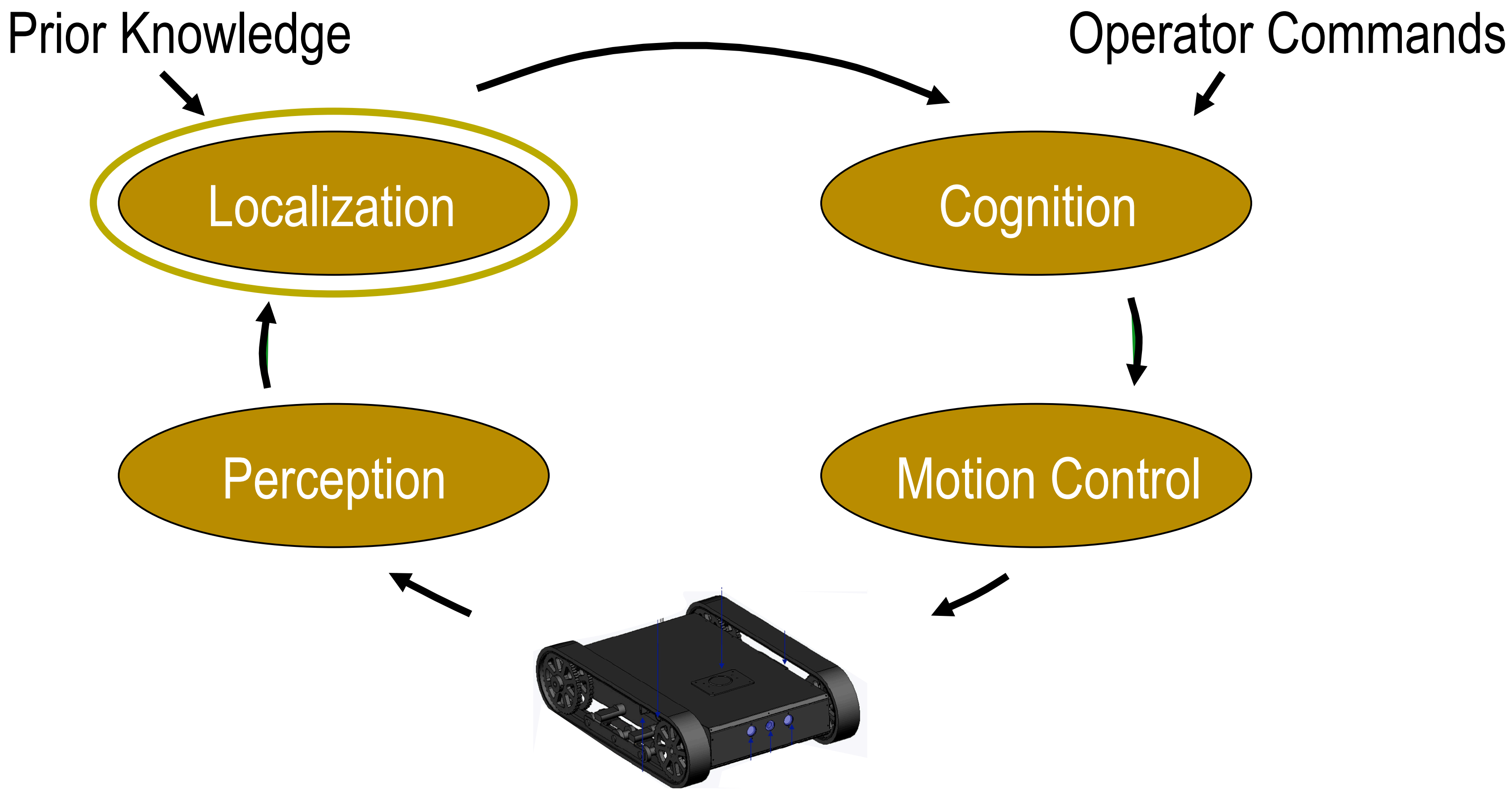
E190Q – Lecture 11
Autonomous Robot Navigation

Instructor: Chris Clark

Semester: Spring 2013

Figures courtesy of Siegwart & Nourbakhsh

Control Structures
Planning Based Control



Kalman Filter Localization

- Introduction to Kalman Filters
  1. KF Representations
  2. Two Measurement Sensor Fusion
  3. Single Variable Kalman Filtering
  4. Multi-Variable KF Representations
- Kalman Filter Localization

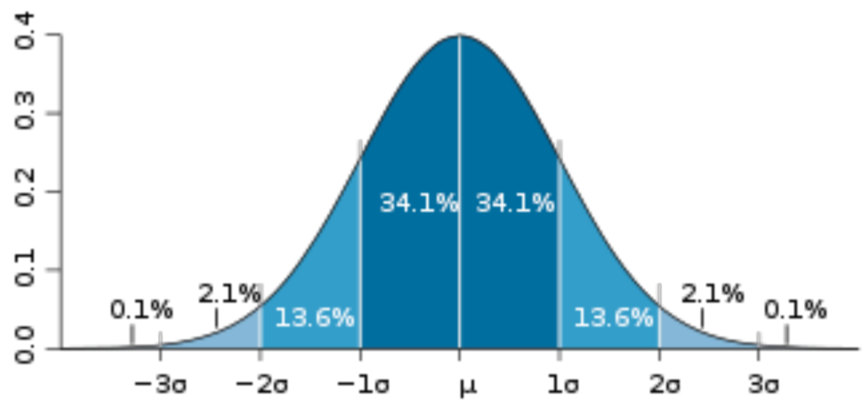
KF Representations

- What do Kalman Filters use to represent the states being estimated?

Gaussian Distributions!

KF Representations

- Single variable Gaussian Distribution
  - Symmetrical
  - Uni-modal
  - Characterized by
    - Mean μ
    - Variance σ^2
  - Properties
    - Propagation of errors
    - Product of Gaussians



KF Representations

- Single Var. Gaussian Characterization
  - Mean
    - Expected value of a random variable with a continuous Probability Density Function $p(x)$

$$\mu = \mathrm{E}[X] = \int x\, p(x)\, dx$$

    - For a discrete set of K samples

$$\mu = \sum_{k=1}^{K} x_k / K$$

KF Representations

- Single Var. Gaussian Characterization
  - Variance
    - Expected value of the difference from the mean squared

$$\sigma^2 = \mathrm{E}[(X-\mu)^2] = \int (x - \mu)^2 p(x)\, dx$$

    - For a discrete set of K samples

$$\sigma^2 = \sum_{k=1}^{K} (x_k - \mu)^2 / K$$

KF Representations

- Single variable Gaussian Properties
  - Propagation of Errors

$$\left.\begin{array}{l} X \sim N(\mu, \sigma^2) \\ Y = aX + b \end{array}\right\} \Rightarrow \quad Y \sim N(a\mu + b, a^2\sigma^2)$$

KF Representations

- Single variable Gaussian Properties
  - Product of Gaussians

$$\left.\begin{matrix} X_1 \sim N(\mu_1, \sigma_1^{\,2}) \\ X_2 \sim N(\mu_2, \sigma_2^{\,2}) \end{matrix}\right\} \Rightarrow$$

$$p(X_1) \cdot p(X_2) \sim N\left(\frac{\sigma_2^{\,2}}{\sigma_1^{\,2} + \sigma_2^{\,2}} \mu_1 + \frac{\sigma_1^{\,2}}{\sigma_1^{\,2} + \sigma_2^{\,2}} \mu_2, \quad \frac{1}{\sigma_1^{\,-2} + \sigma_2^{\,-2}} \right)$$

KF Representations

- Single variable Gaussian Properties…
  - We stay in the “Gaussian world” as long as we start with Gaussians and perform only linear transformations.

Kalman Filter Localization

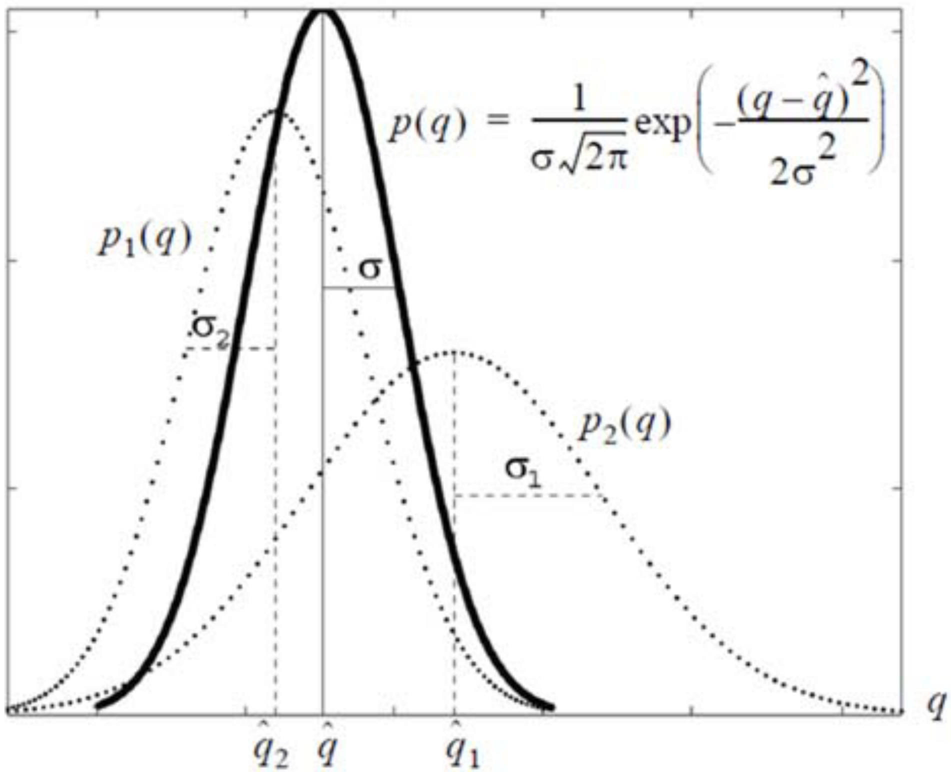
- Introduction to Kalman Filters
  1. KF Representations
  2. Two Measurement Sensor Fusion
  3. Single Variable Kalman Filtering
  4. Multi-Variable KF Representations
- Kalman Filter Localization

Fusing Two Measurements

- Example
  - Given two measurements q_1 and q_2, how do we fuse them to obtain an estimate $\hat{q}$?
  - Assume measurements are modeled as random variables that follow a Gaussian distribution with variance σ_1^2 and σ_2^2 respectively

Fusing Two Measurements

- Example (cont'):



Fusing Two Measurements

- Example (cont'):
  - Lets frame the problem as minimizing a weighted least squares cost function:

$$S = \sum_{i=1}^{n} w_i(\hat{q} - q_i)^2$$

$$\frac{\partial S}{\partial \hat{q}} = \frac{\partial}{\partial \hat{q}} \sum_{i=1}^{n} w_i(\hat{q} - q_i)^2 = 2\sum_{i=1}^{n} w_i(\hat{q} - q_i) = 0$$

Fusing Two Measurements

- Example (cont'):
  - If $n=2$ and $w_i = 1/\sigma_i^2$

$$\hat{q} = q_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}(q_2 - q_1)$$

Kalman Filter Localization

- Introduction to Kalman Filters
  1. KF Representations
  2. Two Measurement Sensor Fusion
  3. Single Variable Kalman Filtering
  4. Multi-Variable KF Representations
- Kalman Filter Localization

Single Variable KF

- Example: Fusing two Measurements

$$\hat{q} = q_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}(q_2 - q_1)$$

  - We can reformulate this in KF notation

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

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

Single Variable KF

- KF for a Discrete Time System

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

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

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

  - Where

$\hat{x}_t$ is the current state estimate

σ_t^2 is the associated variance

z_t^2 is the most recent measurement

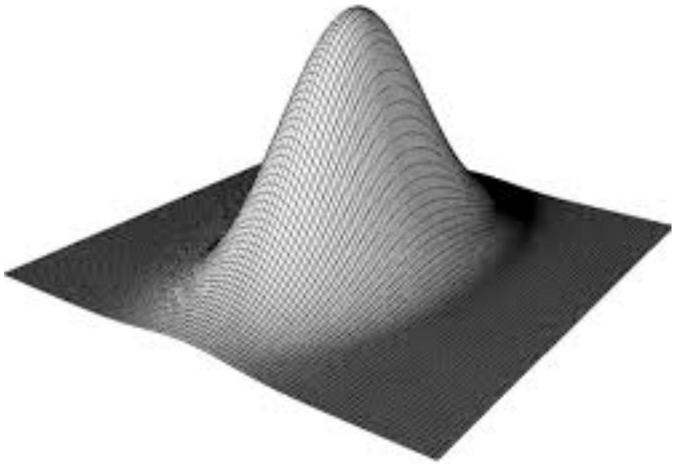
K is the Kalman Gain

Kalman Filter Localization

- Introduction to Kalman Filters
  1. KF Representations
  2. Two Measurement Sensor Fusion
  3. Single Variable Kalman Filtering
  4. Multi-Variable KF Representations
- Kalman Filter Localization

Representations in KF

- Multi-variable Gaussian Distribution
  - Symmetrical
  - Uni-modal
  - Characterized by
    - Mean Vector $\boldsymbol{\mu}$
    - Covariance Matrix Σ
  - Properties
    - Propagation of errors
    - Product of Gaussians



Representations in KF

- Multi-Var. Gaussian Characterization
  - Mean Vector
    - Vector of expected values of n random variables

$$\boldsymbol{\mu} = \mathrm{E}[\boldsymbol{X}] = [\, \mu_0 \;\; \mu_1 \;\; \mu_2 \; \ldots \; \mu_n \;\,]^T$$

$$\mu_i = \int x_i \, p(x_i) \; dx_i$$

Representations in KF

- Multi-Var. Gaussian Characterization
  - Covariance
    - Expected value of the difference from the means squared

$$\sigma_{ij} = \mathrm{Cov}[X_i, X_j] = \mathrm{E}[(X_i - \mu_i)\,(X_j - \mu_j)\,]$$

    - **Covariance** is a measure of how much two random variables change together.
    - **Positive** $\boldsymbol{\sigma_{ij}}$ – when variable i is **above** its expected value, then the other variable j tends to also be **above** its μ_j
    - **Negative** $\boldsymbol{\sigma_{ij}}$ – when variable i is **above** its expected value, then the other variable j tends to be **below** its μ_j

Representations in KF

- Multi-Var. Gaussian Characterization
  - Covariance
    - For continuous random variables

$$\sigma_{ij} = \iint (x_i - \mu_i)\,(x_j - \mu_j)\,p(x_i\,,\,x_j)\,dx_i\,dx_j$$

    - For discrete set of K samples

$$\sigma_{ij} = \sum_{k=1}^{K} (x_{i,k} - \mu_i)(x_{j,k} - \mu_j)/K$$

Representations in KF

- Multi-Var. Gaussian Characterization
  - Covariance Matrix
    - Covariance between each pair of random variables

$$\Sigma = \begin{bmatrix} \sigma_{00} & \sigma_{01} & \ldots & \sigma_{0n} \\ \sigma_{10} & \sigma_{11} & \ldots & \sigma_{1n} \\ & \vdots & & \\ \sigma_{n0} & \sigma_{n1} & \ldots & \sigma_{nn} \end{bmatrix}$$

Note: $\sigma_{ii} = \sigma_i^2$

Representations in KF

- Multi variable Gaussian Properties
  - Propagation of Errors

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

Representations in KF

- Multi variable Gaussian Properties
  - Product of Gaussians

$$\left.\begin{array}{l} X_1 \sim N(\mu_1, \Sigma_1) \\ X_2 \sim N(\mu_2, \Sigma_2) \end{array}\right\}$$

$$\Rightarrow p(X_1)\cdot p(X_2) \sim N\left(\frac{\Sigma_2}{\Sigma_1+\Sigma_2}\mu_1 + \frac{\Sigma_1}{\Sigma_1+\Sigma_2}\mu_2,\quad \frac{1}{\Sigma_1^{-1}+\Sigma_2^{-1}}\right)$$

Next…

Apply the Kalman Filter to multiple variables in the form of a KF.

Kalman Filter Localization

- Introduction to Kalman Filters
- 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, $\mathbf{x}$

    e.g.

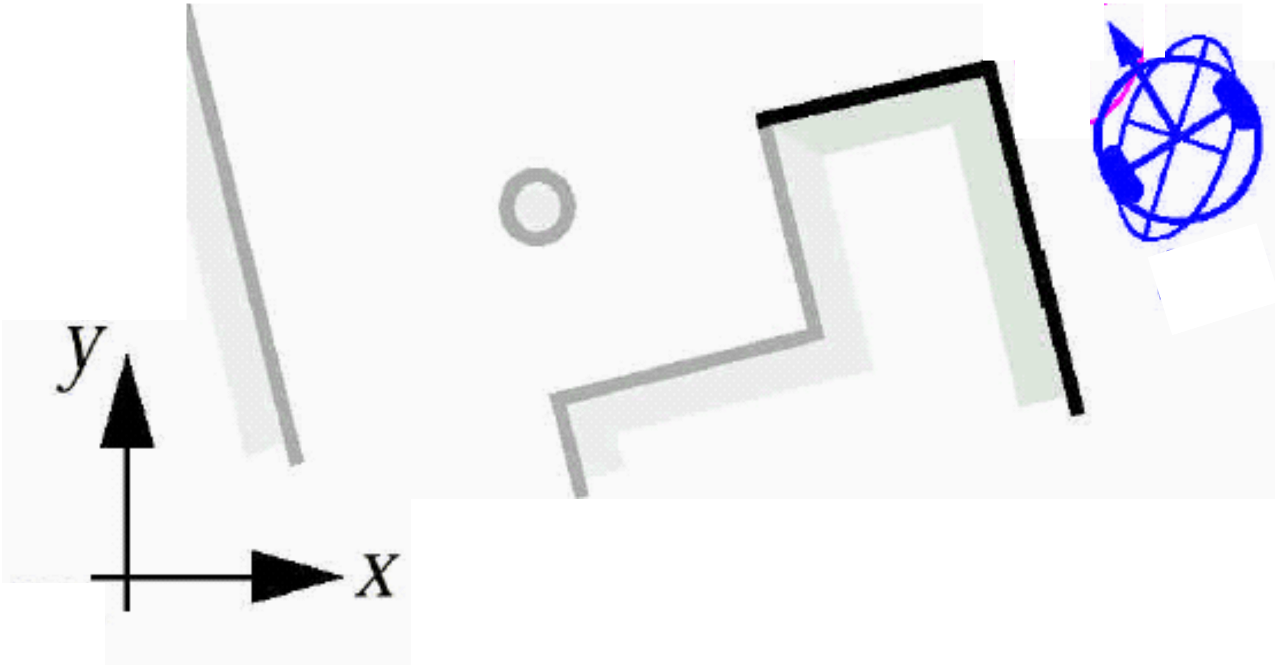
$$\mathbf{x} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$$

  - Associated Covariance, $\mathbf{P}$

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

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

     $$\mathbf{x}'_t \quad \mathbf{P}'_t$$

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

     $$\mathbf{x}_t \quad \mathbf{P}_t$$

Kalman Filter Localization

- Introduction to Kalman Filters
- 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 $\mathbf{x}_{t-1}$ and the odometry $\mathbf{u}_t$

$$\mathbf{x}'_t = f(\mathbf{x}_{t-1}, \mathbf{u}_t)$$

EKFL Prediction Step

- Motion model
  - For our differential drive robot…

$$\mathbf{x}_{t\text{-}1} = \begin{bmatrix} x_{t\text{-}1} \\ y_{t\text{-}1} \\ \theta_{t\text{-}1} \end{bmatrix}$$

$$\mathbf{u}_t = \begin{bmatrix} \Delta s_{r,t} \\ \Delta s_{l,t} \end{bmatrix}$$

EKFL Prediction Step

- Motion model
  - And the model we derived…

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

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

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

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

$$\mathbf{x}'_t = f(\mathbf{x}_{t-1}, \mathbf{u}_t)$$
$$\approx \mathbf{F}_{x,t}\,\mathbf{x}_{t-1} + \mathbf{F}_{u,t}\,\mathbf{u}_t$$

where

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

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

EKFL Prediction Step

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

$$\mathbf{P'}_t = \mathbf{F}_{x,t}\mathbf{P}_{t-1}\mathbf{F}_{x,t}^T + \mathbf{F}_{u,t}\mathbf{Q}_t\mathbf{F}_{u,t}^T$$

where

$\mathbf{Q}_t$ = *Motion Error Covariance Matrix*

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

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

EKFL Prediction Step

- Covariance

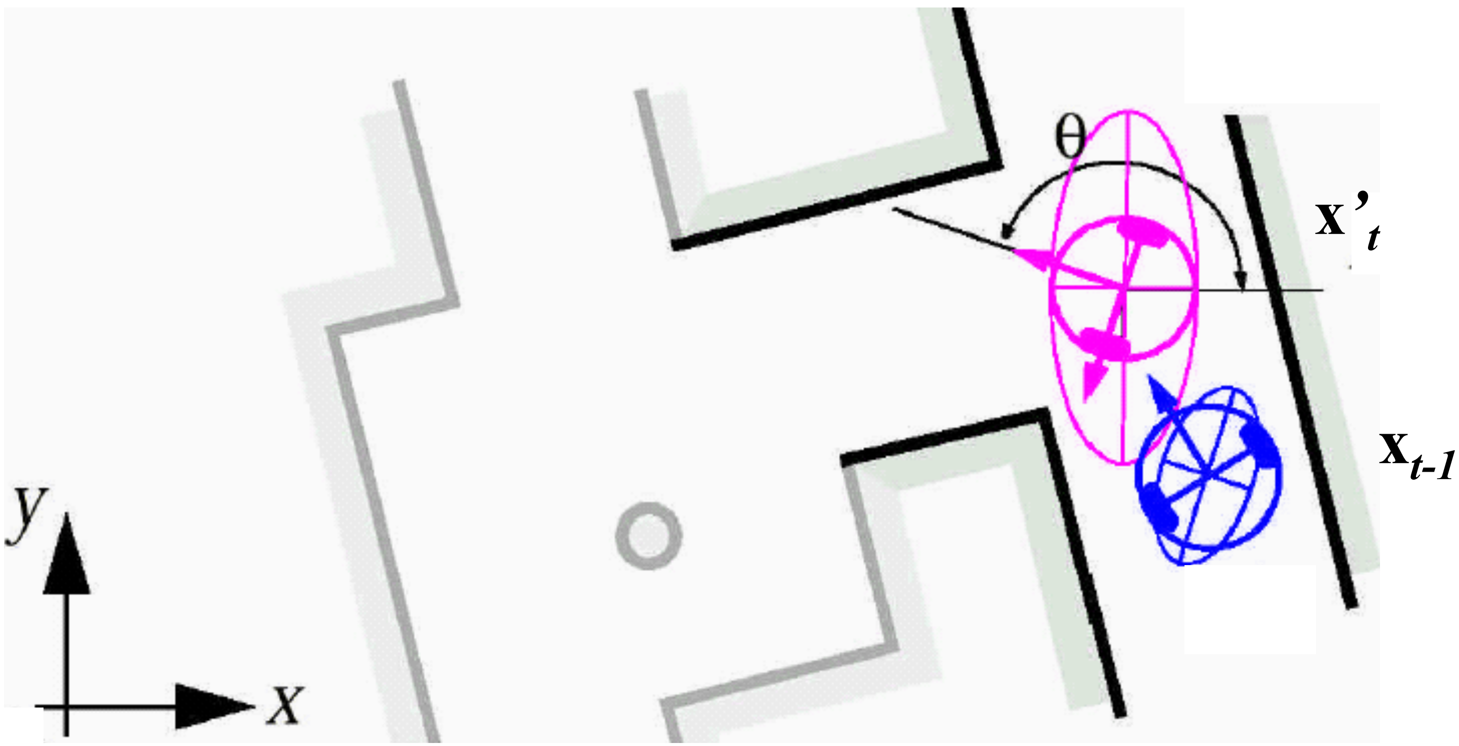
$$\mathbf{Q}_t = \begin{bmatrix} k\,|\Delta s_{r,t}| & 0 \\ 0 & k\,|\Delta s_{l,t}| \end{bmatrix}$$

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

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

EKFL Prediction Step

1. Motion Model



Kalman Filter Localization

- Introduction to Kalman Filters
- Kalman Filter Localization
  1. EKF Localization Overview
  2. EKF Prediction
  3. EKF Correction
  4. Algorithm Summary

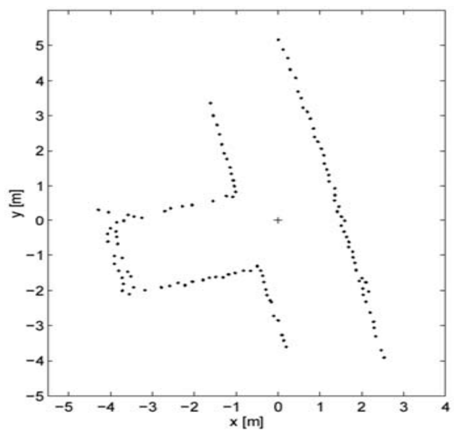
EKFL Correction Step

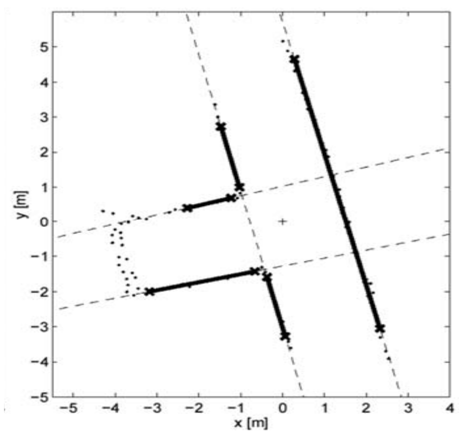
- Innovation
  - We **correct** by comparing current measurements $\mathbf{z}_t$ with what we expect to observe $\mathbf{z}_{exp,t}$ given our predicted location in the map $\mathbf{M}$.
  - The amount we correct our state is proportional to the **innovation** $\mathbf{v}_t$

$$\mathbf{v}_t = \mathbf{z}_t - \mathbf{z}_{exp,t}$$

EKFL Correction Step

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





$$\mathbf{z}^{i}_{t} = \begin{bmatrix} \alpha^{i}_{t} \\ r^{i}_{t} \end{bmatrix} \qquad \mathbf{R}^{i}_{t} = \begin{bmatrix} \sigma^{i}_{\alpha\alpha,t} & \sigma^{i}_{\alpha r,t} \\ \sigma^{i}_{r\alpha,t} & \sigma^{i}_{rr,t} \end{bmatrix}$$

EKFL Correction Step

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

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

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

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

EKFL Correction Step

- The Measurement

$$\mathbf{R}^{i}_{t} = \begin{bmatrix} \sigma^{i}_{\alpha\alpha,t} & \sigma^{i}_{\alpha r,t} \\ \sigma^{i}_{r\alpha,t} & \sigma^{i}_{rr,t} \end{bmatrix}$$

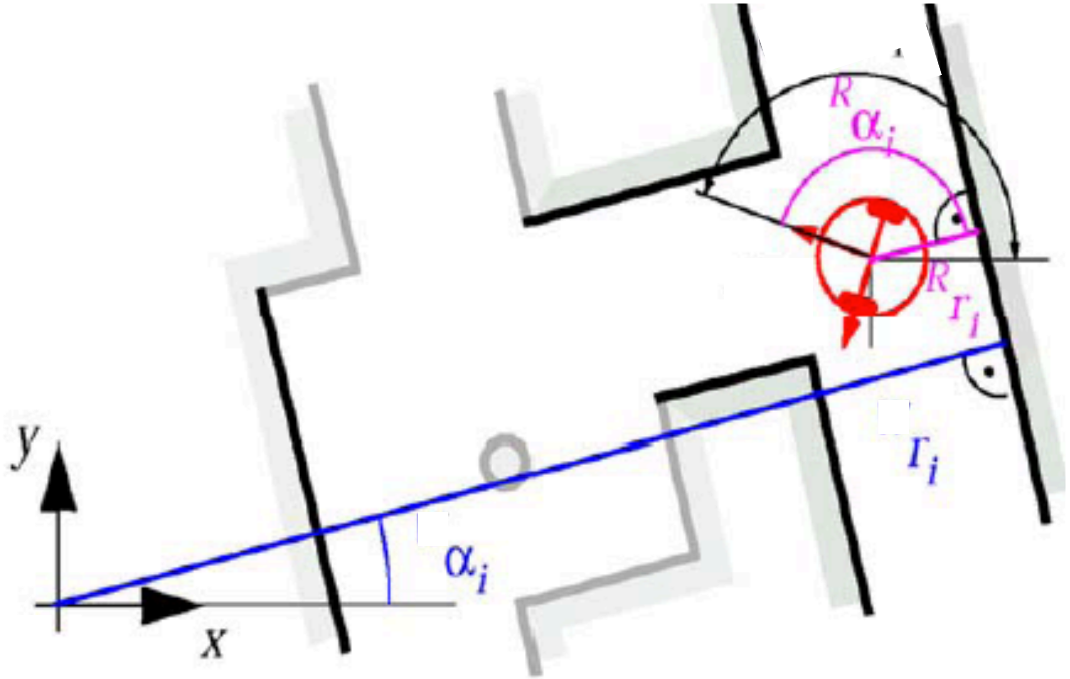
$$= \mathbf{G}_{\rho\beta,t}\, \mathbf{\Sigma}_{z,t}\, \mathbf{G}_{\rho\beta,t}^{\ T}$$

where

$\mathbf{\Sigma}_{z,t}$ = *Sensor Error Covariance Matrix*

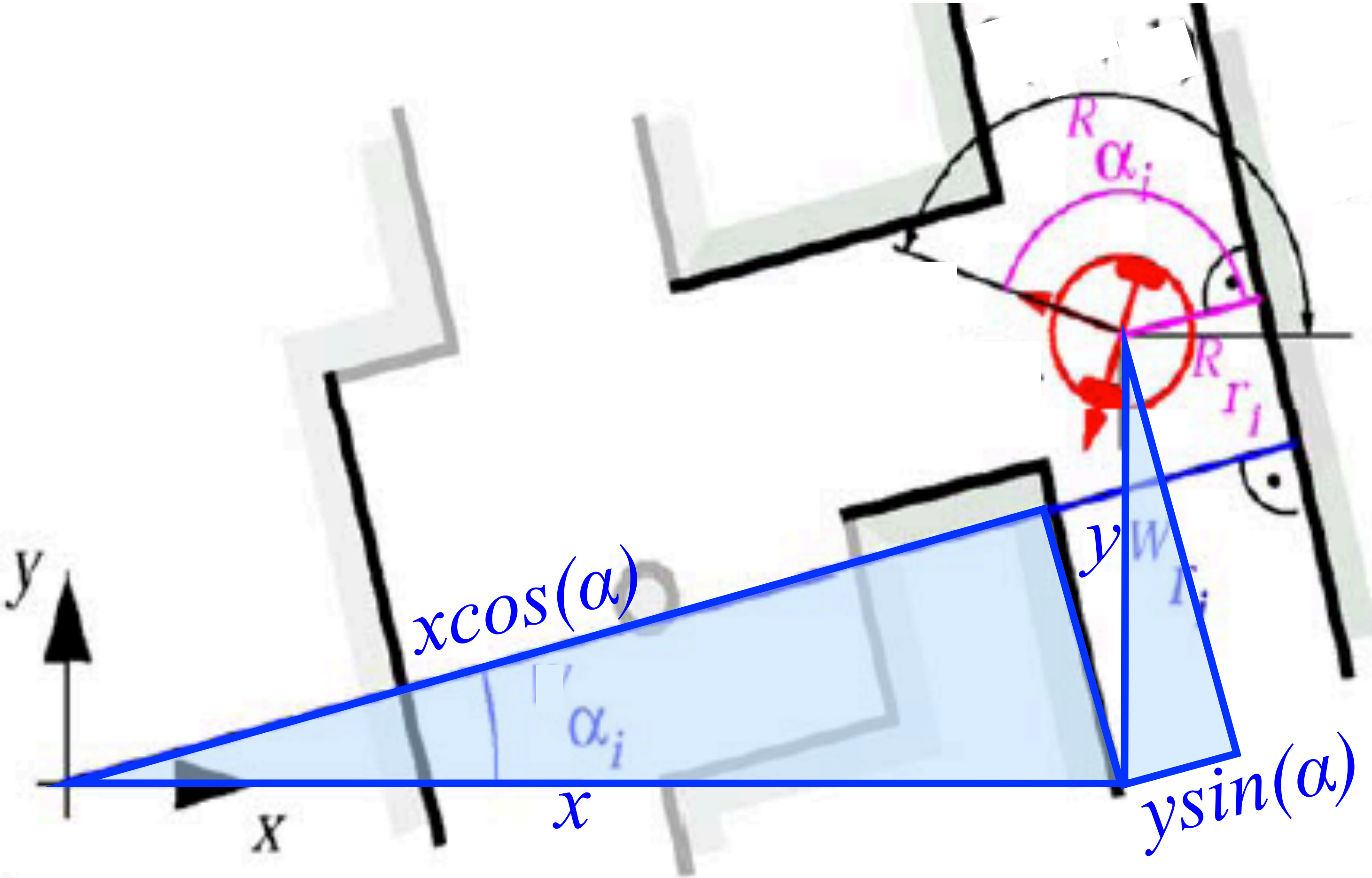
$\mathbf{G}_{\rho\beta,t}$ = *Derivative of g() wrt measurements* $\boldsymbol{\rho}_t$, $\boldsymbol{\beta}_t$

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

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

where

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

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

EKFL Correction Step

- Final updates
  - Update the state estimate

$$\mathbf{x}_t = \mathbf{x'}_t + \mathbf{K}_t\,\mathbf{v}_t$$

  - Update the associated covariance matrix

$$\mathbf{P}_t = \mathbf{P'}_t - \mathbf{K}_t\,\Sigma_{IN,t}\,\mathbf{K}_t^T$$

  - Both use the Kalman gain Matrix

$$\mathbf{K}_t = \mathbf{P'}_t\,\mathbf{H}_{x',t}^T\,(\Sigma_{IN,t})^{-1}$$

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

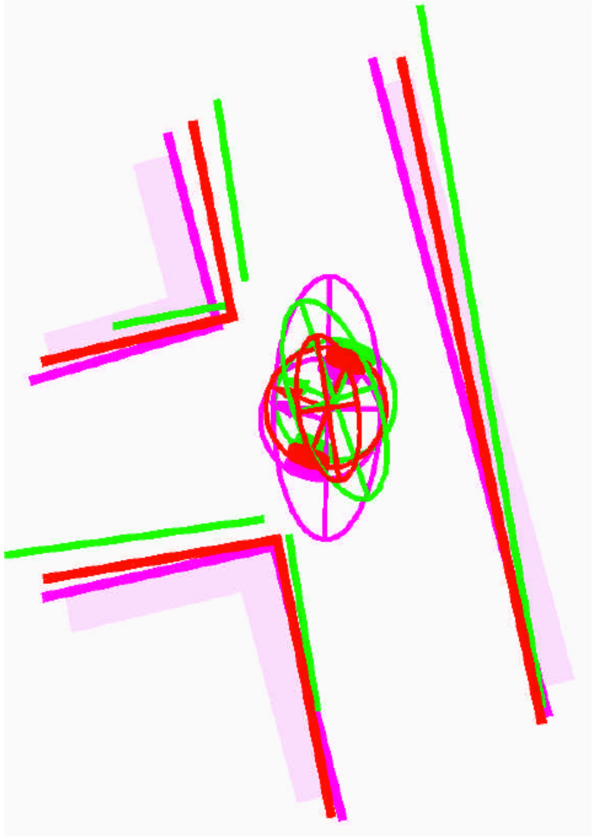
$$\sigma_t^2 = \sigma_{t-1}^2 - K_t \sigma_{t-1}^2$$

Both use the Kalman gain Matrix

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

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)



Kalman Filter Localization

- Introduction to Kalman Filters
- Kalman Filter Localization
  1. EKF Localization Overview
  2. EKF Prediction
  3. EKF Correction
  4. Algorithm Summary

EKFL Summary

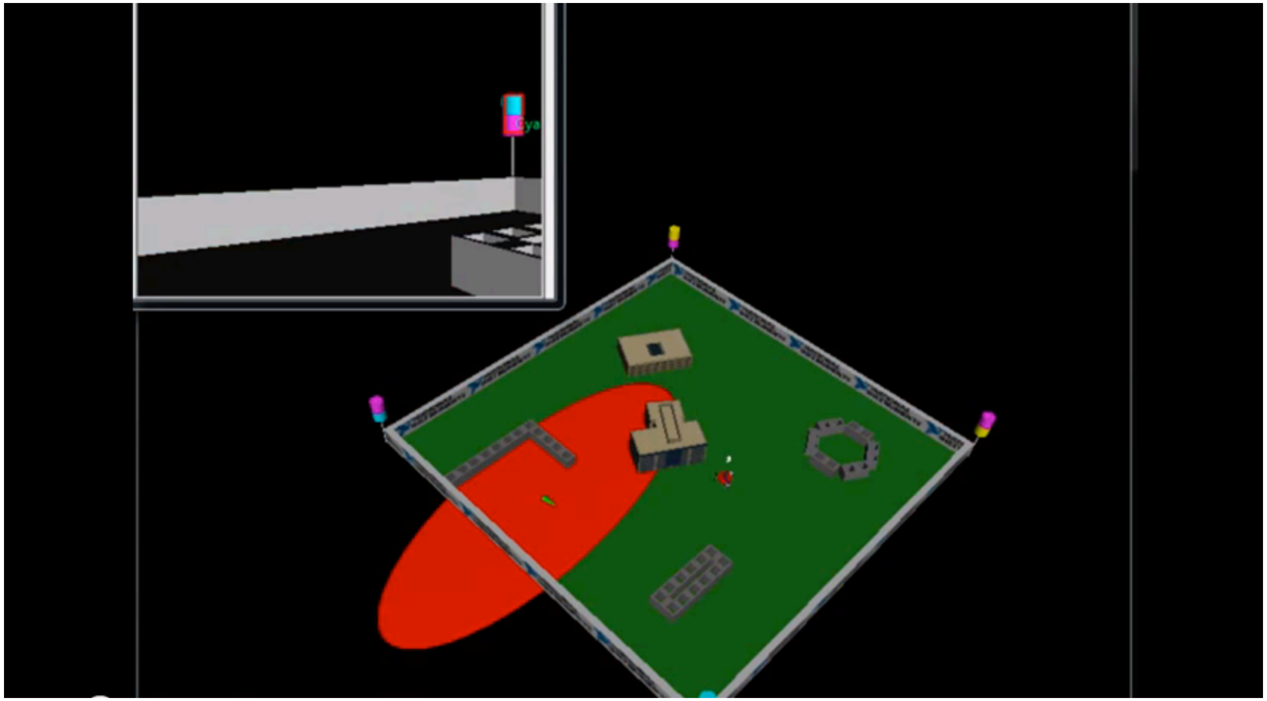
Prediction

1. $\mathbf{x}'_t = f(\mathbf{x}_{t-1}, \mathbf{u}_t)$
2. $\mathbf{P}'_t = \mathbf{F}_{x,t} \mathbf{P}_{t-1} \mathbf{F}_{x,t}^T + \mathbf{F}_{u,t} \mathbf{Q}_t \mathbf{F}_{u,t}^T$

Correction

3. $\mathbf{z}^i_{exp,t} = h^i(\mathbf{x}'_t, \mathbf{M})$
4. $\mathbf{v}_t = \mathbf{z}_t - \mathbf{z}_{exp,t}$
5. $\Sigma_{IN,t} = \mathbf{H}^i_{x',t} \mathbf{P}'_t \mathbf{H}^i_{x',t}{}^T + \mathbf{R}^i_t$
6. $\mathbf{x}_t = \mathbf{x}'_t + \mathbf{K}_t \mathbf{v}_t$
7. $\mathbf{P}_t = \mathbf{P}'_t - \mathbf{K}_t \Sigma_{IN,t} \mathbf{K}_t^T$
8. $\mathbf{K}_t = \mathbf{P}'_t \mathbf{H}_{x',t}^T (\Sigma_{IN,t})^{-1}$

EKFL Example



http://www.youtube.com/watch?v=8mYWutaCaL4