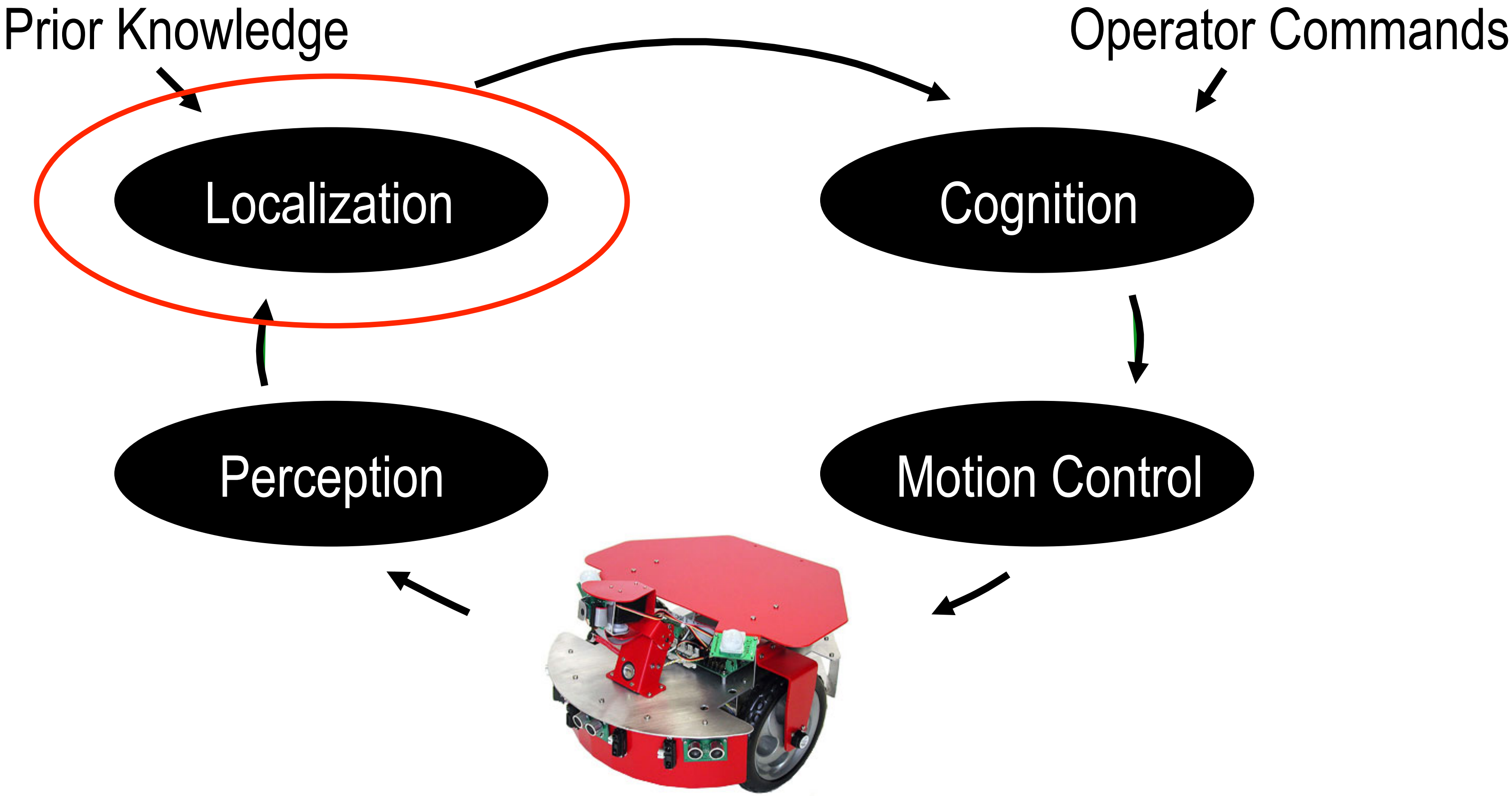
COS 495 - Lecture 17
Autonomous Robot Navigation

Instructor: Chris Clark

Semester: Fall 2011

Figures courtesy of Siegwart & Nourbakhsh

Control Structure



Extended Kalman Filter Localization

1. EKF Localization Overview
2. EKF Prediction
3. EKF Correction
4. Algorithm Summary

Extended Kalman Filter Localization

- Robot State Representation
  - State vector to be estimated, $\boldsymbol{x}$

    e.g.

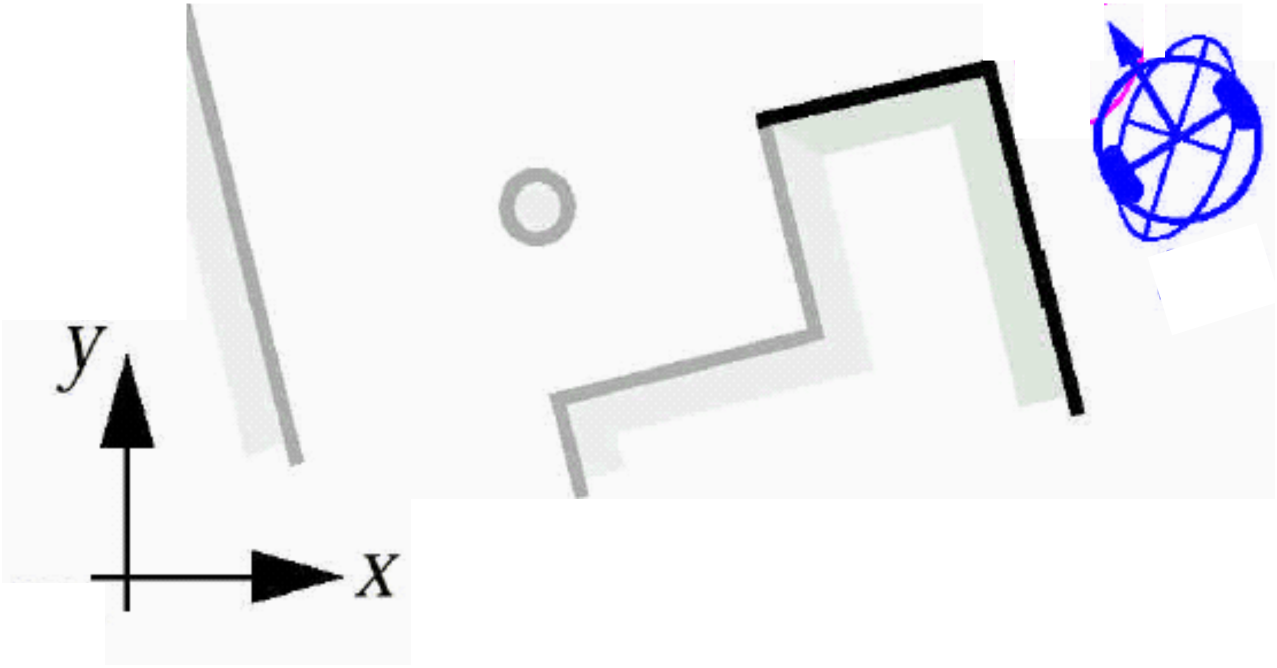
$$\boldsymbol{x} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$$

  - Associated Covariance, $\boldsymbol{P}$

$$\boldsymbol{P} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{x\theta} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{y\theta} \\ \sigma_{\theta x} & \sigma_{\theta y} & \sigma_{\theta\theta} \end{bmatrix}$$

Extended Kalman Filter Localization

1. Robot State Representation



Extended Kalman Filter Localization

- Iterative algorithm
  1. Prediction – Use a motion model and odometry to predict the state of the robot and its covariance

     $$x'_t \quad P'_t$$

  2. Correction - Use a sensor model and measurement to predict the state of the robot and its covariance

     $$x_t \quad P_t$$

Extended Kalman Filter Localization

1. EKF Localization Overview
2. EKF Prediction
3. EKF Correction
4. Algorithm Summary

EKFL Prediction Step

- Motion Model
  - Lets use a general form of a motion model as a discrete time equation that predicts the current state of the robot given the previous state x_{t-1} and the odemetry u_t

$$x'_t = f(x_{t-1}, u_t)$$

EKFL Prediction Step

- Motion model
  - For our differential drive robot…

$$\boldsymbol{x_{t-1}} = \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ \theta_{t-1} \end{bmatrix}$$

$$\boldsymbol{u_t} = \begin{bmatrix} \Delta s_{r,t} \\ \Delta s_{l,t} \end{bmatrix}$$

EKFL Prediction Step

- Motion model
  - And the model we derived…

$$x'_t = f(x_{t-1}, u_t) = \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ \theta_{t-1} \end{bmatrix} + \begin{bmatrix} \Delta s_t \cos(\theta_{t-1} + \Delta\theta_t/2) \\ \Delta s_t \sin(\theta_{t-1} + \Delta\theta_t/2) \\ \Delta\theta_t \end{bmatrix}$$

$$\Delta s_t = (\Delta s_{r,t} + \Delta s_{l,t})/2$$

$$\Delta\theta_t = (\Delta s_{r,t} - \Delta s_{l,t})/b$$

EKFL Prediction Step

- Covariance
  - Recall, the propagation of error equation…

$$\left.\begin{array}{r} X \sim N(\mu,\Sigma) \\ Y = AX + B \end{array}\right\} \Rightarrow \quad Y \sim N(A\mu + B, A\Sigma A^T)$$

EKFL Prediction Step

- Covariance
  - Our equation *f()* is not linear, so to use the property we will linearize with first order approximation

$$x'_t = f(x_{t-1}, u_t)$$
$$\approx F_{x,t} x_{t-1} + F_{u,t} u_t$$

where

$F_{x,t}$ = *Derivative of f with respect to state* x_{t-1}

$F_{u,t}$ = *Derivative of f with respect to control*

EKFL Prediction Step

- Covariance
  - Here, we linearize the motion model f to obtain

$$P'_t = F_{x,t} P_{t-1} F_{x,t}^T + F_{u,t} Q_t F_{u,t}^T$$

where

Q_t = *Motion Error Covariance Matrix*

$F_{x,t}$ = *Derivative of f with respect to state* $\mathbf{x}_{t-1}$

$F_{u,t}$ = *Derivative of f with respect to control* $\mathbf{u}_t$

EKFL Prediction Step

- Covariance

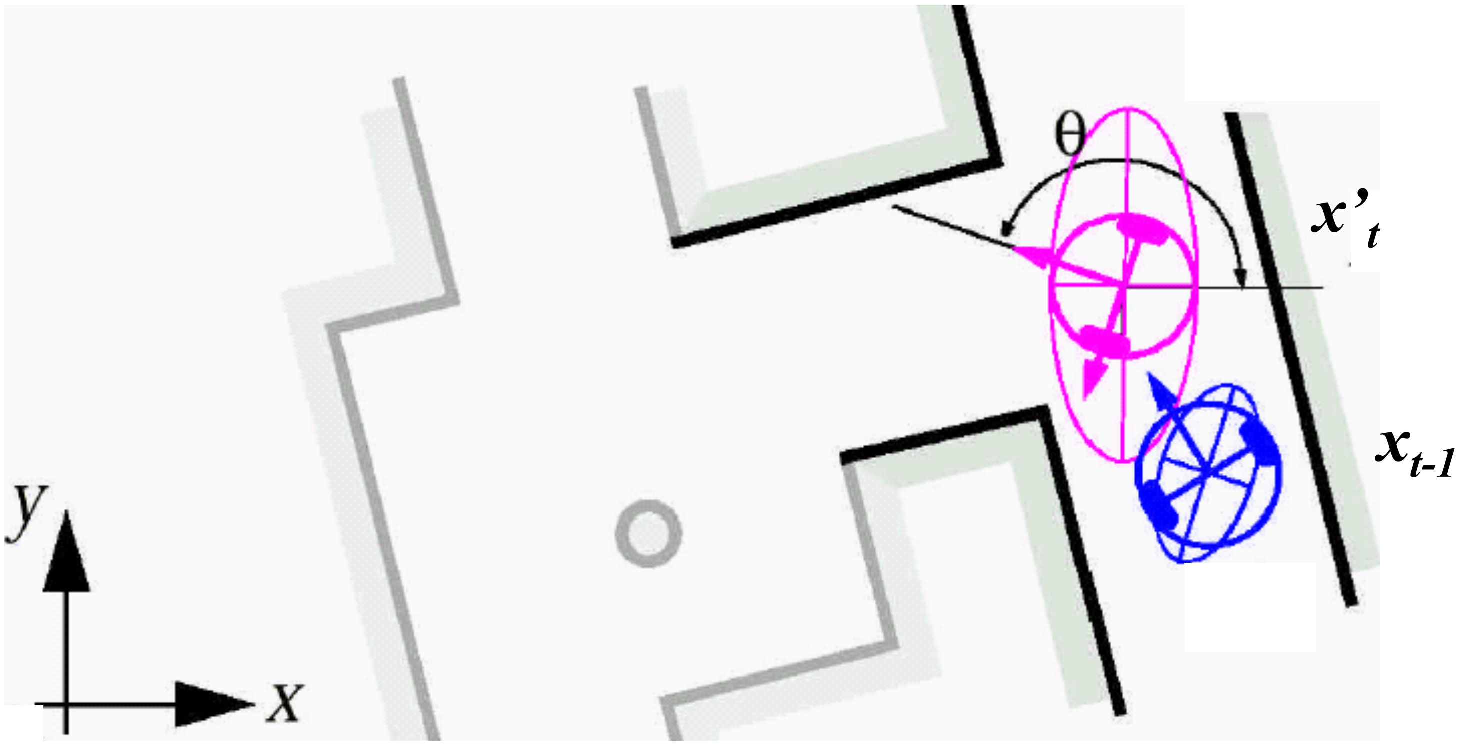
$$Q_t = \begin{bmatrix} k\,|\Delta s_{r,t}| & 0 \\ 0 & k\,|\Delta s_{l,t}| \end{bmatrix}$$

$$F_{x,t} = \begin{bmatrix} df/dx_t & df/dy_t & df/d\theta_t \end{bmatrix}$$

$$F_{u,t} = \begin{bmatrix} df/d\Delta s_{r,t} & df/d\Delta s_{l,t} \end{bmatrix}$$

EKFL Prediction Step

1. Motion Model



Extended Kalman Filter Localization

1. EKF Localization Overview
2. EKF Prediction
3. EKF Correction
4. Algorithm Summary

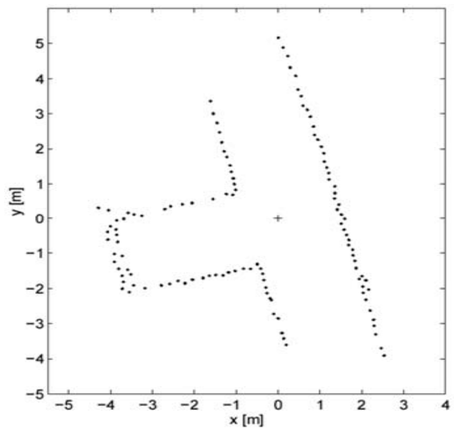
EKFL Correction Step

- Innovation
  - We **correct** by comparing current measurements z_t with what we expect to observe $z_{exp,t}$ given our predicted location in the map M.
  - The amount we correct our state is proportional to the **innovation** v_t

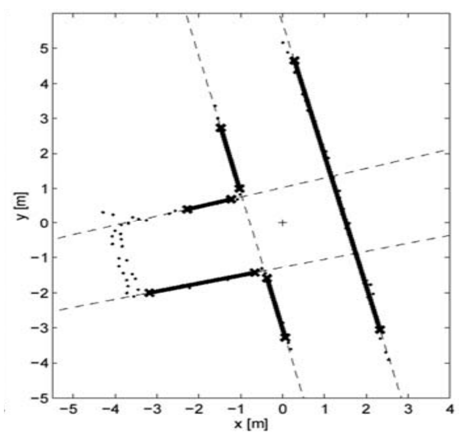
$$v_t = z_t - z_{exp,t}$$

EKFL Correction Step

- The Measurement
  - Assume our robot measures the relative location of a wall i extracted as line



$$\boldsymbol{z}^i_t = \begin{bmatrix} \alpha^i_t \\ r^i_t \end{bmatrix} \qquad \boldsymbol{R}^i_t = \begin{bmatrix} \sigma^i_{\alpha\alpha,t} & \sigma^i_{\alpha r,t} \\ \sigma^i_{r\alpha,t} & \sigma^i_{rr,t} \end{bmatrix}$$



EKFL Correction Step

- The Measurement
  - Assume our robot measures the relative location of a wall i extracted as line

$$z^i_t = \begin{bmatrix} \alpha^i_t \\ r^i_t \end{bmatrix} = g(\rho_1, \rho_2, \ldots, \rho_n, \beta_1, \beta_2, \ldots, \beta_n)$$

$$\alpha = \frac{1}{2}\operatorname{atan}\left(\frac{\sum w_i \rho_i^2 \sin 2\beta_i - \frac{2}{\sum w_i} \sum\sum w_i w_j \rho_i \rho_j \cos\beta_i \sin\beta_j}{\sum w_i \rho_i^2 \cos 2\beta_i - \frac{1}{\sum w_i} \sum\sum w_i w_j \rho_i \rho_j \cos(\beta_i + \beta_j)} \right)$$

$$r = \frac{\sum w_i \rho_i \cos(\beta_i - \alpha)}{\sum w_i}$$

EKFL Correction Step

- The Measurement

$$R^i_t = \begin{bmatrix} \sigma^i_{\alpha\alpha,t} & \sigma^i_{\alpha r,t} \\ \sigma^i_{r\alpha,t} & \sigma^i_{rr,t} \end{bmatrix}$$

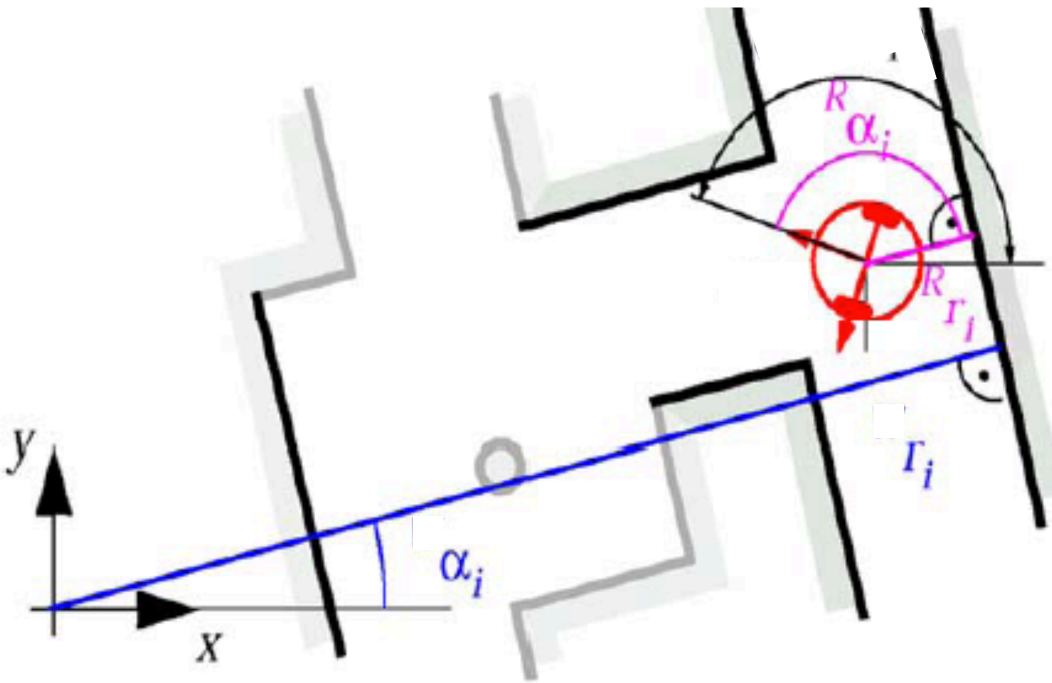
$$= G_{\rho\beta,t}\, \Sigma_{z,t}\, G_{\rho\beta,t}^{\;T}$$

where

$\Sigma_{z,t}$ = *Sensor Error Covariance Matrix*

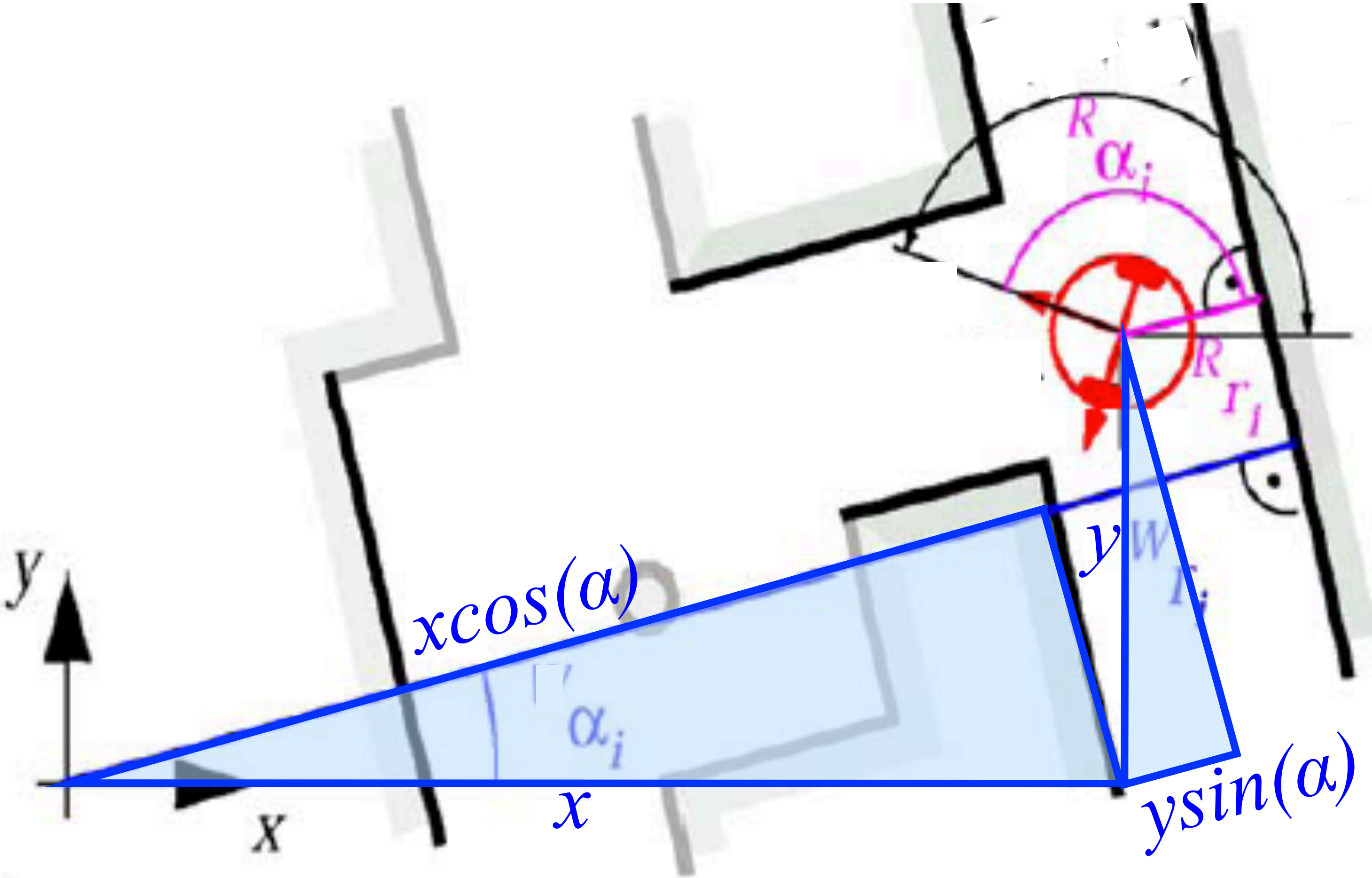
$G_{\rho\beta,t}$ = *Derivative of g() wrt measurements* ρ_t, β_t

EKFL Correction Step



$$\mathbf{z}^{i}_{exp,t} = h^{i}(\mathbf{x'}_{t}, \mathbf{M}) = \begin{bmatrix} \alpha^{i}_{M} - \theta'_{t} \\ r^{i} - x'_{t}\cos(\alpha^{i}_{M}) - y'_{t}\sin(\alpha^{i}_{M}) \end{bmatrix}$$

EKFL Correction Step



EKFL Correction Step

- The covariance associate with the innovation is

$$\Sigma_{IN,t} = H^{i}_{x,t} P'_{t} H^{i\,T}_{x,t} + R^{i}_{t}$$

where

R^{i}_{t} = *Line Measurement Error Covariance Matrix*

$H^{i}_{x,t}$ = *Derivative of h with respect to state* $\mathbf{x}_t$

EKFL Correction Step

- Final updates
  - Update the state estimate

$$x_t = x'_t + K_t\, v_t$$

  - Update the associated covariance matrix

$$P_t = P'_t - K_t\, \Sigma_{IN,t} K_t^T$$

  - Both use the Kalman gain Matrix

$$K_t = P'_t\, H_{x',t}^T\, (\Sigma_{IN,t})^{-1}$$

EKFL Correction Step

- Compare with single var. KF
  - Update the state estimate

$$\hat{x}_t = \hat{x}_{t-1} + K_t (z_t - \hat{x}_{t-1})$$

  - Update the associated covariance matrix

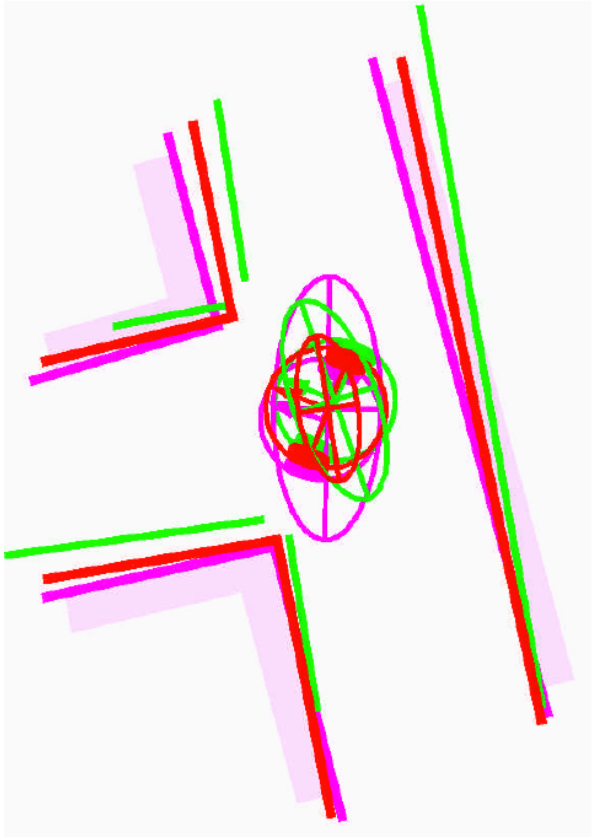
$$\sigma_t^2 = \sigma_{t-1}^2 - K_t \sigma_{t-1}^2$$

Both use the Kalman gain Matrix

$$K_t = \frac{\sigma_{t-1}^2}{\sigma_{t-1}^2 + \sigma_t^2}$$

EKFL Correction Step

- Final updates
  - By fusing the prediction of robot position (magenta) with the innovation gained by the measurements (green) we get the updated estimate of the robot position (red)



Extended Kalman Filter Localization

1. EKF Localization Overview
2. EKF Prediction
3. EKF Correction
4. Algorithm Summary

EKFL Summary

Prediction

1. $x'_t = f(x_{t-1}, u_t)$
2. $P'_t = F_{x,t} P_{t-1} F_{x,t}^T + F_{u,t} Q_t F_{u,t}^T$

Correction

3. $z^i_{exp,t} = h^i(x'_t, M)$
4. $v_t = z_t - z_{exp,t}$
5. $\Sigma_{IN,t} = H^i_{x',t} P'_t H^i_{x',t}{}^T + R^i_t$
6. $x_t = x'_t + K_t v_t$
7. $P_t = P'_t - K_t \Sigma_{IN,t} K_t^T$
8. $K_t = P'_t H_{x',t}^T (\Sigma_{IN,t})^{-1}$