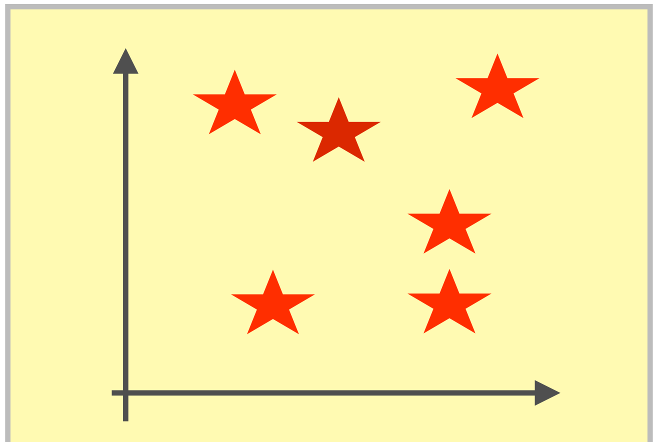
16-311-Q Introduction to Robotics Fall'17

Lecture 22:
Map Building (unknown # of landmarks)
EKF SLAM

Instructor:
Gianni A. Di Caro

PROBLEM: WHAT IF THE NUMBER OF LANDMARKS IS *UKNOWN*?



Where(?) the M(?) landmarks?

State vector $\boldsymbol{\xi}$ *of the EKF:*

(Coordinates of the M landmarks)

Usually, both the locations and the number M of the landmarks existing in the environment are not known a priori

The state vector $\boldsymbol{\xi}_k$ must be **incrementally expanded** by two new coordinate components $[\lambda^i_x \ \lambda^i_y]$ each time a *new* landmark λ^i is observed

Formally, the expansion of the state vector $\boldsymbol{\xi}_k$ amounts to the following *process dynamics* (before process dynamics was stationary), implemented through the use of the *auxiliary* (non linear!) function ***q()***:

$$\boldsymbol{\xi}^*_{k+1} = \boldsymbol{q}(\boldsymbol{\xi}_k, \boldsymbol{z}_{k+1}; x_k, y_k, \theta_k) = \begin{bmatrix} \boldsymbol{\xi}_k \\ \boxed{\boldsymbol{g}(x_k, y_k, \theta_k, \boldsymbol{z}_{k+1})} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\xi}_k \\ x_k + \rho_k \cos(\theta_k + \beta_k) \\ y_k + \rho_k \sin(\theta_k + \beta_k) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\xi}_k \\ \lambda^i_{kx} \\ \lambda^i_{ky} \end{bmatrix}$$

If $\boldsymbol{\xi}_k$ is $2n \times 1, \ n < M \Rightarrow \boldsymbol{\xi}^*_{k+1}$ is $2(n+1) \times 1$.

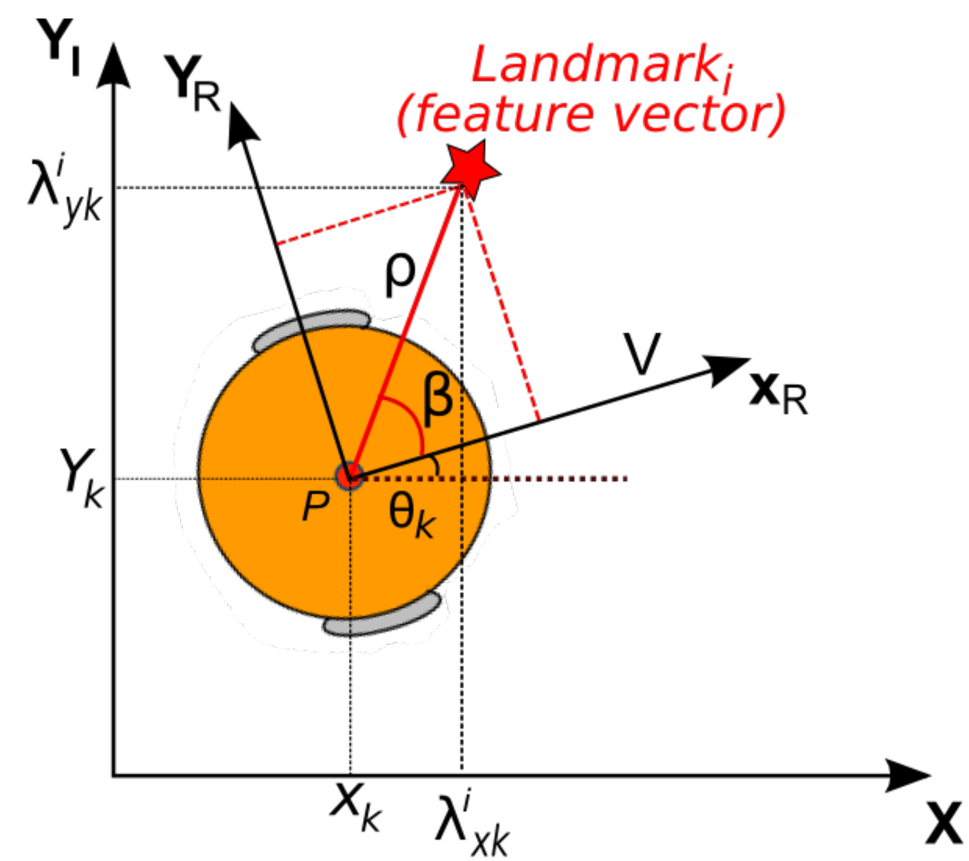
The order of the landmarks in the state vector depends on the order of observation

THE STATE EXPANSION FUNCTION

$$\boldsymbol{\xi}^*_{k+1} = \boldsymbol{q}(\boldsymbol{\xi}_k, \boldsymbol{z}_{k+1}; x_k, y_k, \theta_k) = \begin{bmatrix} \boldsymbol{\xi}_k \\ \boldsymbol{g}(x_k, y_k, \theta_k, \boldsymbol{z}_{k+1}) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\xi}_k \\ x_k + \rho_k \cos(\theta_k + \beta_k) \\ y_k + \rho_k \sin(\theta_k + \beta_k) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\xi}_k \\ \lambda^i_{kx} \\ \lambda^i_{ky} \end{bmatrix}$$

- The vector function **g**(x, y, θ, **z**), which is the <u>inverse of the observation function</u> $\ell()$, gives the measured coordinates (in the world frame) of the observed landmark.
- Inputs: known vehicle pose $[x\ y\ \theta]^T$, sensor observation $\mathbf{z} = [\rho\ \beta]^T$ at time step k.
- **g() gets the local robot measure and outputs the world coordinates of the observed landmark**:

$$\boldsymbol{g}(x, y, \theta, \boldsymbol{z}) = \begin{bmatrix} x + \rho\cos(\theta + \beta) \\ y + \rho\sin(\theta + \beta) \end{bmatrix}$$



EXPANDING BOTH STATE AND COVARIANCE MATRIX

- After the *new* landmark *i* has been observed, at step *k + 1*, the **expanded state vector:**

$$\boldsymbol{\xi}^*_{k+1} \leftarrow \begin{bmatrix} \boldsymbol{\xi}_k & \boldsymbol{\lambda}^i \end{bmatrix}^T = \begin{bmatrix} \lambda^t_x & \lambda^t_y & \lambda^s_x & \lambda^s_y & \lambda^j_x & \lambda^j_y & \dots & \dots & \lambda^i_x & \lambda^i_y \end{bmatrix}^T$$

- Without losing generality, let's assume that the landmarks are **ordered incrementally in the state vector** $\rightarrow$ landmark *i* is the *i*-th discovered landmark:

$$\boldsymbol{\xi}^*_{k+1} \leftarrow \begin{bmatrix} \boldsymbol{\xi}_k & \boldsymbol{\lambda}^i \end{bmatrix}^T = \begin{bmatrix} \lambda^1_x & \lambda^1_y & \lambda^2_x & \lambda^2_y & \lambda^3_x & \lambda^3_y & \dots & \dots & \lambda^i_x & \lambda^i_y \end{bmatrix}^T$$

- The new **expanded covariance matrix**, of dimension $2i \times 2i$:

$$P^*_{k+1|k} = \begin{bmatrix} \sigma_{\lambda^1_x\lambda^1_x} & \sigma_{\lambda^1_x\lambda^1_y} & \sigma_{\lambda^1_x\lambda^2_x} & \sigma_{\lambda^1_x\lambda^2_y} & \dots & \sigma_{\lambda^1_x\lambda^i_x} & \sigma_{\lambda^1_x\lambda^i_y} \\ \sigma_{\lambda^1_y\lambda^1_x} & \sigma_{\lambda^1_y\lambda^1_y} & \sigma_{\lambda^1_y\lambda^2_x} & \sigma_{\lambda^1_y\lambda^2_y} & \dots & \sigma_{\lambda^1_y\lambda^i_x} & \sigma_{\lambda^1_y\lambda^i_y} \\ \sigma_{\lambda^2_x\lambda^1_x} & \sigma_{\lambda^2_x\lambda^1_y} & \sigma_{\lambda^2_x\lambda^2_x} & \sigma_{\lambda^2_x\lambda^2_y} & \dots & \sigma_{\lambda^2_x\lambda^i_x} & \sigma_{\lambda^2_x\lambda^i_y} \\ \sigma_{\lambda^2_y\lambda^1_x} & \sigma_{\lambda^2_y\lambda^1_y} & \sigma_{\lambda^2_y\lambda^2_x} & \sigma_{\lambda^2_y\lambda^2_y} & \dots & \sigma_{\lambda^2_y\lambda^i_x} & \sigma_{\lambda^2_y\lambda^i_y} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \sigma_{\lambda^i_x\lambda^1_x} & \sigma_{\lambda^i_x\lambda^1_y} & \sigma_{\lambda^i_x\lambda^2_x} & \sigma_{\lambda^i_x\lambda^2_y} & \dots & \sigma_{\lambda^i_x\lambda^i_x} & \sigma_{\lambda^i_x\lambda^i_y} \\ \sigma_{\lambda^i_y\lambda^1_x} & \sigma_{\lambda^i_y\lambda^1_y} & \sigma_{\lambda^i_y\lambda^2_x} & \sigma_{\lambda^i_y\lambda^2_y} & \dots & \sigma_{\lambda^i_y\lambda^i_x} & \sigma_{\lambda^i_y\lambda^i_y} \end{bmatrix} = \begin{bmatrix} \Sigma_{\boldsymbol{\lambda}^1\boldsymbol{\lambda}^1} & \Sigma_{\boldsymbol{\lambda}^1\boldsymbol{\lambda}^2} & \dots & \Sigma_{\boldsymbol{\lambda}^1\boldsymbol{\lambda}^i} \\ \Sigma_{\boldsymbol{\lambda}^2\boldsymbol{\lambda}^1} & \Sigma_{\boldsymbol{\lambda}^2\boldsymbol{\lambda}^2} & \dots & \Sigma_{\boldsymbol{\lambda}^2\boldsymbol{\lambda}^i} \\ \dots & \dots & \dots & \dots \\ \Sigma_{\boldsymbol{\lambda}^i\boldsymbol{\lambda}^1} & \Sigma_{\boldsymbol{\lambda}^i\boldsymbol{\lambda}^2} & \dots & \Sigma_{\boldsymbol{\lambda}^i\boldsymbol{\lambda}^i} \end{bmatrix}$$

THE EXPANDED COVARIANCE MATRIX

- What are the covariance values for the new predicted coordinates $(\hat{\lambda}^i_x, \hat{\lambda}^i_y)$?

The estimation error associated to the measured coordinates of the newly observed landmark corresponds to the **error associated to the sensing measure**

$\Downarrow$

The covariance for the estimates $(\hat{\lambda}^i_x, \hat{\lambda}^i_y)$ is the same as the **covariance matrix W_k modeling the sensing errors**

$$\Sigma_{\boldsymbol{\lambda}^i\boldsymbol{\lambda}^i} = W_k$$

- What are the cross-covariance values between the new predicted coordinates and those previously in the state?

Cross-covariances are all 0 in the considered model:
the observation of one landmark does not provide information about another landmark

- The resulting expanded covariance matrix:

$$\boldsymbol{P}^*_{k+1|k} = \begin{bmatrix} \boldsymbol{P}_{k|k} & \mathbf{0}_{2(i-1)\times 2} \\ \mathbf{0}_{2\times 2(i-1)} & \boldsymbol{W}_k \end{bmatrix}$$

PREDICTION UPDATE FOLLOWING A STATE EXPANSION

- The process dynamics defined through $\boldsymbol{q}()$ is <u>not linear</u> $\rightarrow$ the *Jacobian* of $\boldsymbol{q}()$ needs to be computed to define the EKF process updating equations based on the expanded state and covariance

$$\boldsymbol{q}_k(\boldsymbol{\xi}_{k|k}, z_{k+1}; x_k, y_k, \theta_k) = \begin{bmatrix} \boldsymbol{\xi}_{k|k} \\ \boldsymbol{g}(x_k, y_k, \theta_k, z_{k+1}) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\xi}_{k|k} \\ x_k + \rho_k \cos(\theta_k + \beta_k) \\ y_k + \rho_k \sin(\theta_k + \beta_k) \end{bmatrix} = \begin{bmatrix} \lambda_{kx}^1 \\ \lambda_{ky}^1 \\ \lambda_{kx}^2 \\ \lambda_{ky}^2 \\ \cdots \\ \lambda_{kx}^{i-1} \\ \lambda_{ky}^{i-1} \\ g_x^k \\ g_y^k \end{bmatrix}$$

JACOBIAN MATRIX

$$\boldsymbol{Q}_k(\boldsymbol{\lambda}_k, z_{k+1},\ w_k^{\rho}, w_k^{\beta};\ x_k, y_k, \theta_k) = \begin{bmatrix} \nabla q_{\lambda^1_{kx}} & \nabla q_{\lambda^1_{ky}} & \dots & \nabla q_{\lambda^n_{kx}} & \nabla q_{\lambda^n_{ky}} & \nabla g_{kx} & \nabla g_{ky} \end{bmatrix}^T$$

$$\boldsymbol{Q}_{k\xi} = \left[\begin{array}{ccccccc|cc}
\frac{\partial \lambda^1_{kx}}{\partial \lambda^1_{kx}} & \frac{\partial \lambda^1_{kx}}{\partial \lambda^1_{ky}} & \frac{\partial \lambda^1_{kx}}{\partial \lambda^2_{kx}} & \frac{\partial \lambda^1_{kx}}{\partial \lambda^2_{ky}} & \cdots & \frac{\partial \lambda^1_{kx}}{\partial \lambda^{i-1}_{kx}} & \frac{\partial \lambda^1_{kx}}{\partial \lambda^{i-1}_{ky}} & \frac{\partial \lambda^1_{kx}}{\partial \rho_k} & \frac{\partial \lambda^1_{kx}}{\partial \beta_k} \\
\frac{\partial \lambda^1_{ky}}{\partial \lambda^1_{kx}} & \frac{\partial \lambda^1_{ky}}{\partial \lambda^1_{ky}} & \frac{\partial \lambda^1_{ky}}{\partial \lambda^2_{kx}} & \frac{\partial \lambda^1_{ky}}{\partial \lambda^2_{ky}} & \cdots & \frac{\partial \lambda^1_{ky}}{\partial \lambda^{i-1}_{kx}} & \frac{\partial \lambda^1_{ky}}{\partial \lambda^{i-1}_{ky}} & \frac{\partial \lambda^1_{ky}}{\partial \rho_k} & \frac{\partial \lambda^1_{ky}}{\partial \beta_k} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\frac{\partial \lambda^{i-1}_{kx}}{\partial \lambda^1_{kx}} & \frac{\partial \lambda^{i-1}_{kx}}{\partial \lambda^1_{ky}} & \frac{\partial \lambda^{i-1}_{kx}}{\partial \lambda^2_{kx}} & \frac{\partial \lambda^{i-1}_{kx}}{\partial \lambda^2_{ky}} & \cdots & \frac{\partial \lambda^{i-1}_{kx}}{\partial \lambda^{i-1}_{kx}} & \frac{\partial \lambda^{i-1}_{kx}}{\partial \lambda^{i-1}_{ky}} & \frac{\partial \lambda^{i-1}_{kx}}{\partial \rho_k} & \frac{\partial \lambda^{i-1}_{kx}}{\partial \beta_k} \\
\frac{\partial \lambda^{i-1}_{ky}}{\partial \lambda^1_{kx}} & \frac{\partial \lambda^{i-1}_{ky}}{\partial \lambda^1_{ky}} & \frac{\partial \lambda^{i-1}_{ky}}{\partial \lambda^2_{kx}} & \frac{\partial \lambda^{i-1}_{ky}}{\partial \lambda^2_{ky}} & \cdots & \frac{\partial \lambda^{i-1}_{ky}}{\partial \lambda^{i-1}_{kx}} & \frac{\partial \lambda^{i-1}_{ky}}{\partial \lambda^{i-1}_{ky}} & \frac{\partial \lambda^{i-1}_{ky}}{\partial \rho_k} & \frac{\partial \lambda^{i-1}_{ky}}{\partial \beta_k} \\
\hline
\frac{\partial g^k_x}{\partial \lambda^1_{kx}} & \frac{\partial g^k_x}{\partial \lambda^1_{ky}} & \frac{\partial g^k_x}{\partial \lambda^2_{kx}} & \frac{\partial g^k_x}{\partial \lambda^2_{ky}} & \cdots & \frac{\partial g^k_x}{\partial \lambda^{i-1}_{kx}} & \frac{\partial g^k_x}{\partial \lambda^{i-1}_{ky}} & \frac{\partial g^k_x}{\partial \rho_k} & \frac{\partial g^k_x}{\partial \beta_k} \\
\frac{\partial g^k_y}{\partial \lambda^1_{kx}} & \frac{\partial g^k_y}{\partial \lambda^1_{ky}} & \frac{\partial g^k_y}{\partial \lambda^2_{kx}} & \frac{\partial g^k_y}{\partial \lambda^2_{ky}} & \cdots & \frac{\partial g^k_y}{\partial \lambda^{i-1}_{kx}} & \frac{\partial g^k_y}{\partial \lambda^{i-1}_{ky}} & \frac{\partial g^k_y}{\partial \rho_k} & \frac{\partial g^k_y}{\partial \beta_k}
\end{array}\right]$$

JACOBIANS FOR LINEARIZATION OF $q_k()$

- Said $n = (i-1)$, the number of landmarks before the current new observation, the dimension of the covariance matrix before the expansion is $2n \times 2n$, while the dimension of the Jacobian $\boldsymbol{Q}$ is $(2n+2) \times (2n+3+2)$
- $\boldsymbol{Q}_{k\xi}$ is evaluated at the current observation z_{k+1}, therefore is also referred to as $\boldsymbol{Q}_{z_{k+1}}$
- The Jacobian $\boldsymbol{Q}_{k\xi}$ of $\boldsymbol{q}_k()$ for the **linearization of the state dynamics in the case of state expansion** can be also expressed in a more compact form:

$$\boldsymbol{Q}_{k\xi} = \begin{bmatrix} \boldsymbol{I}_{2n\times 2n} & \boldsymbol{0}_{2n\times 2} \\ \boldsymbol{G}_\xi & \boldsymbol{G}_z \end{bmatrix}_{\hat{\xi}_{k|k}, z_{k+1}, x_k, y_k, \theta_k, \boldsymbol{w}=0}$$

$$\boldsymbol{G}_\xi = \frac{\partial \boldsymbol{g}_k}{\partial \boldsymbol{\xi}_k} = \begin{bmatrix} 0 & \dots & 0 \\ 0 & \dots & 0 \end{bmatrix} \qquad \boldsymbol{G}_z = \frac{\partial \boldsymbol{g}_k}{\partial z_k} = \begin{bmatrix} \cos(\theta_k + \beta_k) & -\rho_k \sin(\theta_k + \beta_k) \\ \sin(\theta_k + \beta_k) & \rho_k \cos(\theta_k + \beta_k) \end{bmatrix}$$

- **The Jacobian $Q_{k\xi}$ allows to linearly transform (expand) the covariance matrix P_k when adding a new landmark to the state vector:**

$$P^*_{k|k+1} = \boldsymbol{Q}_{k\xi} \begin{bmatrix} P_{k|k} & \boldsymbol{0}_{2n\times 2} \\ \boldsymbol{0}_{2\times 2n} & W_k \end{bmatrix} \boldsymbol{Q}^T_{k\xi} = \begin{bmatrix} P_{k|k} & \boldsymbol{0}_{2n\times 2} \\ \boldsymbol{0}_{2\times 2n} & \boldsymbol{G}_z W_k \boldsymbol{G}_z^T \end{bmatrix}$$

which results in an expanded $2(n+1) \times 2(n+1)$ matrix

EKF INCLUDING STATE EXPANSION

- The Jacobians $\mathbf{L}_{k\xi}$ and $\mathbf{L}_{kw}$ of $\boldsymbol{\ell}_k()$, computed before, accounts for the **linearization of the observation model** when an already discovered landmark is observed (x_k and y_k are parameters):

$$\mathbf{L}_{k\xi} = \begin{bmatrix} 0 & 0 & \dots & \dfrac{\lambda^i_{kx} - x_k}{r^i_k} & \dfrac{\lambda^i_{ky} - y_k}{r^i_k} & \dots & 0 & 0 \\ 0 & 0 & \dots & -\dfrac{\lambda^i_{ky} - y_k}{(r^i_k)^2} & \dfrac{\lambda^i_{kx} - x_k}{(r^i_k)^2} & \dots & 0 & 0 \end{bmatrix}_{\hat{\xi}_{k+1|k}, \mathbf{0}} \qquad \mathbf{L}_{kw} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

- The complete EKF equations:

When a *new landmark i* is observed:

$$\hat{\xi}^*_{k+1|k} = \mathbf{q}_k(\hat{\boldsymbol{\lambda}}_{k|k}, z_{k+1}, x_k, y_k, \theta_k)$$

$$\mathbf{P}^*_{k+1|k} = \begin{bmatrix} \mathbf{P}_{k|k} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_z \mathbf{W}_k \mathbf{G}_z^T \end{bmatrix}$$

At every time step $k+1$ *a landmark i* is observed

$$\hat{\xi}_{k+1} = \hat{\xi}_{k+1|k} + \mathbf{G}_{k+1}(z_{k+1} - \boldsymbol{\ell}_k(\hat{\boldsymbol{\lambda}}_{k+1|k}, \mathbf{0}; x_k, y_k, \theta_k))$$

$$\mathbf{P}_{k+1} = \mathbf{P}_{k+1|k} - \mathbf{G}_{k+1}\mathbf{L}_{k\xi}\mathbf{P}_{k+1|k}$$

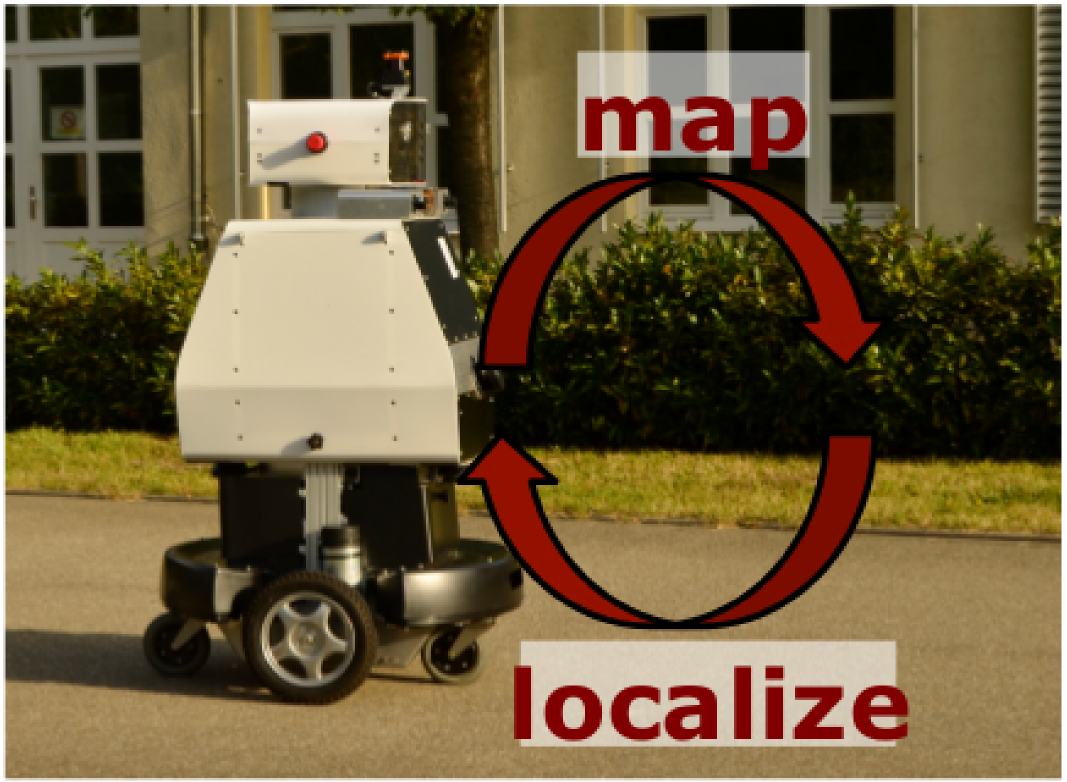
$$\mathbf{G}_{k+1} = \mathbf{P}_{k+1|k}\mathbf{L}_{k\xi}{}^T \mathbf{S}^{-1}_{k+1}$$

$$\mathbf{S}_{k+1} = \mathbf{L}_{k\xi}\mathbf{P}_{k+1|k}\mathbf{L}_{k\xi}{}^T + \mathbf{L}_{kw}\mathbf{W}_{k+1}\mathbf{L}_{kw}{}^T$$

SOLVING LOCALIZATION AND MAPPING TOGETHER

- The previous derivation was based on the fact that robot's localization was perfect
- Since this is not (usually) the case, let's solve the joint problem:

Simultaneous Localization and Mapping (SLAM)

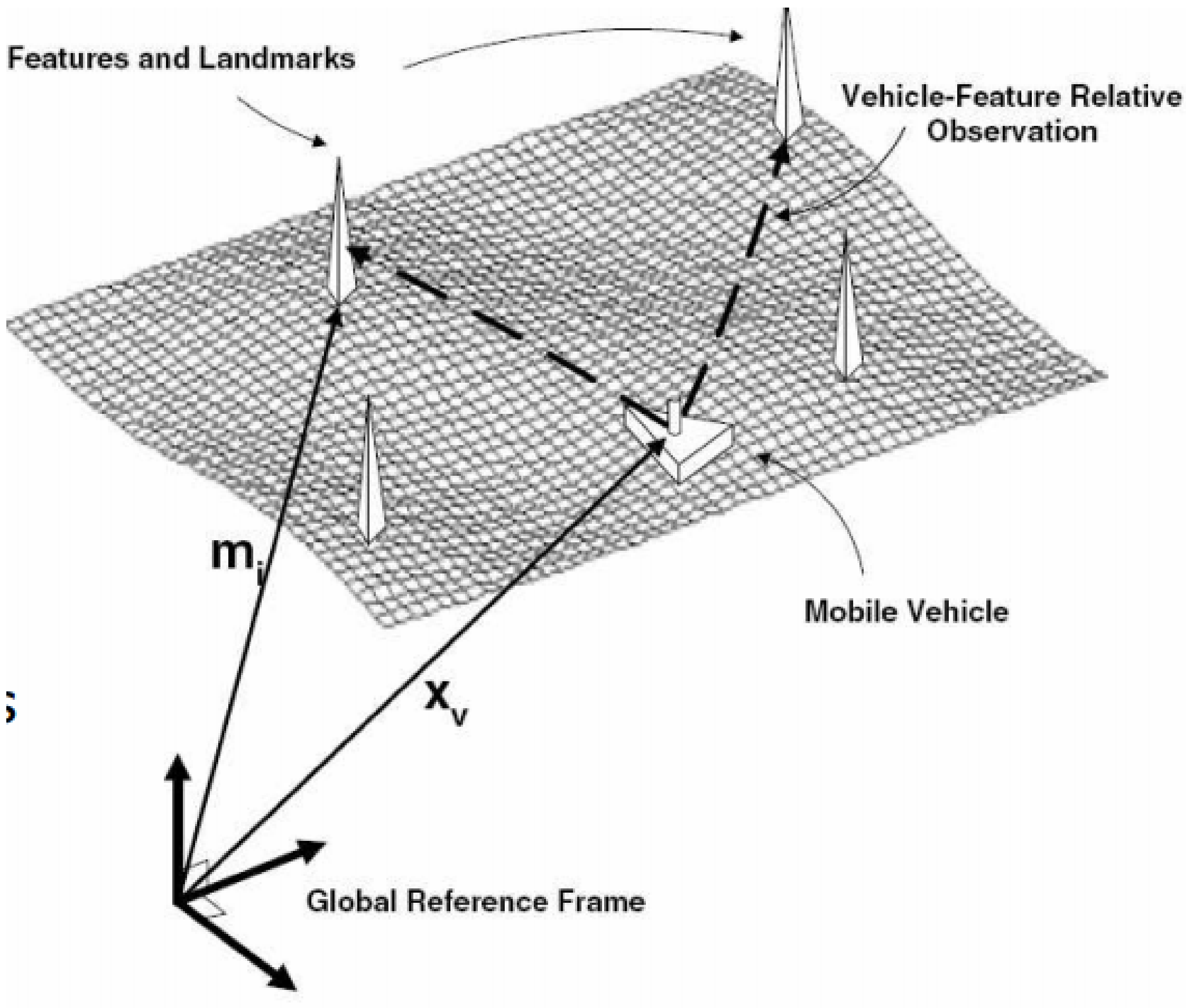


The state vector needs to include also robot's coordinates, since also robot's pose needs to be estimated

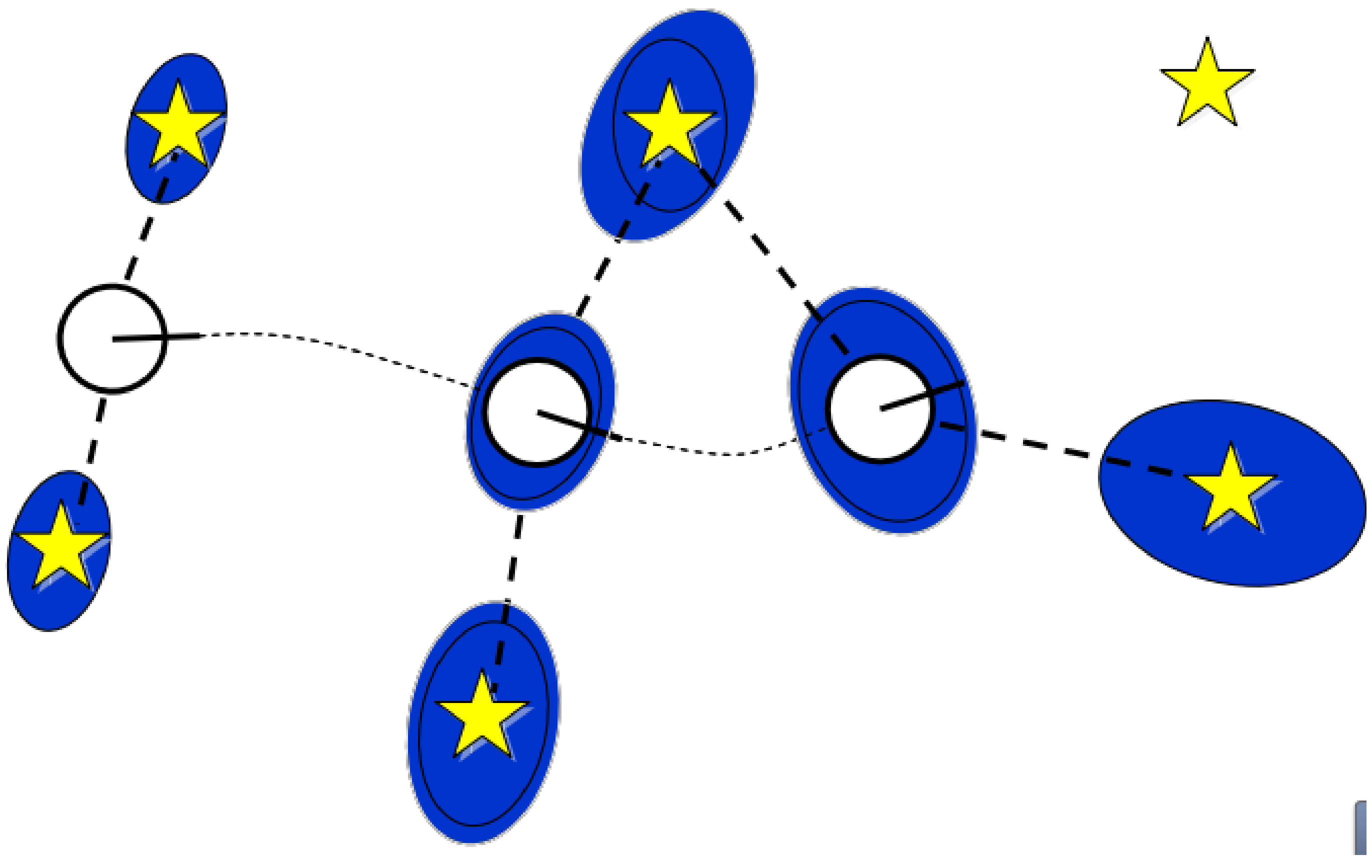
$$\boldsymbol{\xi} = \left[x, y, \theta, \lambda_x^1\ \lambda_y^1\ \lambda_x^2\ \lambda_y^2\ \ldots\ \lambda_x^M\ \lambda_y^M\right]^T$$

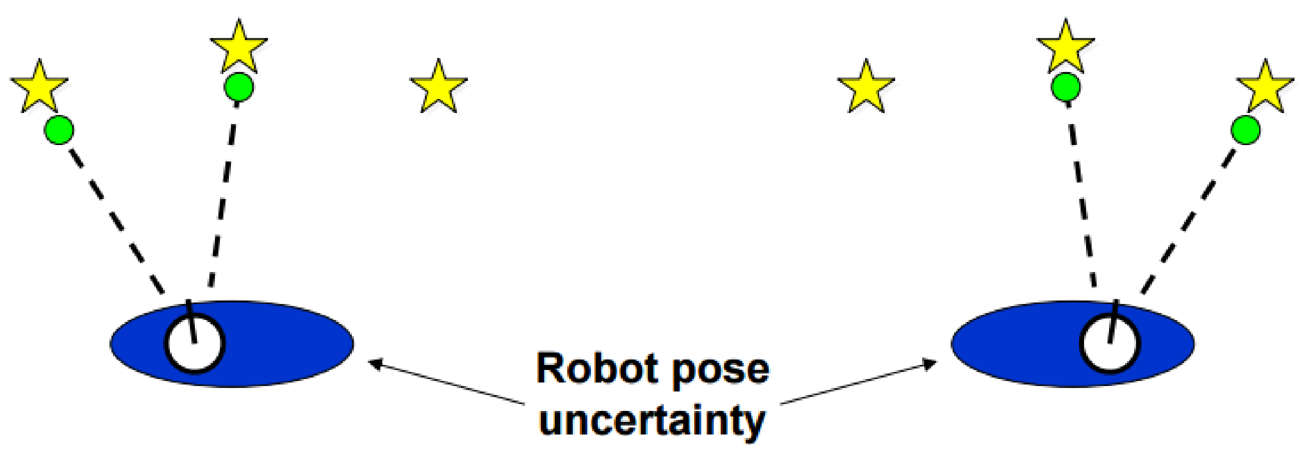
$\boldsymbol{\xi}$ has now dimension $(2M+3) \times 1$, and $\boldsymbol{P}$ is a $(2M+3) \times (2M+3)$ matrix

THE SLAM CHALLENGE IN A NUTSHELL



- Looking for **absolute robot pose**
- Looking for **absolute landmark positions**
- But only relative measurements of landmarks are available





- Robot path and the landmark map are both unknown
- Errors in map and pose estimates are correlated
- Mapping between observations and landmarks is unknown
- *Data Association:* Selecting wrong data association can be disastrous (divergence)

STATE VECTOR AND COVARIANCE MATRIX

- The state vector and the covariance matrix: the state vector includes both the robot's generalized coordinates and landmarks' coordinates; said $n = (i-1)$ the number of discovered landmarks at step k:

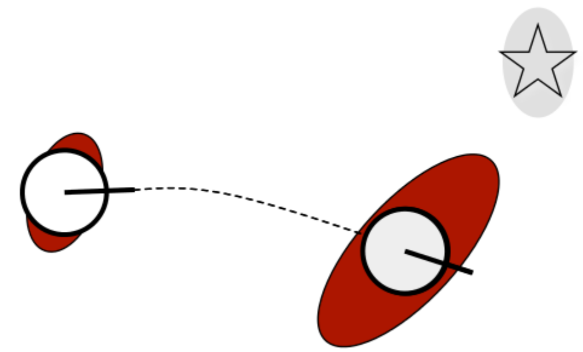
$$\boldsymbol{\xi}_k = \begin{bmatrix} x_k & y_k & \theta_k & \boldsymbol{\lambda}_k^1 & \boldsymbol{\lambda}_k^2 & \dots & \boldsymbol{\lambda}_k^n \end{bmatrix}^T = \begin{bmatrix} \boldsymbol{\xi}_{R_k} & \boldsymbol{\xi}_{\lambda_k^n} \end{bmatrix}^T$$

and the resulting covariance matrix (omitting the index k) is:

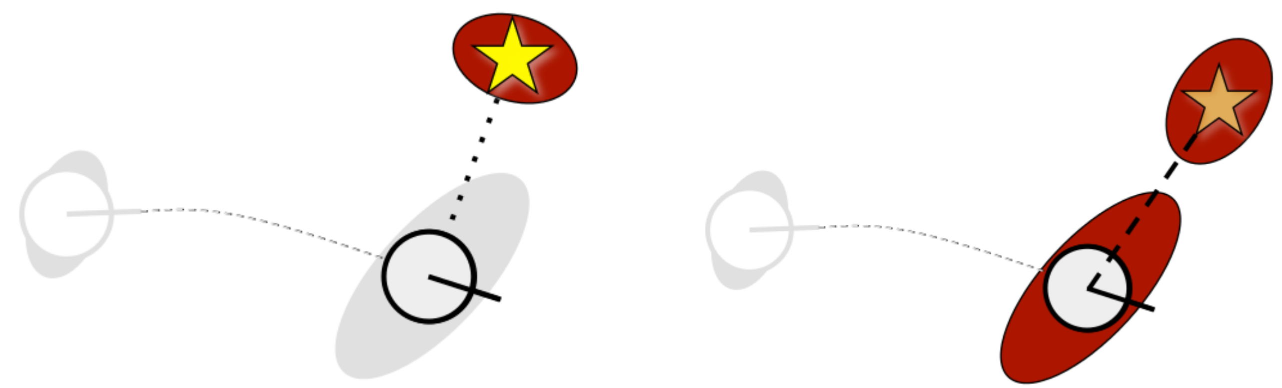
$$\boldsymbol{P}_{(3+2n)\times(3+2n)} = \left[\begin{array}{ccc|ccccc}
\sigma_{xx} & \sigma_{xy} & \sigma_{x\theta} & \sigma_{x\lambda_x^1} & \sigma_{x\lambda_y^1} & \dots & \sigma_{x\lambda_x^n} & \sigma_{x\lambda_y^n} \\
\sigma_{yx} & \sigma_{yy} & \sigma_{y\theta} & \sigma_{y\lambda_x^1} & \sigma_{y\lambda_y^1} & \dots & \sigma_{y\lambda_x^n} & \sigma_{y\lambda_y^n} \\
\sigma_{\theta x} & \sigma_{\theta y} & \sigma_{\theta\theta} & \sigma_{\theta\lambda_x^1} & \sigma_{\theta\lambda_y^1} & \dots & \sigma_{\theta\lambda_x^n} & \sigma_{\theta\lambda_y^n} \\
\hline
\sigma_{\lambda_x^1 x} & \sigma_{\lambda_x^1 y} & \sigma_{\lambda_x^1 \theta} & \sigma_{\lambda_x^1\lambda_x^1} & \sigma_{\lambda_x^1\lambda_y^1} & \dots & \sigma_{\lambda_x^1\lambda_x^n} & \sigma_{\lambda_x^1\lambda_y^n} \\
\sigma_{\lambda_y^1 x} & \sigma_{\lambda_y^1 y} & \sigma_{\lambda_y^1 \theta} & \sigma_{\lambda_y^1\lambda_x^1} & \sigma_{\lambda_y^1\lambda_y^1} & \dots & \sigma_{\lambda_y^1\lambda_x^n} & \sigma_{\lambda_y^1\lambda_y^n} \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
\sigma_{\lambda_x^n x} & \sigma_{\lambda_x^n y} & \sigma_{\lambda_x^n \theta} & \sigma_{\lambda_x^n\lambda_x^1} & \sigma_{\lambda_x^n\lambda_y^1} & \dots & \sigma_{\lambda_x^n\lambda_x^n} & \sigma_{\lambda_x^n\lambda_y^n} \\
\sigma_{\lambda_y^n x} & \sigma_{\lambda_y^n y} & \sigma_{\lambda_y^n \theta} & \sigma_{\lambda_y^n\lambda_x^1} & \sigma_{\lambda_y^n\lambda_y^1} & \dots & \sigma_{\lambda_y^n\lambda_x^n} & \sigma_{\lambda_y^n\lambda_y^n}
\end{array}\right] = \begin{bmatrix} \Sigma_{RR} & \Sigma_{R\boldsymbol{\lambda}} \\ \Sigma_{R\boldsymbol{\lambda}}^T & \Sigma_{\boldsymbol{\lambda\lambda}} \end{bmatrix}$$

THE SLAM *MOVIE*

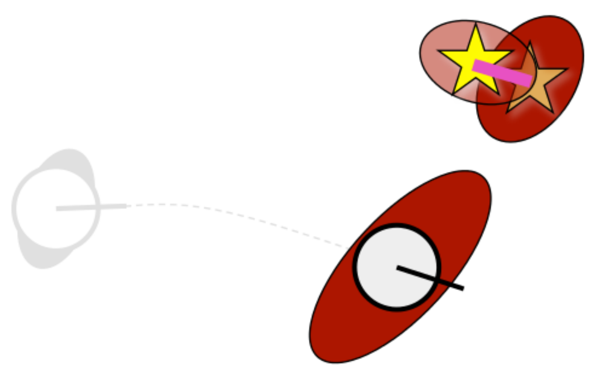
- **Robot's pose prediction** using the odometry model



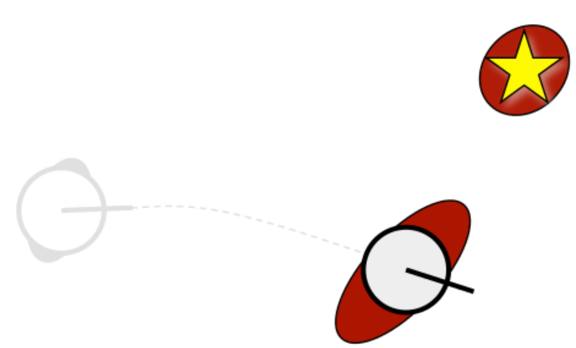
- **Observe** the environment while moving:
  1. Get a landmark observation
  2. Perform a measurement prediction using the map (being) created



- Compare predicted and observed measure to perform **data association** (is a new landmark?) and compute **innovation**



- **Update predictions** for robot pose *and* landmark's position



EKF EQUATIONS FOR SLAM?

Combination of all equations from previous cases!

1. Robot's pose prediction using the odometry model: EKF based on the linearization of the motion equations using the Jacobians $\boldsymbol{F}_{k\boldsymbol{\xi}}$ and $\boldsymbol{F}_{k\boldsymbol{v}}$ computed previously, and the new (constant) Jacobian $\boldsymbol{F}_{k\boldsymbol{\lambda}}$, to reduce the motion equation to a linear form of the type $\boldsymbol{\xi}_{k+1} = \boldsymbol{A}_k\boldsymbol{\xi}_k + \boldsymbol{\nu}_k \rightarrow$ EKF where *non linearity refers to process dynamics*

2. Robot's pose prediction *correction* after the sensory observation of a landmark, whose position is reported on a *map*: EKF based on the linearization of the observation equations using the Jacobians $\boldsymbol{H}_{k\boldsymbol{\xi}}$ and $\boldsymbol{H}_{k\boldsymbol{w}}$ to reduce the observation equation to a linear form of the type $\boldsymbol{z}_{k+1} = \boldsymbol{C}_k\boldsymbol{\xi}_k + \boldsymbol{w}_k \rightarrow$ EKF where *non linearity refers to measurement process*

3. Landmark's position estimation using robot's sensory data and known pose (map building): EKF based on the linearization of the observation equations using the Jacobians $\boldsymbol{L}_{k\boldsymbol{\xi}}$ and $\boldsymbol{L}_{k\boldsymbol{w}}$ to reduce the observation equation to a linear form of the type $\boldsymbol{z}_{k+1} = \boldsymbol{C}_k\boldsymbol{\xi}_k + \boldsymbol{w}_k \rightarrow$ EKF where *non linearity refers to measurement process*

4. State expansion after observing a previously unseen landmark: EKF based on the linearization of the process dynamics using the Jacobian $\boldsymbol{Q}_{kz}$ to reduce the process equation to a linear form of the type $\boldsymbol{\xi}_{k+1} = \boldsymbol{A}_k\boldsymbol{\xi}_k + \boldsymbol{\nu}_k \rightarrow$ EKF where *non linearity refers to system evolution*

STATE AND OBSERVATION EQUATIONS FOR SLAM

- State equation for robot motion, including also the $\boldsymbol{\lambda}$ in the state vector:

At each step k: $$\boldsymbol{\xi}_{k+1} = \begin{bmatrix} x_k \\ y_k \\ \theta_k \\ \boldsymbol{\lambda}_k^1 \\ \cdots \\ \boldsymbol{\lambda}_k^n \end{bmatrix} + \begin{bmatrix} (\Delta S_k + \nu_k^s)\cos(\theta_k + \frac{\Delta\theta_k}{2} + \nu_k^\theta) \\ (\Delta S_k + \nu_k^s)\sin(\theta_k + \frac{\Delta\theta_k}{2} + \nu_k^\theta) \\ \Delta\theta_k + \nu_k^\theta \\ \mathbf{0} \\ \cdots \\ \mathbf{0} \end{bmatrix} = \boldsymbol{f}_k(\boldsymbol{\xi}_k, \boldsymbol{\nu}_k; \Delta S_k, \Delta\theta_k)$$

- State equation for state expansion, when a previously unseen landmark i is observed:

$$\boldsymbol{\xi}_{k+1}^* = \begin{bmatrix} \boldsymbol{\xi}_k \\ \boldsymbol{g}_k(x_k, y_k, \theta_k, z_k) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\xi}_k \\ x_k + \rho_k\cos(\theta_k + \beta_k) \\ y_k + \rho_k\sin(\theta_k + \beta_k) \end{bmatrix} = \boldsymbol{q}_k(\boldsymbol{\xi}_k, z_{k+1})$$

- Observation model, which is unchanged from the previous scenario, but now $\lambda_{kx}^i, \lambda_{ky}^i, x_k, y_k, \theta_k$ are variables, such that z_{k+1} can be seen as the overlapping of the functions $\boldsymbol{h}_k()$ and $\boldsymbol{\ell}_k()$ that have been considered for the localization-only and mapping-only cases before:

$$z_{k+1} = \begin{bmatrix} \rho_{k+1}^i \\ \beta_{k+1}^i \end{bmatrix} = \begin{bmatrix} \sqrt{(\lambda_{kx}^i - x_k)^2 + (\lambda_{ky}^i - y_k)^2} \\ \arctan\left((\lambda_{ky}^i - y_k)/(\lambda_{kx}^i - x_k)\right) - \theta_k \end{bmatrix} + \begin{bmatrix} w_k^\rho \\ w_k^\beta \end{bmatrix} = \boldsymbol{o}_k(\boldsymbol{\xi}_k, \boldsymbol{w}_k)$$

- The Jacobians $\boldsymbol{F}_{k\boldsymbol{\xi}}$, and $\boldsymbol{F}_{k\boldsymbol{\nu}}$ of $f_k()$, to be evaluated in $(\boldsymbol{\xi}_k = \hat{\boldsymbol{\xi}}_{k|k}, \boldsymbol{\nu}_k = 0)$, for the **linearization of the motion dynamics**, which is robot's motion since landmarks do not move (Jacobians are (re)computed because of the new sub-matrix $\boldsymbol{F}_{k\boldsymbol{\xi}_{\boldsymbol{\lambda}^n}}$ in the $\boldsymbol{F}_{k\boldsymbol{\xi}}$ Jacobian):

$$
\begin{aligned}
f_{kx} &= x_k + (\Delta S_k + \nu_k^s)\cos(\theta_k + \tfrac{\Delta\theta_k}{2} + \nu_k^\theta)\\
f_{ky} &= y_k + (\Delta S_k + \nu_k^s)\sin(\theta_k + \tfrac{\Delta\theta_k}{2} + \nu_k^\theta)\\
f_{k\theta} &= \theta_k + \Delta\theta_k + \nu_k^\theta\\
f_{k\boldsymbol{\lambda}^1} &= \boldsymbol{\lambda}_k^1\\
\cdots & \quad \cdots\\
f_{k\boldsymbol{\lambda}^n} &= \boldsymbol{\lambda}_k^n
\end{aligned}
\Rightarrow
\begin{bmatrix}
\frac{\partial f_{kx}}{\partial x_k} & \frac{\partial f_{kx}}{\partial y_k} & \frac{\partial f_{kx}}{\partial \theta_k} & \frac{\partial f_{kx}}{\partial \boldsymbol{\lambda}_k^1} & \cdots & \frac{f_{kx}}{\partial \boldsymbol{\lambda}_k^n} & \frac{\partial f_{kx}}{\partial \nu_k^s} & \frac{\partial f_{kx}}{\partial \nu_k^\theta}\\
\frac{\partial f_{ky}}{\partial x_k} & \frac{\partial f_{ky}}{\partial y_k} & \frac{\partial f_{ky}}{\partial \theta_k} & \frac{\partial f_{ky}}{\partial \boldsymbol{\lambda}_k^1} & \cdots & \frac{f_{ky}}{\partial \boldsymbol{\lambda}_k^n} & \frac{\partial f_{ky}}{\partial \nu_k^s} & \frac{\partial f_{ky}}{\partial \nu_k^\theta}\\
\frac{\partial f_{k\theta}}{\partial x_k} & \frac{\partial f_{k\theta}}{\partial y_k} & \frac{\partial f_{k\theta}}{\partial \theta_k} & \frac{\partial f_{k\theta}}{\partial \boldsymbol{\lambda}_k^1} & \cdots & \frac{f_{k\theta}}{\partial \boldsymbol{\lambda}_k^n} & \frac{\partial f_{k\theta}}{\partial \nu_k^s} & \frac{\partial f_{k\theta}}{\partial \nu_k^\theta}\\
\frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial x_k} & \frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial y_k} & \frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial \theta_k} & \frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial \boldsymbol{\lambda}_k^1} & \cdots & \frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial \boldsymbol{\lambda}_k^n} & \frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial \nu_k^s} & \frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial \nu_k^\theta}\\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots\\
\frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial x_k} & \frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial y_k} & \frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial \theta_k} & \frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial \boldsymbol{\lambda}_k^1} & \cdots & \frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial \boldsymbol{\lambda}_k^n} & \frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial \nu_k^s} & \frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial \nu_k^\theta}
\end{bmatrix}
= \begin{bmatrix}\boldsymbol{F}_{k\boldsymbol{\xi}_R} & \boldsymbol{F}_{k\boldsymbol{\xi}_{\boldsymbol{\lambda}^n}} & \boldsymbol{F}_{k\boldsymbol{\nu}}\end{bmatrix}
$$

JACOBIANS FOR ROBOT POSE

$$
\begin{bmatrix}
\frac{\partial f_{kx}}{\partial x_k} & \frac{\partial f_{kx}}{\partial y_k} & \frac{\partial f_{kx}}{\partial \theta_k} & \frac{\partial f_{kx}}{\partial \boldsymbol{\lambda}_k^1} & \cdots & \frac{f_{kx}}{\partial \boldsymbol{\lambda}_k^n} & \frac{\partial f_{kx}}{\partial \nu_k^s} & \frac{\partial f_{kx}}{\partial \nu_k^\theta} \\
\frac{\partial f_{ky}}{\partial x_k} & \frac{\partial f_{ky}}{\partial y_k} & \frac{\partial f_{ky}}{\partial \theta_k} & \frac{\partial f_{ky}}{\partial \boldsymbol{\lambda}_k^1} & \cdots & \frac{f_{ky}}{\partial \boldsymbol{\lambda}_k^n} & \frac{\partial f_{ky}}{\partial \nu_k^s} & \frac{\partial f_{ky}}{\partial \nu_k^\theta} \\
\frac{\partial f_{k\theta}}{\partial x_k} & \frac{\partial f_{k\theta}}{\partial y_k} & \frac{\partial f_{k\theta}}{\partial \theta_k} & \frac{\partial f_{k\theta}}{\partial \boldsymbol{\lambda}_k^1} & \cdots & \frac{f_{k\theta}}{\partial \boldsymbol{\lambda}_k^n} & \frac{\partial f_{k\theta}}{\partial \nu_k^s} & \frac{\partial f_{k\theta}}{\partial \nu_k^\theta} \\
\frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial x_k} & \frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial y_k} & \frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial \theta_k} & \frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial \boldsymbol{\lambda}_k^1} & \cdots & \frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial \boldsymbol{\lambda}_k^n} & \frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial \nu_k^s} & \frac{\partial f_{k\boldsymbol{\lambda}^1}}{\partial \nu_k^\theta} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
\frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial x_k} & \frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial y_k} & \frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial \theta_k} & \frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial \boldsymbol{\lambda}_k^1} & \cdots & \frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial \boldsymbol{\lambda}_k^n} & \frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial \nu_k^s} & \frac{\partial f_{k\boldsymbol{\lambda}^n}}{\partial \nu_k^\theta}
\end{bmatrix}
= \begin{bmatrix} \boldsymbol{F}_{k\xi_R} & \boldsymbol{F}_{k\xi_{\lambda^n}} & \boldsymbol{F}_{k\nu} \end{bmatrix}
$$

$$
\boldsymbol{F}_{k\xi} = \left[\begin{array}{ccc|c}
1 & 0 & -\Delta S_k \sin(\theta_k + \frac{\Delta\theta_k}{2}) & \\
0 & 1 & \Delta S_k \cos(\theta_k + \frac{\Delta\theta_k}{2}) & \mathbf{0}_{3\times 2n} \\
0 & 0 & 1 & \\
\hline
 & \mathbf{0}_{2n\times 3} & & \boldsymbol{I}_{2n\times 2n}
\end{array}\right]_{\hat{\boldsymbol{\xi}}_{k|k},\boldsymbol{\nu}=0}
$$

$$
\boldsymbol{F}_{k\nu} = \begin{bmatrix}
\cos(\theta_k + \frac{\Delta\theta_k}{2}) & -\Delta S_k \sin(\theta_k + \frac{\Delta\theta_k}{2}) \\
\sin(\theta_k + \frac{\Delta\theta_k}{2}) & \Delta S_k \cos(\theta_k + \frac{\Delta\theta_k}{2}) \\
0 & 1 \\
\multicolumn{2}{c}{\mathbf{0}_{2n\times 2}}
\end{bmatrix}_{\hat{\boldsymbol{\xi}}_{k|k},\boldsymbol{\nu}=0}
$$

JACOBIANS FOR THE OBSERVATION MODELS (LOCALIZATION AND MAPPING)

- The Jacobians $\boldsymbol{O}_{k\xi}$ and $\boldsymbol{O}_{kw}$ of $\boldsymbol{o}_k()$ for the **linearization of the observation model** for both localization and mapping, to be evaluated in $(\boldsymbol{\xi}_k = \hat{\boldsymbol{\xi}}_{k+1|k}, \boldsymbol{w}_k = 0)$. The Jacobians are derived from the combination of the Jacobians $\boldsymbol{H}_{k\xi}$ and $\boldsymbol{L}_{k\xi}$ computed for the localization-only and mapping-only cases:

$$
\begin{aligned}
o_{k\rho} &= \sqrt{(\lambda^i_{kx} - x_k)^2 + (\lambda^i_{ky} - y_k)^2} + w_k^{\rho} \\
o_{k\beta} &= \arctan\Big((\lambda^i_{yx} - y_k)/(\lambda^i_{kx} - x_k)\Big) - \theta_k + w_k^{\beta}
\end{aligned}
\qquad
\boldsymbol{O}_k(x_k, y_k, \theta_k,\ \boldsymbol{\lambda}_k,\ w_k^{\rho}, w_k^{\beta}) = \left[\nabla h_{k\rho} \quad \nabla h_{k\beta}\right]^T
$$

$$
\boldsymbol{O}_k = \begin{bmatrix}
\frac{\partial o_{k\rho}}{\partial x_k} & \frac{\partial o_{k\rho}}{\partial y_k} & \frac{\partial o_{k\rho}}{\partial \theta_k} & \frac{\partial o_{k\rho}}{\partial \lambda^1_{kx}} & \frac{\partial o_{k\rho}}{\partial \lambda^2_{ky}} & \cdots & \frac{\partial o_{k\rho}}{\partial \lambda^n_{kx}} & \frac{\partial o_{k\rho}}{\partial \lambda^n_{ky}} & \frac{\partial o_{k\rho}}{\partial w_k^{\rho}} & \frac{\partial o_{k\rho}}{\partial w_k^{\beta}} \\
\frac{\partial o_{k\beta}}{\partial x_k} & \frac{\partial o_{k\beta}}{\partial y_k} & \frac{\partial o_{k\beta}}{\partial \theta_k} & \frac{\partial o_{k\beta}}{\partial \lambda^1_{kx}} & \frac{\partial o_{k\beta}}{\partial \lambda^2_{ky}} & \cdots & \frac{\partial o_{k\beta}}{\partial \lambda^n_{kx}} & \frac{\partial o_{k\beta}}{\partial \lambda^n_{ky}} & \frac{\partial o_{k\beta}}{\partial w_k^{\rho}} & \frac{\partial o_{k\beta}}{\partial w_k^{\beta}}
\end{bmatrix} = \left[\boldsymbol{O}_{k\xi_R} \quad \boldsymbol{O}_{k\xi_{\lambda^n}} \quad \boldsymbol{O}_{kw}\right]
$$

$$
\boldsymbol{O}_{k\xi} = \begin{bmatrix}
-\frac{\lambda^i_{kx} - x_k}{r^i_k} & -\frac{\lambda^i_{ky} - y_k}{r^i_k} & 0 & 0 \; 0 & \ldots & \frac{\lambda^i_{kx} - x_k}{r^i_k} & \frac{\lambda^i_{ky} - y_k}{r^i_k} & \ldots & 0 \; 0 \\
\frac{\lambda^i_{ky} - y_k}{(r^i_k)^2} & -\frac{\lambda^i_{kx} - x_k}{(r^i_k)^2} & -1 & 0 \; 0 & \ldots & -\frac{\lambda^i_{ky} - y_k}{(r^i_k)^2} & \frac{\lambda^i_{kx} - x_k}{(r^i_k)^2} & \ldots & 0 \; 0
\end{bmatrix}
\qquad
\boldsymbol{O}_{kw} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
$$

$$
r^i_k = \sqrt{(\lambda^i_{kx} - x_k)^2 + (\lambda^i_{ky} - y_k)^2}, \qquad \boldsymbol{O}_{k\xi} \text{ is sparse: } \boldsymbol{O}_{k\xi} = \left[\boldsymbol{H}_{k\xi_R} \ \boldsymbol{0} \cdots \boldsymbol{0} \ \boldsymbol{L}_{k\xi_{\lambda^i}} \ \boldsymbol{0} \cdots \boldsymbol{0}\right]^T
$$

JACOBIANS FOR LANDMARK STATE EXPANSION

$$\boldsymbol{q}_k(\boldsymbol{\xi}_{k|k}, \boldsymbol{z}_{k+1}, x_k, y_k, \theta_k) = \begin{bmatrix} \boldsymbol{\xi}_{k|k} \\ \boldsymbol{g}(x_k, y_k, \theta_k, \boldsymbol{z}_{k+1}) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\xi}_{k|k} \\ x_k + \rho_k \cos(\theta_k + \beta_k) \\ y_k + \rho_k \sin(\theta_k + \beta_k) \end{bmatrix} = \begin{bmatrix} \lambda^1_{kx} \\ \lambda^1_{ky} \\ \lambda^2_{kx} \\ \lambda^2_{ky} \\ \cdots \\ \lambda^{i-1}_{kx} \\ \lambda^{i-1}_{ky} \\ g^k_x \\ g^k_y \end{bmatrix}$$

- The Jacobian $\boldsymbol{Q}_{k\xi}$ of $\boldsymbol{q}_k()$ for the **linearization of the state dynamics when a new landmark is observed (state expansion with landmark initialization)**; $\boldsymbol{Q}_{k\xi}$ is obtained as in the mapping-only case + the fact that now $\boldsymbol{G}_{\xi} = \begin{bmatrix} \boldsymbol{G}_{\xi_R} & \boldsymbol{G}_{\xi_\lambda} \end{bmatrix}$ is non-zero since $\boldsymbol{\xi}_R$ is part of the state vector, making $\boldsymbol{G}_{\xi_R} \neq \boldsymbol{0}$:

$$\boldsymbol{Q}_{k\xi} = \begin{bmatrix} \boldsymbol{I}_{(3+2n)\times(3+2n)} & \boldsymbol{0}_{(3+2n)\times 2} \\ \boldsymbol{G}_{\xi_R} \quad \boldsymbol{0}_{2\times 2n} & \boldsymbol{G}_z \end{bmatrix}_{\hat{\boldsymbol{\xi}}_{k|k}, \boldsymbol{z}_k, \boldsymbol{w}=0}$$

$$\boldsymbol{G}_{\xi_R} = \frac{\partial \boldsymbol{g}_k}{\partial \boldsymbol{\xi}_R} = \begin{bmatrix} 1 & 0 & -\rho_k \sin(\theta_k + \beta_k) \\ 0 & 1 & \rho_k \cos(\theta_k + \beta_k) \end{bmatrix} \qquad \boldsymbol{G}_z = \frac{\partial \boldsymbol{g}_k}{\partial z} = \begin{bmatrix} \cos(\theta_k + \beta_k) & -\rho_k \sin(\theta_k + \beta_k) \\ \sin(\theta_k + \beta_k) & \rho_k \cos(\theta_k + \beta_k) \end{bmatrix}$$

- The SLAM EKF equations:

At every time step k:

$$\hat{\boldsymbol{\xi}}_{k+1|k} = \boldsymbol{f}_k(\hat{\boldsymbol{\xi}}_{k|k}, \boldsymbol{0}; \Delta S_k, \Delta\theta_k)$$

$$\boldsymbol{P}_{k+1|k} = \boldsymbol{F}_{k\xi}\boldsymbol{P}_k\boldsymbol{F}_{k\xi}{}^T + \boldsymbol{F}_{k\nu}\boldsymbol{V}_k\boldsymbol{F}_{k\nu}{}^T$$

When a *new landmark i* is observed:

$$\hat{\boldsymbol{\xi}}^*_{k+1|k} = \boldsymbol{q}_k(\hat{\boldsymbol{\xi}}_{k|k}, z_{k+1})$$

$$\boldsymbol{P}^*_{k+1|k} = \boldsymbol{Q}_{k\xi}\begin{bmatrix}\boldsymbol{P}_{k|k} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{W}_k\end{bmatrix}\boldsymbol{Q}_{k\xi}{}^T$$

At every time step $k+1$ *a landmark i* is observed

$$\hat{\boldsymbol{\xi}}_{k+1} = \hat{\boldsymbol{\xi}}_{k+1|k} + \boldsymbol{G}_{k+1}(z_{k+1} - \boldsymbol{o}_k(\hat{\boldsymbol{\xi}}_{k+1|k}, \boldsymbol{0}))$$

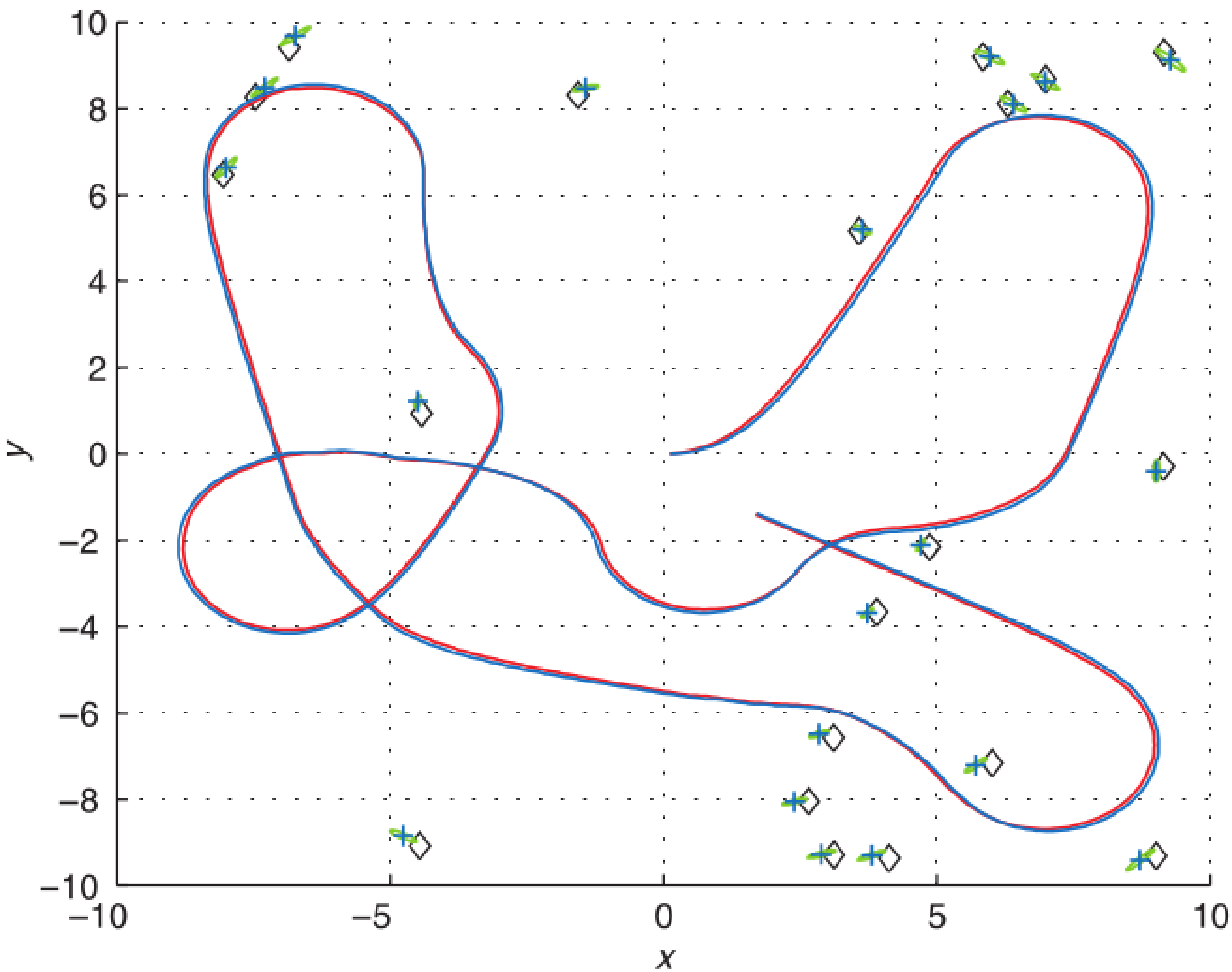
$$\boldsymbol{P}_{k+1} = \boldsymbol{P}_{k+1|k} - \boldsymbol{G}_{k+1}\boldsymbol{O}_{k\xi}\boldsymbol{P}_{k+1|k}$$

$$\boldsymbol{G}_{k+1} = \boldsymbol{P}_{k+1|k}\boldsymbol{O}_{k\xi}{}^T\boldsymbol{S}_{k+1}^{-1}$$

$$\boldsymbol{S}_{k+1} = \boldsymbol{O}_{k\xi}\boldsymbol{P}_{k+1|k}\boldsymbol{O}_{k\xi}{}^T + \boldsymbol{O}_{kw}\boldsymbol{W}_{k+1}\boldsymbol{O}_{kw}{}^T$$

- The Kalman gain matrix $\boldsymbol{G}$ multiplies innovation from the landmark observation, a 2-vector, so as to update **every element of the state vector**: the pose of the vehicle and the position of every map feature.

SLAM PERFORMANCE IN SIMULATION



COVARIANCE VS. TIME