determine configuration ← $\boxed{\text{MULTI-POSE REGISTRATION}}$ → variant of the multi-point registration
of SE(2) and SE(3)
configurations that are mutually aligning each other!

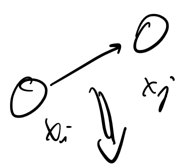
core of ✓
modern-SLAM system
├ problem as a graph
└ node = position ⊕ measurement that the robot has seen at that position

→ being able to compute relative poses of two nodes based on the measurements at two nodes and optimize that would enable graph-based SLAM!!

edge gives graph — constellation of masses (robot positions)
⊕
springs (equilibrium of positions) – 2 connected nodes

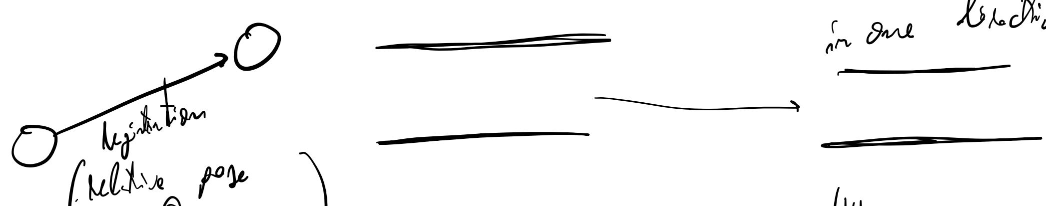
loop-closure constraint → relation between 2 poses
from scan matching e.g.
(in the case of 2D laser)

graph-construction → NOT ADDRESSED IN THIS LECTURE
↓
backend ⇒ focus → how to optimize the graph (minimum energy configuration)
compute the



registration of the environment information of the two poses x_i and x_j
or
odometry (relative motion)

Ω_{ij} encodes certainty of measurement along different directions!
└ e.g., laser scanner on featureless corridor



in one direction, cannot retrieve information
(environment has not enough structure)

[Θ information matrix] → VERY IMPORTANT IN THIS CASE !!!!

Now

z is become $SE(3)$ object as Rotation matrix $\oplus$ translation vector

$\boxminus$ → by taking 2 transformation matrices and multiplying by the prediction

$\equiv$ (prediction expressed in the $z^{[i,j]}$ measurement coordinate frame)

(relative position of the prediction relative to the measurement) $\oplus$

map it back to Euler angles ...

(undoing the effects of boxplus operation)

h depends on $X^{[i]}$, $X^{[j]}$ → influence the structure of the Jacobian

$\Omega^{i,j}$ = information matrix

12 numbers → need to project it back to an error with dimension 6

CHORDAL DISTANCE

+2π is the bad bad part of the derivation until now.....

what does it mean to subtract 2 rotation matrices?

- 2 translations $\equiv$ difference in position
- 2 rotations $\equiv$ well, 1 rotation $\equiv$ 3 orthogonal vectors

difference in chordal distances
(less encountered when talking about direction vectors)

12-D vector → in an euclidean space !

⇓

easier to compute w/ pen and paper

12×6

↑

active variables remain 6 (Euler angle @ 3 position ones)

$$\cancel{\theta}2t(\Delta v_i)^{-1} = v2t(-\Delta v_i)$$

⇒ Conclusions (Further people in the continuation of the set of slides!)

- we managed to build to least-squares formulation
- pose-pose constraints ~ from a position we sense the position of another robot