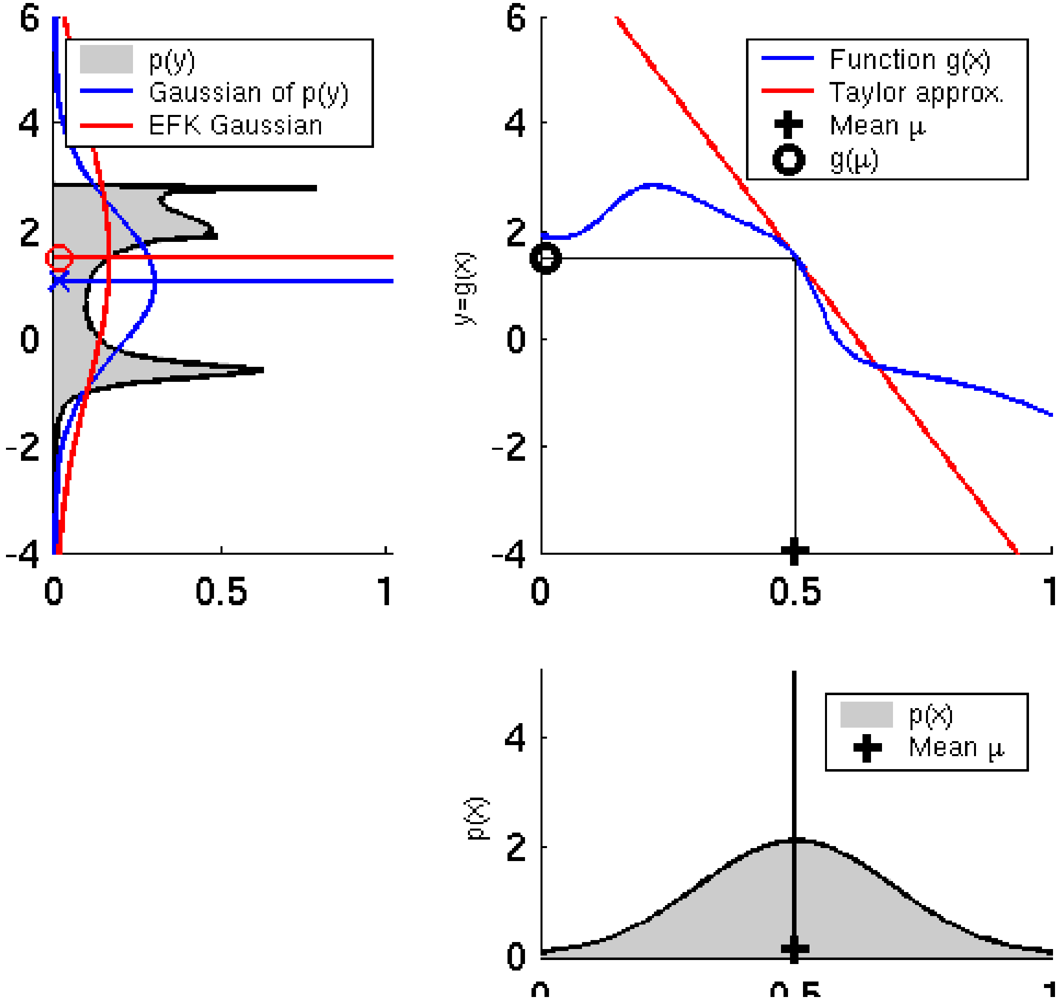
Robótica Móvil
un enfoque probabilístico

Filtro de Bayes– Unscented Kalman Filter (UKF)

Ignacio Mas

KF, EKF



p(y)
Gaussian of p(y)
EFK Gaussian
Function g(x)
Taylor approx.
Mean μ
g(μ)
y=g(x)
p(x)
Mean μ
p(x)
6
4
2
0
-2
-4
0
0.5
1

KF, EKF y UKF

- El filtro de Kalman requiere modelos lineales
- EKF linealiza con expansión de Taylor

¿Hay una **mejor** manera de **linealizar**?

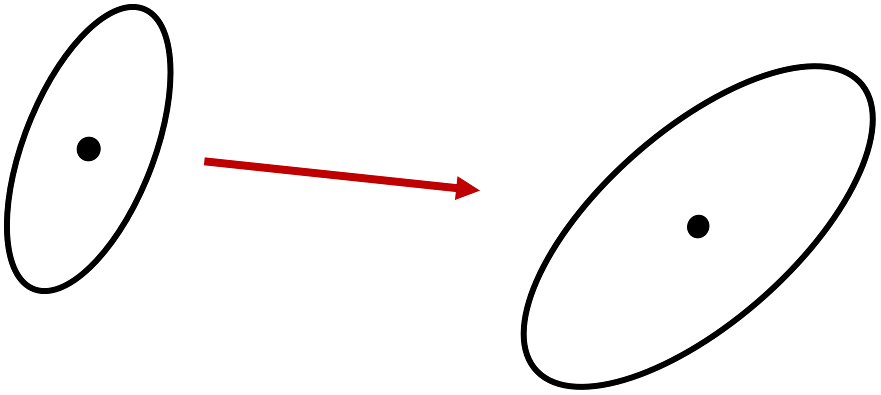
Unscented Transform

⬇

Unscented Kalman Filter (UKF)

Aproximación de Taylor (EKF)

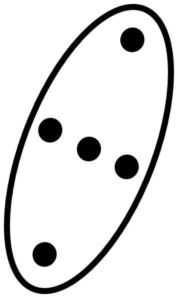
- Linealización de una función no lineal a través de una expansión de Taylor



Unscented Transform

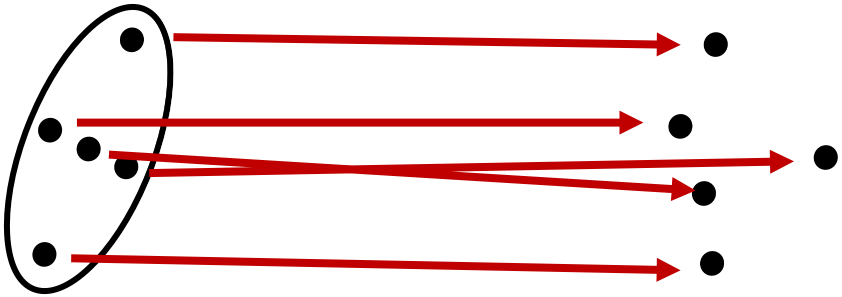
- Calcular un conjunto de puntos:

Puntos Sigma



Unscented Transform

- Transformar cada punto sigma con la función no lineal



Unscented Transform

- Calcular una Gaussiana con los puntos transformados y pesados

Unscented Transform - Pasos

- Calcular un conjunto de Puntos Sigma
- A cada Punto Sigma se le asocia un peso
- Se transforma cada punto con la función no lineal
- Se calcula una Gaussiana a partir de los puntos pesados

- Se evita linealizar **alrededor de la media**, como en el caso de Taylor (EKF)

Puntos Sigma

- ¿Cómo elegir los Puntos Sigma?
- ¿Cómo calcular sus pesos?

- Seleccionar $\mathcal{X}^{[i]}, w^{[i]}$ tales que:

$$\begin{aligned}\sum_i w^{[i]} &= 1 \\ \mu &= \sum_i w^{[i]} \mathcal{X}^{[i]} \\ \Sigma &= \sum_i w^{[i]} (\mathcal{X}^{[i]} - \mu)(\mathcal{X}^{[i]} - \mu)^T\end{aligned}$$

- No hay una única solución a $\mathcal{X}^{[i]}, w^{[i]}$

Puntos Sigma

- Elección de Puntos Sigma

primer punto: la media

$$\begin{aligned}
\mathcal{X}^{[0]} &= \mu \\
\mathcal{X}^{[i]} &= \mu + \left(\sqrt{(n+\lambda)\,\Sigma}\right)_i \qquad \text{for } i = 1, \ldots, n \\
\mathcal{X}^{[i]} &= \mu - \left(\sqrt{(n+\lambda)\,\Sigma}\right)_{i-n} \qquad \text{for } i = n+1, \ldots, 2n
\end{aligned}$$

Vector columna

Raíz cuadrada de
una matriz

Parámetro de escala

dimensionalidad

Raíz cuadrada de una matriz

- Definiendo *S* como $\Sigma = SS$
- Se calcula diagonalizando

$$
\begin{aligned}
\Sigma &= VDV^{-1} \\
&= V \begin{pmatrix} d_{11} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & d_{nn} \end{pmatrix} V^{-1} \\
&= V \begin{pmatrix} \sqrt{d_{11}} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \sqrt{d_{nn}} \end{pmatrix} \begin{pmatrix} \sqrt{d_{11}} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \sqrt{d_{nn}} \end{pmatrix} V^{-1}
\end{aligned}
$$

Raíz cuadrada de una matriz

- Se puede definir

$$S = V \underbrace{\begin{pmatrix} \sqrt{d_{11}} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \sqrt{d_{nn}} \end{pmatrix}}_{D^{1/2}} V^{-1}$$

- Tal que

$$SS = (VD^{1/2}V^{-1})(VD^{1/2}V^{-1}) = VDV^{-1} = \Sigma$$

Raíz cuadrada de Cholesky de una matriz

- Definición alternativa de raíz cuadrada

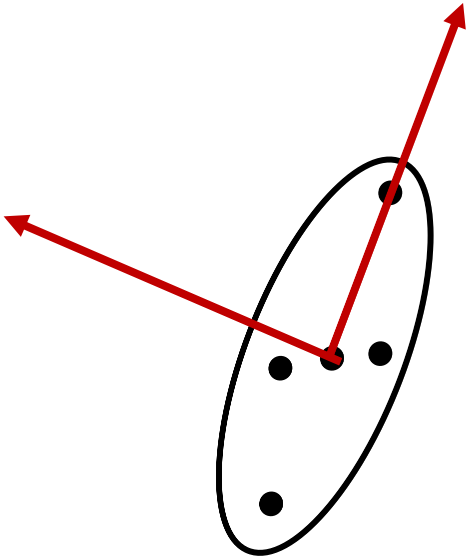
$$L \quad \text{con} \quad \Sigma = LL^T$$

- Resulta de la descomposición de Cholesky
- Numéricamente estable
- Se usa para implementar el UKF
- L y Σ tienen los mismos autovectores

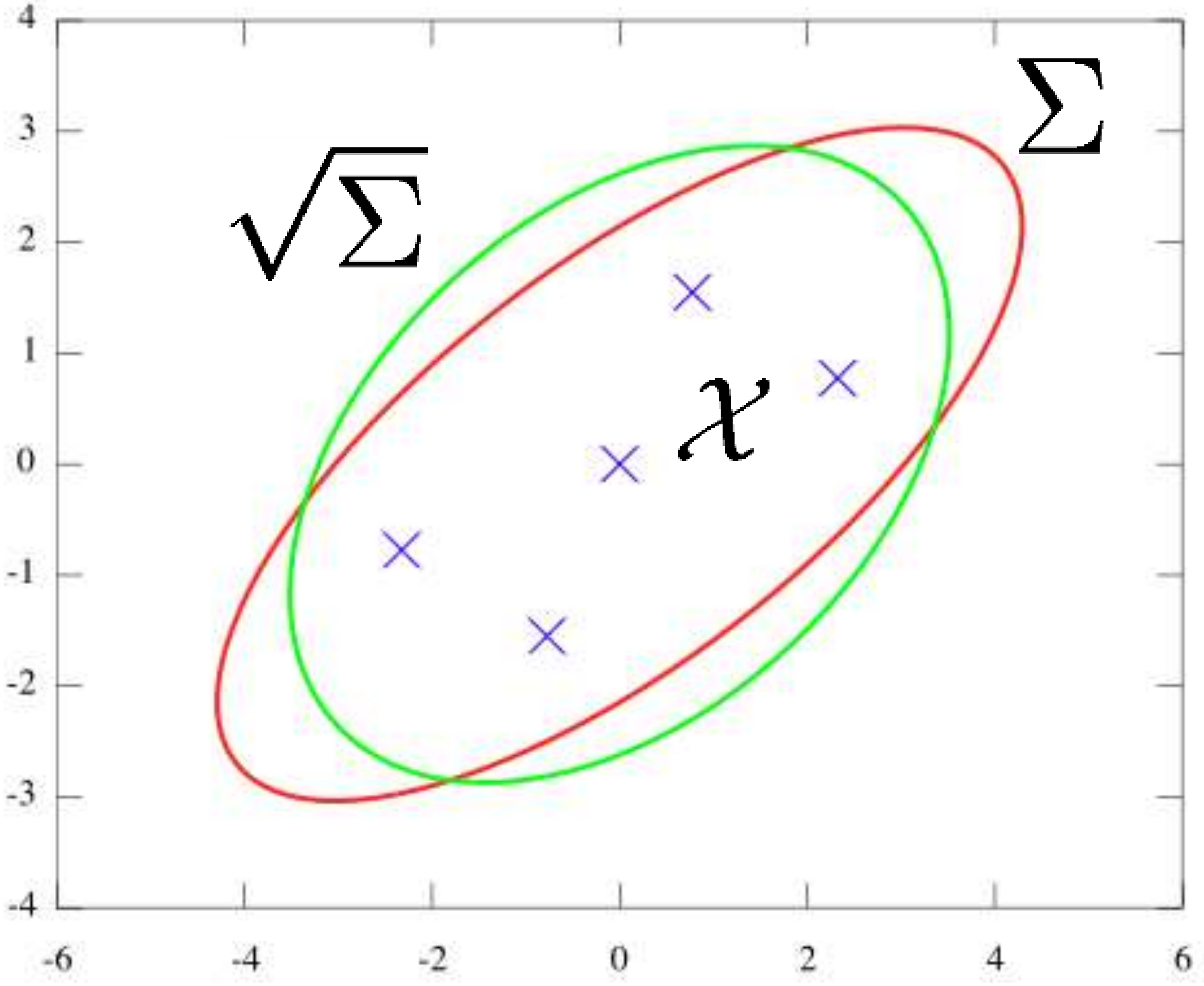
Puntos Sigma y autovectores

- Los Puntos Sigma no necesariamente están en los ejes principales de Σ

$$\begin{aligned}
\mathcal{X}^{[i]} &= \mu + \left(\sqrt{(n+\lambda)\,\Sigma}\right)_i \quad \text{for } i = 1, \ldots, n \\
\mathcal{X}^{[i]} &= \mu - \left(\sqrt{(n+\lambda)\,\Sigma}\right)_{i-n} \quad \text{for } i = n+1, \ldots, 2n
\end{aligned}$$



Puntos Sigma: Ejemplo



Pesos de los Puntos Sigma

Cálculo para la media

parámetros

$$
\begin{aligned}
w_m^{[0]} &= \frac{\lambda}{n+\lambda} \\
w_c^{[0]} &= w_m^{[0]} + (1 - \alpha^2 + \beta) \\
w_m^{[i]} = w_c^{[i]} &= \frac{1}{2(n+\lambda)} \quad \text{for } i = 1, \ldots, 2n
\end{aligned}
$$

Cálculo para la covarianza

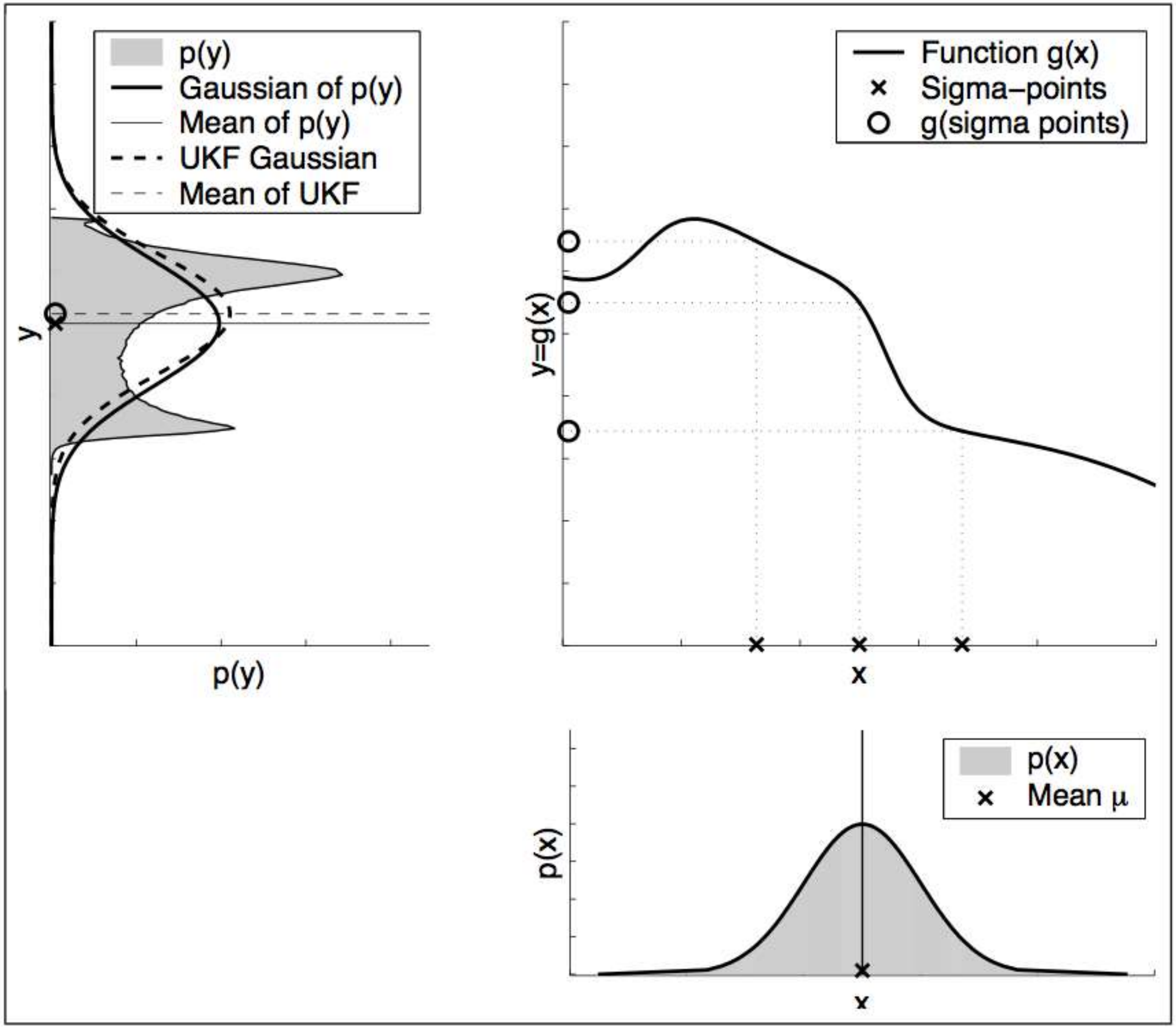
Recuperar la Gaussiana

- Calcular una Gaussiana de los **puntos transformados** y sus pesos

$$\mu' = \sum_{i=0}^{2n} w_m^{[i]}\, g(\mathcal{X}^{[i]})$$

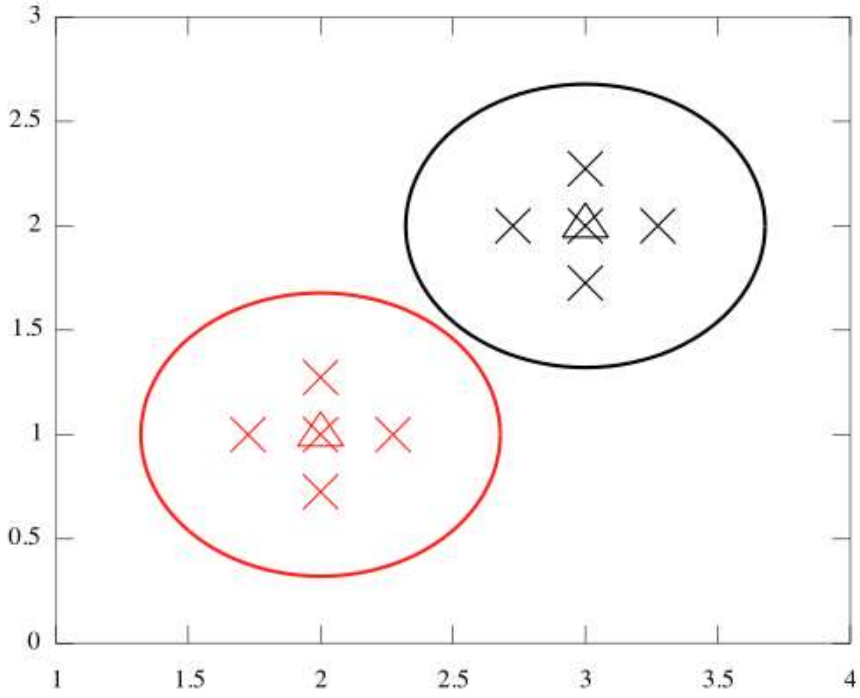
$$\Sigma' = \sum_{i=0}^{2n} w_c^{[i]}\, (g(\mathcal{X}^{[i]}) - \mu')(g(\mathcal{X}^{[i]}) - \mu')^T$$

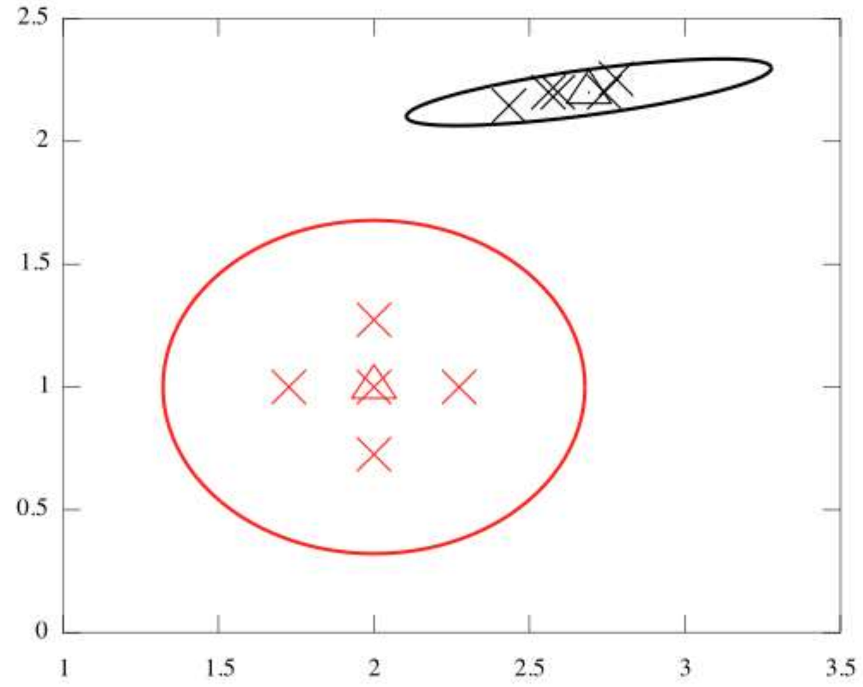
Ejemplo



p(y)
Gaussian of p(y)
Mean of p(y)
UKF Gaussian
Mean of UKF
y
p(y)
Function g(x)
Sigma–points
g(sigma points)
y=g(x)
x
p(x)
Mean μ
p(x)
x

Ejemplos





$$g((x,y)^T) = \begin{pmatrix} x+1 \\ y+1 \end{pmatrix}^T$$

$$g((x,y)^T) = \begin{pmatrix} 1+x+\sin(2x)+\cos(y) \\ 2+0.2y \end{pmatrix}^T$$

Resumen: Unscented Transform

- Puntos Sigma

$$
\begin{aligned}
\mathcal{X}^{[0]} &= \mu \\
\mathcal{X}^{[i]} &= \mu + \left(\sqrt{(n+\lambda)\,\Sigma}\right)_i \qquad \text{for } i = 1, \ldots, n \\
\mathcal{X}^{[i]} &= \mu - \left(\sqrt{(n+\lambda)\,\Sigma}\right)_{i-n} \qquad \text{for } i = n+1, \ldots, 2n
\end{aligned}
$$

- Pesos

$$
\begin{aligned}
w_m^{[0]} &= \frac{\lambda}{n+\lambda} \\
w_c^{[0]} &= w_m^{[0]} + (1 - \alpha^2 + \beta) \\
w_m^{[i]} = w_c^{[i]} &= \frac{1}{2(n+\lambda)} \qquad \text{for } i = 1, \ldots, 2n
\end{aligned}
$$

Parámetros

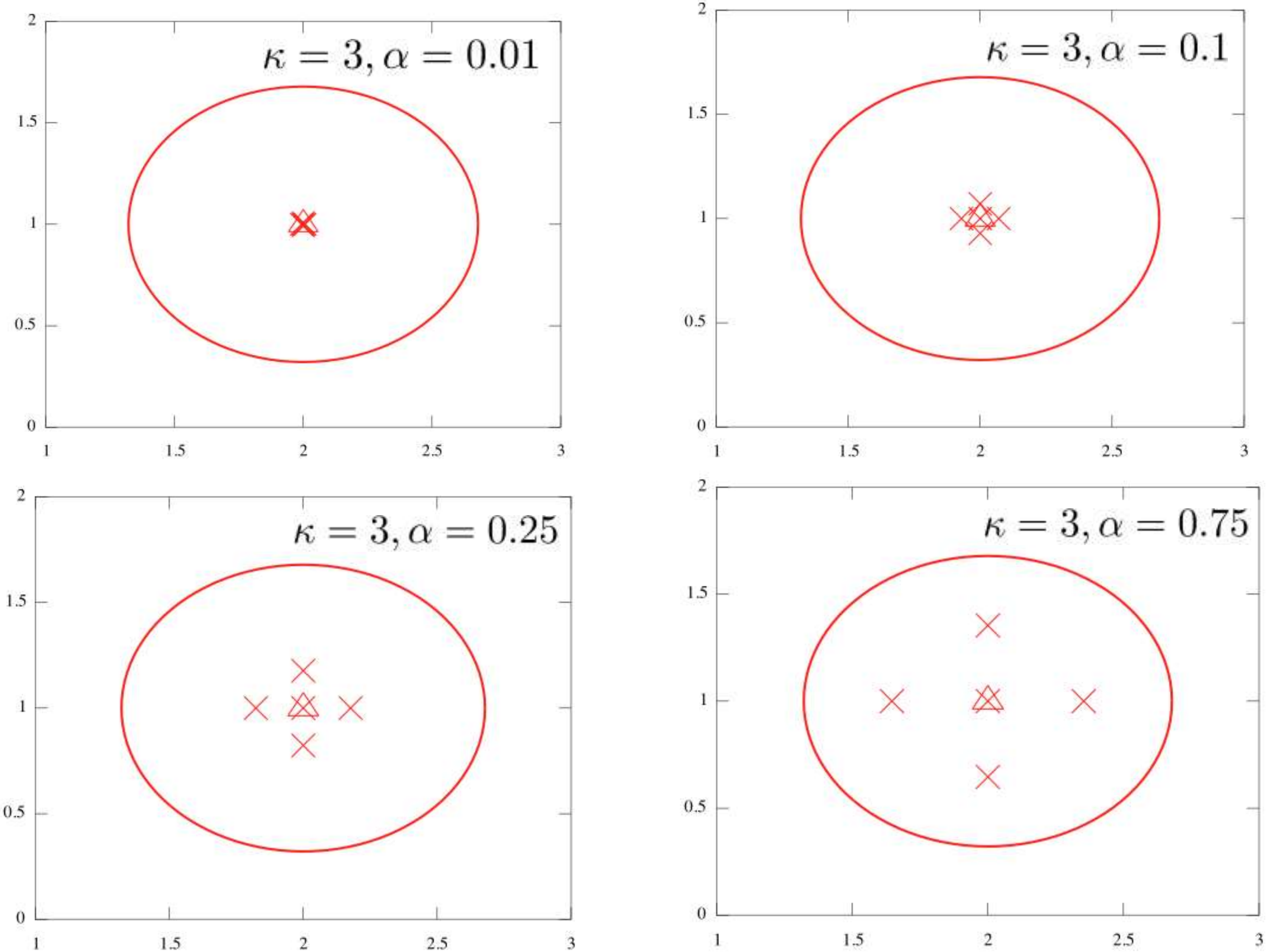
- Parámetros libres (no hay solución única)
- Comúnmente se usa:

$$
\begin{aligned}
\kappa &\geq 0 \\
\alpha &\in (0, 1] \\
\lambda &= \alpha^2(n + \kappa) - n \\
\beta &= 2
\end{aligned}
$$

Definen que tan lejos están los puntos de la media

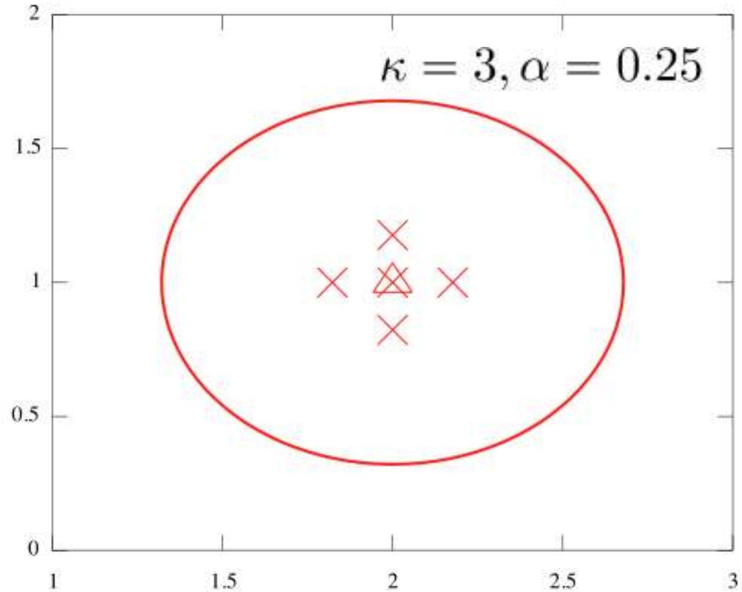
Valor óptimo para Gaussianas

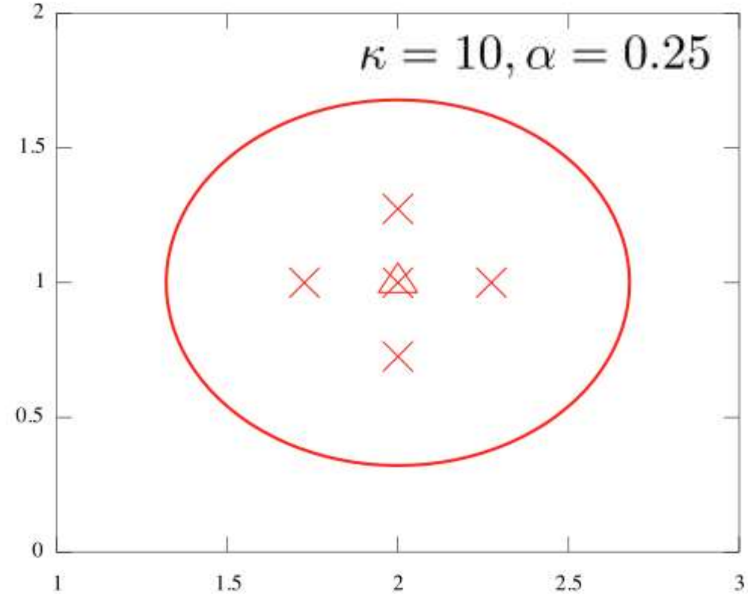
Ejemplos

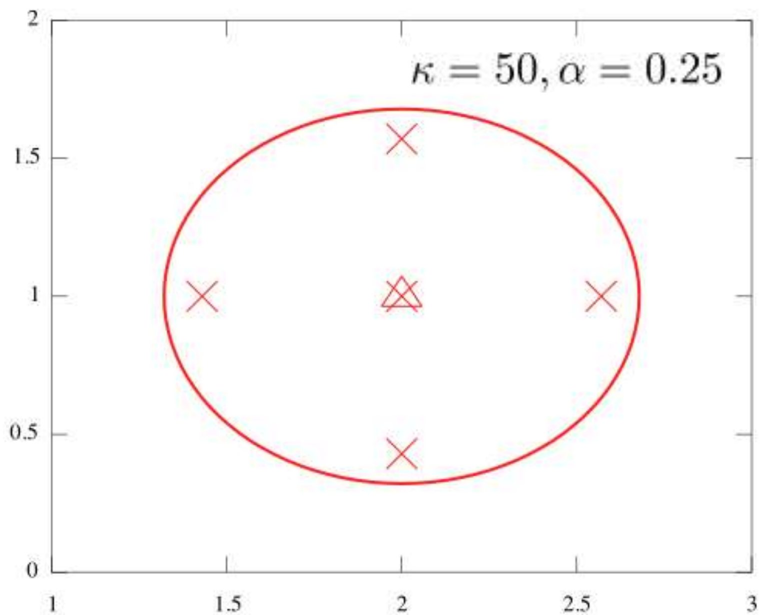


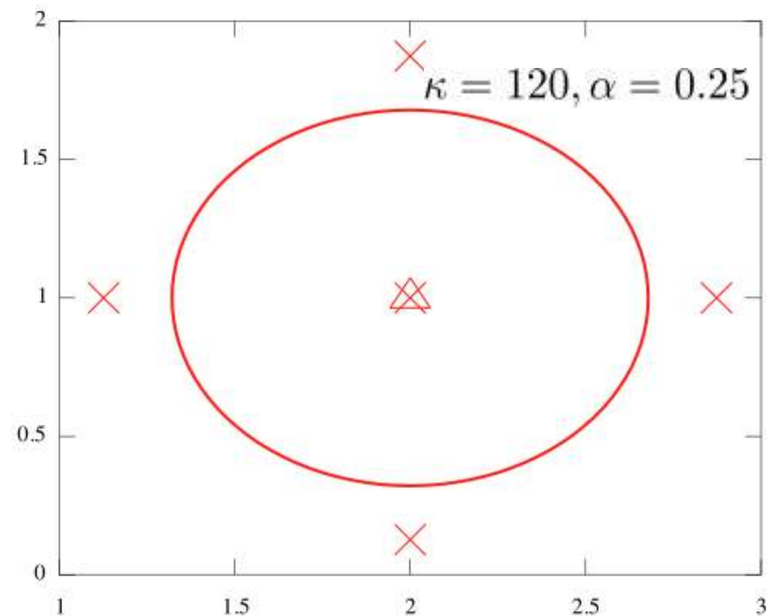
$\kappa = 3, \alpha = 0.01$
$\kappa = 3, \alpha = 0.1$
$\kappa = 3, \alpha = 0.25$
$\kappa = 3, \alpha = 0.75$

Ejemplos


$\kappa = 3, \alpha = 0.25$


$\kappa = 10, \alpha = 0.25$


$\kappa = 50, \alpha = 0.25$


$\kappa = 120, \alpha = 0.25$

Algoritmo EKF

Extended_Kalman_filter$(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t)$**:**

$$\bar{\mu}_t = g(u_t, \mu_{t-1})$$
$$\bar{\Sigma}_t = G_t\, \Sigma_{t-1}\, G_t^T + R_t$$

$$K_t = \bar{\Sigma}_t\, H_t^T (H_t\, \bar{\Sigma}_t\, H_t^T + Q_t)^{-1}$$
$$\mu_t = \bar{\mu}_t + K_t(z_t - h(\bar{\mu}_t))$$
$$\Sigma_t = (I - K_t\, H_t)\, \bar{\Sigma}_t$$

return μ_t, Σ_t

de EKF hacia UKF

Unscented

~~Extended~~_Kalman_filter$(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t)$**:**

$\bar{\mu}_t =$ Se reemplaza por la propagación

$\bar{\Sigma}_t =$ de movimiento de los Puntos Sigma

$K_t = \bar{\Sigma}_t\, H_t^T (H_t\, \bar{\Sigma}_t\, H_t^T + Q_t)^{-1}$

$\mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t))$

$\Sigma_t = (I - K_t\, H_t)\, \bar{\Sigma}_t$

return μ_t, Σ_t

Algoritmo UKF - Predicción

1: **Unscented_Kalman_filter**$(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t)$**:**

2: $\mathcal{X}_{t-1} = (\mu_{t-1} \quad \mu_{t-1} + \sqrt{(n+\lambda)\Sigma_{t-1}} \quad \mu_{t-1} - \sqrt{(n+\lambda)\Sigma_{t-1}})$

3: $\bar{\mathcal{X}}_t^* = g(u_t, \mathcal{X}_{t-1})$

4: $$\bar{\mu}_t = \sum_{i=0}^{2n} w_m^{[i]} \bar{\mathcal{X}}_t^{*[i]}$$

5: $$\bar{\Sigma}_t = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{X}}_t^{*[i]} - \bar{\mu}_t)(\bar{\mathcal{X}}_t^{*[i]} - \bar{\mu}_t)^T + R_t$$

de EKF hacia UKF

Unscented

1: $\cancel{\textbf{Extended}}$**_Kalman_filter**$(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t)$**:**

2: $\bar{\mu}_t =$ Se reemplaza por la propagación

3: $\bar{\Sigma}_t =$ de movimiento de los Puntos Sigma

4: Propagar los Puntos Sigma para la observación esperada y la ganancia del filtro

5: $\mu_t = \bar{\mu}_t + K_t(z_t - h(\bar{\mu}_t))$

6: $\Sigma_t = (I - K_t\, H_t)\, \bar{\Sigma}_t$

7: $return\ \mu_t, \Sigma_t$

Algoritmo UKF – Corrección (1)

6: $\bar{\mathcal{X}}_t = (\bar{\mu}_t \quad \bar{\mu}_t + \sqrt{(n+\lambda)\bar{\Sigma}_t} \quad \bar{\mu}_t - \sqrt{(n+\lambda)\bar{\Sigma}_t})$

7: $\bar{\mathcal{Z}}_t = h(\bar{\mathcal{X}}_t)$

8: $\hat{z}_t = \sum_{i=0}^{2n} w_m^{[i]} \bar{\mathcal{Z}}_t^{[i]}$

9: $S_t = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T + Q_t$

10: $\bar{\Sigma}_t^{x,z} = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{X}}_t^{[i]} - \bar{\mu}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T$

11: $K_t = \bar{\Sigma}_t^{x,z}\ S_t^{-1}$

$$K_t = \overbrace{\bar{\Sigma}_t\ H_t^T}^{\bar{\Sigma}_t^{x,z}} (\overbrace{H_t\ \bar{\Sigma}_t\ H_t^T + Q_t}^{S_t})^{-1}$$

Algoritmo UKF – Corrección (2)

$$\bar{\mathcal{X}}_t = (\bar{\mu}_t \quad \bar{\mu}_t + \sqrt{(n+\lambda)\bar{\Sigma}_t} \quad \bar{\mu}_t - \sqrt{(n+\lambda)\bar{\Sigma}_t})$$

$$\bar{\mathcal{Z}}_t = h(\bar{\mathcal{X}}_t)$$

$$\hat{z}_t = \sum_{i=0}^{2n} w_m^{[i]} \bar{\mathcal{Z}}_t^{[i]}$$

$$S_t = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T + Q_t$$

$$\bar{\Sigma}_t^{x,z} = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{X}}_t^{[i]} - \bar{\mu}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T$$

$$K_t = \bar{\Sigma}_t^{x,z} \; S_t^{-1}$$

$$\mu_t = \bar{\mu}_t + K_t(z_t - \hat{z}_t)$$

$$\Sigma_t = \bar{\Sigma}_t - K_t \; S_t \; K_t^T$$

$$\text{return } \mu_t, \Sigma_t$$

Algoritmo UKF – Corrección (2)

$$\bar{\mathcal{X}}_t = (\bar{\mu}_t \quad \bar{\mu}_t + \sqrt{(n+\lambda)\bar{\Sigma}_t} \quad \bar{\mu}_t - \sqrt{(n+\lambda)\bar{\Sigma}_t})$$

$$\bar{\mathcal{Z}}_t = h(\bar{\mathcal{X}}_t)$$

$$\hat{z}_t = \sum_{i=0}^{2n} w_m^{[i]} \bar{\mathcal{Z}}_t^{[i]}$$

$$S_t = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T + Q_t$$

$$\bar{\Sigma}_t^{x,z} = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{X}}_t^{[i]} - \bar{\mu}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T$$

$$K_t = \bar{\Sigma}_t^{x,z} \; S_t^{-1}$$

$$\mu_t = \bar{\mu}_t + K_t(z_t - \hat{z}_t)$$

$$\Sigma_t = \bar{\Sigma}_t - K_t \; S_t \; K_t^T$$

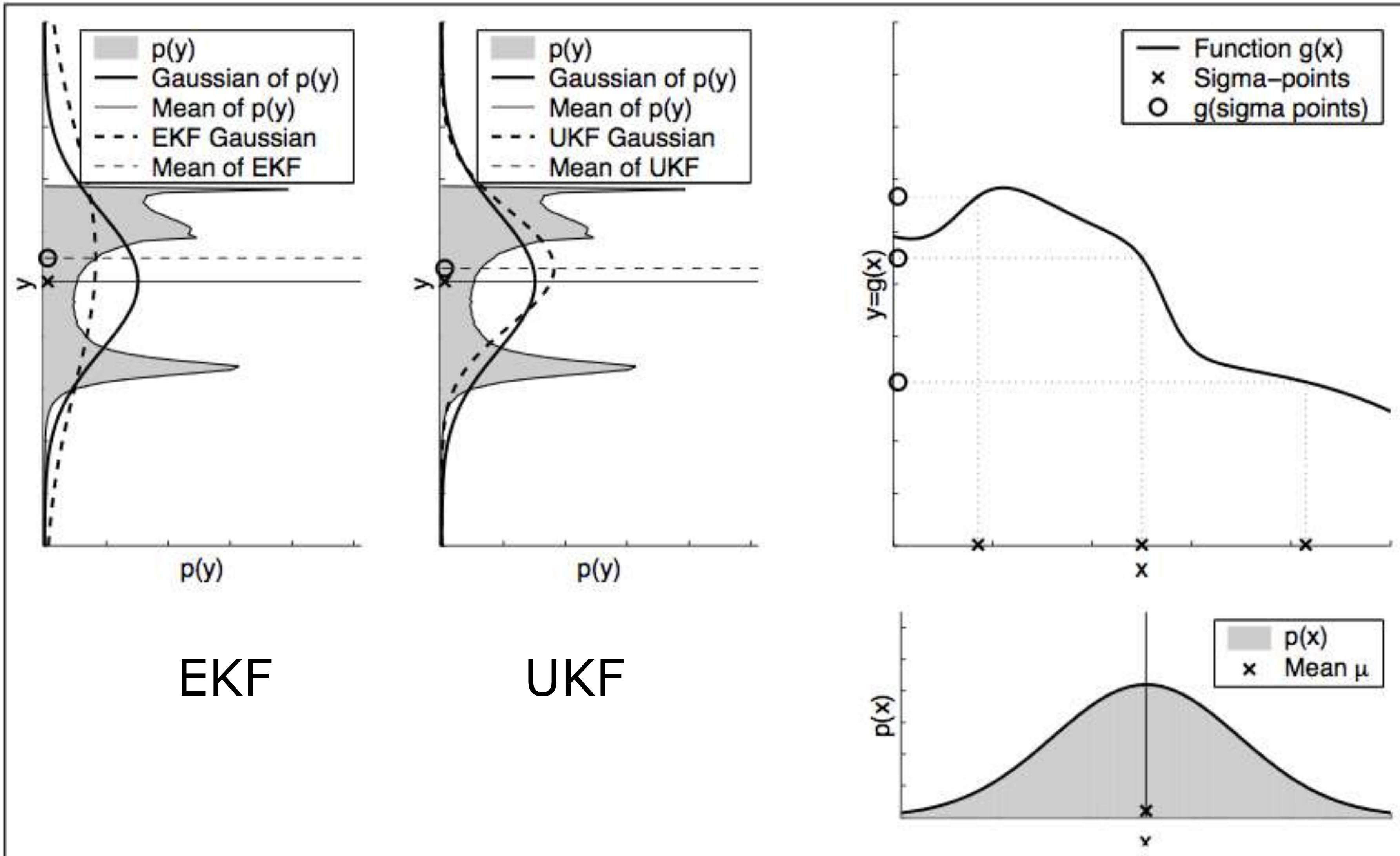
return μ_t, Σ_t

$$\begin{aligned}
\Sigma_t &= (I - K_t H_t)\bar{\Sigma}_t \\
&= \bar{\Sigma}_t - K_t H_t \bar{\Sigma}_t \\
&= \bar{\Sigma}_t - K_t \, (\Sigma^{x,z})^T \\
&= \bar{\Sigma}_t - K_t \, (\Sigma^{x,z} s_t^{-1} S_t)^T \\
&= \bar{\Sigma}_t - K_t \, (K_t S_t)^T \\
&= \bar{\Sigma}_t - K_t S_t^T K_t^T \\
&= \bar{\Sigma}_t - K_t S_t K_t^T
\end{aligned}$$

Algoritmo UKF – Covarianza

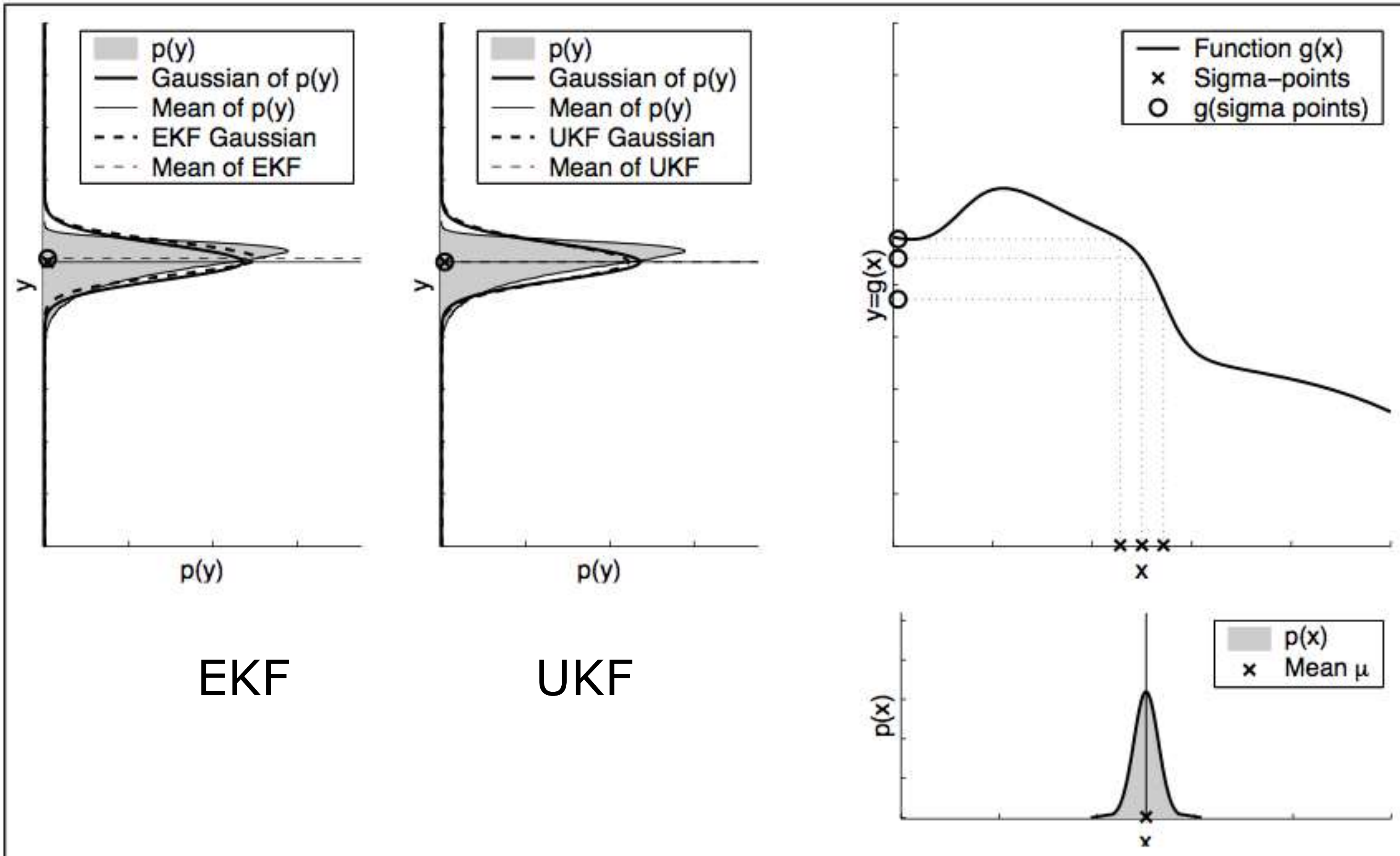
$$
\begin{aligned}
\Sigma_t &= (I - K_t H_t)\bar{\Sigma}_t \\
&= \bar{\Sigma}_t - K_t \underbrace{H_t \bar{\Sigma}_t} \\
&= \bar{\Sigma}_t - K_t \left(\bar{\Sigma}^{x,z}\right)^T \\
&= \bar{\Sigma}_t - K_t \left(\underbrace{\bar{\Sigma}^{x,z} S_t^{-1}} S_t\right)^T \\
&= \bar{\Sigma}_t - K_t \left(K_t S_t\right)^T \\
&= \bar{\Sigma}_t - K_t S_t^T K_t^T \\
&= \bar{\Sigma}_t - K_t S_t K_t^T
\end{aligned}
$$

UKF vs. EKF

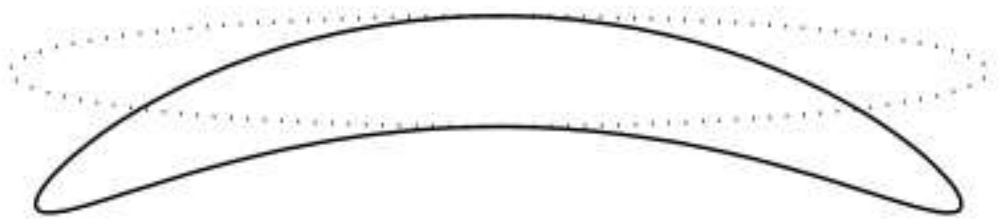


p(y)
Gaussian of p(y)
Mean of p(y)
EKF Gaussian
Mean of EKF
y
p(y)
p(y)
Gaussian of p(y)
Mean of p(y)
UKF Gaussian
Mean of UKF
y
p(y)
Function g(x)
Sigma–points
g(sigma points)
y=g(x)
x
p(x)
Mean μ
p(x)
x
EKF
UKF

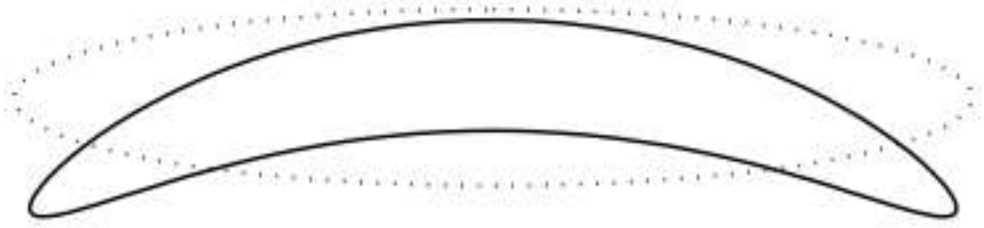
UKF vs. EKF (baja covarianza)



UKF vs. EKF (movimiento)

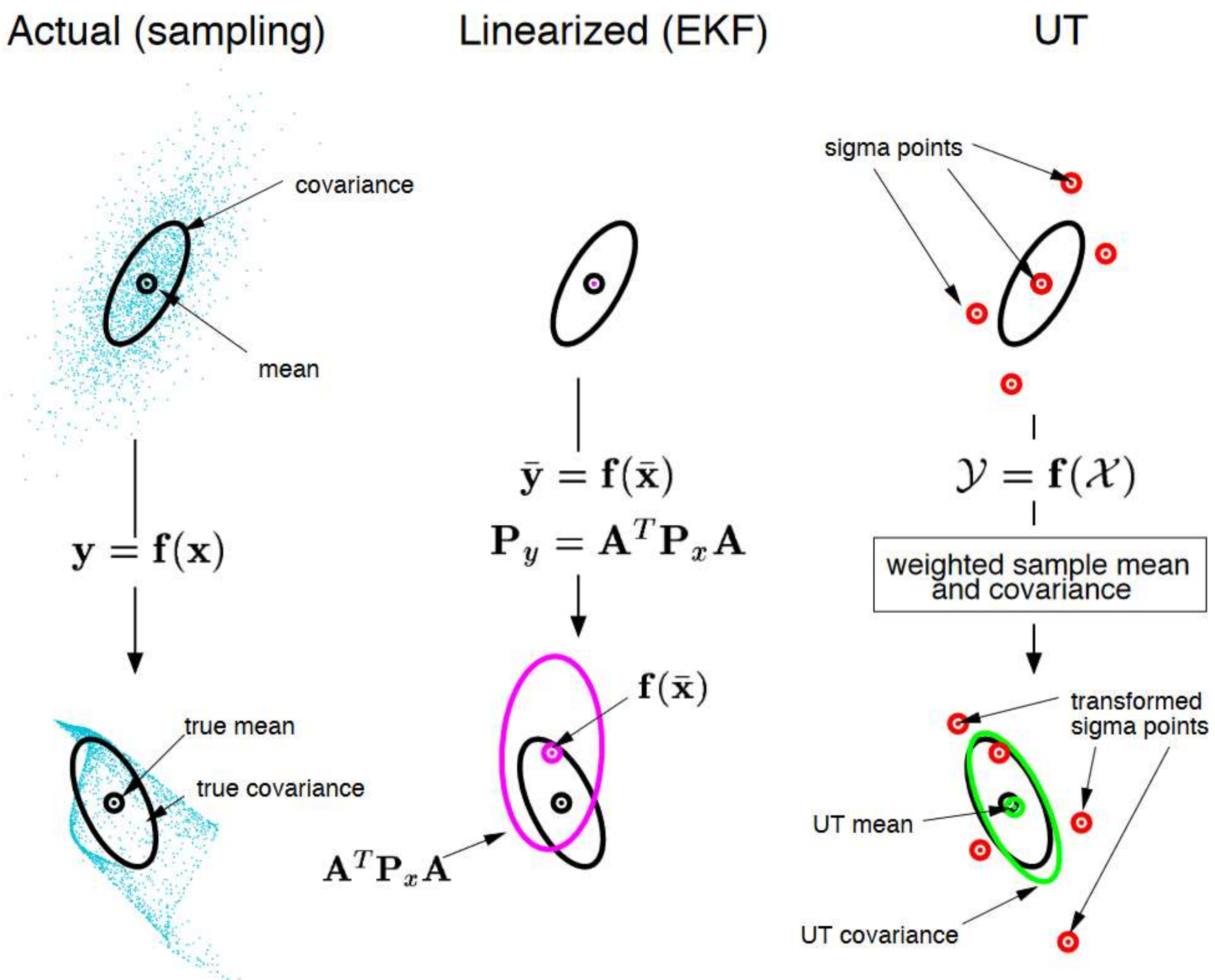


Aproximación EKF



Aproximación UKF

UKF vs. EKF



Cortesía de Wan y Van Der Merwe, "The unscented Kalman filter for nonlinear estimation"

Resumen UT y UKF

- UT es una alternativa de linealización
- UT es una mejor aproximación que la expansión de Taylor
- UT usa la propagación de Puntos Sigma
- Existen parámetros libres
- UKF usa UT en los pasos de predicción y corrección

UKF vs EKF

- Resultados iguales para modelos lineales
- UKF aproxima mejor que EKF para modelos no lineales
- En general, las diferencias son pequeñas
- No se necesita calcular Jacobianos
- Mismo orden de complejidad
- UKF es levemente más lento que EKF
- UKF también esta restringido a distribuciones Gaussianas