

# 11. The Unscented Kalman Filter

In the previous chapter we discussed a first approach to applying the Kalman filter to nonlinear systems. There, the approach was to apply the Kalman filter to the first-order approximation of the nonlinear system equations. In this chapter we will present an alternative approach, which uses a specially selected set of points to approximate the random variables. These points are then transformed through the (full) nonlinear functions, and from the transformed points the relevant statistics are recovered. We will see that the resulting *Unscented Kalman Filter* (UKF) is often easier to implement than the EKF, and often provides better performance.

## 11.1 Yet another derivation of the Kalman filter gain

Given a random variable  $x$  and an observation of that variable  $z$ , corrupted by measurement noise  $w$  as

$$z = h(x, w)$$

where  $w$  is independent of  $x$ , and we know:

$$\begin{aligned} \mathbb{E}[x] &= \hat{x}_p & \text{Var}[x] &= P_p \\ \mathbb{E}[w] &= 0 & \text{Var}[w] &= R \end{aligned}$$

We derived the Kalman filter gain  $K$  as that linear update rule that minimizes the trace of the estimation error's variance, where the update estimate is given by

$$\hat{x}_m = \hat{x}_p + K(z - \hat{z}) \tag{11.1}$$

where  $z$  is the observation, and  $\hat{z}$  is the expected measurement.

---

Last update: 2018-02-22 at 12:36:35

For linear measurements, with  $z = Hx + w$ , the gain is computed as

$$K = P_p H^T (H P_p H^T + R)^{-1}$$

This can be rewritten as follows, introducing the cross-covariances  $P_{xz}$  and  $P_{zz}$  as

$$\begin{aligned} P_{xz} &:= \mathbb{E} [(x - \hat{x}_p)(z - \hat{z})^T] \\ &= \mathbb{E} [(x - \hat{x}_p)(Hx + w - H\hat{x}_p)^T] \\ &= \mathbb{E} [(x - \hat{x}_p)(x - \hat{x}_p)^T] H^T + \mathbb{E} [(x - \hat{x}_p)w^T] \\ &= P_p H^T \quad (\text{by independence of } w \text{ and } x) \\ P_{zz} &:= \mathbb{E} [(z - \hat{z})(z - \hat{z})^T] \\ &= \mathbb{E} [(Hx + w - H\hat{x}_p)(Hx + w - H\hat{x}_p)^T] \\ &= H \mathbb{E} [(x - \hat{x}_p)(x - \hat{x}_p)^T] H^T + \mathbb{E} [ww^T] + H \mathbb{E} [(x - \hat{x}_p)w^T] + \mathbb{E} [w(x - \hat{x}_p)^T] H^T \\ &= H P_p H^T + R \end{aligned}$$

This allows us to write

$$K = P_{xz} P_{zz}^{-1}. \tag{11.2}$$

Note that the covariance update can also be rewritten then, as

$$\begin{aligned} P_m &= (I - KH) P_p = P_p - K H P_p \\ &= P_p - P_{xz} P_{zz}^{-1} P_{xz}^T \\ &= P_p - P_{xz} P_{zz}^{-1} P_{zz} P_{zz}^{-1} P_{xz}^T \\ &= P_p - K P_{zz} K^T \end{aligned} \tag{11.3}$$

We can now use (11.1), (11.2), and (11.3) to do the Kalman filter update if we have an alternative method to compute the expected measurement  $\hat{z}$  and the statistics  $P_{xz}$  and  $P_{zz}$ . This would then not require the linearization  $H$ .

### 11.1.1 Nonlinear systems

Application of this Kalman filter gain to nonlinear systems requires the computation of the statistics  $P_{xz}$  and  $P_{zz}$  for nonlinear functions  $h$ . We have seen that this can be done

using the change of variables formula, though it typically is intractable to compute (and furthermore, using the change of variables formula requires the full PDF of the random variables – often we only have access to their mean and covariance).

For the extended Kalman filter, we computed  $P_{xz}$  and  $P_{zz}$  through a first order Taylor series expansion of the measurement model, and used  $\hat{z} = h(\hat{x}_p, 0)$  for the measurement. The unscented filter is simply an alternative approach to computing these quantities, that does not require an approximation of the function  $h(\cdot, \cdot)$ .

## 11.2 The Unscented Transform

The unscented transform is a method for approximating the mean and variance of a random variable transformed through a nonlinear function. Before we state the transform, we provide a motivating example.

### 11.2.1 Motivating example: polar to Cartesian coordinates

Consider a radar that measures the bearing  $\theta$  and range  $r$  to a target (that is, its position in polar coordinates). The bearing and range measurements are corrupted by (independent), Gaussian noises – a representative sensor may have 2cm standard deviation on the range, and 15° standard deviation on the bearing.

We may be interested in the target's position in the Cartesian plane, that is its position  $x = (x_1, x_2)$  computed from the measurements as

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = f(\xi) = \begin{bmatrix} r \cos \theta \\ r \sin \theta \end{bmatrix}$$

where  $\xi = (r, \theta)$ . The problem is to now compute the mean and variance of  $x$ , given knowledge of  $r$  and  $\theta$ .

For this example, the true target is located at  $x_1 = 1$  and  $x_2 = 0$ .

#### Sample-based approximation

One approach is to generate samples of  $r$  and  $\theta$ , transform the samples through  $f$ , and then compute the resulting sample statistics. Let  $\tilde{\xi}_i$  be samples drawn from a distribution

so that the sample mean and variance match the underlying variable's mean and variance:

$$\lim_{N \rightarrow \infty} \sum_{i=0}^N \frac{1}{N} \check{\xi}_i = \text{E} [\xi]$$

$$\lim_{N \rightarrow \infty} \sum_{i=0}^N \frac{1}{N-1} (\check{\xi}_i - \text{E} [\xi]) (\check{\xi}_i - \text{E} [\xi])^T = \text{Var} [\xi]$$

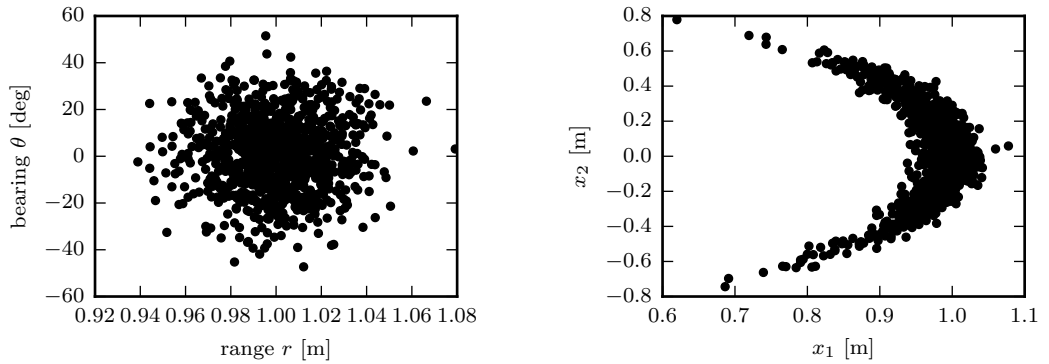
For each sample  $\check{\xi}_i$  we compute a transformed sample  $\check{x}_i = f(\check{\xi}_i)$ . Under the law of large numbers (to be made precise later), we can then say

$$\lim_{N \rightarrow \infty} \sum_{i=0}^N \frac{1}{N} \check{x}_i = \text{E} [x]$$

$$\lim_{N \rightarrow \infty} \sum_{i=0}^N \frac{1}{N-1} (\check{x}_i - \text{E} [x]) (\check{x}_i - \text{E} [x])^T = \text{Var} [x]$$

That is, we can approximate the mean and variance of  $x$  from the samples  $\check{x}_i$ . Note that, for finite  $N$ , these approximations are themselves random variables (as they are composed of random variables).

Below, we show the position solutions for  $N = 1000$  range and bearing measurements, for a true target at the position  $(x_1, x_2) = (1, 0)$ . On the left, the samples are shown in the  $(r, \theta)$  plane, and on the right the transformed samples in  $(x_1, x_2)$ . We note the crescent shape of the position samples.



Though this is a good way of approximating the statistics of the transformed variable, sampling many points is computationally expensive, and the the resulting estimates are themselves random.

### First-order approximation

We can try to approximate the mean and variance of  $x_1$  and  $x_2$  using a first-order approach, similar to what we did for the EKF.

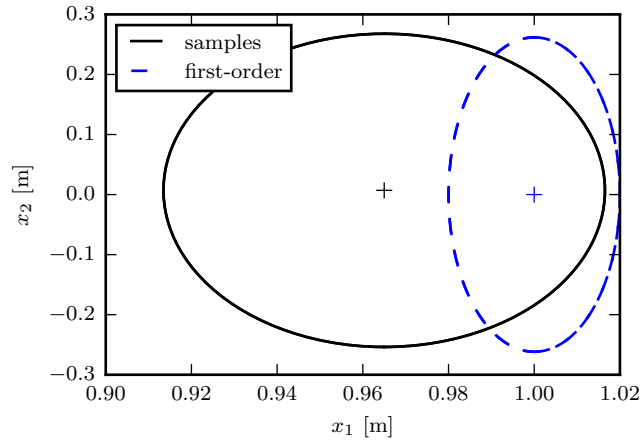
$$x = f(\xi) = f(\hat{\xi}) + \frac{\partial f(\hat{\xi})}{\partial \xi} (\xi - \hat{\xi}) + \dots \quad (11.4)$$

$$A := \frac{\partial f(\hat{\xi})}{\partial \xi} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathbb{E}[x] \approx f(\hat{\xi}) = \begin{bmatrix} \hat{r} \cos \hat{\theta} \\ \hat{r} \sin \hat{\theta} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (11.5)$$

$$\text{Var}[x] \approx A \text{Var}[\xi] A^T = \begin{bmatrix} 0.02^2 & 0 \\ 0 & 15^2 \end{bmatrix} \quad (11.6)$$

We can now compare these predicted statistics to the sample statistics for the samples from the distribution, this is shown in the below figure:



We note two properties of the approximation:

- the mean is incorrect – the sample mean lies more than one estimated standard deviation away from the estimated mean, and
- the variance in  $x_1$  is significantly under-estimated.

Therefore, in this case, the first-order approximation fails to capture important information of the problem. The unscented transform, below, presents an alternative method for computing the mean and variance of the transformed variables. It will be shown to perform much better for this example.

### 11.2.2 The unscented transform

The *unscented transform*<sup>\*</sup> proceeds by selecting a deterministic set of points to represent the mean and variance of a random variable. These points are then transformed through the nonlinear function in question, and the mean and variance of the resulting points are used as an approximation for the mean and variance of the true random variable transformed through the nonlinear function. The intuition is that *it is easier to approximate a distribution than to approximate a nonlinear function*.

Note that we will select only a small set of points (unlike the example before), and we will select them *deterministically*.

Here, we will only present the most basic point selection scheme. We will call these points “sigma-points”.

Let  $x \in \mathbb{R}^n$  and  $y = f(x) \in \mathbb{R}^m$ , with the mean and variance of  $x$  known:

$$\mathbb{E}[x] = \mu_x \quad \text{Var}[x] = P_x.$$

We define a set of  $2n$  sigma-points,  $s_{x,i}$  for  $i \in \{0, 1, 2, \dots, 2n - 1\}$ , computed as below, for

---

<sup>\*</sup>About the name (by Jeffrey Uhlmann, from [http://ethw.org/First-Hand:The\\_Unscented\\_Transform](http://ethw.org/First-Hand:The_Unscented_Transform)):

*Initially I only referred to it as the “new filter.” Needing a more specific name, people in my lab began referring to it as the “Uhlmann filter,” which obviously isn’t a name that I could use, so I had to come up with an official term. One evening everyone else in the lab was at the Royal Opera House, and as I was working I noticed someone’s deodorant on a desk. The word “unscented” caught my eye as the perfect technical term. At first people in the lab thought it was absurd – which is okay because absurdity is my guiding principle – and that it wouldn’t catch on. My claim was that people simply accept technical terms as technical terms: for example, does anyone think about why a tree is called a tree?*

$$i = \{0, 1, \dots, n-1\}$$

$$s_{x,i} = \mu_x + (\sqrt{nP_x})_i$$

$$s_{x,n+i} = \mu_x - (\sqrt{nP_x})_i$$

where  $A = \sqrt{A}\sqrt{A}^T$  for  $A = A^T > 0$  (one way to compute this is with the Cholesky decomposition) and  $(A)_i$  represents the  $i$ 'th column of the matrix  $A$ . These sigma-points lie on the covariance contour of  $x$ .

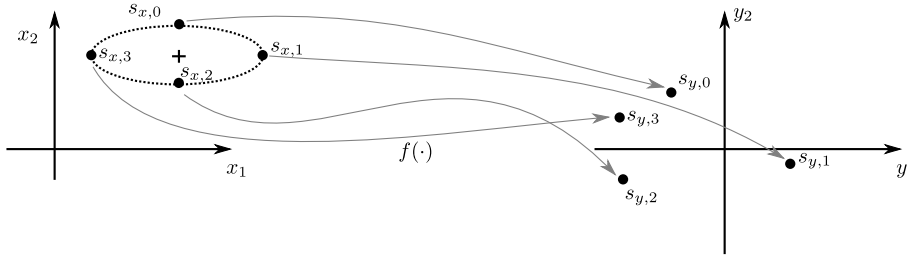
We now define the transformed sigma-points  $s_{y,i}$  as

$$s_{y,i} := f(s_{x,i}) \quad \text{for } i \in \{0, 1, 2, \dots, 2n-1\}$$

from which we compute an approximation for the mean and variance of  $y$ :

$$\mathbb{E}[y] \approx \mu_y := \sum_{i=0}^{2n-1} \frac{1}{2n} s_{y,i} \quad (11.7)$$

$$\text{Var}[y] \approx P_y := \sum_{i=0}^{2n-1} \frac{1}{2n} (s_{y,i} - \mu_y)(s_{y,i} - \mu_y)^T \quad (11.8)$$



Remarks:

- This allows for easy ‘rapid-prototyping’: we don’t need to explicitly compute the derivatives (or any other properties) of the function  $f$ , so it is easy to evaluate changes in  $f$ .
- This requires to compute the matrix square root of an  $n \times n$  matrix, which may be computationally expensive, with complexity  $\mathcal{O}(n^3)$ . However, the matrix multiplications for the first-order approach have the same complexity.
- The function  $f$  need not be differentiable (or even continuous) for this approximation.

### 11.2.3 Revisiting the example

We now apply the unscented transform to the example of Section 11.2.1. Recall  $\xi = (r, \theta)$  with  $E[\xi] = \mu_\xi = (1, 0)$  and  $\text{Var}[\xi] = P_\xi = \text{diag}(0.02^2, 15^2)$ . We defined the corresponding Cartesian position as  $x = (x_1, x_2) = f(\xi) = (r \cos \theta, r \sin \theta)$ .

We first compute the sigma points  $s_{\xi,i}$ , for which we require the matrix square root of  $2P_\xi$ , (easy here)  $\sqrt{2P_\xi} = \text{diag}(0.02\sqrt{2}, 15\sqrt{2})$ .

$$s_{\xi,0} = (1.028\dots, 0), \quad s_{\xi,1} = (1, 21.2\dots), \quad s_{\xi,2} = (0.971\dots, 0), \quad s_{\xi,3} = (1, -21.2\dots).$$

Each sigma point is transformed to generate a corresponding  $s_{x,i}$ :

$$\begin{aligned} s_{x,0} &= f(s_{\xi,0}) = (1.028\dots, 0), & s_{x,1} &= f(s_{\xi,1}) = (0.932\dots, 0.361\dots), \\ s_{x,2} &= f(s_{\xi,2}) = (0.971\dots, 0), & s_{x,3} &= f(s_{\xi,3}) = (0.932\dots, -0.361\dots), \end{aligned}$$

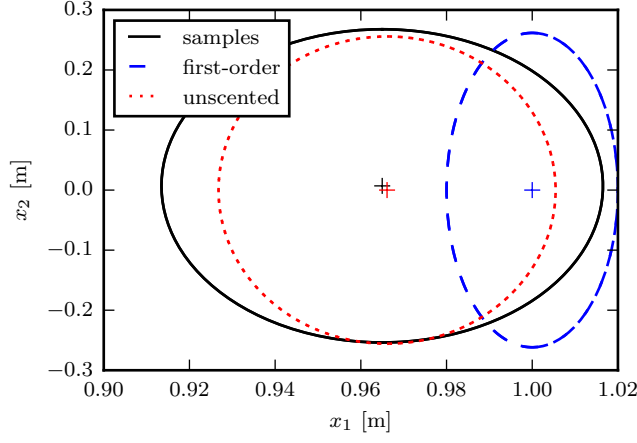
and we can compute the mean and variance approximations as

$$E[x] \approx \mu_x := \sum_{i=0}^3 \frac{1}{4} s_{x,i} = \begin{bmatrix} 0.966\dots \\ 0 \end{bmatrix} \quad (11.9)$$

$$\text{Var}[x] \approx P_x := \sum_{i=0}^3 \frac{1}{4} (s_{x,i} - \mu_x)(s_{x,i} - \mu_x)^T = \begin{bmatrix} 0.0015\dots & 0 \\ 0 & 0.065\dots \end{bmatrix} \quad (11.10)$$

The resulting estimate is shown in the figure below, where it is compared to the first-order estimate, and the empirical sample mean and variance. We note that the unscented transform mean estimate is extremely close to the empirical mean, and that the variance ellipse matches the empirical sample variance very closely. It is clear that the unscented transform yields a much better approximation of the true statistics than the first order methods.





#### 11.2.4 Properties of the Unscented transform

Consider the transformation  $y = f(x)$ , where we assume that  $f$  is analytic. We will apply the unscented transform to approximate properties of  $y$ , which we will compare to the first-order approximation (as used in the EKF). For this analysis, we will assume both  $x$  and  $y$  are scalar, though the analysis generalizes to higher dimensions.

We introduce the Taylor expansion for  $f$ , about a point  $\bar{x}$ :

$$\begin{aligned} f(x) &= \sum_{i=0}^{\infty} \frac{1}{i!} \frac{\partial^i f(\bar{x})}{\partial x^i} (x - \bar{x})^i \\ &= f(\bar{x}) + \frac{\partial f(\bar{x})}{\partial x} (x - \bar{x}) + \frac{1}{2} \frac{\partial^2 f(\bar{x})}{\partial x^2} (x - \bar{x})^2 + \mathcal{O}(x^3) \end{aligned}$$

The exact statistics of  $y$  can be computed from this series, exploiting linearity, as below. We use  $\mu_x = \mathbb{E}[x]$  as the linearization point:

$$\begin{aligned} \mathbb{E}[y] &= \mathbb{E}[f(x)] \\ &= \mathbb{E}[f(\mu_x)] + \mathbb{E}\left[\frac{\partial f(\mu_x)}{\partial x} (x - \mu_x)\right] + \mathbb{E}\left[\frac{1}{2} \frac{\partial^2 f(\mu_x)}{\partial x^2} (x - \mu_x)^2\right] + \dots \\ &= f(\mu_x) + \frac{\partial f(\mu_x)}{\partial x} \mathbb{E}[(x - \mu_x)] + \frac{1}{2} \frac{\partial^2 f(\mu_x)}{\partial x^2} \mathbb{E}[(x - \mu_x)^2] + \dots \\ &= f(\mu_x) + \frac{1}{2} \frac{\partial^2 f(\mu_x)}{\partial x^2} \text{Var}[x] + \mathbb{E}[\mathcal{O}(x^3)] \end{aligned} \tag{11.11}$$

We will compute the variance as  $\text{Var}[y] = \text{Var}[f(x)] = \mathbb{E}[f(x)^2] - \mathbb{E}[f(x)]^2$ . We compute these components as:

$$\begin{aligned}
\mathbb{E}[f(x)^2] &= \mathbb{E}\left[\left(\sum_{i=0}^{\infty} \frac{1}{i!} \frac{\partial^i f(\mu_x)}{\partial x^i} (x - \mu_x)^i\right) \left(\sum_{j=0}^{\infty} \frac{1}{j!} \frac{\partial^j f(\mu_x)}{\partial x^j} (x - \mu_x)^j\right)\right] \\
&= f(\mu_x)^2 + \mathbb{E}\left[2f(\mu_x) \frac{\partial f(\mu_x)}{\partial x} (x - \mu_x)\right] + \mathbb{E}\left[\left(\frac{\partial f(\mu_x)}{\partial x} (x - \mu_x)\right)^2\right] \\
&\quad + \mathbb{E}\left[2\frac{1}{2}f(\mu_x) \frac{\partial^2 f(\mu_x)}{\partial x^2} (x - \mu_x)^2\right] + \mathbb{E}[\mathcal{O}(x^3)] \\
&= f(\mu_x)^2 + \left(\frac{\partial f(\mu_x)}{\partial x}\right)^2 \text{Var}[x] + f(\mu_x) \frac{\partial^2 f(\mu_x)}{\partial x^2} \text{Var}[x] + \mathbb{E}[\mathcal{O}(x^3)]
\end{aligned} \tag{11.12}$$

$$\begin{aligned}
\mathbb{E}[f(x)]^2 &= f(\mu_x)^2 + 2f(\mu_x) \frac{1}{2} \frac{\partial^2 f(\mu_x)}{\partial x^2} \mathbb{E}[(x - \mu_x)^2] + \mathbb{E}[\mathcal{O}(x^3)] \\
&= f(\mu_x)^2 + f(\mu_x) \frac{\partial^2 f(\mu_x)}{\partial x^2} \text{Var}[x] + \mathbb{E}[\mathcal{O}(x^3)]
\end{aligned} \tag{11.13}$$

And therefore:

$$\begin{aligned}
\text{Var}[y] &= \text{Var}[f(x)] = \mathbb{E}[f(x)^2] - \mathbb{E}[f(x)]^2 \\
&= \left(\frac{\partial f(\mu_x)}{\partial x}\right)^2 \text{Var}[x] + \mathbb{E}[\mathcal{O}(x^3)]
\end{aligned} \tag{11.14}$$

We can compare this to the result of the unscented transform. The system is scalar (i.e.  $n = 1$ ), so we will have two points:

$$s_{x,0} := \mu_x + \Delta, \quad s_{x,1} := \mu_x - \Delta \quad \text{where } \Delta := \sqrt{\text{Var}[x]}$$

These points are transformed through  $f$ , to yield the points  $s_{y,i}$ , expressed through the series expansion as:

$$\begin{aligned}
s_{y,0} &:= f(s_{x,0}) \\
&= f(\mu_x) + \frac{\partial f(\mu_x)}{\partial x} (s_{x,0} - \mu_x) + \frac{1}{2} \frac{\partial^2 f(\mu_x)}{\partial x^2} (s_{x,0} - \mu_x)^2 + \mathcal{O}((s_{x,0} - \mu_x)^3) \\
&= f(\mu_x) + \frac{\partial f(\mu_x)}{\partial x} \Delta + \frac{1}{2} \frac{\partial^2 f(\mu_x)}{\partial x^2} \Delta^2 + \mathcal{O}(\Delta^3) \\
s_{y,1} &= f(\mu_x) - \frac{\partial f(\mu_x)}{\partial x} \Delta + \frac{1}{2} \frac{\partial^2 f(\mu_x)}{\partial x^2} \Delta^2 + \mathcal{O}(\Delta^3)
\end{aligned}$$

The mean approximation of the unscented transform, which we will here call  $\mu_y$ , is:

$$\begin{aligned}
\mu_y &= \sum_{i=0}^1 \frac{1}{2} s_{y,i} \\
&= f(\mu_x) + \frac{1}{2} \frac{\partial^2 f(\mu_x)}{\partial x^2} \Delta^2 + \mathcal{O}(\Delta^3) \\
&= f(\mu_x) + \frac{1}{2} \frac{\partial^2 f(\mu_x)}{\partial x^2} \text{Var}[x] + \mathcal{O}(\Delta^3)
\end{aligned} \tag{11.15}$$

Comparing now (11.11) and (11.15) we see that the unscented transform approximates the mean correctly up to second order. Recall that linearization, as was done for the EKF, was correct only to first order.

The unscented transform's variance approximation, which we will here call  $P_y$ , is:

$$\begin{aligned}
P_y &:= \sum_{i=0}^1 \frac{1}{2} (s_{y,i} - \mu_y)^2 \\
&= \frac{1}{2} \left( \frac{\partial f(\mu_x)}{\partial x} \Delta + \mathcal{O}(\Delta^3) \right)^2 + \frac{1}{2} \left( -\frac{\partial f(\mu_x)}{\partial x} \Delta + \mathcal{O}(\Delta^3) \right)^2 \\
&= \left( \frac{\partial f(\mu_x)}{\partial x} \right)^2 \Delta^2 + \mathcal{O}(\Delta^4) = \left( \frac{\partial f(\mu_x)}{\partial x} \right)^2 \text{Var}[x] + \mathcal{O}(\Delta^4)
\end{aligned} \tag{11.16}$$

Comparing (11.14) and (11.16) we notice that the variance estimate is correct to second order – this is the *same order* as for a first order approximate of  $f$ . Even though the errors are on the same order, the errors are not equal, and the unscented transform typically outperforms a first order approximation, as the unscented approximation includes some approximation of higher order terms (unlike a first order approximation).

*Remarks:*

- The errors are on the order of the expectation of the random variable cubed. This make sense: we only had information about the variable squared (its variance) – we could not possibly do any better without more information.
- These errors may still be arbitrarily large. You can generate distributions for which the errors will be unbounded for both the first-order and unscented transformation approximations.
- If the function  $f$  is affine in  $x$ , then the estimates from the unscented transform are exact (as is the first order approximation), for any distribution of  $x$ .

### 11.2.5 Extensions of the unscented transform

This is the simplest version of the unscented transform. Specifically, we have  $2n$  sigma-points, and we've weighted them all equally, (the weights we've chosen are  $1/(2n)$ ). This can be extended as follows:

- The most common scheme involves augmenting this set of points with an additional sigma-point, equal to the mean.
- The points need not be weighted equally – as long as the weights sum up to one.
- There are also schemes that have fewer points (as few as  $n + 1$ ), which are more computationally efficient, but do not provide as accurate an estimate.

## 11.3 The Unscented Kalman Filter

The unscented filter is the application of the unscented transform to the nonlinear system dynamics to approximate the prior mean and variance; and its application to the nonlinear measurement model to compute the measurement statistics from which we compute the Kalman filter gain using the standard equations. We will distinguish between two cases: systems where the noise is purely additive, and systems where the noise enters in a nonlinear fashion.

### 11.3.1 Additive noise

We consider systems of the following form:

$$x(k) = q_{k-1}(x(k-1)) + v(k-1) \quad \mathbb{E}[x(0)] = x_0, \text{Var}[x(0)] = P_0 \quad (11.17)$$

$$\mathbb{E}[v(k-1)] = 0, \text{Var}[v(k-1)] = Q(k-1)$$

$$z(k) = h_k(x(k)) + w(k) \quad \mathbb{E}[w(k)] = 0, \text{Var}[w(k)] = R(k) \quad (11.18)$$

for  $k = 1, 2, \dots$ , and where  $x(0)$ ,  $\{v(\cdot)\}$ , and  $\{w(\cdot)\}$  are mutually independent. The state is  $n$  dimensional.

We note the following properties, by independence of the noise:

$$\begin{aligned} \mathbb{E}[x(k)] &= \mathbb{E}[q_{k-1}(x(k-1))] \\ \text{Var}[x(k)] &= \text{Var}[q_{k-1}(x(k-1))] + Q(k-1) \\ \mathbb{E}[z(k)] &= \mathbb{E}[h_k(x(k))] \\ \text{Var}[z(k)] &= \text{Var}[h_k(x(k))] + R(k) \end{aligned}$$

We will now use the unscented transform to approximate these expectations and variances.

**Initialization:**  $\hat{x}_m(0) = x_0$ ,  $P_m(0) = P_0$ .

**Step 1 (S1): Prior update/Prediction step**

Generate  $2n$  sigma-points:

$$\begin{aligned} s_{x_m(k-1),i} &= \hat{x}_m(k-1) + \left(\sqrt{nP_m(k-1)}\right)_i \\ s_{x_m(k-1),n+i} &= \hat{x}_m(k-1) - \left(\sqrt{nP_m(k-1)}\right)_i \end{aligned}$$

Using these, compute the prior sigma-points:

$$s_{x_p(k),i} = q_{k-1} \left( s_{x_m(k-1),i} \right) \quad \text{for } i \in \{0, 1, \dots, 2n-1\}$$

Compute the prior statistics (note the additive noise):

$$\begin{aligned} \hat{x}_p(k) &= \sum_{i=0}^{2n-1} \frac{1}{2n} s_{x_p(k),i} \\ P_p(k) &= \sum_{i=0}^{2n-1} \frac{1}{2n} (s_{x_p(k),i} - \hat{x}_p(k)) (s_{x_p(k),i} - \hat{x}_p(k))^T + Q(k-1) \end{aligned}$$

**Step 2 (S2): A posteriori update/Measurement update step**

Using the sigma points from Step 1, compute sigma points for the measurements

$$s_{z(k),i} = h_{k-1} \left( s_{x_p(k),i} \right) \quad \text{for } i \in \{0, 1, \dots, 2n-1\}$$

and from that the expected measurement and the associated covariance  $P_{zz}(k)$ :

$$\begin{aligned} \hat{z}(k) &= \sum_{i=0}^{2n-1} \frac{1}{2n} s_{z(k),i} \\ P_{zz}(k) &= \sum_{i=0}^{2n-1} \frac{1}{2n} (s_{z(k),i} - \hat{z}(k)) (s_{z(k),i} - \hat{z}(k))^T + R(k) \end{aligned}$$

We also compute the cross covariance  $P_{xz}(k)$ :

$$P_{xz}(k) = \sum_{i=0}^{2n-1} \frac{1}{2n} (s_{x_p(k),i} - \hat{x}_p(k)) (s_{z(k),i} - \hat{z}(k))^T$$

From this, we apply the Kalman filter gain:

$$\begin{aligned} K(k) &= P_{xz}(k) P_{zz}(k)^{-1} \\ \hat{x}_m(k) &:= \hat{x}_p(k) + K(k) (z(k) - \hat{z}(k)) \\ P_m(k) &= P_p(k) - K(k) P_{zz}(k) K(k)^T \end{aligned}$$

### 11.3.2 Nonlinear noise

The UKF can also straight-forwardly deal with more general nonlinear systems:

$$x(k) = q_{k-1}(x(k-1), v(k-1)) \quad \mathbb{E}[x(0)] = x_0, \text{Var}[x(0)] = P_0 \quad (11.19)$$

$$\mathbb{E}[v(k-1)] = 0, \text{Var}[v(k-1)] = Q(k-1)$$

$$z(k) = h_k(x(k), w(k)) \quad \mathbb{E}[w(k)] = 0, \text{Var}[w(k)] = R(k) \quad (11.20)$$

for  $k = 1, 2, \dots$ , and where  $x(0)$ ,  $\{v(\cdot)\}$ , and  $\{w(\cdot)\}$  are mutually independent. The state is  $n$  dimensional.

We will need an augmented state vector,  $\xi = (x, v, w) \in \mathbb{R}^{n_\xi}$ , with  $n_\xi = (n + \dim(v) + \dim(w))$ . The augmented system functions are defined as

$$\begin{aligned} \bar{q}_{k-1}(\xi) &:= (q_{k-1}(x, v), v, w) \\ \bar{h}_k(\xi) &= h_k(x, w) \end{aligned}$$

The following algorithm is a generalization of that of the previous section, but will be more computationally expensive.

**Initialization:**  $\hat{x}_m(0) = x_0$ ,  $P_m(0) = P_0$ .

**Step 1 (S1): Prior update/Prediction step**

Let

$$\begin{aligned} \hat{\xi}_m(k-1) &:= (\hat{x}_m(k-1), 0, 0) \\ \text{Var}[\xi_m(k-1)] &:= \text{diag}(P_m(k-1), Q(k-1), R(k)) \end{aligned}$$

Generate  $2n_\xi$  sigma-points:

$$\begin{aligned} s_{\xi_m(k-1),i} &= \hat{\xi}_m(k-1) + \left( \sqrt{n_\xi \text{Var} [\xi_m(k-1)]} \right)_i \\ s_{\xi_m(k-1),i} &= \hat{\xi}_m(k-1) - \left( \sqrt{n_\xi \text{Var} [\xi_m(k-1)]} \right)_i \end{aligned}$$

Note that, because the variance is block-diagonal, the computation its square root can be done block-wise.

Using these, and our augmented dynamics equation, compute the prior sigma-points:

$$\begin{aligned} s_{\xi_p(k-1),i} &:= \bar{q}_{k-1} \left( s_{\xi_m(k-1),i} \right) \quad \text{for } i \in \{0, 1, \dots, 2n_\xi - 1\} \\ &=: \left( s_{x_p(k-1),i}, *, * \right) \end{aligned}$$

Compute the prior statistics:

$$\begin{aligned} \hat{x}_p(k) &= \sum_{i=0}^{2n_\xi-1} \frac{1}{2n_\xi} s_{x_p(k),i} \\ P_p(k) &= \sum_{i=0}^{2n_\xi-1} \frac{1}{2n_\xi} \left( s_{x_p(k),i} - \hat{x}_p(k) \right) \left( s_{x_p(k),i} - \hat{x}_p(k) \right)^T \end{aligned}$$

## Step 2 (S2): A posteriori update/Measurement update step

Using the sigma points from Step 1, compute sigma points for the measurements

$$s_{z(k),i} = \bar{h}_{k-1} \left( s_{\xi_p(k),i} \right) \quad \text{for } i \in \{0, 1, \dots, 2n_\xi - 1\}$$

and from that the expected measurement and the associated covariance  $P_{zz}(k)$ :

$$\begin{aligned} \hat{z}(k) &= \sum_{i=0}^{2n_\xi-1} \frac{1}{2n_\xi} s_{z(k),i} \\ P_{zz}(k) &= \sum_{i=0}^{2n_\xi-1} \frac{1}{2n_\xi} \left( s_{z(k),i} - \hat{z}(k) \right) \left( s_{z(k),i} - \hat{z}(k) \right)^T \end{aligned}$$

We also compute the cross covariance  $P_{xz}(k)$ :

$$P_{xz}(k) = \sum_{i=0}^{2n_\xi-1} \frac{1}{2n_\xi} \left( s_{x_p(k),i} - \hat{x}_p(k) \right) \left( s_{z(k),i} - \hat{z}(k) \right)^T$$

From this, we apply the Kalman filter gain:

$$\begin{aligned} K(k) &= P_{xz}(k) P_{zz}(k)^{-1} \\ \hat{x}_m(k) &:= \hat{x}_p(k) + K(k) (z(k) - \hat{z}(k)) \\ P_m(k) &= P_p(k) - K(k) P_{zz}(k) K(k)^T \end{aligned}$$

### 11.3.3 Remarks

- At the beginning of step 2 (for either additive or nonlinear noise), we can compute new sigma points using  $\hat{x}_p$  and  $P_p$ . This will be more accurate than what we have described above, but will require more computation.
- Notice that Section 11.3.2 allows us to straight-forwardly handle the situation where  $v(k-1)$  and  $w(k-1)$  are not independent.
- If either the measurement model or dynamics equation is linear, we don't need to use the unscented transform for the corresponding statistics.
- If only one of the process noise and measurement noise enters in a nonlinear way, we can adapt the equations to capture this with the minimum number of sigma-points.

## 11.4 Revisiting the pathological example

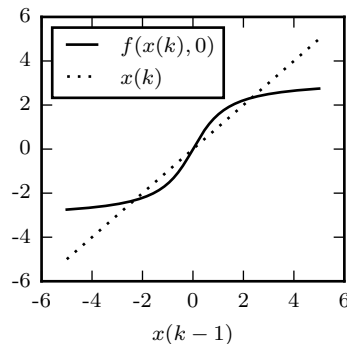
We revisit the pathological example from the EKF notes. Recall the system dynamics:

$$\begin{aligned} x(k) &= q(x(k-1), v(k-1)) = 2 \arctan(x(k-1) + v(k-1)) \\ z(k) &= h(x(k), w(k)) = x(k) + w(k) \end{aligned}$$

and  $x(0) \sim \mathcal{N}(x_0, 1)$ ,  $v(k-1) \sim \mathcal{N}(0, 0.1)$ , and  $w(k) \sim \mathcal{N}(0, 10)$ . We notice that the measurement is affine in the state and noise, and that the dynamics are smooth. The process uncertainty is small, and the measurement uncertainty large.

The dynamics are illustrated below:





The deterministic system (when we set  $v(k) = 0$ ) has three equilibria, at  $x(k) = 0$  and  $x(k) \approx \pm 2.33$ . The equilibrium at zero is unstable, the other two are stable.

We implement the UKF algorithm for nonlinear noise, and run it many times to get a feeling for the estimator's performance. We can compare the performance for  $x_0 = 4$ , and  $x_0 = 0$ .

The performance is roughly the same as we had for the EKF. For  $x_0 = 4$ , the UKF performs well, but for  $x_0 = 0$  it does not. Usually, it's OK, but sometimes the true state goes to one equilibrium, and the estimator to the other. Note

- this is true even though we can measure the state,
- it doesn't matter how long we run the filter, it doesn't correct itself, and
- there is no model mismatch.

Just like the EKF, the UKF has a danger of *diverging*.

## 11.5 Further reading

An excellent paper is S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004 (written by the creators of the Unscented Kalman Filter), which was also the primary source for this lecture. It is very accessible, and provides in-depth discussion.