Decision Making for Robots and Autonomous Systems

Fall 2015

DUKE COMPUTER SCIENCE

George Konidaris
gdk@cs.duke.edu

Kalman Filters

Algorithm for using a series of measurements (with noise) to estimate the state of a system over time.

Comes from similar model to POMDP (but no commitment to how actions are chosen).

Very wide use.

Aircraft

Google Maps

OPTUS
3:40 PM
Restaurants
Blue Chillies
St David Street
Brunswick St
King William St
Condell St
Hanover St
Napier St
Search
Directions

Kalman Filters

General scheme:

- System has a true state, x_t.
- There is a control input, u_t.
- Next state is a linear function of x_t and u_t.
- Observation z_t, linear function of x_t.
- Transition and observation effected by zero-mean Gaussian noise.

Would like to track state of the system (x_t).

Why Gaussian noise?

Measurement noise vs. unknown dynamics.

More Formally

System is in state x_t at time t.

Agent inputs control u_{t+1}.

*k*x*k* matrices

Transition:

$$x_{t+1} = \underbrace{Fx_t}_{\text{passive dynamics}} + \underbrace{Bu_{t+1}}_{\text{effect of control}} + \underbrace{\mathcal{N}(0, Q)}_{\text{control noise}}$$

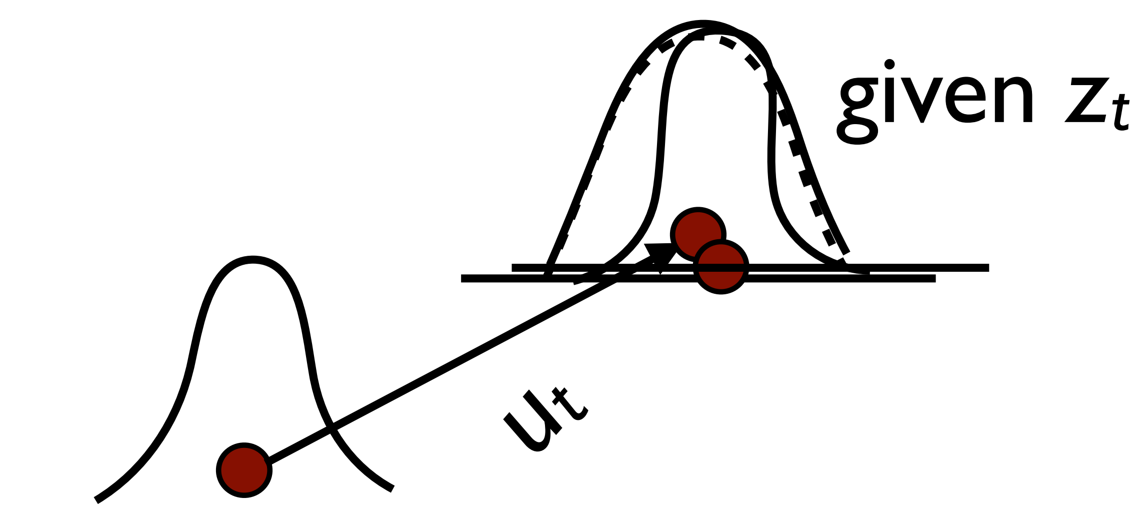
Observation:

$$z_{t+1} = \underbrace{Hx_{t+1}}_{\text{observation function}} + \underbrace{\mathcal{N}(0, R)}_{\text{noise}}$$

Kalman Filter

The algorithm itself:

- Maintain mean and covariance estimates ($\hat{x}_t$ and P_t)
- First update using dynamics
- Then update using observation



Kalman: Predict

Mean is pretty straightforward:

$$\hat{x}^p_{t+1} = F\hat{x}_t + Bu_{t+1}$$

Covariance:

$$P^p_{t+1} = \underbrace{FP_tF^T}_{\text{forward uncertainty}} + \underbrace{Q}_{\text{“new” uncertainty}}$$

Kalman: Update

Update mean and covariance given observation:

$$\hat{x}_{t+1} = \hat{x}^p_{t+1} + K_{t+1} y_{t+1}$$

$$P_{t+1} = (I - K_{t+1} H) P^p_{t+1}$$

where:

$$y_{t+1} = z_{t+1} - H \hat{x}^p_{t+1}$$ observation - expected

$$K_{t+1} = P^p_{t+1} H_{t+1} S^{-1}_{t+1}$$ observation covariance

$$S_{t+1} = H P^p_{t+1} H^T + R$$

Demo

…

Apollo

DUKE COMPUTER SCIENCE

Generalization

The relevant matrices can be functions of time (but not state).

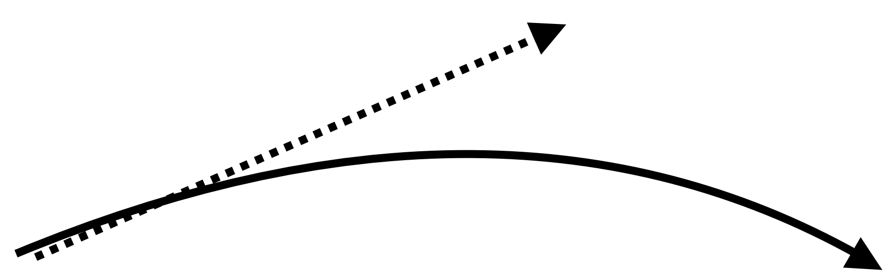
$$x_{t+1} = F(t)x_t + B(t)u_{t+1} + \mathcal{N}(0, Q(t))$$

Generalization

What if forward model is not linear?

Extended Kalman Filter

- Linearize control about current x_t estimate.
- Proceed as if linear!



Generalization

What if the distributions are not Gaussian?

Unscented Kalman Filter

- Pretend they are Gaussian! (sort of)
- Compute moments:
    - Mean
    - Variance
- … and use these as mean and variance of Gaussian estimate.
- "Moment matching"