Probabilistic Robotics Course

Unscented Transform

$$x^{(0)} = \frac{1}{\sqrt{2}}\left[0, 2\right]^\top \quad x^{(1)} = -\frac{1}{\sqrt{2}}\left[\sqrt{3}, 1\right]^\top \quad x^{(2)} = -\frac{1}{\sqrt{2}}\left[-\sqrt{3}, 1\right]^\top$$

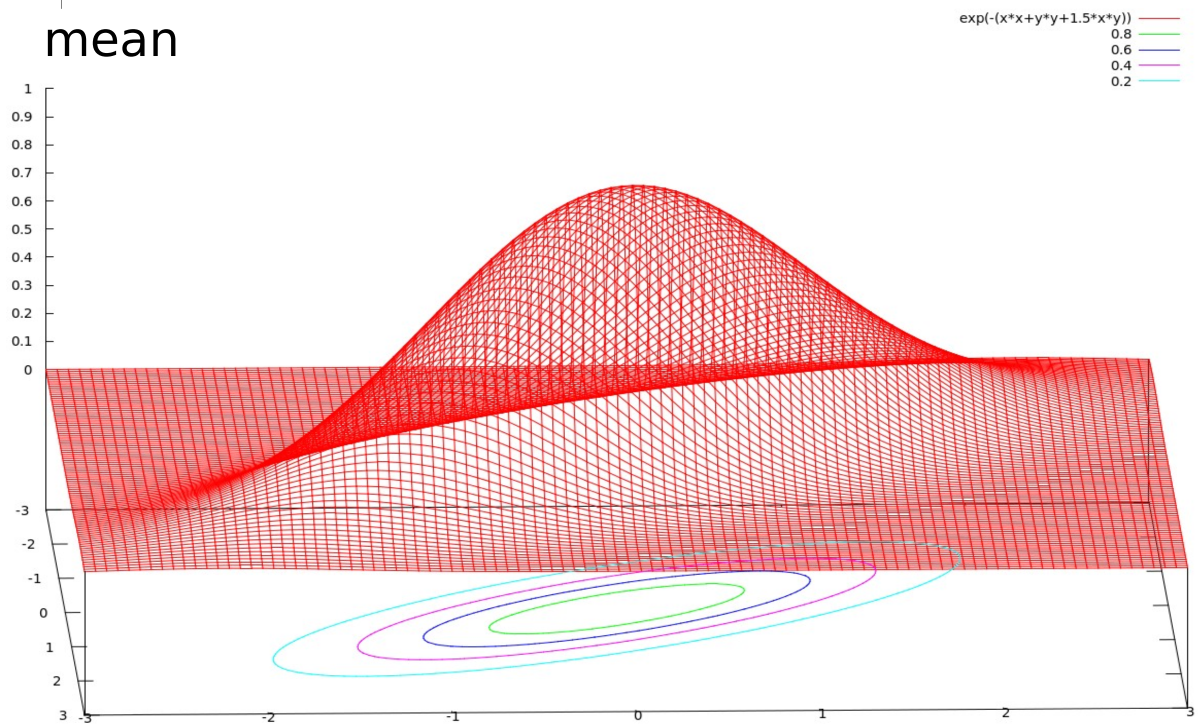
Giorgio Grisetti
grisetti@diag.uniroma1.it

Department of Computer, Control and Management Engineering
Sapienza University of Rome

Gaussian

The pdf of a Gaussian distribution has the following form:

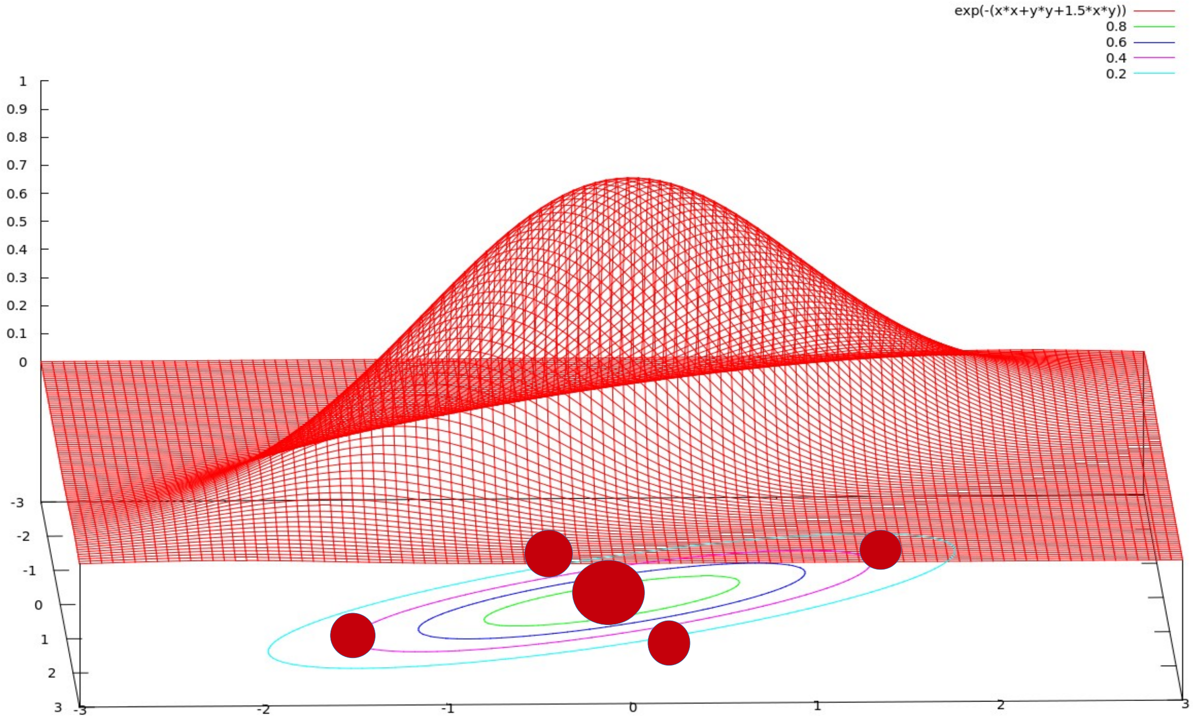
$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \underbrace{\mu}_{\text{mean}}, \underbrace{\mathbf{\Sigma}}_{\text{covariance}}) = \frac{1}{(2\pi)^{n/2}\sqrt{\det \mathbf{\Sigma}}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^T \mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)\right)$$



Unscented Gaussian

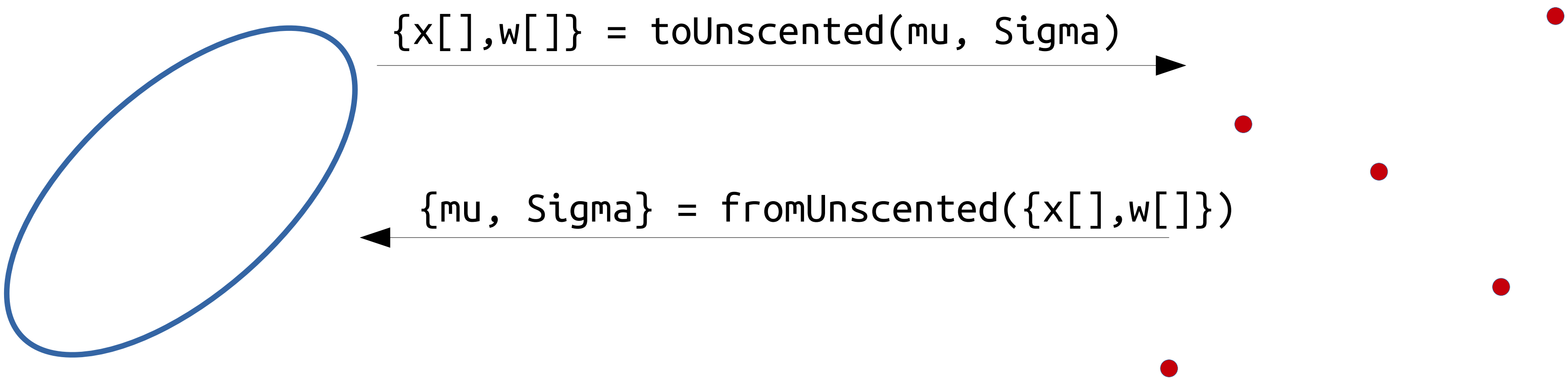
We can represent the Gaussian as a set of

- control points
- weights



Sigma Points from Parameters

The location of the samples and the weight should be computed such that the transformation can be inverted!



Parameters from Sigma Points

Each sigma point is characterized by

- a position $\mathbf{x}^{(i)} \in \Omega$
- a weight for the mean $w_m^{(i)} \in \Re^+$
- a weight for the covariance $w_c^{(i)} \in \Re^+$

$$\left.\begin{array}{l} \mathbf{x}^{(i)} \in \Omega \\ w_m^{(i)} \in \Re^+ \\ w_c^{(i)} \in \Re^+ \end{array}\right\} \chi^{(i)}$$

We can imagine the sigma points as weighed control points that control the shape of the Gaussian.

We can reconstruct the parameters from the sigma points as follows:

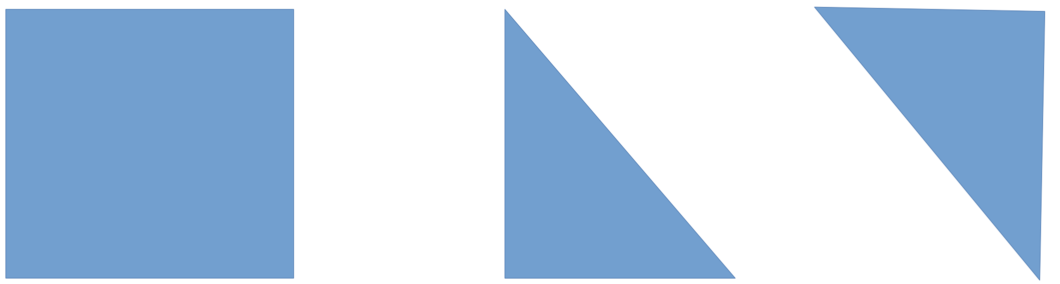
$$\mu = \sum w_m^{(i)} \mathbf{x}^{(i)}$$

$$\mathbf{\Sigma} = \sum w_c^{(i)} (\mathbf{x}^{(i)} - \mu)(\mathbf{x}^{(i)} - \mu)^T$$

Cholesky Decomposition

Can be seen as the square root of a symmetric positive definite matrix:

$$\mathbf{A} = \mathbf{L}\mathbf{L}^T$$



We call $\mathbf{L}$ the square root of $\mathbf{A}$

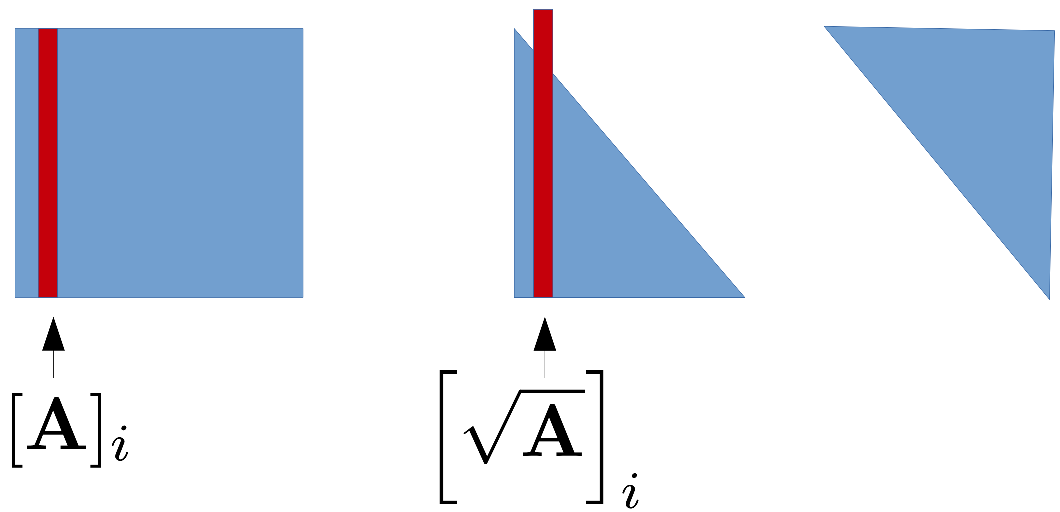
$$\mathbf{L} = \sqrt{\mathbf{A}}$$

Computed by using a variant of Gaussian elimination

Cholesky Decomposition

Can be seen as the square root of a symmetric positive definite matrix:

$$\mathbf{A} = \mathbf{L}\mathbf{L}^T$$



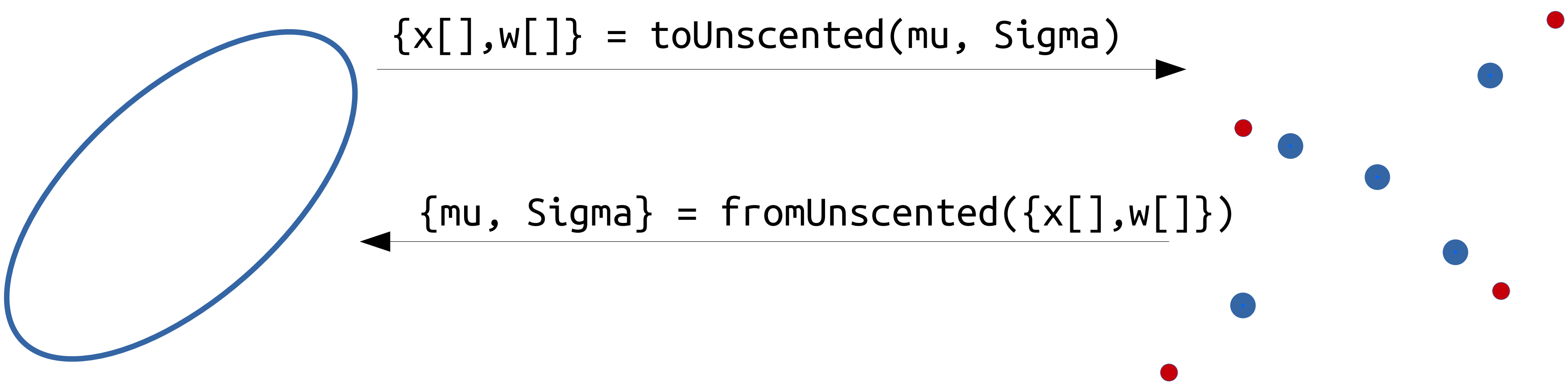
we will call the i-th column a matrix $[\mathbf{A}]_i$

and the i-th column if its square root $\left[\sqrt{\mathbf{A}}\right]_i$

Exercise: implement the Cholesky decomposition in C/C++, no libraries allowed

Sigma Points from Parameters

Different combination of weights/sample locations might allow to go forth and back from the representations, since in the sigma points we have more DoF than in the mean and covariance:



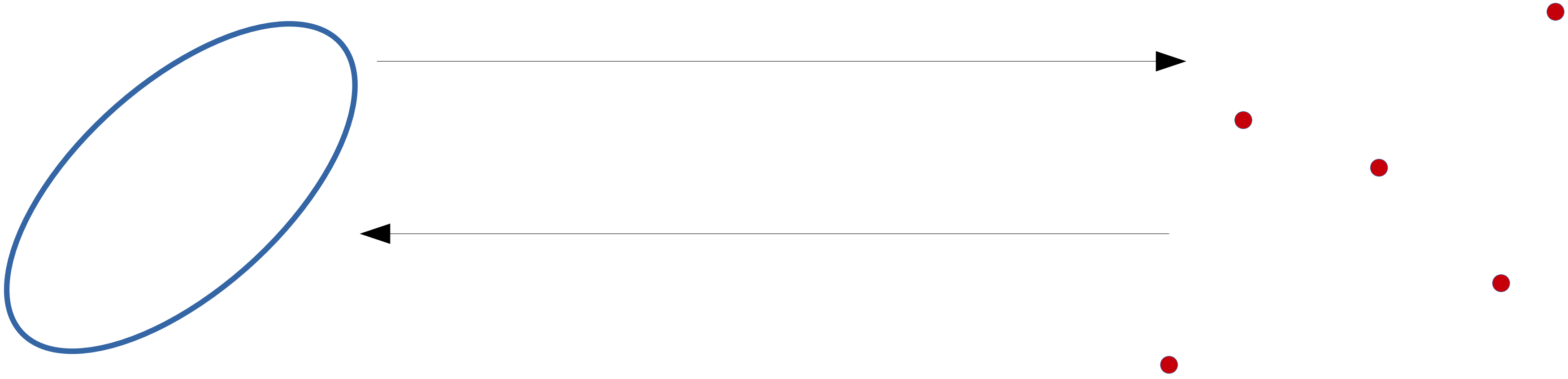
We can control these additional degrees of freedom through some "parameters":

$$\alpha \in (0, 1]$$ Influences how far the points are from the mean

$$\beta = 2$$

Sigma Points from Parameters

Different combination of weights/sample locations might allow to go forth and back from the representations, since in the sigma points we have more DoF than in the mean and covariance:



Use these:

$$
\begin{aligned}
\alpha &= 10^{-3} \\
\beta &= 2
\end{aligned}
$$

Sigma Points from Parameters

Let ***n*** be the dimension, we will have ***2n+1*** sigma points

- Compute the following quantities

$$
\begin{aligned}
\lambda &= \alpha^2 n \\
\mathbf{L}' &= \sqrt{(n+\lambda)\mathbf{\Sigma}}
\end{aligned}
$$

- Compute the positions of the sigma points

$$
\begin{aligned}
\mathbf{x}^{(0)} &= \mu \\
\mathbf{x}^{(i)} &= \mu + [\mathbf{L}']_i \ \text{ for } i \in [1..n] \\
\mathbf{x}^{(i)} &= \mu - [\mathbf{L}']_{n-i} \ \text{ for } i \in [n+1..2n]
\end{aligned}
$$

Sigma Points from Parameters

Let ***n*** be the dimension, we will have ***2n+1*** sigma points

- Compute the weights:

$$
\begin{aligned}
w_m^{(0)} &= \frac{\lambda}{n+\lambda} \\
w_c^{(0)} &= w_m^{(0)} + (1 - \alpha^2 + \beta) \\
w_c^{(i)} &= w_m^{(i)} = \frac{1}{2(n+\lambda)}
\end{aligned}
$$

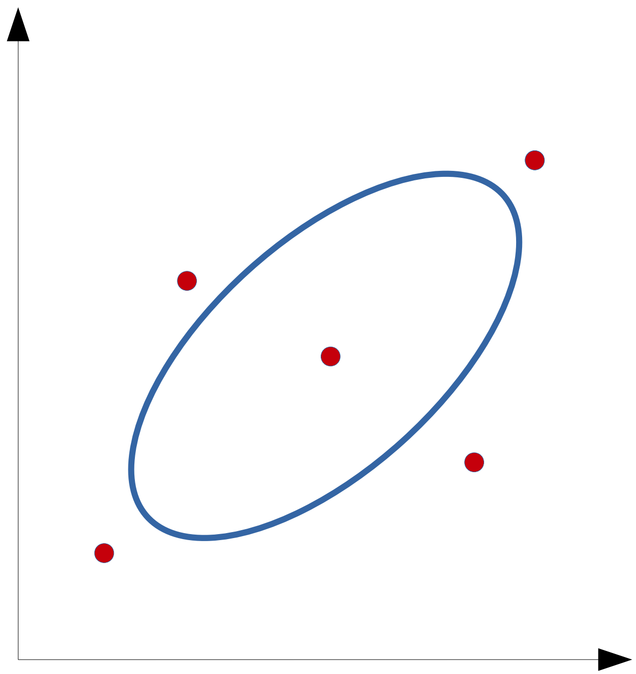
Exercise:

Implement both functions for an N dimensional case and verify that the implementation works

Unscented Marginalization

Let $\mathbf{x}^T = (\mathbf{x}_a^T \; \mathbf{x}_b^T)$ be a random variable represented by an UT:

$$\mathbf{x} \sim \mathcal{UT}(\mathbf{x}; \mathbf{x}^{(i)}, w_m^{(i)}, w_c^{(i)})$$



The marginal

$$p(\mathbf{x}_a) = \int_{\mathbf{x}_b} p(\mathbf{x}_a, \mathbf{x}_b) d\mathbf{x}_b$$

is obtained as:

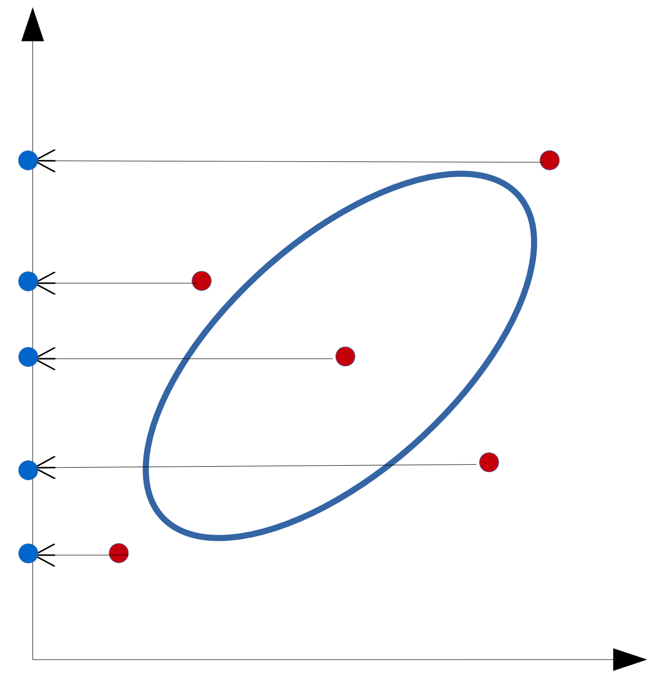
$$\mathbf{x}_a \sim \mathcal{UT}(\mathbf{x}_a; \mathbf{x}_a^{(i)}, w_m^{(i)}, w_c^{(i)})$$

Unscented Marginalization

Keep the sigma points and suppress the marginalized dimensions.

The weights remain the same.

If there are more sigma points than needed, you can always transform it to parametric form and transform it back.



Unscented Functions

Let $\mathbf{x}_a$ be a Gaussian random variable such that:

$$\mathbf{x}_a \sim \mathcal{UT}(\mathbf{x}_a; \mathbf{x}_a^{(i)}, w_m^{(i)}, w_c^{(i)})$$

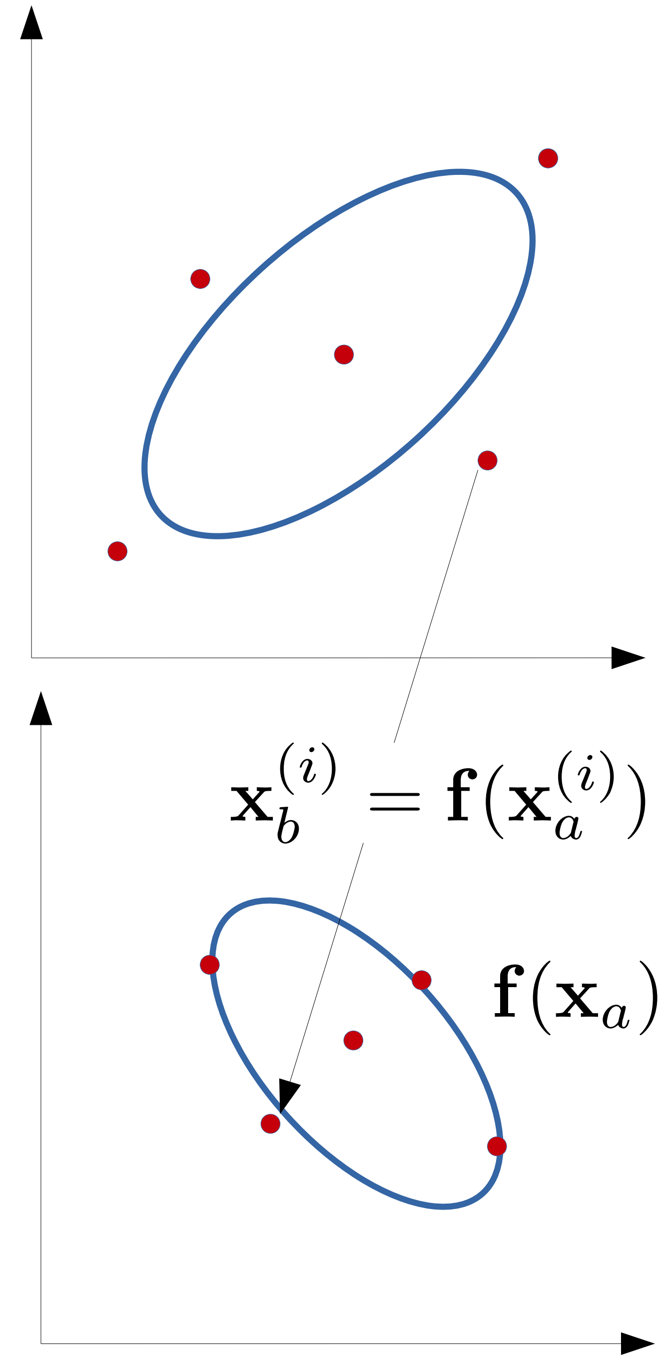
Let $\mathbf{x}_b = \mathbf{f}(\mathbf{x}_a)$ be the transformation of $\mathbf{x}_a$

An unscented approximation of $\mathbf{x}_b$ can be obtained as:

$$\mathbf{x}_b \sim \mathcal{UT}(\mathbf{x}_b; \mathbf{x}_b^{(i)}, w_m^{(i)}, w_c^{(i)})$$

where $\mathbf{x}_b^{(i)} = \mathbf{f}(\mathbf{x}_a^{(i)})$

we write this as $\mathrm{X}_b^{(i)} = \mathbf{f}(\mathcal{X}_a^{(i)})$



Unscented Chain Rule

We know

$$\mathbf{x}_a \sim \mathcal{UT}(\mathbf{x}_a; \mathbf{x}_a^{(i)}, w_m^{(i)}, w_c^{(i)})$$

$$p(\mathbf{x}_b|\mathbf{x}_a) = \mathcal{N}(\mathbf{x}_b; \mathbf{f}(\mathbf{x}_a), \mathbf{\Sigma}_{b|a})$$

We want to compute

$$p(\mathbf{x}_a, \mathbf{x}_b) = \mathcal{N}\left(\mathbf{x}_{a,b}; \mu_{a,b}, \mathbf{\Sigma}_{a,b}\right)$$

The parameters are

$$\begin{aligned} \mu_{a,b} &= \begin{pmatrix} \mu_a \\ \mu_b \end{pmatrix} = \begin{pmatrix} \mu_a \\ \mathbf{f}(\mu_a) \end{pmatrix} \\ \mathbf{\Sigma}_{a,b} &= \begin{pmatrix} \mathbf{\Sigma}_a & \mathbf{\Sigma}_{ab} \\ \mathbf{\Sigma}_{ba} & \mathbf{\Sigma}_{b|a} + \mathbf{\Sigma}_{bb} \end{pmatrix} \end{aligned}$$

We need:

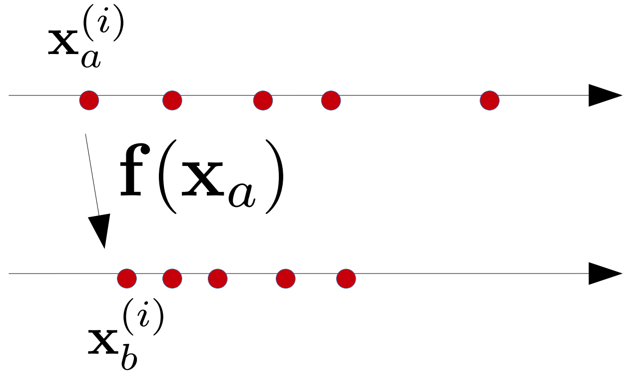
- the cross correlation coefficients
- the covariance term due to the projection of $\mathbf{x}_a$ through $\mathbf{f}(\mathbf{x}_a)$

Unscented Chain Rule

Projection of $\mathbf{x}_a \sim \mathcal{UT}(\mathbf{x}_a; \mathbf{x}_a^{(i)}, w_m^{(i)}, w_c^{(i)})$ through $\mathbf{f}(\mathbf{x}_a)$:

$$\mathbf{x}_b^{(i)} = \mathbf{f}(\mathbf{x}_a^{(i)})$$

$$\mu_b = \sum_i w_m^{(i)} \mathbf{f}(\mathbf{x}_a^{(i)})$$



Unscented (cross) correlations:

$$\mathbf{\Sigma}_{ab} = \sum_i w_c^{(i)} (\mathbf{x}_a^{(i)} - \mu_a)(\mathbf{x}_b^{(i)} - \mu_b)^T$$

$$\mathbf{\Sigma}_{bb} = \sum_i w_c^{(i)} (\mathbf{x}_b^{(i)} - \mu_b)(\mathbf{x}_b^{(i)} - \mu_b)^T$$

Unscented Conditioning

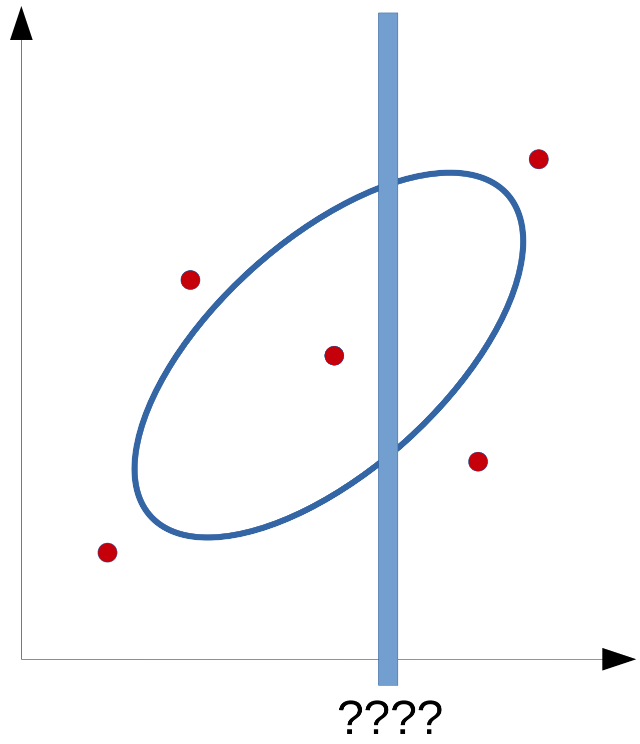
Not easy

Reason:

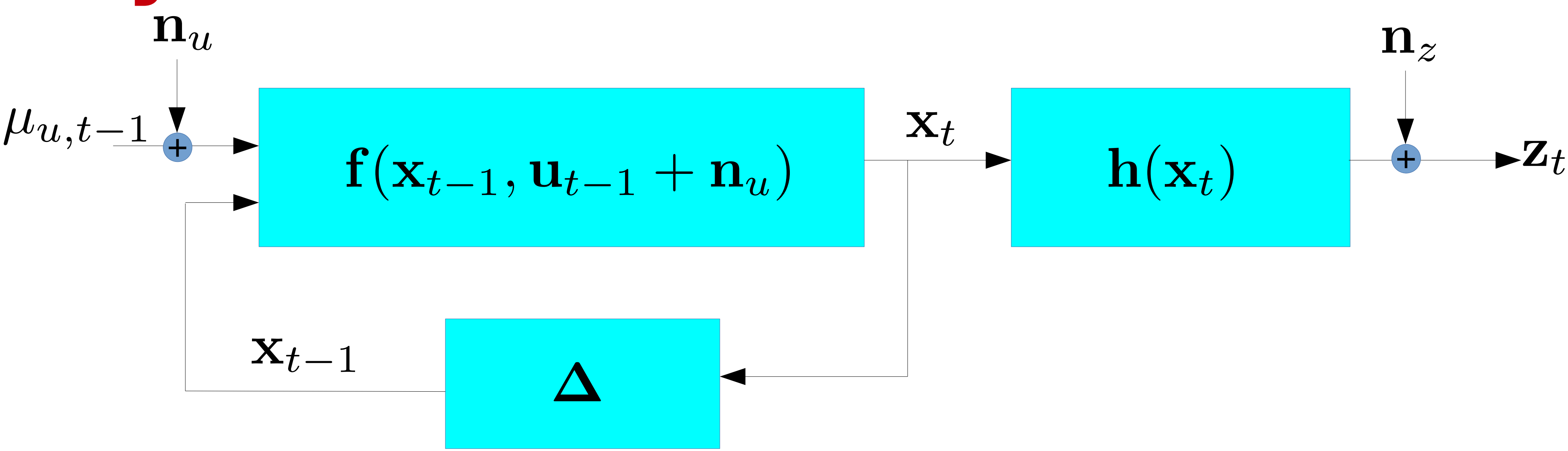
- control points do not like to be sliced

What to do:

go back to the usual parameterization



System with Gaussian Noise



Inputs and Observations are affected by zero mean Gaussian noise. Initial belief is Gaussian

$$\mathbf{n}_u \sim \mathcal{N}(\mathbf{n}_u | 0, \boldsymbol{\Sigma}_u)$$

$$\mathbf{n}_z \sim \mathcal{N}(\mathbf{n}_z | 0, \boldsymbol{\Sigma}_z)$$

$$p(\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_0 | \mu_0, \boldsymbol{\Sigma}_0)$$

For compactness we embed the noise in the control

$$\mathbf{u}_{t-1} \sim \mathcal{N}(\mathbf{n}_z | \mu_{u,t-1}, \boldsymbol{\Sigma}_{u,t})$$

Unscented Filtering

Some considerations

- The unscented transform behaves better under non-linear transformations
- We can replace
  - Unscented projection through function to compute the prediction
  - Unscented chain rule to compute the joint distribution over measurements and predicted states
  - The conditioning remains the same as in (E)KF

No Jacobians!

Predict

Incorporate the control by computing the Gaussian distribution of the next state given the input

The next state is a nonlinear function of the past state and the controls:

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_{t-1})$$

The previous states and controls are distributed according to

$$\begin{pmatrix} \mathbf{x}_{t-1|t-1} \\ \mathbf{u}_{t-1} \end{pmatrix} \sim \mathcal{N}\left[\begin{pmatrix} \mu_{t-1|t-1} \\ \mu_{u,t-1} \end{pmatrix}, \begin{pmatrix} \mathbf{\Sigma}_{t-1|t-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_{u,t-1} \end{pmatrix}\right]$$

Predict

Compute the sigma points from the joint over previous states and controls

$$\begin{pmatrix} \mathbf{x}_{t-1|t-1} \\ \mathbf{u}_{t-1} \end{pmatrix} \sim \mathcal{N}\left[\begin{pmatrix} \mu_{t-1|t-1} \\ \mu_{u,t-1} \end{pmatrix}, \begin{pmatrix} \mathbf{\Sigma}_{t-1|t-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_{u,t-1} \end{pmatrix}\right]$$

They have dimension dim(x) +dim(u)

$$\mathcal{X}^{(i)}_{t-1|t-1}$$

Project the point through transition function

$$\mathcal{X}^{(i)}_{t|t-1} = \mathbf{f}(\mathcal{X}^{(i)}_{t-1|t-1})$$

They have dimension dim(x)

Update: Chain Rule

Given the following distributions

$$\begin{aligned} p(\mathbf{z}_t|\mathbf{x}_t) &= \mathcal{N}(\mathbf{z}_t; \mathbf{h}(\mathbf{x}_t), \mathbf{\Sigma}_{z|x}) \\ p(\mathbf{x}_t) &= \mathcal{UT}(\mathbf{x}_t; \mathcal{X}^{(i)}_{t|t-1}) \end{aligned}$$

We want to compute the joint p(x,z)

$$\begin{aligned} \mu_{x,z} &= \begin{pmatrix} \mu_x \\ \mu_z \end{pmatrix} \\ \mathbf{\Sigma}_{x,z} &= \begin{pmatrix} \mathbf{\Sigma}_x & \mathbf{\Sigma}_{xz} \\ \mathbf{\Sigma}_{zx} & \mathbf{\Sigma}_{z|x} + \mathbf{\Sigma}_{zz} \end{pmatrix} \end{aligned}$$

Update Chain Rule

Projection of $\mathbf{x}_{t|t-1} \sim \mathcal{UT}(\mathbf{x}_t; \mathcal{X}^{(i)}_{t|t-1})$ through $\mathbf{h}(\mathbf{x}_t)$

$$\mathbf{z}_t^{(i)} = \mathbf{h}(\mathbf{x}^{(i)}_{t|t-1})$$

$$\mu_z = \sum_i w_m^{(i)} (\mathbf{z}_t^i)$$

Unscented (cross) correlations:

$$\mathbf{\Sigma}_{xz} = \sum_i w_c^{(i)} (\mathbf{x}^{(i)}_{t|t-1} - \mu_{t|t-1})(\mathbf{z}_t^{(i)} - \mu_z)^T$$

$$\mathbf{\Sigma}_{zz} = \sum_i w_c^{(i)} (\mathbf{z}_t^{(i)} - \mu_z)(\mathbf{z}_t^{(i)} - \mu_z)^T$$

Update: Conditioning

Given the joint properties

$$\mu_{x,z} = \begin{pmatrix} \mu_x \\ \mu_z \end{pmatrix}$$

computed from chain rule

$$\mathbf{\Sigma}_{x,z} = \begin{pmatrix} \mathbf{\Sigma}_x & \mathbf{\Sigma}_{xz} \\ \mathbf{\Sigma}_{zx} & \mathbf{\Sigma}_{z|x} + \mathbf{\Sigma}_{zz} \end{pmatrix}$$

we condition on the actual measurement:

$$\begin{aligned} \mu_{t|t} &= \mu_{t|t-1} + \mathbf{\Sigma}_{xz} \left(\mathbf{\Sigma}_{z|x} + \mathbf{\Sigma}_{zz} \right)^{-1} \left(\mathbf{z}_t - \mu_z \right) \\ \mathbf{\Sigma}_{t|t} &= \mathbf{\Sigma}_{t|t-1} - \mathbf{\Sigma}_{xz} \left(\mathbf{\Sigma}_{z|x} + \mathbf{\Sigma}_{zz} \right)^{-1} \mathbf{\Sigma}_{zx} \end{aligned}$$

Wrapup (UKF)

$$\begin{pmatrix} \mathbf{x}_{t-1|t-1} \\ \mathbf{u}_{t-1} \end{pmatrix} \sim \mathcal{N}\left[\begin{pmatrix} \mu_{t-1|t-1} \\ \mu_{u,t-1} \end{pmatrix}, \begin{pmatrix} \mathbf{\Sigma}_{t-1|t-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_{u,t-1} \end{pmatrix}\right]$$

compute sigma points

$$\mathcal{X}^{(i)}_{t-1|t-1}$$

apply transition

$$\mathcal{X}^{(i)}_{t|t-1} = \mathbf{f}(\mathcal{X}^{(i)}_{t-1|t-1})$$

Predict

compute sigma points and mean of measurement

$$\mathbf{z}^{(i)}_t = \mathbf{h}(\mathbf{x}^{(i)}_{t|t-1})$$

$$\mu_z = \sum_i w^{(i)}_m \mathbf{z}^{(i)}_t$$

compute cross correlation of joint distribution

$$\mathbf{\Sigma}_{xz} = \sum_i w^{(i)}_c (\mathbf{x}^{(i)}_{t|t-1} - \mu_{t|t-1})(\mathbf{z}^{(i)}_t - \mu_z)^T$$

$$\mathbf{\Sigma}_{zz} = \sum_i w^{(i)}_c (\mathbf{z}^{(i)}_t - \mu_z)(\mathbf{z}^{(i)}_t - \mu_z)^T$$

conditioning

$$\begin{aligned} \mu_{t|t} &= \mu_{t|t-1} + \mathbf{\Sigma}_{xz}\left(\mathbf{\Sigma}_{z|x} + \mathbf{\Sigma}_{zz}\right)^{-1}(\mathbf{z}_t - \mu_z) \\ \mathbf{\Sigma}_{t|t} &= \mathbf{\Sigma}_{t|t-1} - \mathbf{\Sigma}_{xz}\left(\mathbf{\Sigma}_{z|x} + \mathbf{\Sigma}_{zz}\right)^{-1}\mathbf{\Sigma}_{zx} \end{aligned}$$

Update

References

Further details are here (warmly recommended)

- *Thomas Schoen*, On Manipulating the Multivariate Gaussian Density
- *Simon J. Julier*, The Scaled Unscented Transform

Summary

- UKF is a non-linear extension of EKF
- It is reported to better deal with non-linear transition and observation functions
- Still relies on the Gaussian assumption and uses pure Gaussian conditioning
- Derivative free