

Lecture 1: Overview of Bayesian Modeling of Time-Varying Systems

Simo Särkkä

Department of Biomedical Engineering and Computational Science
Aalto University

January 20, 2011

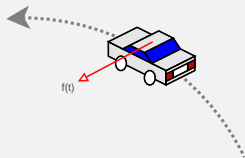
- 1 Overview and History
- 2 Bayesian Estimation of Dynamic Processes
- 3 Optimal Filtering
- 4 Applications
- 5 Probabilistic State Space Models
 - Regression Models
 - Time Series Models
 - Target Tracking Models
 - Further State Space Models
- 6 Summary and Demonstration

- **Modeling** with stochastic state space models.
- **Bayesian theory** of optimal filtering.
- **Gaussian approximations**: Derivation of Kalman, extended Kalman and unscented Kalman filters, Gauss-Hermite and cubature Kalman filters from the general theory.
- **Monte Carlo methods**: Particle filtering, Rao-Blackwellized filtering.
- **Optimal smoothing**: Bayesian theory of optimal smoothing and related Kalman (=Gaussian) and particle type methods.
- Various **illustrative applications** to backup the theory.
- Various **exercises** to practice modeling and estimation.

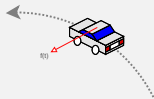
Some History

- In 40's, **Wiener's** work on **stochastic analysis and optimal filtering** (and “cybernetics”)
- In late 50's, state space models, **Bellman's dynamic programming**, **Swerling's filter**.
- In early 60's, **Kalman filter** and **Kalman-Bucy filter**, stability analysis of linear state space models (mostly by Kalman).
- In mid 60's, **Rauch-Tung-Striebel smoother**, **extended Kalman filters** (EKF).
- In late 60's, **Bayesian approach** to optimal filtering, first practical applications (e.g. Apollo program).
- In 70's and 80's, first **particle filters**, square root Kalman filters, new algorithms and applications.
- In 90's, rebirth of **particle filters**, sigma-point and **unscented Kalman filters** (UKF), new applications.
- In 00's, new algorithm variations and applications.

Recursive Estimation of Dynamic Process

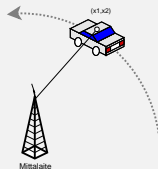


- **Dynamic**, that is, time varying phenomenon - for example the motion state of a car.
- The phenomenon is **measured** - for example by a radar or by acceleration and angular velocity sensors.
- The purpose is to **compute the state of the phenomenon** when only the **measurements are observed**.
- The solution should be **recursive**, where the information in new measurements is used for **updating** the old information.



- The laws of physics, biology, epidemiology etc. are typically differential equations.
- Uncertainties and unknown sub-phenomena are modeled as stochastic processes:
 - Physical phenomena: differential equations + uncertainty \Rightarrow stochastic differential equations.
 - Discretized physical phenomena: Stochastic differential equations \Rightarrow stochastic difference equations.
 - Naturally discrete-time phenomena: Systems jumping from step to another.
- Stochastic differential and difference equations can be represented in stochastic state space form.

Bayesian Modeling of Measurements



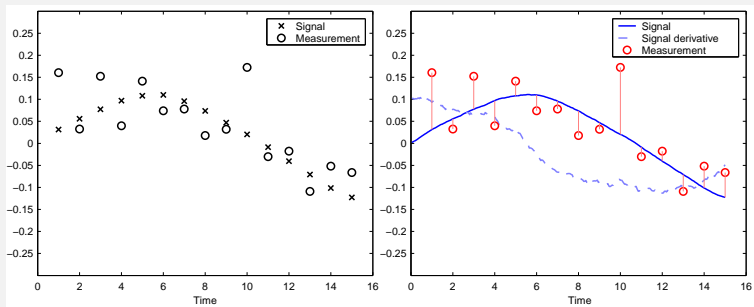
- The relationship between measurements and phenomenon is mathematically modeled as a **probability distribution**.
- The **measurements** could be (in ideal world) computed if the **phenomenon was known** (forward model).
- The **uncertainties** in measurements and model are modeled as random processes.
- The measurement model is the **conditional distribution of measurements** given the state of the phenomenon.

Why Bayesian Approach?

- Theory of optimal filtering can be formulated in many ways:
 - 1 **Least squares** optimization framework \Rightarrow hard to extend recursive estimation beyond linear models, uncertainties cannot be modeled.
 - 2 **Maximum likelihood** framework \Rightarrow the theoretical basis of dynamic models is somewhat heuristic, uncertainties cannot be modeled.
 - 3 **Bayesian** framework \Rightarrow theory is quite complete, but the computational complexity can be unbounded.
 - 4 **Other approaches** \Rightarrow typically applicable to restricted special cases.
- For practical “engineering” reasons, **Bayesian approach** is used here (because it works!).
- **Kalman filter** (1960) was originally derived in **least squares** framework
- **Non-linear filtering theory** has been **Bayesian** from the beginning (about 1964).

Bayesian Estimation of Dynamic Process

Time-varying **process** and noisy **measurements** from it:



- Generally, **Markov model** for the state:

$$\mathbf{x}_k \sim p(\mathbf{x}_k | \mathbf{x}_{k-1}).$$

- **Likelihood distribution** of the measurement:

$$\mathbf{y}_k \sim p(\mathbf{y}_k | \mathbf{x}_k).$$

- In principle, we could simply use the **Bayes' rule**

$$\begin{aligned} p(\mathbf{x}_1, \dots, \mathbf{x}_T | \mathbf{y}_1, \dots, \mathbf{y}_T) \\ = \frac{p(\mathbf{y}_1, \dots, \mathbf{y}_T | \mathbf{x}_1, \dots, \mathbf{x}_T) p(\mathbf{x}_1, \dots, \mathbf{x}_T)}{p(\mathbf{y}_1, \dots, \mathbf{y}_T)}, \end{aligned}$$

- **Curse of computational complexity:** complexity grows more than linearly with number of measurements.



- The classical recursive (efficient) solution to the dynamic estimation problem is called the **optimal filter**.
- The **optimal filter** computes the (marginal) **posterior distribution of the state** given the measurements:

$$p(\mathbf{x}(t_k) | \mathbf{y}_1, \dots, \mathbf{y}_k).$$

- The **“filtered” state** $\hat{\mathbf{x}}(t_k)$ typically is the posterior mean

$$\hat{\mathbf{x}}(t_k) = E(\mathbf{x}(t_k) | \mathbf{y}_1, \dots, \mathbf{y}_k).$$

- The solution is called **filter** for historical reasons.

Optimal Filtering, Prediction and Smoothing

- Recursively computable **marginal distributions**:

- **Filtering distributions**:

$$p(\mathbf{x}_k | \mathbf{y}_1, \dots, \mathbf{y}_k), \quad k = 1, \dots, T.$$

- **Prediction distributions**:

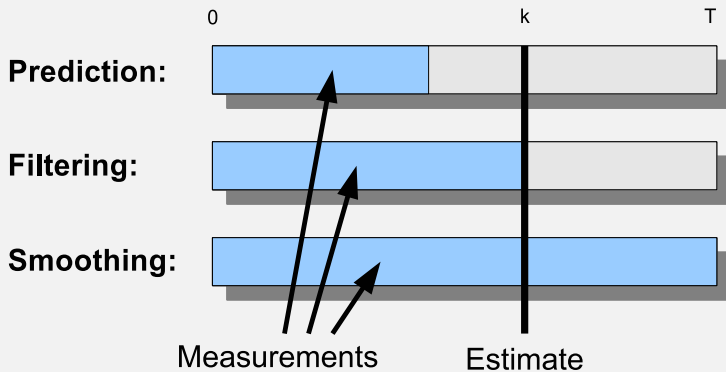
$$p(\mathbf{x}_{k+n} | \mathbf{y}_1, \dots, \mathbf{y}_k), \quad k = 1, \dots, T, \quad n = 1, 2, \dots,$$

- **Smoothing distributions**:

$$p(\mathbf{x}_k | \mathbf{y}_1, \dots, \mathbf{y}_T), \quad k = 1, \dots, T.$$

- The **mean** can be computed exactly, as well as **marginal moments**, but cross-moments need extra effort.

Optimal Filtering, Prediction and Smoothing (cont.)



Algorithms for Computing the Solutions

- **Kalman filter** is the classical optimal filter for linear-Gaussian models.
- **Extended Kalman filter** (EKF) is linearization based extension of Kalman filter to non-linear models.
- **Unscented Kalman filter** (UKF) is sigma-point transformation based extension of Kalman filter.
- **Gauss-Hermite and Cubature Kalman filters** (GHKF/CKF) are numerical integration based extensions of Kalman filter.
- **Particle filter** forms a **Monte Carlo representation** (particle set) to the distribution of the state estimate.
- **Grid based filters** approximate the probability distributions by a finite grid.
- **Mixture Gaussian approximations** are used, for example, in multiple model Kalman filters and Rao-Blackwellized Particle filters.

Navigation of Lunar Module



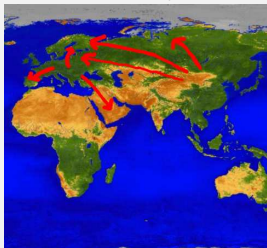
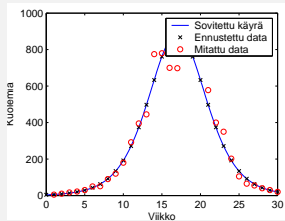
- The **navigation system of Eagle lunar module** AGC was based on an optimal filter - this was **in the year 1969**.
- The dynamic model was Newton's **gravitation law**.
- The measurements at lunar landing were the **radar readings**.
- On rendezvous with the command ship the orientation was computed **with gyroscopes** and their biases were also compensated with the radar.
- The optimal filter was an **extended Kalman filter**.

Satellite Navigation (GPS)

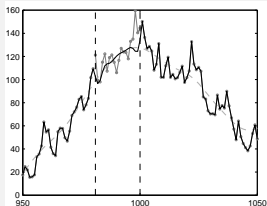
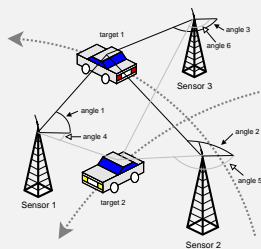


- The dynamic model in GPS receivers is often the Newton's second law where the force is completely random, that is, the **Wiener velocity model**.
- The measurements are **time delays of satellite signals**.
- The optimal filter computes **the position and the accurate time**.
- Also the errors caused by **multi path** can be modeled and compensated.
- **Acceleration and angular velocity measurements** are sometimes used as extra measurements.

Spread of Disease



- Spreading of a disease in population can be modeled by **differential equations**.
- Modeling the unknown parameters and phenomena as random processes leads to **stochastic dynamic model**.
- **The measurements** in this case are the number of **dead or recovered individuals**.
- **Optimal filter** is used for computing the unknown parameters and the number of susceptible and infected individuals.
- It is also possible to **predict** when the **maximum of the epidemic** and how many **casualties there will be**.



- **Target tracking**, where one or many targets are tracked with many passive sensors - **air surveillance**.
- **Time series prediction** where the model parameters are estimated from the measured time series and the unknown part is predicted using these parameters.
- Analysis and restoration of **audio signals**.
- **Telecommunication systems** - optimal receivers, signal detectors.
- **State estimation of control systems** - chemical processes, auto pilots, control systems of cars.

Generic Probabilistic State Space Model

- General form of **probabilistic state space** models:

$$\mathbf{x}_k \sim p(\mathbf{x}_k | \mathbf{x}_{k-1})$$

$$\mathbf{y}_k \sim p(\mathbf{y}_k | \mathbf{x}_k)$$

$$\mathbf{x}_0 \sim p(\mathbf{x}_0).$$

- \mathbf{x}_k is the generalized **state** at time step k , including all physical state variables and parameters.
- \mathbf{y}_k is the **vector of measurements** obtained at time step k .
- The **dynamic model** $p(\mathbf{x}_k | \mathbf{x}_{k-1})$ models the dynamics of the state.
- The **measurement model** $p(\mathbf{y}_k | \mathbf{x}_k)$ models the measurements and their uncertainties.
- The **prior distribution** $p(\mathbf{x}_0)$ models the information known about the state before obtaining any measurements.

Linear Gaussian State Space Models

- General form of **linear Gaussian state space models**:

$$\mathbf{x}_k = \mathbf{A} \mathbf{x}_{k-1} + \mathbf{q}_{k-1}, \quad \mathbf{q}_{k-1} \sim \mathcal{N}(0, \mathbf{Q})$$

$$\mathbf{y}_k = \mathbf{H} \mathbf{x}_k + \mathbf{r}_k, \quad \mathbf{r}_k \sim \mathcal{N}(0, \mathbf{R})$$

$$\mathbf{x}_0 \sim \mathcal{N}(\mathbf{m}_0, \mathbf{P}_0).$$

- In **probabilistic notation** the model is:

$$p(\mathbf{y}_k | \mathbf{x}_k) = \mathcal{N}(\mathbf{y}_k | \mathbf{H} \mathbf{x}_k, \mathbf{R})$$

$$p(\mathbf{x}_k | \mathbf{x}_{k-1}) = \mathcal{N}(\mathbf{x}_k | \mathbf{A} \mathbf{x}_{k-1}, \mathbf{Q}).$$

- **Surprisingly general** class of models – linearity is from measurements to estimates, not from time to outputs.

Non-Linear State Space Models

- General form of **non-linear Gaussian state space models**:

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{q}_k)$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{r}_k).$$

- \mathbf{q}_k and \mathbf{r}_k are typically assumed **Gaussian**.
- Functions $\mathbf{f}(\cdot)$ and $\mathbf{h}(\cdot)$ are **non-linear functions** modeling the dynamics and measurements of the system.
- **Equivalent** to the generic **probabilistic models** of the form

$$\mathbf{x}_k \sim p(\mathbf{x}_k | \mathbf{x}_{k-1})$$

$$\mathbf{y}_k \sim p(\mathbf{y}_k | \mathbf{x}_k).$$

Modeling with State Space Models

- Probabilistic state space models are **very general** – every finite dimensional Bayesian estimation problem has a state space representation.
- The most difficult task is figure out **how to formulate** an estimation problem in state space form.
- Formulating state space representations of physical problems is **engineering** in its basic form.
- Best way to learn this engineering is by examples and practical work – in this lecture we shall give examples.

Linear and Linear in Parameters Models

- Basic **linear regression** model with noise ϵ_k :

$$y_k = a_0 + a_1 x_k + \epsilon_k, \quad k = 1, \dots, N.$$

- First rename x_k to e.g. s_k to avoid confusion:

$$y_k = a_0 + a_1 s_k + \epsilon_k, \quad k = 1, \dots, N.$$

- Define matrix $\mathbf{H}_k = (1 \ s_k)$ and state $\mathbf{x} = (a_0 \ a_1)^T$:

$$y_k = \mathbf{H}_k \mathbf{x} + e_k, \quad k = 1, \dots, N.$$

- For notation sake we can also define $\mathbf{x}_k = \mathbf{x}$ such that $\mathbf{x}_k = \mathbf{x}_{k-1}$:

$$\mathbf{x}_k = \mathbf{x}_{k-1}$$

$$y_k = \mathbf{H}_k \mathbf{x}_k + e_k.$$

- Thus we have a **linear Gaussian state space model**, solvable with the basic **Kalman filter**.

Linear and Linear in Parameters Models (cont.)

- More **general linear regression** models:

$$y_k = a_0 + a_1 s_{k,1} + \cdots + a_d s_{k,d} + \epsilon_k, \quad k = 1, \dots, N.$$

- Defining matrix $\mathbf{H}_k = (1 \ s_{k,1} \ \cdots \ s_{k,d})$ and state $\mathbf{x}_k = \mathbf{x} = (a_0 \ a_1 \ \cdots \ a_d)^T$ gives **linear Gaussian state space model**:

$$\mathbf{x}_k = \mathbf{x}_{k-1}$$

$$y_k = \mathbf{H}_k \mathbf{x}_k + \epsilon_k.$$

- **Linear in parameters models**:

$$y_k = a_0 + a_1 f_1(s_k) + \cdots + a_d f_d(s_k) + \epsilon_k.$$

- Definitions $\mathbf{H}_k = (1 \ f_1(s_k) \ \cdots \ f_d(s_k))$ and $\mathbf{x}_k = \mathbf{x} = (a_0 \ a_1 \ \cdots \ a_d)^T$ again give **linear Gaussian state space model**.

Non-Linear and Neural Network Models

- Non-linearity in measurements models arises in **generalized linear models**, e.g.

$$y_k = g^{-1}(a_0 + a_1 s_k) + \epsilon_k.$$

- The measurement model is now non-linear and if we define $\mathbf{x} = (a_0 \ a_1)^T$ and $h(\mathbf{x}) = g^{-1}(x_1 + x_2 s_k)$ we get **non-linear Gaussian state space model**:

$$\mathbf{x}_k = \mathbf{x}_{k-1}$$

$$y_k = h(\mathbf{x}_k) + \epsilon_k.$$

- Neural network models such as **multi-layer perceptron (MLP)** models can be also transformed into the above form.
- Instead of basic Kalman filter we need **extended Kalman filter** or **unscented Kalman filter** to cope with the non-linearity.

Adaptive Filtering Models

- In digital signal processing, a commonly used signal model is the **autoregressive model**

$$y_k = w_1 y_{k-1} + \dots + w_d y_{k-d} + \epsilon_k,$$

- In **adaptive filtering** the weights w_i are estimated from data.
- If we define matrix $\mathbf{H}_k = (y_{k-1} \dots y_{k-d})$ and state as $\mathbf{x}_k = (w_1 \dots w_d)^T$, we get **linear Gaussian state space model**:

$$\mathbf{x}_k = \mathbf{x}_{k-1}$$

$$y_k = \mathbf{H}_k \mathbf{x}_k + \epsilon_k.$$

- The estimation problem can be solved with **Kalman filter**.
- The **LMS algorithm** can be interpreted as approximate version of this Kalman filter.

Adaptive Filtering Models (cont.)

- In **time varying autoregressive models (TVAR)** models the weights are time-varying:

$$y_k = w_{1,k} y_{k-1} + \dots + w_{d,k} y_{k-d} + \epsilon_k,$$

- Typical model for the **time dependence of weights**:

$$w_{i,k} = w_{i,k-1} + q_{k-1,i}, \quad q_{k-1,i} \sim \mathcal{N}(0, \sigma^2), \quad i = 1, \dots, d.$$

- Can be written as **linear Gaussian state space model** with process noise $\mathbf{q}_{k-1} = (q_{k-1,1} \dots q_{k-1,d})^T$:

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{q}_{k-1}$$

$$y_k = \mathbf{H}_k \mathbf{x}_k + \epsilon_k.$$

- More general **(TV)ARMA models** can be handled similarly.

Spectral and Covariance Models

- Time series can be often modeled in terms of **spectral density**

$$S(\omega) = \{\text{some function of angular velocity } \omega\}.$$

- Or in terms of **mean and covariance function**:

$$\mathbf{m}(t) = E[\mathbf{x}(t)]$$

$$\mathbf{C}(t, t') = E[(\mathbf{x}(t) - \mathbf{m}(t)) (\mathbf{x}(t') - \mathbf{m}(t'))^T]$$

- Such **Gaussian processes** have representations as **outputs of linear Gaussian systems** driven by **white noise**.
- We often can construct a **linear Gaussian state space model** with a given spectral density or covariance function.
- If spectral density is a **rational function**, this is possible.

Stochastic Differential Equation Models

- Physical systems can be often modeled as **differential equations with random terms** such as

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}(\mathbf{x}, t) + \mathbf{L}(t) \mathbf{w}(t),$$

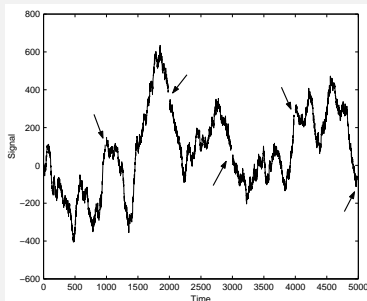
where $\mathbf{w}(t)$ is a **continuous-time white noise process**.

- The noise process can be used for modeling the **deviation from the ideal solution** $d\mathbf{x}(t)/dt = \mathbf{f}(\mathbf{x}, t)$.
- For example, locally (short term) linear functions, almost periodic functions, etc.
- The dynamic model has to be **dicretized** somehow in computations.
- Typically, **measurements** are assumed to be obtained at **discrete instances of time**:

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}(t_k)) + \mathbf{r}_k,$$

Time Series Prediction: CATS Benchmark (1/3)

- An artificial time series with 5 000 data points was given. Within those 100 values were missing:



- The purpose was to predict the 100 missing values based on the other data.
- The performance criterion was the mean square error.

Time Series Prediction: CATS Benchmark (2/3)

- Model was build based on the following principles:
 - Long term behaviour was modeled with locally linear model:

$$\frac{d^2x(t)}{dt^2} = w(t),$$

- The dynamic model is equivalent to the linear difference equation model

$$\begin{pmatrix} x_k \\ \dot{x}_k \end{pmatrix} = \begin{pmatrix} 1 & \Delta t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_{k-1} \\ \dot{x}_{k-1} \end{pmatrix} + \begin{pmatrix} q_{1,k-1}^x \\ q_{2,k-1}^x \end{pmatrix},$$

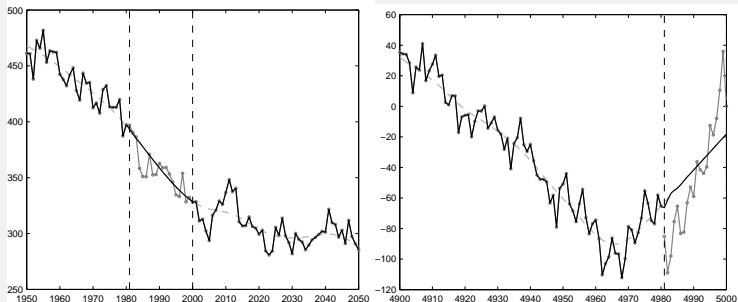
- The measurement model was simply

$$y_k = x_k + r_k^x, \quad r_k^x \sim \mathbf{N}(0, \sigma_x^2).$$

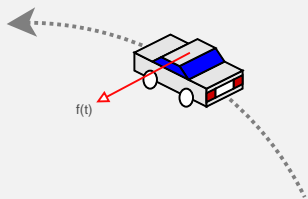
- The remaining residual periodicity was modeled with TVAR model (see the previous slides).

Time Series Prediction: CATS Benchmark (3/3)

- The parameters of linear state space models were estimated with cross-validation.
- Final prediction result was computed with Kalman filter and Rauch-Tung-Striebel smoother:



- This linear filtering solution worked better than any of the non-linear neural network estimators and won the competition!



- The dynamics of the car in 2d (x_1, x_2) are given by the **Newton's law**:

$$\mathbf{f}(t) = m\mathbf{a}(t),$$

where $\mathbf{a}(t)$ is the acceleration, m is the mass of the car, and $\mathbf{f}(t)$ is a vector of (unknown) forces acting the car.

- We shall now model $\mathbf{f}(t)/m$ as a 2-dimensional **white noise process**:

$$d^2x_1/dt^2 = w_1(t)$$

$$d^2x_2/dt^2 = w_2(t).$$

- If we define $x_3(t) = dx_1/dt$, $x_4(t) = dx_2/dt$, then the model can be written as a first order **system of differential equations**:

$$\frac{d}{dt} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \underbrace{\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}}_{\mathbf{F}} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} + \underbrace{\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}}_{\mathbf{L}} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}.$$

- In shorter **matrix form**:

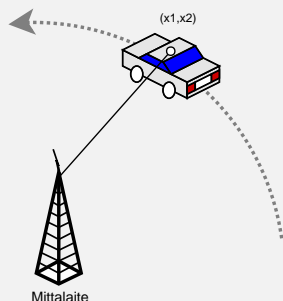
$$\frac{d\mathbf{x}}{dt} = \mathbf{F}\mathbf{x} + \mathbf{L}\mathbf{w}.$$

- If the state of the car is **measured (sampled) with sampling period Δt** it suffices to consider the state of the car only at the time instances $t \in \{0, \Delta t, 2\Delta t, \dots\}$.
- The **dynamic model can be discretized**, which leads to the **linear difference equation** model

$$\mathbf{x}_k = \mathbf{A} \mathbf{x}_{k-1} + \mathbf{q}_{k-1},$$

where $\mathbf{x}_k = \mathbf{x}(t_k)$, \mathbf{A} is the transition matrix and \mathbf{q}_k is a discrete-time Gaussian noise process.

Measurement Model for a Car



- Assume that the **position of the car** (x_1, x_2) is measured and the measurements are corrupted by Gaussian measurement noise $e_{1,k}, e_{2,k}$:

$$y_{1,k} = x_{1,k} + e_{1,k}$$

$$y_{2,k} = x_{2,k} + e_{2,k}.$$

- The **measurement model** can be now written as

$$\mathbf{y}_k = \mathbf{H} \mathbf{x}_k + \mathbf{e}_k, \quad \mathbf{H} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

- The dynamic and measurement models of the car now form a **linear Gaussian filtering model**:

$$\mathbf{x}_k = \mathbf{A} \mathbf{x}_{k-1} + \mathbf{q}_{k-1}$$

$$\mathbf{y}_k = \mathbf{H} \mathbf{x}_k + \mathbf{r}_k,$$

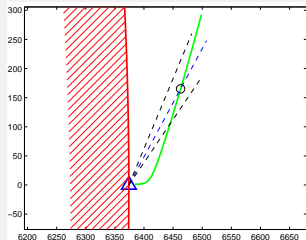
where $\mathbf{q}_{k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ and $\mathbf{r}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$.

- The posterior distribution is **Gaussian**

$$p(\mathbf{x}_k | \mathbf{y}_1, \dots, \mathbf{y}_k) = \mathcal{N}(\mathbf{x}_k | \mathbf{m}_k, \mathbf{P}_k).$$

- The mean \mathbf{m}_k and covariance \mathbf{P}_k of the posterior distribution can be computed by the **Kalman filter**.

Re-Entry Vehicle Model [1/3]



- Gravitation law:

$$\mathbf{F} = m\mathbf{a}(t) = -\frac{GmM\mathbf{r}(t)}{|\mathbf{r}(t)|^3}.$$

- If we also model the friction and uncertainties:

$$\mathbf{a}(t) = -\frac{GM\mathbf{r}(t)}{|\mathbf{r}(t)|^3} - D(\mathbf{r}(t))|\mathbf{v}(t)|\mathbf{v}(t) + \mathbf{w}(t).$$

- If we define $\mathbf{x} = (x_1 \ x_2 \ \frac{dx_1}{dt} \ \frac{dx_2}{dt})^T$, the model is of the form

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}) + \mathbf{L} \mathbf{w}(t).$$

where $\mathbf{f}(\cdot)$ is **non-linear**.

- The **radar measurement**:

$$r = \sqrt{(x_1 - x_r)^2 + (x_2 - y_r)^2} + e_r$$

$$\theta = \tan^{-1} \left(\frac{x_2 - y_r}{x_1 - x_r} \right) + e_\theta,$$

where $e_r \sim N(0, \sigma_r^2)$ and $e_\theta \sim N(0, \sigma_\theta^2)$.

- By suitable numerical integration scheme the model can be approximately written as **discrete-time state space model**:

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{q}_{k-1})$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{r}_k),$$

where \mathbf{y}_k is the vector of measurements, and $\mathbf{q}_{k-1} \sim \mathbf{N}(\mathbf{0}, \mathbf{Q})$ and $\mathbf{r}_k \sim \mathbf{N}(\mathbf{0}, \mathbf{R})$.

- The tracking of the space vehicle can be now implemented by, e.g., **extended Kalman filter (EKF)**, **unscented Kalman filter (UKF)** or **particle filter**.

- *Multiple target tracking (remote surveillance)*
 - Target dynamics are modeled with stochastic difference or differential equations.
 - The measurements arrive at irregular intervals.
 - The number of targets is unknown.
 - Data association indicators are also unknown latent variables.
- *Bus and bus stop tracking*
 - Bus dynamics are modeled with stochastic difference or differential equations.
 - Measurements from GPS, odometer, gyroscope and acceleration sensors.
 - The index of the current bus stop is an unknown variable to be estimated.
 - The known order of bus stops is used as additional information in form of hidden Markov model (HMM).

- *Tracking of objects in images*
 - Object position and pose can be modeled with difference or differential equations.
 - Data representation can be, e.g., spline approximation of the object contour.
 - The measurement model is a complicated likelihood, which depends on object identification algorithm etc.
- *Monitoring of chemical processes*
 - Reaction kinetics are modeled with stochastic differential equations.
 - The measurements can be highly non-linear functions of the state.
 - Processes typically contain unknown physical parameters.

- The purpose of is to estimate the **state of a time-varying system** from **noisy measurements** obtained from it.
- The **linear theory** dates back to **50's**, non-linear **Bayesian theory** was founded in **60's**.
- The efficient computational solutions can be divided into **prediction, filtering and smoothing**.
- **Applications:** tracking, navigation, telecommunications, audio processing, control systems, etc.
- The formal Bayesian estimation equations can be approximated by e.g. **Gaussian approximations, Monte Carlo or Gaussian mixtures**.
- **Formulating** physical systems as **state space models** is a challenging engineering topic as such.

<http://www.lce.hut.fi/research/mm/ekfukf/>