Speaker Localization using the Uncented Kalman Filter [1]

Sharon Gannot

School of Electrical Engineering, Bar-Ilan University

Katholieke Universiteit Leuven, 2007

[1] Joint work with Tsvi Dvorkind, Technion

Outline

1. Problem Formulation

Outline

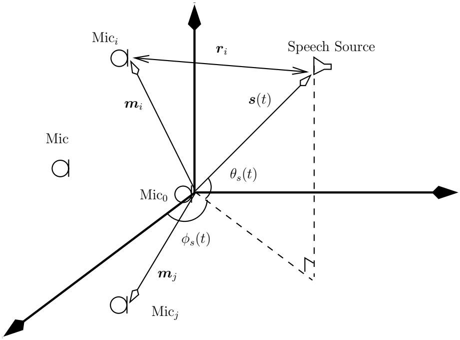
Outline

Outline

Outline

Outline

Problem Formulation



$M+1$ microphones:

$$\mathbf{m}_i^T \triangleq \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}$$
$$i = 0, \ldots, M$$
$$\mathbf{m}_0^T = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$$

Moving speaker:

$$\mathbf{s}^T(t) \triangleq \begin{bmatrix} x_s(t) & y_s(t) & z_s(t) \end{bmatrix}$$
$$\mathbf{s}_p^T(t) \triangleq \begin{bmatrix} \phi_s(t) & \theta_s(t) & \rho_s(t) \end{bmatrix}$$

Methodology

Use a Two Stage Approach

Methodology

Use a Two Stage Approach

- Measure the Time Difference of Arrival (TDOA) between the m_i and m_0: τ_i

Methodology

Use a Two Stage Approach

- Measure the Time Difference of Arrival (TDOA) between the m_i and m_0: τ_i
- Use $r_i(t) = c\tau_i = \|\mathbf{s}(t) - \mathbf{m}_i\| - \|\mathbf{s}(t)\|; \; i = 1 \ldots M$, to estimate the speaker's trajectory.

Methodology

Use a Two Stage Approach

- Measure the Time Difference of Arrival (TDOA) between the m_i and m_0: τ_i
- Use $r_i(t) = c\tau_i = \|\mathbf{s}(t) - \mathbf{m}_i\| - \|\mathbf{s}(t)\|; \; i = 1 \ldots M$, to estimate the speaker's trajectory.

TDOA Estimation

Methodology

Use a Two Stage Approach

- Measure the Time Difference of Arrival (TDOA) between the m_i and m_0: τ_i
- Use $r_i(t) = c\tau_i = \|\mathbf{s}(t) - \mathbf{m}_i\| - \|\mathbf{s}(t)\|;\ i = 1 \dots M$, to estimate the speaker's trajectory.

TDOA Estimation

- Generalized Cross-Correlation (GCC) Knapp and Carter, 1976

Methodology

Use a Two Stage Approach

- Measure the Time Difference of Arrival (TDOA) between the m_i and m_0: τ_i
- Use $r_i(t) = c\tau_i = \|\mathbf{s}(t) - \mathbf{m}_i\| - \|\mathbf{s}(t)\|;\ i = 1 \ldots M$, to estimate the speaker's trajectory.

TDOA Estimation

- Generalized Cross-Correlation (GCC) Knapp and Carter, 1976
- Eigenvalue Decomposition Benesty, 2000; Doclo and Moonen, 2003

Methodology

Use a Two Stage Approach

- Measure the Time Difference of Arrival (TDOA) between the m_i and m_0: τ_i
- Use $r_i(t) = c\tau_i = \|\mathbf{s}(t) - \mathbf{m}_i\| - \|\mathbf{s}(t)\|; \; i = 1 \ldots M$, to estimate the speaker's trajectory.

TDOA Estimation

- Generalized Cross-Correlation (GCC) Knapp and Carter, 1976
- Eigenvalue Decomposition Benesty, 2000; Doclo and Moonen, 2003
- Relative transfer function estimation using speech nonstationarity Dvorkind and Gannot, 2005

Time Difference of Arrival Estimation
Model

Received Signals

$$z_m(t) = a_m(t) * s(t) + n_m(t) \ ; \ m = 1, \ldots, M$$

$$n_m(t) = b_m(t) * n(t) \ ; \ m = 1, \ldots, M$$

Time Difference of Arrival Estimation

Model

Received Signals

$$z_m(t) = a_m(t) * s(t) + n_m(t) \; ; \; m = 1, \ldots, M$$

$$n_m(t) = b_m(t) * n(t) \; ; \; m = 1, \ldots, M$$

Relative Transfer Function (RTF)

$$\mathcal{H}(e^{j\omega}) \triangleq \frac{A_m(e^{j\omega})}{A_1(e^{j\omega})} \overset{\mathrm{DTFT}}{\Leftrightarrow} h_m(t)$$

The peak of $h_m(t)$ corresponds to the desired TDOA!

Power Spectral Density

$$\Phi_{z_i z_j}(e^{j\omega}) = A_i(e^{j\omega}){A_j}^*(e^{j\omega})\Phi_{ss}(e^{j\omega}) + B_i(e^{j\omega}){B_j}^*(e^{j\omega})\Phi_{nn}(e^{j\omega})$$

Biased Estimator

$$\Phi_{z_m z_1}(e^{j\omega}) - \mathcal{H}_m(\omega)\Phi_{z_1 z_1}(e^{j\omega}) = \Phi_{b_m^1}(e^{j\omega})$$

Bias Term:

$$\Phi_{b_m^1}(e^{j\omega}) = \left(\mathcal{G}_m(e^{j\omega}) - \mathcal{H}_m(\omega)\right)\left|B_1(e^{j\omega})\right|^2 \Phi_{nn}(e^{j\omega})$$

where $\mathcal{G}_m(e^{j\omega}) \triangleq \frac{B_m(\omega)}{B_1(\omega)}$.

Least Squares Estimation

S1 Algorithm

Exploit Speech Non-Stationarity

$$\hat{\Phi}_{z_m z_1}(n, e^{j\omega}) = \mathcal{H}_m(\omega)\hat{\Phi}_{z_1 z_1}(n, e^{j\omega}) + \Phi_{b_m^1}(e^{j\omega}) + \xi(n, e^{j\omega})$$

$$\begin{bmatrix} \hat{\mathcal{H}}_m(e^{j\omega}) \\ \hat{\Phi}_{b_m^1}(e^{j\omega}) \end{bmatrix} = \left(A^\dagger(e^{j\omega}) W A(e^{j\omega})\right)^{-1} A^\dagger(e^{j\omega}) W \hat{\mathbf{\Phi}}_{z_m z_1}(e^{j\omega})$$

with

$$A(e^{j\omega}) \triangleq \begin{bmatrix} \hat{\Phi}_{z_1 z_1}(1, e^{j\omega}), 1 \\ \vdots \\ \hat{\Phi}_{z_1 z_1}(N, e^{j\omega}), 1 \end{bmatrix}; \quad \hat{\mathbf{\Phi}}_{z_m z_1}(e^{j\omega}) \triangleq \begin{bmatrix} \hat{\Phi}_{z_m z_1}(1, e^{j\omega}) \\ \vdots \\ \hat{\Phi}_{z_m z_1}(N, e^{j\omega}) \end{bmatrix}.$$

Recursive Least Squares Estimation

RS1 Algorithm

Define, $\boldsymbol{\theta} = [\mathcal{H}_m(\omega), \Phi_{b_m^1}(e^{j\omega})]^T$, $\mathbf{a}_n^T = [\hat{\Phi}_{z_1 z_1}(n, e^{j\omega}), 1]$ and $y_n = \hat{\Phi}_{z_m z_1}(n, e^{j\omega})$

RLS Algorithm

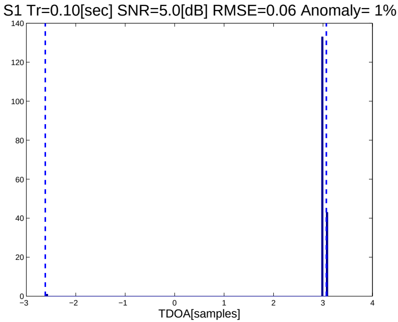
$$\mathbf{K}_n = \frac{P_{n-1}\mathbf{a}_n}{\alpha + \mathbf{a}_n^\dagger P_{n-1}\mathbf{a}_n}$$

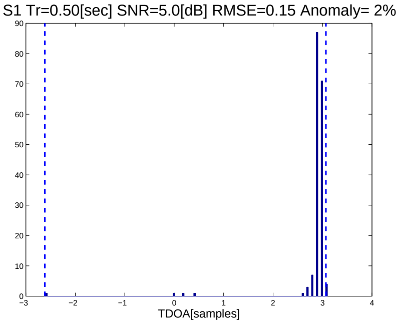
$$\hat{\boldsymbol{\theta}}(n) = \hat{\boldsymbol{\theta}}(n-1) + \mathbf{K}_n \left(y_n - \mathbf{a}_n^\dagger \hat{\boldsymbol{\theta}}(n-1) \right)$$

$$P_n = \left(\sum_{t=1}^{n} \alpha^{n-t} \mathbf{a}_t \mathbf{a}_t^\dagger \right)^{-1} = \left(P_{n-1} - \mathbf{K}_n \mathbf{a}_n^\dagger P_{n-1} \right) \frac{1}{\alpha}$$

Static Scenario - Influence of T_{60}

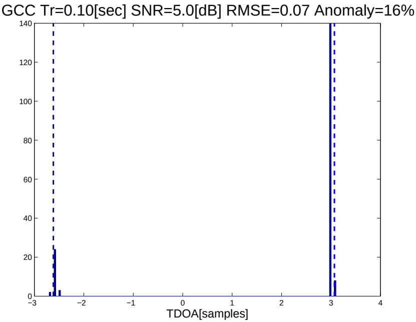
S1 Algorithm

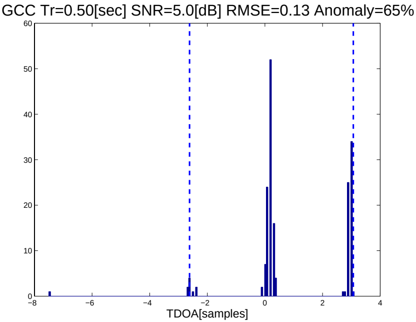




Static Scenario - Influence of T_{60}

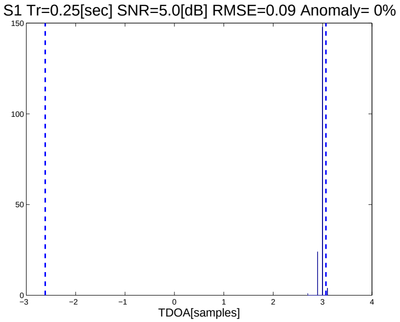
GCC Algorithm

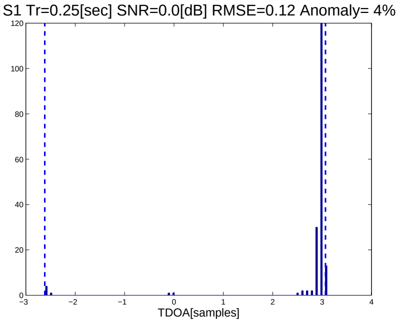




Static Scenario - Influence of SNR

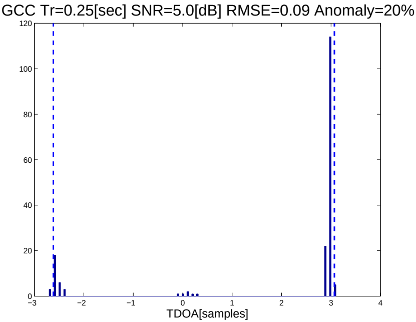
S1 Algorithm

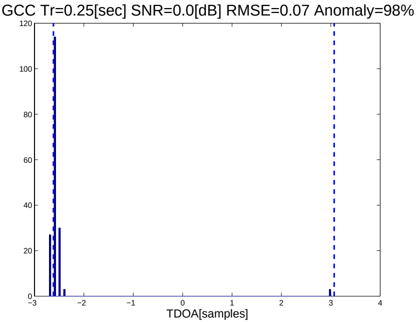




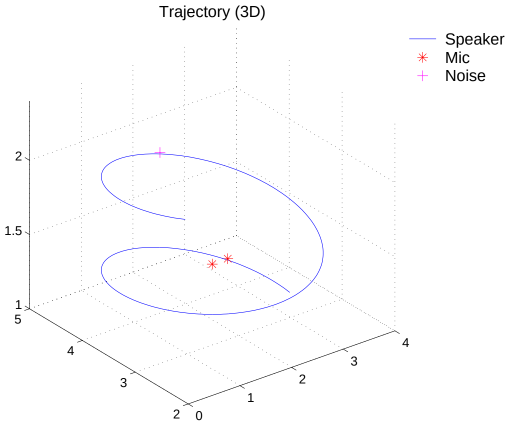
Static Scenario - Influence of SNR

GCC Algorithm





Tracking Scenario

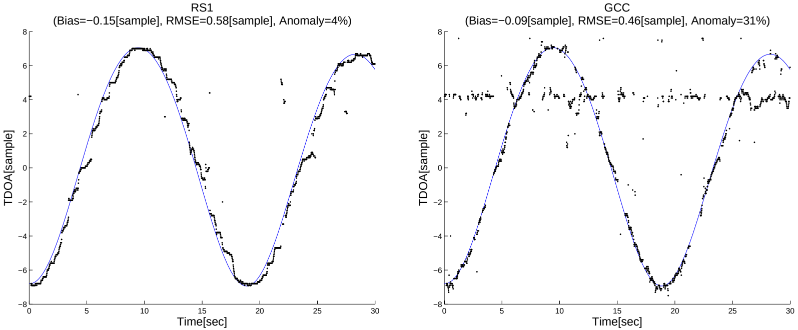


$$\mathbf{m}_1^T = \begin{bmatrix} 2 & 3.5 & 1.375 \end{bmatrix}$$
$$\mathbf{m}_2^T = \begin{bmatrix} 2.3 & 3.5 & 1.375 \end{bmatrix}$$

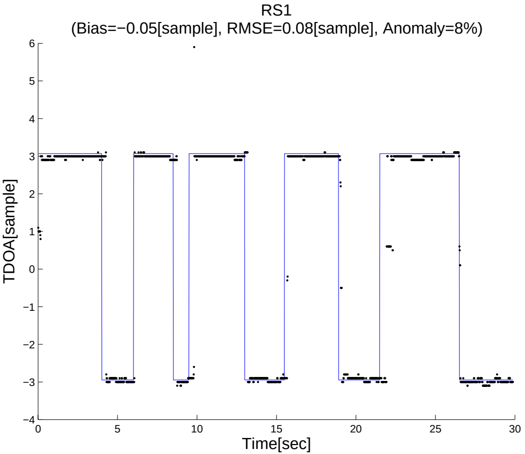
$$x_s(t) = 2 + R\cos(2\pi f t)$$
$$y_s(t) = y(t) = 3.5 + R\sin(2\pi f t)$$
$$z_s(t) = z(t) = 1 + \frac{t}{T}.$$

$R = 1.5\mathrm{m}$ $\quad T = 30\mathrm{Sec}$
$f = 0.0529\mathrm{Hz}$ $\quad T_{60} = 0.25\mathrm{Sec}$
$SNR = 10\mathrm{dB}$

Tracking Results
RS1 vs. GCC Algorithm



Switching Scenario
RS1 Algorithm



Localization

Measurement vector

$$\mathbf{r}(t) = c\boldsymbol{\tau} = \begin{bmatrix} \|\mathbf{s}(t) - \mathbf{m}_1\| - \|\mathbf{s}(t)\| \\ \vdots \\ \|\mathbf{s}(t) - \mathbf{m}_M\| - \|\mathbf{s}(t)\| \end{bmatrix} + \mathbf{v}(t) \triangleq \mathbf{h}(\mathbf{s}(t)) + \mathbf{v}(t).$$

Localization

Measurement vector

$$\mathbf{r}(t) = c\boldsymbol{\tau} = \begin{bmatrix} \|\mathbf{s}(t) - \mathbf{m}_1\| - \|\mathbf{s}(t)\| \\ \vdots \\ \|\mathbf{s}(t) - \mathbf{m}_M\| - \|\mathbf{s}(t)\| \end{bmatrix} + \mathbf{v}(t) \triangleq \mathbf{h}(\mathbf{s}(t)) + \mathbf{v}(t).$$

A nonlinear problem

Extracting $\mathbf{s}(t)$ is a nonlinear problem !

Methods

Non-Temporal Methods

Methods

Non-Temporal Methods

- Linear Correction Least-Squares (LCLS) Huang *et al.*, 2001.

Methods

Non-Temporal Methods

- Linear Correction Least-Squares (LCLS) Huang *et al.*, 2001.
- Iterative Gauss Gannot and Dvorkind, 2005

Methods

Non-Temporal Methods

- Linear Correction Least-Squares (LCLS) Huang *et al.*, 2001.
- Iterative Gauss Gannot and Dvorkind, 2005

Temporal Methods

Methods

Non-Temporal Methods

- Linear Correction Least-Squares (LCLS) Huang *et al.*, 2001.
- Iterative Gauss Gannot and Dvorkind, 2005

Temporal Methods

- Recursive Gauss Gannot and Dvorkind, 2005

Methods

Non-Temporal Methods

- Linear Correction Least-Squares (LCLS) Huang *et al.*, 2001.
- Iterative Gauss Gannot and Dvorkind, 2005

Temporal Methods

- Recursive Gauss Gannot and Dvorkind, 2005
- Bayesian methods (Kalman based).

Bayesian Methods

Simplified Dynamic Model

$$\mathbf{s}(t+1) = \Phi \mathbf{s}(t) + \mathbf{w}(t).$$

Bayesian Methods

Simplified Dynamic Model

$$\mathbf{s}(t+1) = \Phi\mathbf{s}(t) + \mathbf{w}(t).$$

How mean & covariance propagate through nonlinearities ?

Bayesian Methods

Simplified Dynamic Model

$$\mathbf{s}(t+1) = \Phi\mathbf{s}(t) + \mathbf{w}(t).$$

How mean & covariance propagate through nonlinearities ?

- Linearizing the dynamic equations
  $\Longrightarrow$ Extended Kalman filter Schmidt, 1970.

Bayesian Methods

Simplified Dynamic Model

$$\mathbf{s}(t+1) = \Phi\mathbf{s}(t) + \mathbf{w}(t).$$

How mean & covariance propagate through nonlinearities ?

- Linearizing the dynamic equations
  $\Longrightarrow$ Extended Kalman filter Schmidt, 1970.
- "Monte Carlo" method Djurić *et al.*, 2001.

Bayesian Methods

Simplified Dynamic Model

$$\mathbf{s}(t+1) = \Phi\mathbf{s}(t) + \mathbf{w}(t).$$

How mean & covariance propagate through nonlinearities ?

- Linearizing the dynamic equations
  $\Longrightarrow$ Extended Kalman filter Schmidt, 1970.
- "Monte Carlo" method Djurić *et al.*, 2001.
- Unscented transform S.J. Julier and J.K. Uhlmann, 1997.

Extended Kalman Filter

Propagation equations:

$$\hat{\mathbf{s}}(t|t-1) = \Phi\hat{\mathbf{s}}(t-1|t-1)$$
$$P(t|t-1) = \Phi P(t-1|t-1)\Phi^T + Q(t)$$

Extended Kalman Filter

Propagation equations:

$$\hat{\mathbf{s}}(t|t-1) = \Phi\hat{\mathbf{s}}(t-1|t-1)$$
$$P(t|t-1) = \Phi P(t-1|t-1)\Phi^T + Q(t)$$

Kalman gain:

$$K(t) = P(t|t-1)H^T(t)\left(H(t)P(t|t-1)H^T(t) + R(t)\right)^{-1}$$

Extended Kalman Filter (cont.

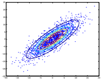
Update equations:

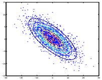
$$\hat{\mathbf{s}}(t|t) = \hat{\mathbf{s}}(t|t-1) + K(t)\left(\mathbf{r}(t) - \mathbf{h}(\hat{\mathbf{s}}(t|t-1))\right)$$

$$H(t) \triangleq \nabla_{\mathbf{s}(t)}\mathbf{h}(\hat{\mathbf{s}}(t|t-1)) = \begin{bmatrix} \left(\frac{\hat{\mathbf{s}}(t|t-1)-\mathbf{m}_1}{\|\hat{\mathbf{s}}(t|t-1)-\mathbf{m}_1\|} - \frac{\hat{\mathbf{s}}(t|t-1)}{\|\hat{\mathbf{s}}(t|t-1)\|}\right)^T \\ \vdots \\ \left(\frac{\hat{\mathbf{s}}(t|t-1)-\mathbf{m}_M}{\|\hat{\mathbf{s}}(t|t-1)-\mathbf{m}_M\|} - \frac{\hat{\mathbf{s}}(t|t-1)}{\|\hat{\mathbf{s}}(t|t-1)\|}\right)^T \end{bmatrix}$$

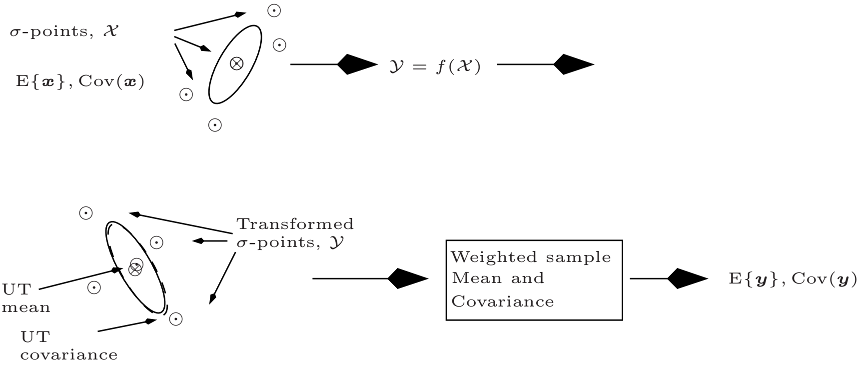
$$P(t|t) = \left(I - K(t)H(t)\right)P(t|t-1)$$

"Monte Carlo" Propagation

Random Number Generator $f_{\boldsymbol{X}}(\boldsymbol{x})$ →  → Ensemble Mean & Covariance → $\mathrm{E}\{\boldsymbol{x}\}, \mathrm{Cov}(\boldsymbol{x})$

$\boldsymbol{y}_i = f(\boldsymbol{x}_i)$, $i = 0, 1, \ldots$ →  → Ensemble Mean & Covariance → $\mathrm{E}\{\boldsymbol{y}\}, \mathrm{Cov}(\boldsymbol{y})$

Unscented Transform



Unscented Transform (cont.)

Calculate σ-points

$$\mathcal{X}_0 = \bar{\mathbf{x}}$$

$$\mathcal{X}_l = \bar{\mathbf{x}} + \left(\sqrt{(L+\lambda)P_{xx}}\right)_l ; \; l = 1, \ldots, L$$

$$\mathcal{X}_{l+L} = \bar{\mathbf{x}} - \left(\sqrt{(L+\lambda)P_{xx}}\right)_l ; \; l = 1, \ldots, L$$

Unscented Transform (cont.)

Calculate σ-points

$$\mathcal{X}_0 = \bar{\mathbf{x}}$$

$$\mathcal{X}_l = \bar{\mathbf{x}} + \left(\sqrt{(L+\lambda)P_{xx}}\right)_l ; \; l = 1, \ldots, L$$

$$\mathcal{X}_{l+L} = \bar{\mathbf{x}} - \left(\sqrt{(L+\lambda)P_{xx}}\right)_l ; \; l = 1, \ldots, L$$

Calculate Weights

$$W_0^{(m)} = \lambda/(L+\lambda)$$

$$W_0^{(c)} = \lambda/(L+\lambda) + (1 - \alpha^2 + \beta)$$

$$W_l^{(m)} = W_l^{(c)} = 1/2(L+\lambda); \; l = 1, 2, \ldots, 2L$$

Unscented Transform (cont.)

Summary

Unscented Transform (cont.)

Summary

1. Construct **x** σ-points: $\mathcal{X}_l, \ l = 0, \ldots, 2L$.

Unscented Transform (cont.)

Summary

1. Construct **x** σ-points: $\mathcal{X}_l, \ l = 0, \ldots, 2L$.
2. Transform each point to the respective **y** σ-points: $\mathcal{Y}_l = f\left(\mathcal{X}_l\right), \ l = 0, \ldots, 2L$.

Unscented Transform (cont.)

Summary

1. Construct $\mathbf{x}$ σ-points: $\mathcal{X}_l, \ l = 0, \ldots, 2L$.
2. Transform each point to the respective $\mathbf{y}$ σ-points: $\mathcal{Y}_l = f\left(\mathcal{X}_l\right), \ l = 0, \ldots, 2L$.
3. Use weighted averaging, $\bar{\mathbf{y}} \approx \sum_{l=0}^{2L} W_l^{(m)} \mathcal{Y}_l$ to estimate $\mathbf{y}$ mean.

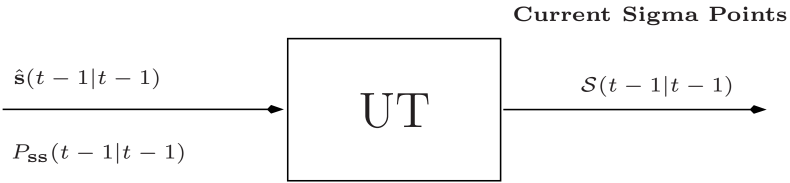
Unscented Transform (cont.)

Summary

1. Construct $\mathbf{x}$ σ-points: $\mathcal{X}_l,\ l = 0, \ldots, 2L$.
2. Transform each point to the respective $\mathbf{y}$ σ-points: $\mathcal{Y}_l = f\left(\mathcal{X}_l\right),\ l = 0, \ldots, 2L$.
3. Use weighted averaging, $\bar{\mathbf{y}} \approx \sum_{l=0}^{2L} W_l^{(m)} \mathcal{Y}_l$ to estimate $\mathbf{y}$ mean.
4. Use weighted outer product, $P_{yy} \approx \sum_{l=0}^{2L} W_l^{(c)} \left(\mathcal{Y}_l - \bar{\mathbf{y}}\right)\left(\mathcal{Y}_l - \bar{\mathbf{y}}\right)^T$ to estimate $\mathbf{y}$ covariance and $P_{xy} \approx \sum_{l=0}^{2L} W_l^{(c)} \left(\mathcal{X}_l - \bar{\mathbf{x}}\right)\left(\mathcal{Y}_l - \bar{\mathbf{y}}\right)^T$ to estimate the cross-covariance between $\mathbf{x}$ and $\mathbf{y}$.

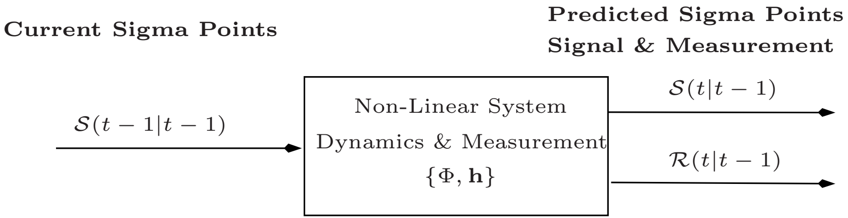
The Unscented Kalman Filter (UKF)

Unscented Transform



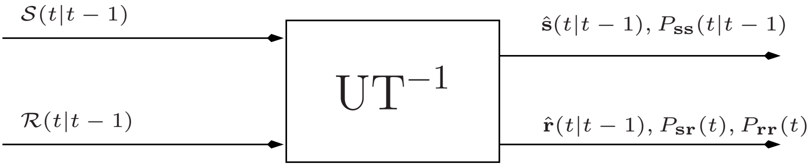
The Unscented Kalman Filter (UKF)

Propagation Stage



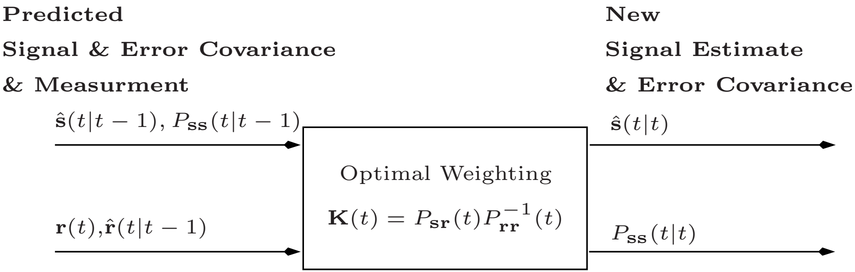
The Unscented Kalman Filter (UKF)

Inverse Unscented Transform

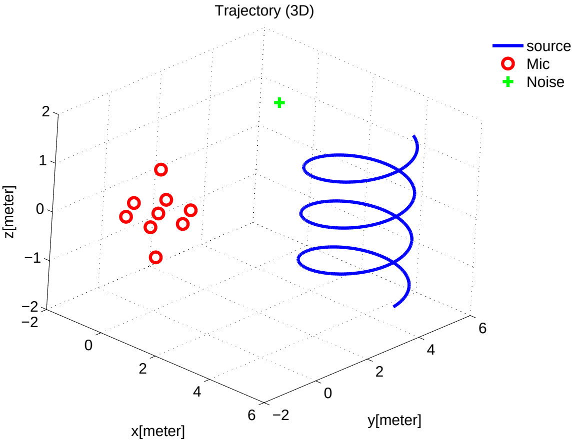


The Unscented Kalman Filter (UKF)

Update Stage



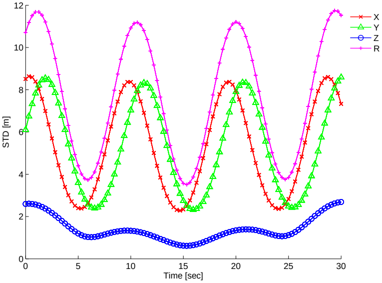
Test Scenario

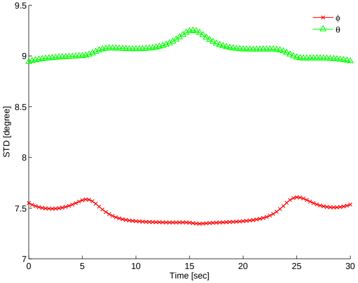


$$
\begin{aligned}
\mathbf{m}_1^T &= \begin{bmatrix}0.9 & 0 & 0\end{bmatrix}\\
\mathbf{m}_2^T &= \begin{bmatrix}0.45 & 0.7794 & 0\end{bmatrix}\\
\mathbf{m}_3^T &= \begin{bmatrix}-0.45 & 0.7794 & 0\end{bmatrix}\\
\mathbf{m}_4^T &= \begin{bmatrix}-0.9 & 0 & 0\end{bmatrix}\\
\mathbf{m}_5^T &= \begin{bmatrix}-0.45 & -0.7794 & 0\end{bmatrix}\\
\mathbf{m}_6^T &= \begin{bmatrix}0.45 & -0.7794 & 0\end{bmatrix}\\
\mathbf{m}_7^T &= \begin{bmatrix}0 & 0 & 0.9\end{bmatrix}\\
\mathbf{m}_8^T &= \begin{bmatrix}0 & 0 & -0.9\end{bmatrix}.
\end{aligned}
$$

$$
\begin{aligned}
x_s(t) &= R(\cos(\tfrac{t}{R}) + 2.5)\\
y_s(t) &= R(\sin(\tfrac{t}{R}) + 2.5)\\
z_s(t) &= \tfrac{t}{10} - 1.5.
\end{aligned}
$$

Cramér Rao Lower Bound



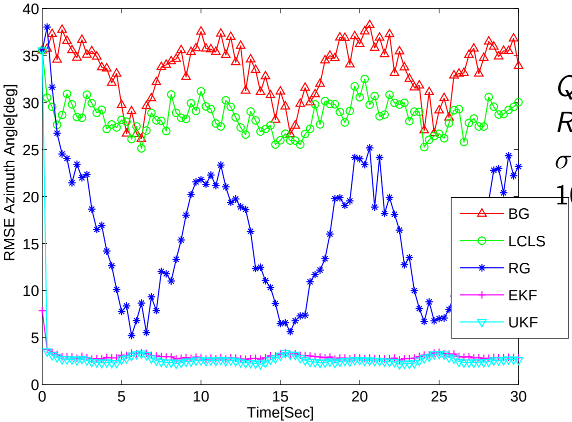


Conclusions

Azimuth and Elevation angles can be better estimated than the Cartesian coordinates and the Distance.

Tracking Scenario

Gaussian Noise



$Q(t) = 0.5^2 I;$

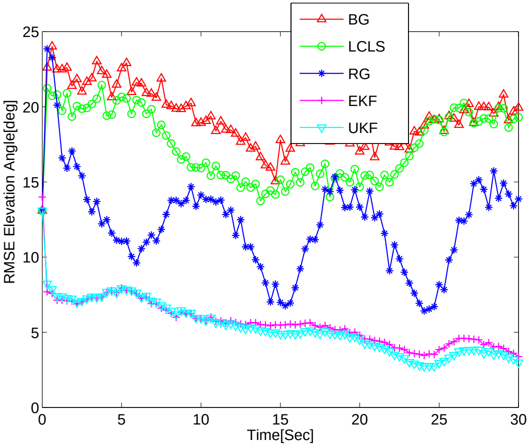
$R(t) = 10\sigma_v^2 I;$

$\sigma_v = 0.2\text{m};$

1000 Monte-Carlo trials

Tracking Scenario

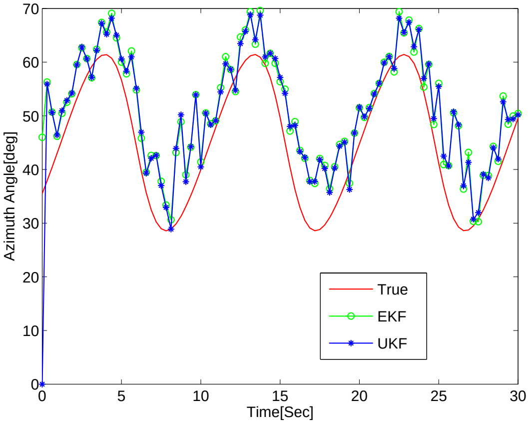
Gaussian Noise and Anomalies



$Q(t) = 0.5^2 I$; $R(t) = 10^{-2} I$; ... = 0.2m; 1000 Monte Carlo trials

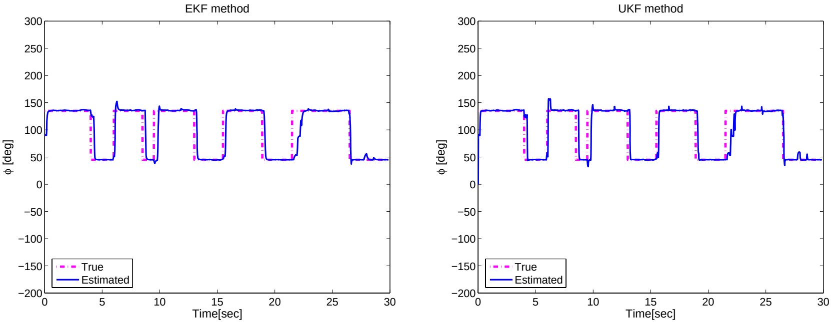
Tracking Scenario

Typical Realization



$$Q(t) = 0.5^2 I$$
$$R(t) = 10\sigma_v^2 I$$
$$\sigma_v = 0.2\text{m}$$

Switching Scenario



Two sources:

$\left[\ \phi = \frac{\pi}{4}\text{rad} \quad \theta = \frac{\pi}{4}\text{rad} \quad R = 1.5\text{m}\ \right]\ \&\ \left[\ \phi = \frac{3\pi}{4}\text{rad} \quad \theta = \frac{\pi}{3}\text{rad} \quad R = 1.5\text{m}\ \right].$

Summary

- Two stage approach for speaker localization.

Summary

- Two stage approach for speaker localization.
- Polar coordinates are better estimated than Cartesian coordinates.

Summary

- Two stage approach for speaker localization.
- Polar coordinates are better estimated than Cartesian coordinates.
- Temporal methods outperforms non-temporal methods.

Summary

- Two stage approach for speaker localization.
- Polar coordinates are better estimated than Cartesian coordinates.
- Temporal methods outperforms non-temporal methods.
- Advantage of Bayesian methods even with naive propagation scheme.

Summary

- Two stage approach for speaker localization.
- Polar coordinates are better estimated than Cartesian coordinates.
- Temporal methods outperforms non-temporal methods.
- Advantage of Bayesian methods even with naive propagation scheme.
- EKF and UKF have comparable performance (and computational complexity).