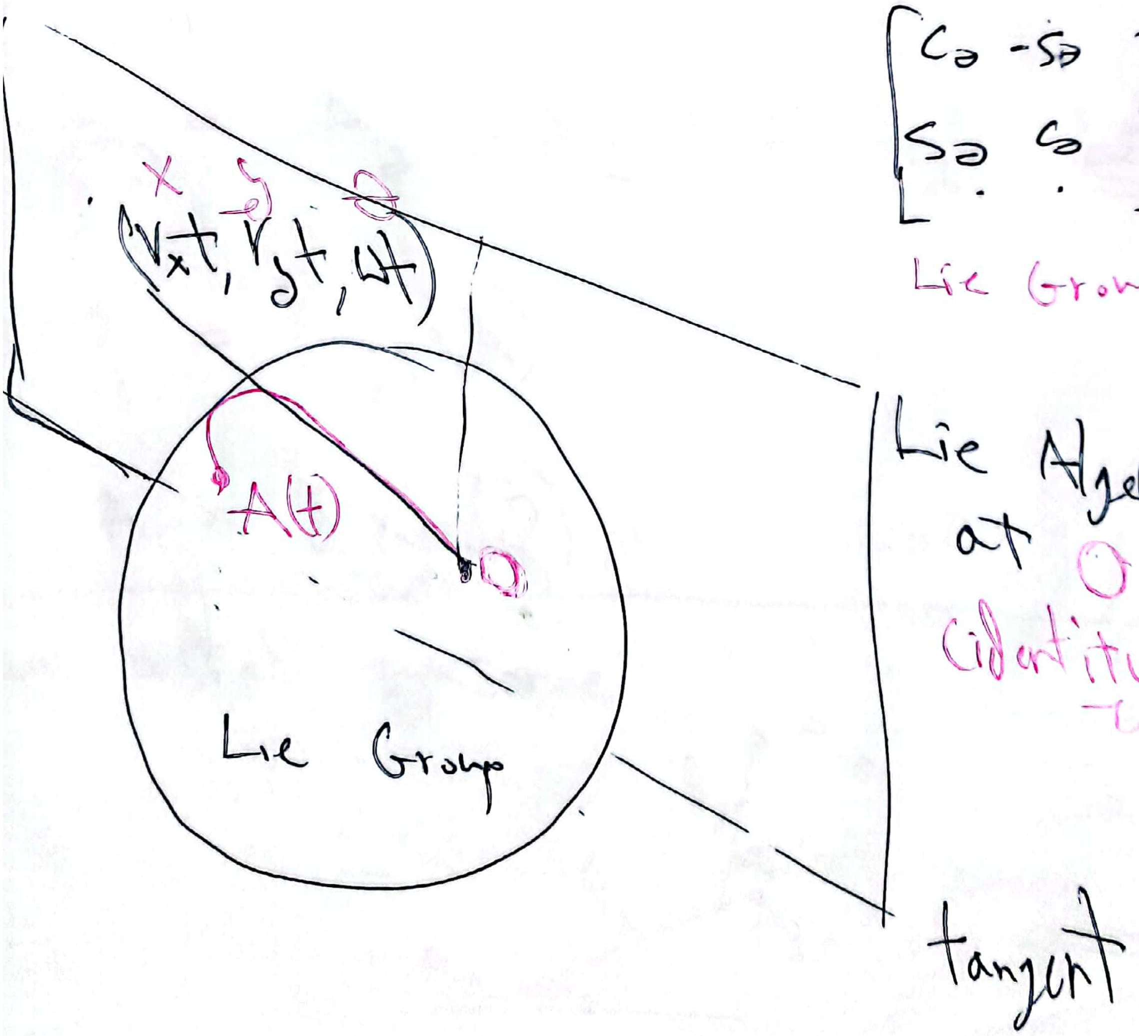
vector

Lie Algebra

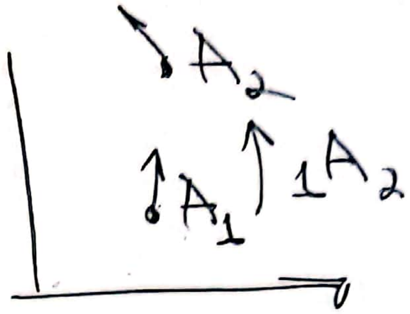
log $\uparrow$ $\downarrow$ exp

$$\begin{bmatrix} c_\theta & -s_\theta & x \\ s_\theta & c_\theta & y \\ \cdot & \cdot & 1 \end{bmatrix}$$

Lie Group



Lie Group: SE2



$$A_2 = A_1 \oplus {}_1A_2 = A_1 * {}_1A_2$$ Compose

$${}_2A_1 = {}_2A_1^{-1}$$ Inverse

$${}_1A_2 = A_2 \ominus A_1 = A_1^{-1} A_2$$ between

$$\tau = \mathrm{Log}(A)$$

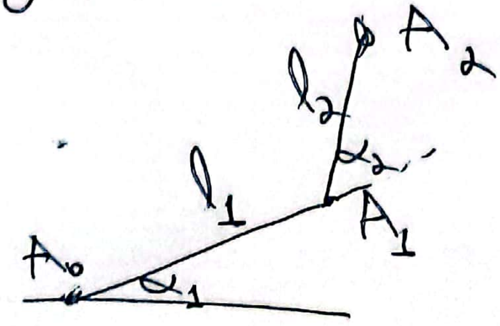
$$A = \mathrm{Exp}(\tau)$$

---

pip install symforce

משימה: פרק 13

לצייר את
הרובוט הבא:



לפתור קינמטיקה ישירה והפוכה