RMP- Extended Kalman Filter

Key words	Data association, EKF, EKF with SLAM, Jacobian, Mahalanobis distance, incremental/local localization, indirect kalman filter, non-linear, sub-optimal
Status	note complete

Extended Kalman Filter: non-linear models, sub-optimal

- Ideas & Assumptions:
  - **non-linear model:** motion model or/and measurement model can be non-linear
  - zero-mean Gaussian noise

  —> **linearization !!**

  —> sub-optimal

- Prediction
  1. prediction $\hat{x}_{k+1}^{-}$ based on **non-linear motion model,** with $w_k = 0$
  2. project the covariance P_k forward to P_{k+1}^{-}, here A and W are **Jacobians to state and to error vectors**

  —> **a priori estimate** $(\hat{x}_{k+1}^{-}, P_{k+1}^{-})$

- Correction
  1. compute Kalman Gain K_{k+1}, here H and V are **Jacobians to state and to sensor errors**
  2. update prediction $\hat{x}_{k+1}^{-}$ with measurements z_{k+1} to $\hat{x}_{k+1}$
  3. update error covariance to P_{k+1}

  —> **a posteriori estimate** ($\hat{x}_{k+1}, P_{k+1}$)

- **non-linear** motion and measurement model:
  - motion model:

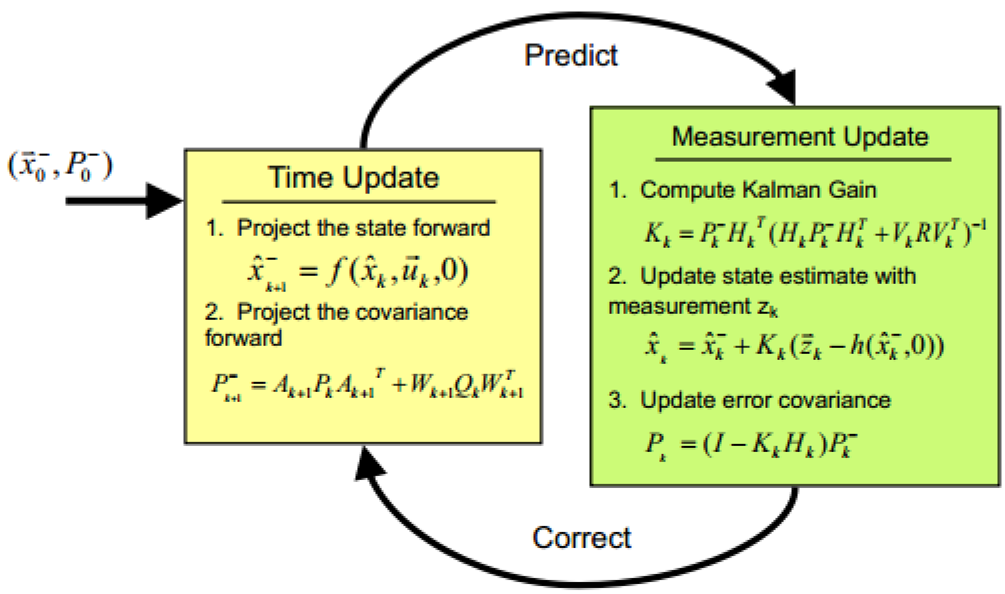
$$x_{k+1} = f(x_k, u_k, 0)$$

$$\text{with } A = \frac{\partial f}{\partial x_k}, W = \frac{\partial f}{\partial w_k}$$

- measurement model:

$$z_{k+1} = h(x_{k+1}^{-}, 0)$$

$$\text{with } H = \frac{\partial h}{\partial x_{k+1}^{-}}, V = \frac{\partial h}{\partial v_{k+1}}$$



Indirect Kalman Filter: estimation of state error

- estimate error between true value and current estimated value —> error propagation

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{bmatrix}_{k+1} = \begin{bmatrix} x_{k+1} - \hat{x}_{k+1} \\ y_{k+1} - \hat{y}_{k+1} \\ \phi_{k+1} - \hat{\phi}_{k+1} \end{bmatrix}$$

Extended Kalman Filter for SLAM:

- SLAM: Simultaneous Localizaiton and Mapping
- idea: look at the distribution of errors —> find out discrepancies between landmarks
    - discrepancy due to state error of robot (**constant offset)**
    - discrepancy due to measurement noise **(random)**
- state vector: **robot position + landmark positions**

$$x = [x_R^T, x_{L1}^T, \dots, x_{Ln}^T]^T$$

- state covariance matrix

$$P = \begin{bmatrix} P_{R,R} & P_{R,L1} & \dots & P_{R,Ln} \\ P_{R,L1} & P_{L1,L1} & \dots & P_{L1,Ln} \\ \vdots & \vdots & \ddots & \vdots \\ P_{Ln,R} & P_{Ln,l1} & \dots & P_{Ln,Ln} \end{bmatrix}$$

- motion model:
    - robot position
    - landmark: **constant**
- measurement vector:
    - **landmark visible** and measured: $H_{Li} \neq 0$
    - **landmark invisible** and not measured: $H_{Li} = 0$

  —> **data association and matching necessary!!!**

$$H = [H_R, 0, \dots, 0, H_{Li}, 0, \dots, 0]^T$$

▼ **Data association:** assign measurements to the predictions

- **Mahalanobis distance:** distance with **uncertainty considered**

  If D_m < threshold: **data associatio**n!! —> **update H matrix** to the corresponding position.

$$D_m(x) = \sqrt{(x-\mu)^T \Sigma^{-1} (x-\mu)}$$

- similarity measurement between x and y

$$D_m(x,y) = \sqrt{(x-y)^T\Sigma^{-1}(x-y)}$$

- associate **i-th measurement** z_i with **j-th estimated state vector** $x^-_{k+1,j}$

$$\Sigma = H_i(x^-_{k+1,j}) \cdot P^-_{k+1} \cdot H_i(x^-_{k+1,j})^T + R_{i,k+1}$$

$$D_m(z_{k+1,i}, x^-_{k+1,j}) = \sqrt{(z_i - h_i(x^-_{k+1,j}))^T\Sigma^{-1}(z_i - h_i(x^-_{k+1,j}))}$$