

NOTES DE COURS SLAM

M2 AURO/IAFA

Patrick Danès
LAAS-CNRS & Université Toulouse III - Paul Sabatier
`patrick.danes@laas.fr`

Version du 24 septembre 2022

I Introduction

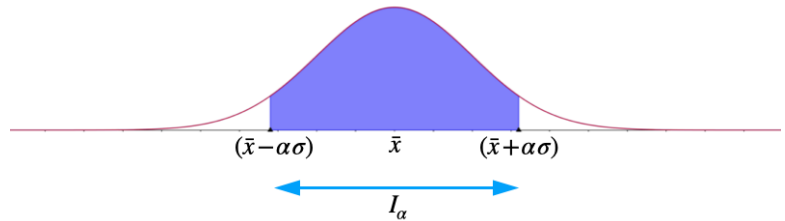
- Simultaneous Localization and Mapping : one of the most emblematic problems of mobile robotics
 - A robot evolving in an unknown environment concurrently builds a map of the environment and localizes itself into this map.
 - Huge literature - Huge number of special issues in Robotics Journal and of dedicated sessions in Robotics Conferences - Multiple industrial applications too!
 - Some references (cf. “so many books, so little time” –FZ)
 - H. DURRANT-WHYTE & T. BAILEY, *Simultaneous Localization and Mapping : Part I & Part II*, IEEE RA Magazine 2006.
 - S. THRUN & J.J. LEONARD, Springer Handbook of Robotics 2008.
 - G. GRISETTI *et al.*, *A Tutorial on Graph-Based SLAM*, IEEE ITS Magazine 2010.
 - C. CADENA *et al.*, *Past, Present and Future of Simultaneous Localization and Mapping : Toward the Robust-Perception Age*, IEEE T-RO 2016.
 - F. DELLAERT & M. KAESS, *Factor Graphs for Robot Perception*, Foundations and Trends in Robotics 2017.
 - S. THRUN, W. BURGARD, D. FOX, Probabilistic Robotics, MIT Press.
 - Tutorials and MATLAB toolbox by J. SOLÁ, www.joansola.eu.
- Some features of the problem
 - The robot has at its disposal : (i) control inputs / odometry / self motion ; (ii) relative observations of the environment from its embedded sensors.
 - Aim : incremental/recursive map building and localization.
 - “Chicken-and-Egg problem”, see below.
- Pose graph
 - Robot trajectory (collection of poses) from relative pose measurements and incorporation of loop closure constraints.
 - Those relative pose measurements come from : self-motion (odometry, IMU), visual feature registration, ICP...
- Mapping
 - Synthesis of an environment model from sensor data given the robot pose, by incorporating (ii) above into a map.
 - Several types of map do exist (cf. M. TAÏX’s course)
 - Dense metric map of geometric cues (2D polygons, 3D polyedra,... → obstacles) : hard to build from sensor data ; eases planning ; scalability \Leftarrow # & complexity.
 - Sparse metric map of landmarks (reference points / descriptors → salient features) : easy to build from sensor data ; eases localization ; scalability \Leftarrow #.
 - Sparse point cloud map (3D points → surfaces) : easy to build ; no planning ; scalability \Leftarrow density, surface.
 - Dense metric occupancy grid (segmented space & occupancy posterior probability) : easy to build, eases planning, scalability \Leftarrow surface—volume, resolution.
 - Dense metric elevation grid (terrain height) : easy to build, eases planning, scalability \Leftarrow surface, resolution.
 - Topological and semantic maps : higher level.

- **Localization**
 - Given a known map, estimation of the robot pose from (i) and (ii) above.
 - Various forms of Bayes inference/filtering.
- **SLAM**
 - Neither the robot pose nor the map is given.
 - Two prominent approaches
 - Estimate the state vector of the problem (robot pose and map) with a Bayes filter : EKF/UKF, particle filter.
 - Optimize the robot trajectory and map.

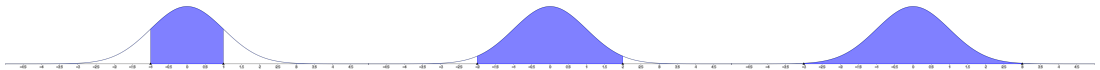
II Handling uncertainty in a probabilistic context : a bird's view

- **Probabilities - Random variables - etc.**
 - Random experiments - Outcome/Events - Probabilities - (Statistical) independence
 - **Random variable/vector** $X \in \mathbb{R}$ or $X \in \mathbb{R}^N$
 - x outcome of X ; $x = X(\omega)$
 - Elementary events $X \leq x$
 - **Cumulative distribution function** $P_X(x) = \mathbb{P}(X \leq x)$
 - **Probability density function** $p_X(x) = \left[\frac{dP_X(\xi)}{d\xi} \right]_{\xi=x} = \lim_{d\xi \rightarrow 0} \frac{\mathbb{P}(x < X \leq x + d\xi)}{d\xi}$
 - $p_X(x) \geq 0$
 - $P_X(x) = \int_{-\infty}^x p_X(\xi) d\xi$; $P_X(+\infty) = 1$; $\mathbb{P}(X \in I) = \int_I p_X(\xi) d\xi$
 - Mind the meaning of $d\xi$ in the multivariate case
 - **Gaussian random variables/vectors**
 - $X \in \mathbb{R}$, $X \sim \mathcal{N}(\bar{x}, \sigma^2)$ iff $p_X(x) = \mathcal{N}(x; \bar{x}, \sigma^2) \triangleq \frac{1}{\sqrt{2\pi}\sigma} \exp -\frac{(x-\bar{x})^2}{2\sigma^2}$
 - $X \in \mathbb{R}^N$, $X \sim \mathcal{N}(\bar{x}, P)$ iff

$$p_X(x) = \mathcal{N}(x; \bar{x}, P) \triangleq \frac{1}{\sqrt{(2\pi)^N \det(P)}} \exp \left(-\frac{1}{2} (x - \bar{x})^T P^{-1} (x - \bar{x}) \right),$$
i.e., iff $p_X(x) \triangleq \frac{1}{\sqrt{\det(2\pi P)}} \exp \left(-\frac{1}{2} (x - \bar{x})^T P^{-1} (x - \bar{x}) \right)$
 - **Confidence sets**
 - For $X \in \mathbb{R}$, $X \sim \mathcal{N}(\bar{x}, \sigma^2)$, I_α of minimum size such that $\mathbb{P}(X \in I_\alpha) = (\text{given}) p_\alpha$ is the confidence set $I_\alpha = [\bar{x} - \alpha\sigma; \bar{x} + \alpha\sigma] = \left\{ \xi : \frac{(\xi - \bar{x})^2}{\sigma^2} \leq \alpha^2 \right\}$



Hence, $Y = \frac{X - \bar{x}}{\sigma} \sim \mathcal{N}(0, 1)$ and the confidence sets I_1, I_2, I_3 are associated to the respective probabilities $p_1 = 0.6827$, $p_2 = 0.9545$, $p_3 = 0.9973$



- For $X \in \mathbb{R}^N$, $X \sim \mathcal{N}(\bar{x}, P)$, \mathcal{E}_α of minimum size such that $\mathbb{P}(X \in \mathcal{E}_\alpha) = (\text{given})p_\alpha$ is the confidence ellipse/ellipsoid/hyperellipsoid $\mathcal{E}_\alpha = \left\{ \xi : (\xi - \bar{x})^T P^{-1} (\xi - \bar{x}) \leq \alpha^2 \right\}$
 - \hookrightarrow Centered on \bar{x}
 - \hookrightarrow Principal axes : (orthogonal) eigenvectors of P
 - \hookrightarrow Extent along eigenvector $v_i : \alpha \sqrt{\lambda_i}$, with λ_i the eigenvalue associated to v_i
 - $\hookrightarrow \mathbb{P}(X \in \mathcal{E}_\alpha)$ depends on α and N
- **The central limit theorem** : roughly speaking, a random vector which is the sum of a large number of independent random excitations is Gaussian
- Expectation operator - Mean/Expectation - Covariance
 - $\mathbb{E}_X[g(X)] = \int g(x) p_X(x) dx$
 - $m_X = \mathbb{E}_X[X] = \int x p_X(x) dx$
 - $\text{Cov}_X = \mathbb{E}_X[(X - m_X)(X - m_X)^T] = \mathbb{E}_X[XX^T] - m_X m_X^T$
 - Note : $\mathbb{E}_X[(X - m_X)^T (X - m_X)] = \text{trace}(\text{Cov}_X)$.
 - Expectation and covariance of $\mathcal{N}(\bar{x}, P)$ are \bar{x} and P !
 - If $X \in \mathbb{R}^N$, $X \sim \mathcal{N}(\bar{x}, P)$, and A, b a matrix, vector pair of suitable dimensions, then $Y = AX + b \sim \mathcal{N}(A\bar{x} + b, APA^T)$
- Joint, marginal, conditional pdfs – Independence, correlation
 - If $X = (X_1, \dots, X_N)^T$, then $p_X(x) = p_{X_1, \dots, X_N}(x_1, \dots, x_N)$ is the joint pdf on X_1, \dots, X_N
 - X_1, \dots, X_N independent $\Leftrightarrow \forall x_1, \dots, x_N, p_{X_1, \dots, X_N}(x_1, \dots, x_N) = p_{X_1}(x_1) \dots p_{X_N}(x_N)$
 - X_1, \dots, X_N independent, identically distributed (i.i.d.) $\Leftrightarrow X_1, \dots, X_N$ independent and $p_{X_1}(x_1) = \dots = p_{X_N}(x_N)$
 - X, Y uncorrelated $\Leftrightarrow \mathbb{E}_{XY}[XY^T] = \mathbb{E}_X[X] \mathbb{E}_Y[Y]^T \Leftrightarrow C_{XY} = 0$
 - X, Y independent $\Rightarrow X, Y$ uncorrelated
 - $[X, Y \text{ independent iff uncorrelated}] \Leftarrow X, Y \text{ jointly Gaussian, i.e., } (X^T, Y^T)^T \text{ Gaussian}$
 - Marginal pdfs

$$p_X(x) = \int p_{XY}(x, y) dy$$

- Conditional pdfs

$$p_{X|Y}(x|y) = \frac{p_{XY}(x, y)}{p_Y(y)} = \frac{p_{XY}(x, y)}{\int p_{XY}(x, y) dx} \text{ if } p_Y(y) \neq 0, \text{ (Bayes pdf)}$$

- If X, Z jointly Gaussian then conditioned on the event $Z = z$, X is Gaussian!

$$\begin{pmatrix} X \\ Z \end{pmatrix} \text{ such that } p_{X,Z}(x, z) = \mathcal{N}\left(\begin{pmatrix} x \\ z \end{pmatrix}; \begin{pmatrix} m_X \\ m_Z \end{pmatrix}, \begin{pmatrix} P_{XX} & P_{XZ} \\ P_{ZX} & P_{ZZ} \end{pmatrix}\right)$$

\Downarrow

$$p_{X|Z}(x|z) = \mathcal{N}(x; m_{X|Z}, P_{X|Z})$$

and we know $m_{X|Z}, P_{X|Z}$ analytically!

- Bayes estimation : general framework to stochastic estimation with prior knowledge
 - Static case : estimation of an hidden vector x from a measurement vector z
 - Statement
 - ω = outcome of ongoing experience; $x = X(\omega)$ hidden parameter vector; $z = Z(\omega)$ measured observation vector
 - $X \sim p_X(\xi)$, $p_X(\xi)$ (given) prior pdf
 - (Given) observation model $p_{Z|X}(\zeta|\xi)$ expresses statistical relationship between any hidden ξ and possible consequent measurement ζ
 - Goal : synthesize an “estimator function” $g(\cdot)$ such that $\hat{x} = g(z)$ is a (good) estimate of hidden x .
 - What is a “good” estimate?
 - z is random $\Rightarrow \hat{x}$ is random \Rightarrow with ω the outcome of ongoing experience, $z = Z(\omega)$ and $\hat{x} = \hat{X}(\omega)$, with \hat{X} the estimator (random vector)
 - $\hookrightarrow X - \hat{X}$ is the estimation error (random vector), $\mathbb{E}[X - \hat{X}]$ is the bias of \hat{X} , $\mathbb{E}[\|X - \hat{X}\|^2]$ is the mean square error of \hat{X} , $\mathbb{E}[(X - \hat{X})(X - \hat{X})^T]$ is the estimation error covariance matrix, etc.
 - “Holy Grail”
 - All the information on hidden X conveyed by measurement z (and prior knowledge) is captured by the posterior pdf $p_{X|Z}(x|z)$
 - \hookrightarrow Expectation \hat{x}_{MMSE} of $p_{X|Z}(x|z)$ is called the Minimum Mean Square Error (MMSE) Estimate as it minimises the expectation of $\mathbb{E}[\|X - \hat{X}\|^2]$, over any estimator $\hat{X} = \hat{g}(Z)$
 - \hookrightarrow Peak of \hat{x}_{MMSE} of $p_{X|Z}(x|z)$, $\hat{x}_{\text{MMSE}} = \arg \max_x p_{X|Z}(x|z) = \arg \min_x (-\ln p_{X|Z}(x|z))$, is called the Maximum a Posteriori (MAP) Estimate
 - Case of no prior knowledge
 - When $p_X(x)$ is a flat prior (\approx no prior knowledge), the MAP estimate boils down to the Maximum Likelihood Estimate (MLE) $\hat{x}_{\text{MLE}} = \arg \max_x p_{Z|X}(z|x) = \arg \min_x (-\ln p_{Z|X}(z|x))$
 - $\hookrightarrow \mathcal{L}(x; z) = p_{Z|X}(z|x)$ (function of x parameterized by z) is called the likelihood of x w.r.t. z ; mind here : this is not a pdf...
 - “Linear Gaussian case”
 - If $Z = HX + V$, with $Z \in \mathbb{R}^N$, $X \in \mathbb{R}^M$, $V \in \mathbb{R}^N$, $\begin{pmatrix} X \\ V \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \hat{x}^- \\ 0 \end{pmatrix}, \begin{pmatrix} P^- & 0 \\ 0 & R \end{pmatrix}\right)$ and $R > 0$, then $p_{X|Z}(x|z) = \mathcal{N}(x; \hat{x}^+, P^+)$, where $\hat{x}^+ = \mathbb{E}_{X|Z}[X|z]$ and $P^+ = \mathbb{E}_{X|Z}[(X - \hat{x}^+)(X - \hat{x}^+)^T|z]$ write as

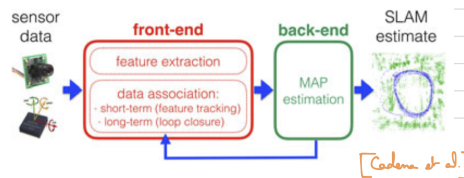
$$\begin{cases} \hat{x}^+ = \hat{x}^- + K(z - H\hat{x}^-) \\ P^+ = P^- - KHP^- \\ K = P^- H^T (R + HP^- H^T)^{-1}. \end{cases}$$

- \hookrightarrow MMSE and MAP estimates are identical in this case.
- \hookrightarrow MLE estimate is $\hat{x}_{\text{MLE}} = (H^T R^{-1} H)^{-1} H^T R^{-1} z$. The covariance of the associated estimator \hat{X}_{MLE} is $\text{Cov}_{\hat{X}_{\text{MLE}}} = (H^T R^{-1} H)^{-1}$.

- **Dynamic case** : estimation of an hidden sequence $x_{0:k}$ from a measurement vector $z_{0:k}$
 - Statement
 - ω = outcome of ongoing experience ; $x_{0:k} = X_{0:k}(\omega)$ hidden state vector ; $z_{0:k} = Z_{0:k}(\omega)$ measured observation vector
 - $X_0 \sim p_{X_0}(\xi_0)$, $p_{X_0}(\xi_0)$ (given) **initial prior pdf**
 - $\{p_{X_k|X_{k-1}}(\xi_k|\xi_{k-1})\}$ (given) **prior dynamics pdf** between times $k-1$ and k (Note : Markov state vector)
 - (Given) **observation model** $\{p_{Z_k|X_k}(\zeta_k|\xi_k)\}$ expresses statistical relationship between any hidden ξ_k and possible consequent measurement ζ_k at time k
 - “Holy Grail”
 - All the information on hidden $X_{0:k}$ conveyed by measurement $z_{0:k}$ (and prior knowledge) is captured by the joint posterior pdf $p_{X_{0:k}|Z_{0:k}}(x_{0:k}|z_{0:k})$; Marginal posterior (filtering) pdf $p_{X_k|Z_{0:k}}(x_k|z_{0:k})$ if often preferred; another marginal posterior pdf $p_{X_k|Z_{0:k-1}}(x_k|z_{0:k-1})$ has to do with prediction (and is named prediction pdf)
- ↪ MMSE or MAP estimates can be sought for
 - “Linear Gaussian case” : closed-form solution is given by **Kalman filtering**

III Variables, models and workflow of landmark-based SLAM

- Variables (static environment)
 - State vector x_k at time k , with $x_k = (r_k^T, m^T)^T$, r_k = robot absolute pose and $m = (m_1^T, \dots, m_M^T)^T$ absolute position of landmarks
 - Control inputs $u_{0:k}$, given (by odometry, etc.)
 - Observations $z_{0:k}$, with z_k measurement at time k = stacking of measurements $\{z_{k,j}\}_j$ from r_k of the set $\{m_j\}_j$ of visible landmarks
- Models (static environment)
 - $r_{k+1} = f(r_k, u_k) + w_k$, $w_k \sim \mathcal{N}(0, Q_k)$ white Gaussian dynamic noise
 - ↪ $p(r_{k+1}|r_k) = p(r_{k+1}|r_k; u_k) = \mathcal{N}(r_{k+1}; f(r_k, u_k), Q_k)$.
 - (m constant noise-free)
 - $z_k = h(r_k, m) + v_k$, $v_k \sim \mathcal{N}(0, R_k)$ white Gaussian measurement noise (noises on individual measurements are assumed mutually independent, R_k diagonal)
 - ↪ $p(z_k|r_k, m) = \mathcal{N}(z_k; h(r_k, m), R_k)$
- Implementation workflow
 - Sensor data
 - **Front-end**
 - **Feature extraction**
 - **Data association**
 - ↪ short-term (feature tracking)
 - ↪ long-term (loop closure)
 - **Back-end** : estimation algorithm



IV Probabilistic SLAM

- Full SLAM : estimate the posterior joint pdf of the robot trajectory together with the map
 $p(r_{0:k}, m|z_{0:k}) = p(r_{0:k}, m|z_{0:k}; u_{0:k-1})$
- Online SLAM : estimate the posterior marginal pdf of the robot pose together with the map
 $p(r_k, m|z_{0:k}) = p(r_k, m|z_{0:k}; u_{0:k-1})$

IV.1 EKF solution

- First solution to SLAM : $p(r_k, m|z_{0:k})$ approximated by a (huge-dimension) Gaussian pdf thanks to EKF approximate computations of posterior moments for nonlinear dynamics/observation model
 - Use of Taylor expansions for covariance and gain equations (\leadsto Jacobian matrices of transition/measurement functions around last state estimates/predictions)
 - \hookrightarrow considering standard assumptions and notations of linear KALMAN filter except the following nonlinear state and measurement stochastic equations

$$x_{k+1} = f(x_k) + w_k, \quad w_k \sim \mathcal{N}(0, Q_k), \quad \text{and} \quad z_k = h(x_k) + v_k, \quad v_k \sim \mathcal{N}(0, R_k),$$

EKF equations write as

$$\begin{aligned} \hat{x}_{0|0} &= m_{X_0} & P_{0|0} &= P_0 \\ \hat{x}_{k+1|k} &= f(\hat{x}_{k|k}) & P_{k+1|k} &= F_k P_{k|k} F_k^T + Q_k \\ \hat{z}_{k+1|k} &= h(\hat{x}_{k+1|k}) & S_{k+1|k} &= R_{k+1} + H_{k+1} P_{k+1|k} H_{k+1}^T \\ K_{k+1} &= P_{k+1|k} H_{k+1}^T S_{k+1|k}^{-1} \\ \hat{x}_{k+1|k+1} &= \hat{x}_{k+1|k} + K_{k+1}(z_{k+1} - \hat{z}_{k+1|k}) & P_{k+1|k+1} &= P_{k+1|k} - K_{k+1} H_{k+1} P_{k+1|k} \\ \text{with } F_k &= \left[\frac{\partial f(x)}{\partial x^T} \right]_{x=\hat{x}_{k|k}} & \text{and } H_{k+1} &= \left[\frac{\partial h(x)}{\partial x^T} \right]_{x=\hat{x}_{k+1|k}} \end{aligned}$$

- Robot Motion \leftrightarrow EKF time update
- Features observation \leftrightarrow
 - State augmentation if new landmark is detected and needs to be initialized in the map
 - EKF measurement update if known landmark is observed and needs to be corrected in the map
 - State reduction if a landmark must be detected in view of map corruption
- Some important facts
 - The robot moves \Rightarrow pose uncertainty increases \Rightarrow (as measurement uncertainty is fixed) growing uncertainty on nearby landmarks
 - As time increases, the covariance matrix of the filtering pdf gets “denser” as the cross-correlations between estimation errors on landmark locations increases
 - \hookrightarrow The relative location between any two m_i, m_j may be accurately known even when their absolute locations are uncertain
 - So, the observation of a landmark already observed in the very beginning of the mapping enables the decrease of the uncertainties in the robot pose and in the location of previously observed landmarks! This is the loop closure phenomenon

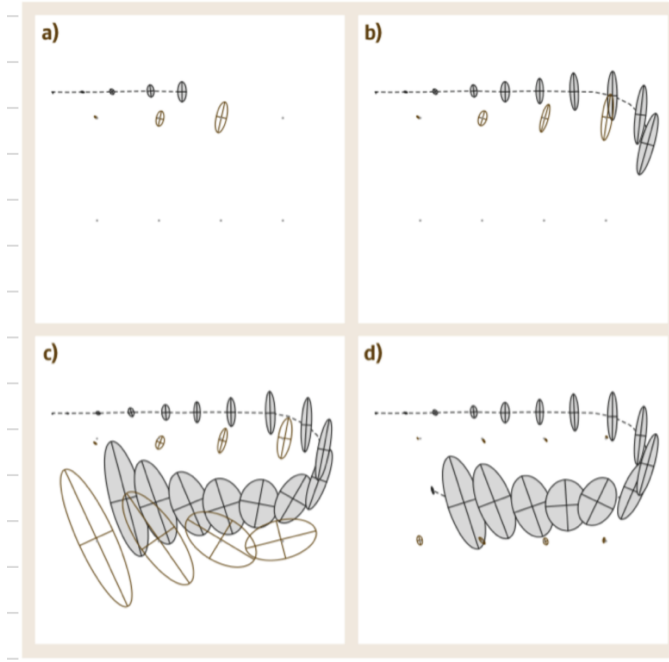


Fig. 37.2a–d EKF applied to the on-line SLAM problem. The robot's path is a dotted line, and its estimates of its own position are shaded ellipses. Eight distinguishable landmarks of unknown location are shown as small dots, and their location estimates are shown as white ellipses. In (a–c) the robot's positional uncertainty is increasing, as is its uncertainty about the landmarks it encounters. In (d) the robot senses the first landmark again, and the uncertainty of all landmarks decreases, as does the uncertainty of its current pose. (Image courtesy of Michael Montemerlo, Stanford University)

[Thrun, Leonard]

- **Data association** may have to be handled, so as to disambiguate which landmark have given rise to observations
- Mind the quadratic complexity of covariance computations. Exploit sparseness.
- Though interesting for small-scale problems and (mostly) for historical reasons, EKF-SLAM is fundamentally limited to scarce maps, involving appropriately selected (perceptually distinct) landmarks.
- Illustration on a (not-so-silly) toy-example : **2D range and/or bearing online EKF-SLAM**
 - State, control input and measurement output variables
 - Planar robot \mathcal{R} with associated frame $\mathcal{F}_k = (O_k, \vec{x}_k, \vec{y}_k)$ at time k . M pointwise landmarks $\mathcal{L}_1, \dots, \mathcal{L}_M$ at (unknown) loci L_1, \dots, L_M , to be referenced in a map \mathcal{M} . World/Map frame is \mathcal{F}_0 .
 - Hidden state vector $x_k = (r_k^T, m_1^T, \dots, m_M^T)^T$: absolute robot pose vector $r_k = (r_{x_k}, r_{y_k}, \theta_k)^T$ with $(r_{x_k}, r_{y_k})^T = \overrightarrow{O_0 O_k(\mathcal{F}_0)}$ and $\theta_k = (\vec{x}_0, \vec{x}_k)$; absolute landmark positions $m_m = (m_{x_m}, m_{y_m})^T = \overrightarrow{O_0 L_m(\mathcal{F}_0)}$, gathered into vector $m = (m_1^T, \dots, m_M^T)^T$.
 - Control input vector $u_k = (\tau_k^T, \rho_k)^T$: robot translation $\tau_k = (\tau_{x_k}, \tau_{y_k})^T = \overrightarrow{O_k O_{k+1}(\mathcal{F}_k)}$ and rotation $\rho_k = (\vec{x}_k, \vec{x}_{k+1})$ from \mathcal{F}_k to \mathcal{F}_{k+1} expressed in \mathcal{F}_k .
 - Observation vector z_k : stacking of $\{z_{k,j}\}_{\{\text{visible } \mathcal{L}_j\}}$, where $z_{k,j}$ terms the range $\|O_k L_j\|$, bearing $(\vec{x}_k, \overrightarrow{O_k L_j}) = \left(\text{atan2}(\overrightarrow{O_k L_j} \cdot \vec{y}_k, \overrightarrow{O_k L_j} \cdot \vec{x}_k) = \text{atan2}(\overrightarrow{O_k L_j} \cdot \vec{y}_0, \overrightarrow{O_k L_j} \cdot \vec{x}_0) - \theta_k \right)$ or their stacking, at time k , of L_j relative to \mathcal{F}_k .
 - Note however—including in the lines below—that subvector m is built incrementally, *i.e.*, M is not known in advance. . .
 - Prior dynamics
 - Robot dynamics $r_{k+1} = f_{\mathcal{R}}(r_k, u_k) + w_k^{\mathcal{R}}$, $w_k^{\mathcal{R}} \sim \mathcal{N}(0, Q_k^{\mathcal{R}})$, $w_{0:k}^{\mathcal{R}}$ white, etc. where

$$f_{\mathcal{R}}(r_k, u_k) = r_k + g_{\mathcal{R}}(r_k, u_k) \text{ with } g_{\mathcal{R}}(r_k, u_k) = g_{\mathcal{R}}(\cancel{r_{x_k}}, \cancel{r_{y_k}}, \theta_k, u_k) = \begin{pmatrix} \tau_{x_k} \cos \theta_k - \tau_{y_k} \sin \theta_k \\ \tau_{x_k} \sin \theta_k + \tau_{y_k} \cos \theta_k \\ \rho_k \end{pmatrix}.$$

↪ Proof : show that the average noise-free model accounts for the rigid-body motion of the robot between times k and $k + 1$, described by

$$T_{0,k+1} = T_{0,k}T_{k,k+1}, \text{ with } T_{0,k} = \begin{pmatrix} \cos \theta_k & -\sin \theta_k & r_{x_k} \\ \sin \theta_k & \cos \theta_k & r_{y_k} \\ 0 & 0 & 1 \end{pmatrix} \text{ and } T_{k,k+1} = \begin{pmatrix} \cos \rho_k & -\sin \rho_k & \tau_{x_k} \\ \sin \rho_k & \cos \rho_k & \tau_{y_k} \\ 0 & 0 & 1 \end{pmatrix}.$$

- Incorporating $m_{k+1} = m_k$ leads to $x_{k+1} = f(x_k, u_k) + w_k$, with

$$f(x_k, u_k) = x_k + E_x^T g_{\mathcal{R}}(r_k, u_k) + w_k, \quad w_k \sim \mathcal{N}(0, Q_k),$$

$$Q_k = \text{blkdiag}(Q_k^{\mathcal{R}}, \underbrace{\mathbb{O}_{2 \times 2}, \dots, \mathbb{O}_{2 \times 2}}_{M \text{ times}}), \quad E_x = \begin{pmatrix} \mathbb{I}_{3 \times 3} & \mathbb{O}_{3 \times 2M} \end{pmatrix}.$$

◦ Observation model

- $z_k = h_k(x_k) + v_k$, $v_k \sim \mathcal{N}(0, R_k)$, composed of measurements $z_{k,j} = h_{k,j}(r_k, m_j) + v_{k,j}$ of \mathcal{L}_j w.r.t. \mathcal{R} , $v_{k,j} \sim \mathcal{N}(0, R_{k,j})$ mutually independent over $\{\mathcal{L}_j\}$ so that $R_k = \text{blkdiag}(\{R_{k,j}\})$, $v_{1:k}$ white, etc. with one of the following options :

$$h_{k,j}(r_k, m_j) = \sqrt{(m_{x_j} - r_{x_k})^2 + (m_{y_j} - r_{y_k})^2} \text{ or } \text{atan2}((m_{y_j} - r_{y_k}), (m_{x_j} - r_{x_k})) - \theta_k,$$

$$\text{or } h_{k,j}(r_k, m_j) = \begin{pmatrix} \sqrt{(m_{x_j} - r_{x_k})^2 + (m_{y_j} - r_{y_k})^2} \\ \text{atan2}((m_{y_j} - r_{y_k}), (m_{x_j} - r_{x_k})) - \theta_k \end{pmatrix}.$$

↪ Proof : rewrite $\begin{pmatrix} \overrightarrow{O_k L_j(\mathcal{F}_k)} \\ 1 \end{pmatrix} = T_{0,k}^{-1} \begin{pmatrix} \overrightarrow{O_0 L_j(\mathcal{F}_0)} \\ 1 \end{pmatrix} \Leftrightarrow \overrightarrow{O_k L_j(\mathcal{F}_k)} = R_{0,k}^T \overrightarrow{O_0 L_j(\mathcal{F}_0)} - R_{0,k}^T P_{0,k}$, with $R_{0,k} = \begin{pmatrix} \cos \theta_k & -\sin \theta_k \\ \sin \theta_k & \cos \theta_k \end{pmatrix}$, $P_{0,k} = \begin{pmatrix} r_{x_k} \\ r_{y_k} \end{pmatrix}$, $\overrightarrow{O_0 L_j(\mathcal{F}_0)} = \begin{pmatrix} m_{x_j} \\ m_{y_j} \end{pmatrix}$, as $\overrightarrow{O_k L_j(\mathcal{F}_k)} = \begin{pmatrix} \cos \theta_k (m_{x_j} - r_{x_k}) + \sin \theta_k (m_{y_j} - r_{y_k}) \\ -\sin \theta_k (m_{x_j} - r_{x_k}) + \cos \theta_k (m_{y_j} - r_{y_k}) \end{pmatrix}$; then, show that the first line of $h_{k,j}(r_k, m_j)$ is the norm of $\overrightarrow{O_k L_j(\mathcal{F}_k)}$ and that the second line of $h_{k,j}(r_k, m_j)$ is its angle, e.g., resorting to the expansion of $\tan(\text{atan2}(\dots) - \theta_k)$.

◦ SLAM Initialization

- The initial pose of the robot is the origin of the map, which contains no landmarks, hence

$$\hat{x}_{0|0} = \hat{r}_{0|0} = \mathbb{O}_{3 \times 1}, \quad P_{0|0} = P_{\mathcal{R} \mathcal{R}_{0|0}} = \mathbb{O}_{3 \times 3}.$$

◦ SLAM Time update between times k and $k + 1$

- Standard EKF equations apply, leading to sparse $F_k = \left[\frac{\partial f(x, u_k)}{\partial x^T} \right]_{x=\hat{x}_{k|k}}$ so that predicted covariance $2M \times 2M$ submatrix $P_{\mathcal{M} \mathcal{M}_{k+1|k}} = P_{\mathcal{M} \mathcal{M}_{k|k}}$ is unchanged.

◦ SLAM Measurement update at time $k + 1$ for already seen landmarks

- Standard EKF equations, with $H_{k+1} = \left[\frac{\partial h_{k+1}(x)}{\partial x^T} \right]_{x=\hat{x}_{k+1|k}}$, and each individual Jacobian matrix $H_{k+1,j} = \left[\frac{\partial h_{k+1,j}(x)}{\partial x^T} \right]_{x=\hat{x}_{k+1|k}}$ also features a sparse structure.
- Interesting sparsity properties also hold in the computation of the residual $z_{k+1,j} - \hat{z}_{k+1|k,j}$ and its covariance matrix.

◦ For unseen landmarks, additional stage to be inserted before SLAM Measurement update

- If \mathcal{R} discovers a landmark, say \mathcal{L}_m , which has not yet been mapped, then the current state vector must be augmented with m_m . Even if, for given r_{k+1} , the output function $h_{k+1,m}(r_{k+1}, m_m)$ is bijective w.r.t. m_m , it is necessary to define a prediction $\hat{m}_{m_{k+1|k}}$ of m_m in order to compute a meaningful individual Jacobian $H_{k+1,m}$. Otherwise this Jacobian may not be a valid approximation of the derivative of $h_{k+1,m}(\cdot, \cdot)$ over the regions where the state vector prediction pdf is high, and EKF may fail!

- Data association handling

- The easiest way to decide whether an already seen landmark in $\{\mathcal{L}_1, \dots, \mathcal{L}_{m-1}\}$ or a new \mathcal{L}_m gives rise to the observation z_{k+1} is “gating”.

Compute $\delta_{k+1,j} \triangleq (z_{k+1} - \hat{z}_{k+1|k,j})^T S_{k+1|k,j}^{-1} (z_{k+1} - \hat{z}_{k+1|k,j})$, $j \in \{1, \dots, m-1\}$;

Detect most likely \mathcal{L}_{j^*} by $j^* = \arg \min_{j \in \{1, \dots, m\}} \delta_{k+1,j}$.

IV.2 Particle Filter solution

- An interesting (and subtle) independence property in the Bayes network of the SLAM problem enables an efficient **Rao-Blackwellized Particle Filter** : **FastSLAM** algorithm.

- The joint posterior pdf is approximated by

$$p(r_{0:k}, m_{0:k} | z_{0:k}) \approx \sum_{i=1}^N \bar{w}_k^{(i)} \left[\prod_{j=1}^M \mathcal{N}(m_j | \hat{m}_{j|k}^{(i)}, P_{j|k}^{(i)}) \right] \delta(r_{0:k} - r_{0:k}^{(i)}).$$

- Some important facts

- The SLAM posterior pdf is factored into low-dimensional estimation problems : a weighted particle cloud (with stochastic support) for the robot pose/trajectory ; for each particle, an associated conditional map posterior pdf through M low-dimensional independent EKFs (one per landmark). Two versions (increasing efficiency) : FastSLAM 1.0, FastSLAM 2.0.
- Inner computations complexities : $\mathcal{O}(N)$ and $\mathcal{O}(N \log M)$ \rightarrow FastSLAM scales up to problems with $M = 10^5 - 10^6$ landmarks.
- Efficient data association mechanism.
- EKF landmark map can be replaced with occupancy grid map.

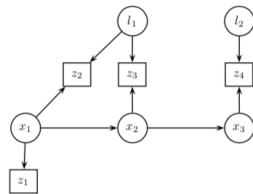
V Optimization SLAM

- SLAM can be viewed as a **sparse graph of constraints** \rightarrow optimization of the robot trajectory and the map

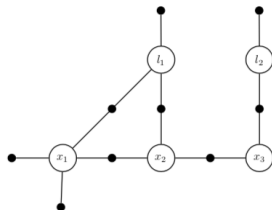
\hookrightarrow this has become the **modern paradigm** to solve full SLAM!

- **Factor graph approach** : **Graph SLAM**

- Factor graph = bipartite graph : variable/states and factors/constraints
- Rewriting of a conditional distribution as a product of factors



$$\begin{aligned} p(X, Z) &= p(x_1)p(x_2|x_1)p(x_3|x_2) \\ &\times p(l_1)p(l_2) \\ &\times p(z_1|x_1) \\ &\times p(z_2|x_1, l_1)p(z_3|x_2, l_1)p(z_4|x_3, l_2). \end{aligned}$$



$$\begin{aligned} p(X|Z) &\propto p(x_1)p(x_2|x_1)p(x_3|x_2) \\ &\times p(l_1)p(l_2) \\ &\times l(x_1; z_1) \\ &\times l(x_1, l_1; z_2)l(x_2, l_1; z_3)l(x_3, l_2; z_4). \end{aligned}$$

- Application to SLAM
 - Nodes = robot poses & landmark locations
 - Arcs = “soft constraints”
 - information conveyed by odometry (between consecutive robot poses)
 - information from observations (between robot poses and landmark locations)
 - loop closure constraints
 - priors
 - The factor graph is built online – Very sparse as constraints number sublinear w.r.t. time and nodes number
- Statement
 - $\hat{r}_{0:k}, \hat{m} = \arg \max_{r_{0:k}, m} \ln p(r_{0:k}, m | z_{0:k}) = \arg \max_{r_{0:k}, m} \ln p(r_{0:k}, m | z_{0:k}; u_{0:k-1})$
 $= \arg \max_{r_{0:k}, m} \ln p(r_0) + \sum_{t=1}^k \ln p(r_t | r_{t-1}; u_{t-1}) + \sum_{t=0}^k \ln p(z_t | r_t, m)$
 - $\Leftrightarrow \hat{r}_{1:k}, \hat{m} = \arg \min_{r_{1:k}, m} \frac{1}{2} \sum_{i=1}^k (r_i - f(r_{i-1}, u_{i-1}))^T Q_i^{-1} (r_i - f(r_{i-1}, u_{i-1}))$
 $+ \frac{1}{2} \sum_{j=0}^k (z_j - h(r_j, m))^T R_j^{-1} (z_j - h(r_j, m))$
 $= \arg \min_{r_{1:k}, m} \frac{1}{2} \sum_p (r_p - f(r_{p-1}, u_{p-1}))^T \Omega_p (r_p - f(r_{p-1}, u_{p-1}))$
 $+ \frac{1}{2} \sum_q (z_q - h(r_q, m))^T \Omega_q (z_q - h(r_q, m))$
 with Ω_p and Ω_q the dynamic and measurement noise information matrices
 - $\Leftrightarrow \hat{x} = (\hat{r}_{1:k}^T, \hat{m}^T)^T = \arg \min_{x=(r_{1:k}^T, m^T)^T} \frac{1}{2} \sum_{l=1}^L e_l^T(x_i, x_j) \Omega_l e_l^T(x_i, x_j)$
 with L the total number of factors, *i.e.*, the number of soft constraints implied by the odometry/landmarks measurements, each of them uniting a pair of unknowns
 - $\Leftrightarrow \hat{x} = (\hat{r}_{1:k}^T, \hat{m}^T)^T = \arg \min_x \frac{1}{2} e^T(x) \Omega e(x)$
 - Sparse factor graph \rightarrow Sparse criterion \rightarrow Solution by sparse optimization
 - Reminder : Newton algorithm to compute the minimum $\hat{x} = \arg \min_x f(x)$ of the real scalar-valued multivariate function $f(\cdot)$
 - Select an initial guess \hat{x}
 - Write the Taylor expansion $f(\hat{x} + \Delta x) \approx \hat{f}(\hat{x} + \Delta x) \triangleq f(\hat{x}) + g_f^T(\hat{x}) \Delta x + \frac{1}{2} \Delta x^T H_f(\hat{x}) \Delta x$,
 with $g_f(\hat{x}) = \left[\frac{\partial f(x)}{\partial x} \right]_{\hat{x}}$ the gradient (column) vector and $H_f(\hat{x}) = \left[\frac{\partial^2 f(x)}{\partial x \partial x^T} \right]_{\hat{x}}$ the Hessian
 (square symmetric) matrix of $f(\cdot)$ at \hat{x}
 - Compute $\Delta x^* = \arg \min_{\Delta x} \hat{f}(\hat{x} + \Delta x) = -H_f^{-1}(\hat{x}) g_f(\hat{x})$
 - Set $\hat{x} \leftarrow \hat{x} + \Delta x^*$ and iterate until convergence
 - Gauss-Newton minimization of $f(x) = e^T(x) \Omega e(x)$ with $e(\cdot)$ a real vector-valued multivariate function
 - Set $J_e(\hat{x}) = \left[\frac{\partial e(x)}{\partial x^T} \right]_{\hat{x}}$ the Jacobian of $e(\cdot)$ at \hat{x} , and deduce $g_f(\hat{x}) = J_e^T(\hat{x}) \Omega e(\hat{x})$
 - Neglect (hence “Gauss-Newton” instead of “Newton” minimization) the second derivatives of $e(\cdot)$ in the second-order Taylor expansion of $f(\cdot)$, so that $H_f(\hat{x}) \approx J_e^T(\hat{x}) \Omega J_e(\hat{x})$
 - Deduce $\Delta x^* = -\left(J_e^T(\hat{x}) \Omega J_e(\hat{x}) \right)^{-1} J_e^T(\hat{x}) \Omega e(\hat{x})$, and iterate
 - $J_e(\hat{x})$ is sparse! Only two nonzero blocs per group of lines corresponding to $e_k(\cdot)$
 - \hookrightarrow The problem can be solved with very high computational performance by matrix QR or Cholesky factorization, *provided good initial guesses can be defined* \rightarrow see open-source solvers **g2o**, **ceres**, **gtsam**
 - Around the MAP x^* , the joint posterior pdf on x is locally approximated by $\mathcal{N}(x; x^*, J_e^T(x^*) \Omega J_e(x^*))$