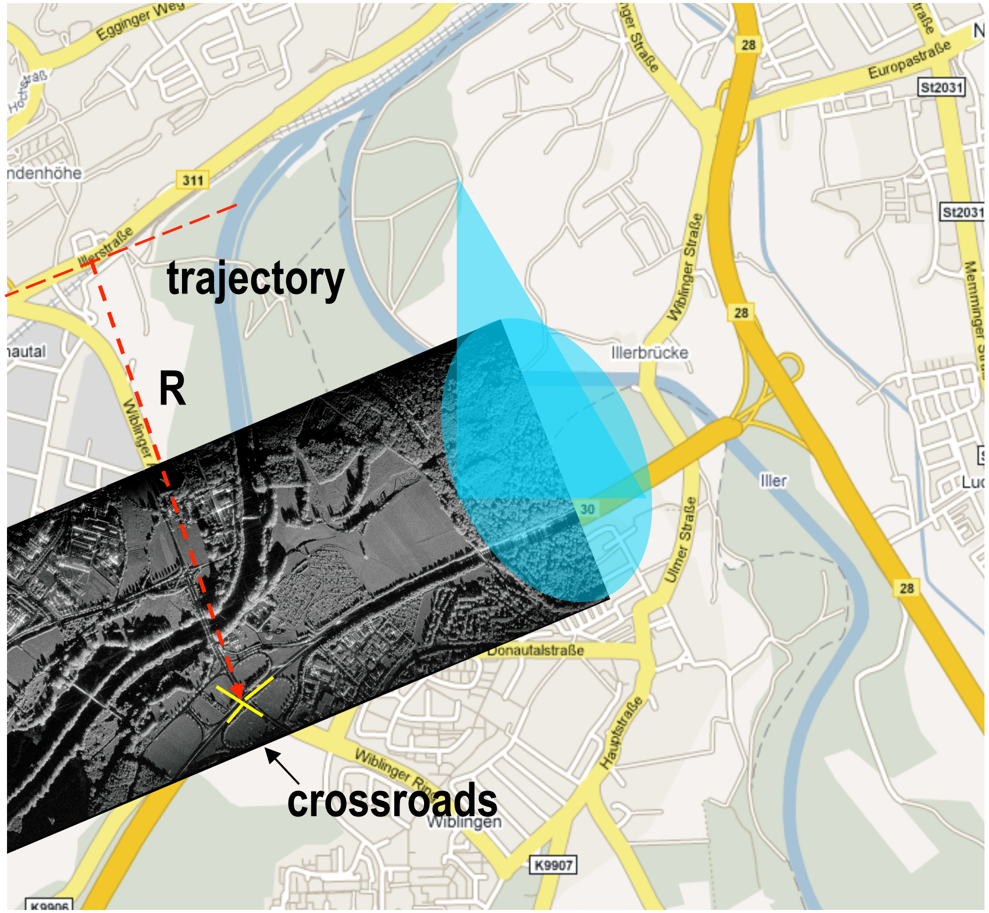
Nonlinear SAR/INS Integration using Sigma-Point Kalman Filter

Andreas Maier, Stefan Kiesel and Gert F. Trommer

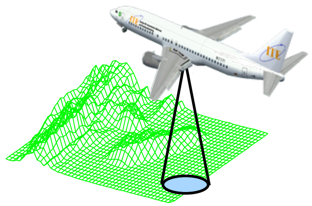
Outline

- Objectives
- SAR/INS System Overview
- Synthetic Aperture Radar
- SAR/INS Integration
- Simulation Results
- Conclusion



Comparison of sensor characteristics

- IMU
  - short term
  - autonomous



- TRN
  - long term
  - autonomous
  - rough terrain

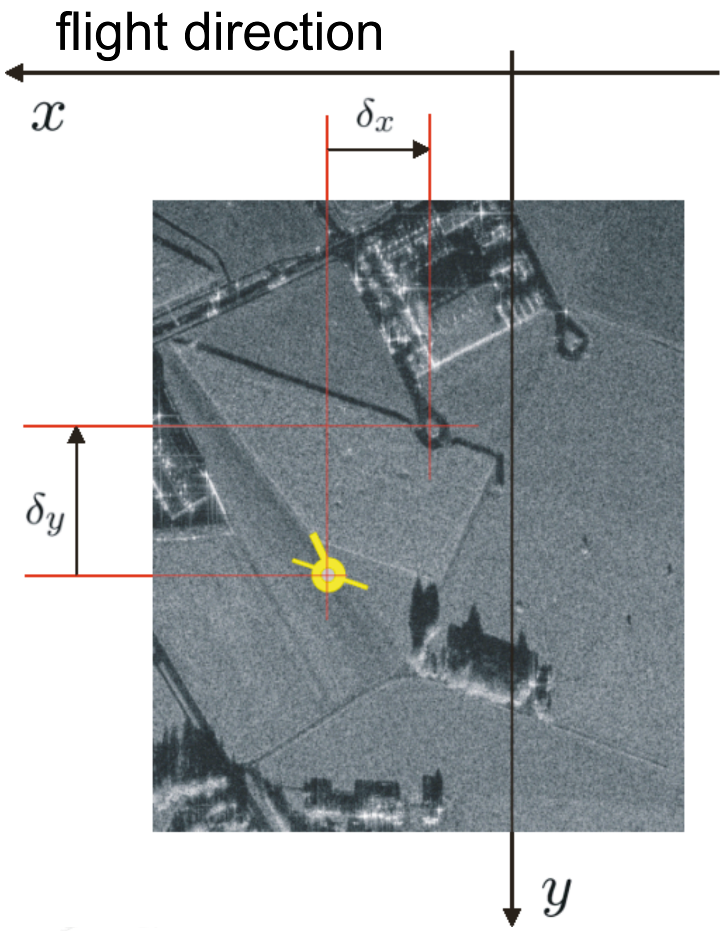
- GPS
  - long term
  - nonautonomous

- SAR (feature based)
  - long term
  - autonomous

Objectives

- Implementation of Sigma-point Kalman filter for SAR/INS integration
- SAR/INS position accuracy analysis
- Investigation of required feature update rates
- SAR/INS in combination with
  - Baromeric altimeter
  - Terrain Referenced Navigation (TRN)

Feature Displacement



- ## Features

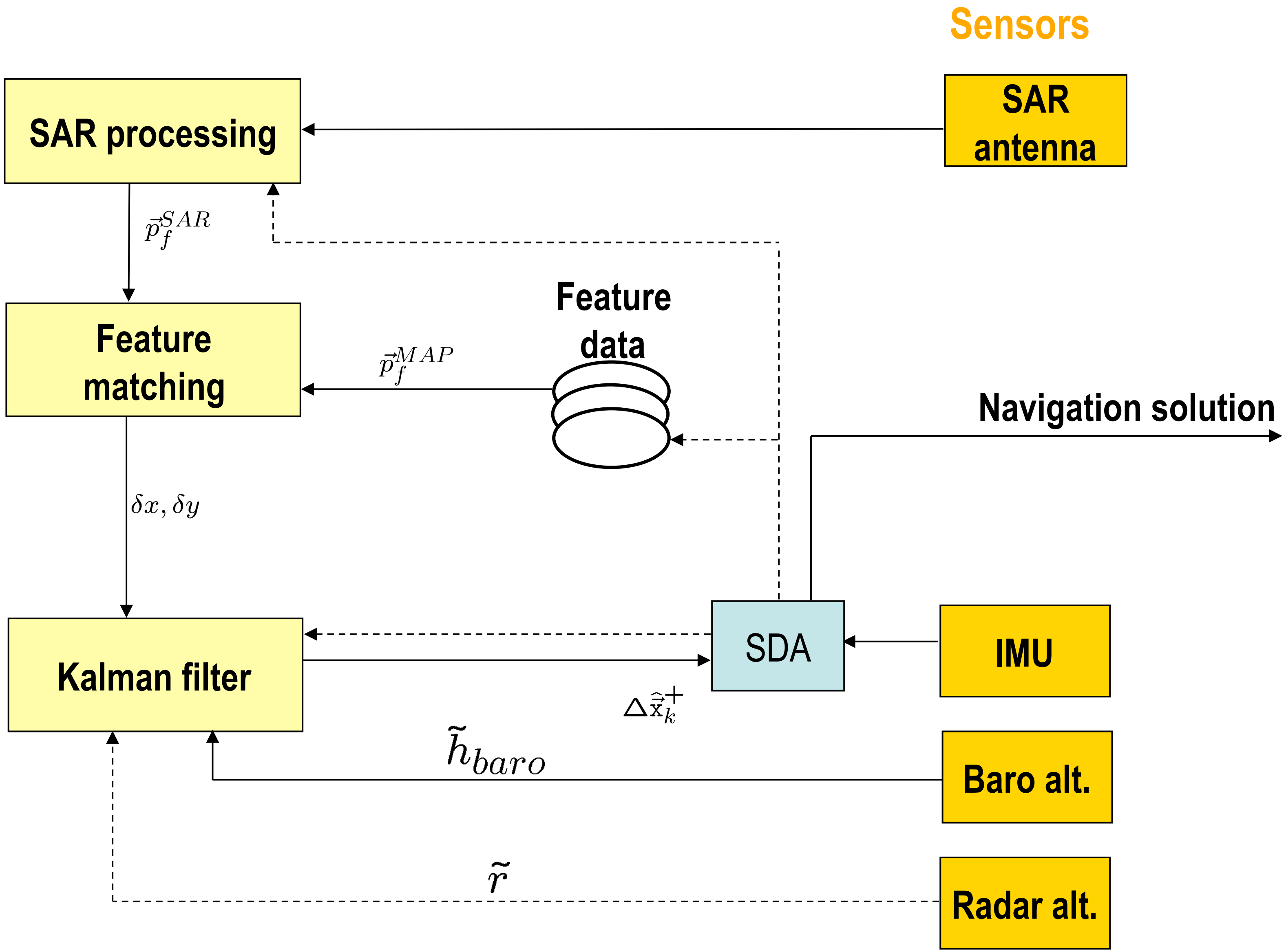
  Unambiguous and well visible e. g.
  - Crossroads
  - Courses of rivers

- ## Feature displacement

  Displacement between imaged feature and map feature is used for navigation update

$$\vec{\delta} = \begin{pmatrix} \delta_x \\ \delta_y \end{pmatrix} = \begin{pmatrix} p_{f,x}^{SAR} - p_{f,x}^{MAP} \\ p_{f,y}^{SAR} - p_{f,y}^{MAP} \end{pmatrix}$$

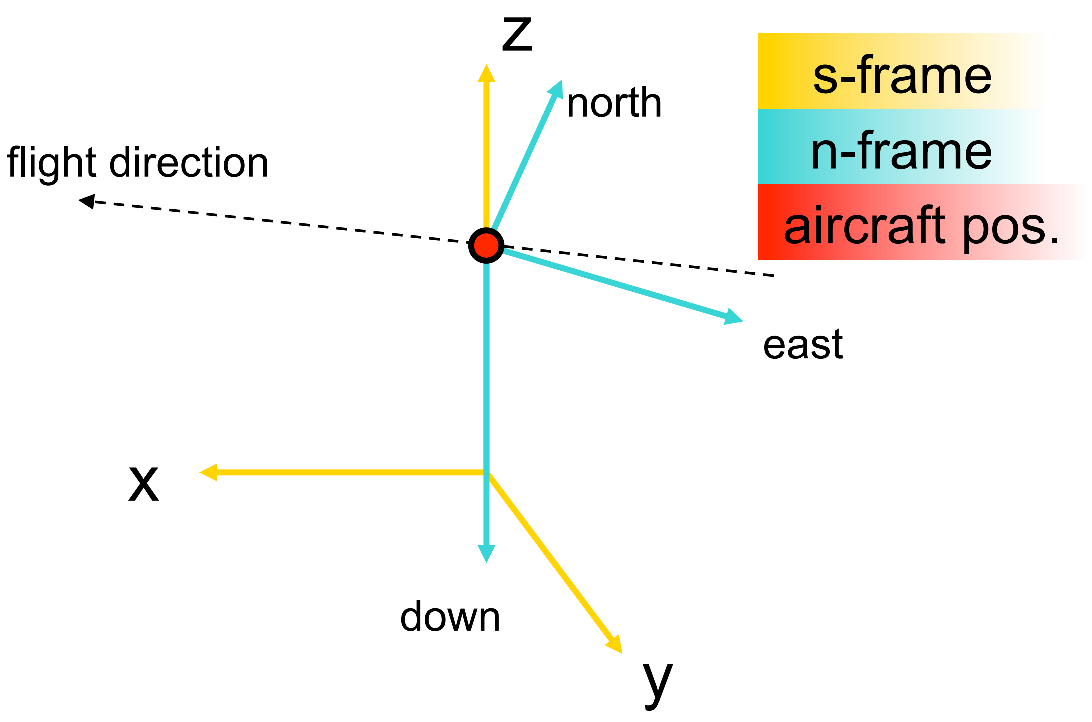
System Overview



Measurement:
Displacement δ of the SAR image with respect to map feature

$$\begin{pmatrix} \delta_x \\ \delta_y \end{pmatrix} = \begin{pmatrix} p_{f,x}^{SAR} - p_{f,x}^{MAP} \\ p_{f,y}^{SAR} - p_{f,y}^{MAP} \end{pmatrix}$$

SAR Sensor



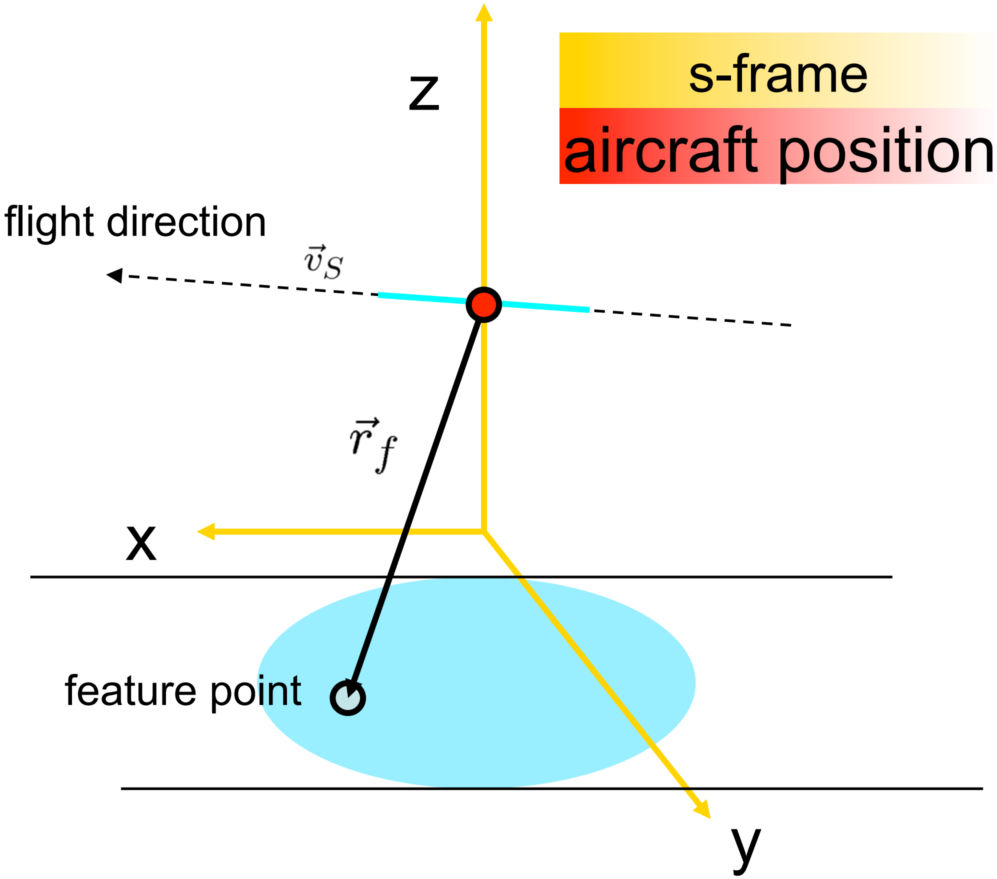
position: $\vec{\hat{p}}^S = \begin{pmatrix} 0 & 0 & \hat{p}_z \end{pmatrix}$

velocity: $\vec{\hat{v}}^S = \begin{pmatrix} \hat{v}_x & 0 & \hat{v}_z \end{pmatrix}^T = C_N^S \vec{\hat{v}}^N$

- Definition of sensor coordinates
  - x in flight direction
  - z upwards
  - y forms a right handed coordinate system
  - Origin is located at ground level
- Transformation matrix
  - Transformation from n to s-frame by rotation around down-axis

$$C_N^S = \begin{pmatrix} \cos(\Psi_v) & \sin(\psi_v) & 0 \\ \sin(\Psi_v) & -\cos(\psi_v) & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

SAR Sensor



$$f_d = \frac{\vec{r}_f}{|\vec{r}_f|} \cdot \frac{2 \cdot \vec{v}^S}{\lambda} \qquad R = |\vec{r}_f|$$

- Information of each reflection point form range and Doppler frequency
- SAR-Processing forms image in xy coordinates

$$\vec{p}_f^{SAR} = \vec{f}\left(f_d, R, \hat{\vec{p}}, \hat{\vec{v}}\right)$$

Displacement

- Map feature has to be transformed into s-frame coordinates
- Nonlinear measurement equation depending on all position and velocity components

Map feature:

$$\vec{p}_f^{\,MAP} = C_N^S \cdot \begin{pmatrix} (\phi_f - \widehat{\phi}) \cdot R_n \\ (\lambda_f - \widehat{\lambda}) \cdot (R_e \cdot \cos(\widehat{\phi})) \\ 0 \end{pmatrix}$$

Measurement equation:

$$\vec{\delta} = \begin{pmatrix} \delta_x \\ \delta_y \end{pmatrix} = \begin{pmatrix} p_{f,x}^{SAR} - p_{f,x}^{MAP} \\ p_{f,y}^{SAR} - p_{f,y}^{MAP} \end{pmatrix} = \begin{pmatrix} h_x(\vec{p}, \vec{v}) \\ h_y(\vec{p}, \vec{v}) \end{pmatrix}$$

Sigma-point Kalman filter

- 15-State SPKF has been implemented
- Measurement noise includes matching errors and map errors
- Sigma-point Kalman filter takes into account higher order terms automatically
- Provides more accurate update in case of nonlinear measurement models

State vector and measurement noise

$$\vec{\mathrm{x}} = \left(\ \Delta\vec{p}^{N}, \Delta\vec{v}^{N}, \Delta\vec{\Psi}, \Delta\vec{b}_a, \Delta\vec{b}_\omega\ \right)$$

$$\vec{\nu} = \left(\nu_{a,x}, \nu_{a,y}, \nu_{f,x}, \nu_{f,y}, \nu_{f,z}\right)$$

Augmented state vector construction

$$\hat{\vec{\mathrm{x}}}^{a} = E[\vec{\mathrm{x}}^{a}] = (\ \hat{\vec{\mathrm{x}}}^{T}\hat{\vec{\nu}}^{T}\ )^{T}$$

$$\mathbf{P^a} = E[(\hat{\vec{\mathrm{x}}}^{a} - \bar{\vec{\mathrm{x}}}^{a})\cdot(\hat{\vec{\mathrm{x}}}^{a} - \bar{\vec{\mathrm{x}}}^{a})^{T}] = \begin{pmatrix} \mathbf{P} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{pmatrix}$$

Sigma-Point Kalman Filter

$$\vec{\chi}_0^a = \hat{\vec{x}}^a$$
$$\vec{\chi}_i^a = \hat{\vec{x}}^a + \zeta\sqrt{\mathbf{P}^a}_i, \qquad i = 1...L$$
$$\vec{\chi}_{i+L}^a = \hat{\vec{x}}^a - \zeta\sqrt{\mathbf{P}^a}_i, \qquad i = 1...L$$

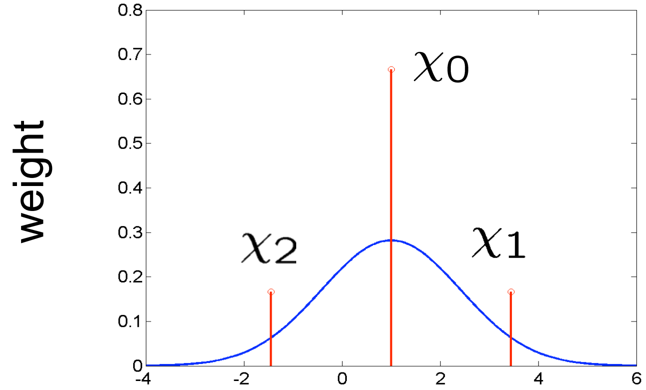
$$\vec{Y}_i = \vec{h}(\vec{\chi}_i^a)$$

$$\hat{\delta} = \sum_{i=0}^{2L} w_i^m \vec{Y}_i$$

$$\mathbf{P_{yy}} = \sum_{i=0}^{2L}\sum_{j=0}^{2L} w_{ij}^c(\vec{Y}_i - \hat{\delta})(\vec{Y}_j - \hat{\delta})$$

$$\mathbf{P_{xy}} = \sum_{i=0}^{2L}\sum_{j=0}^{2L} w_{ij}^c(\vec{\chi}_i - \hat{\vec{x}})(\vec{Y}_j - \hat{\delta})$$

- **Processing steps**
  - Produce sigma-points
  - Transform by measurement equation
  - Calculate **mean**, **covariance** and **correlation**
- **Example of sigma-points**



Sigma-Point Kalman Filter

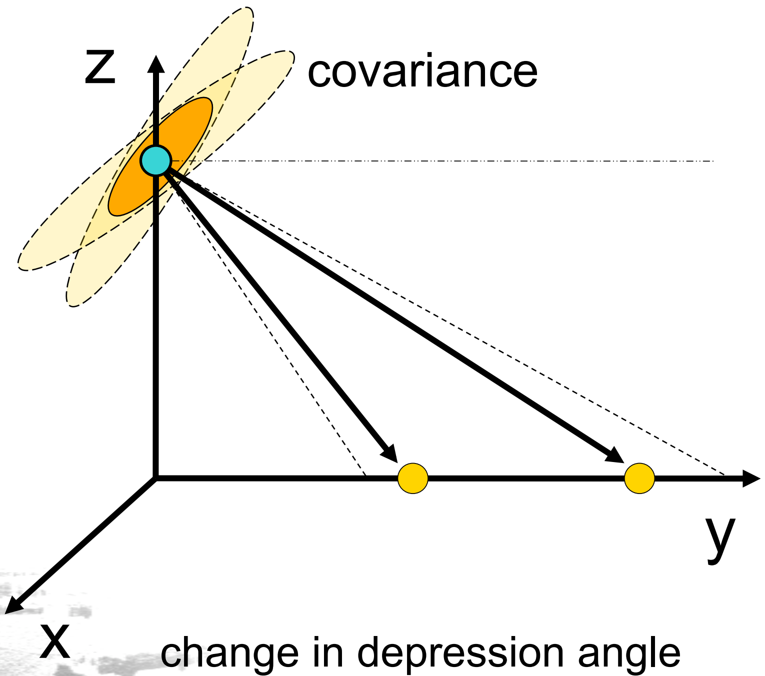
- Sigma-point Kalman filtering is analogous to EKF-processing
  - Calculate gain matrix
  - Calculate navigation error
  - Calculate new covariance matrix
- Correlation and covariance accurate to the second order term

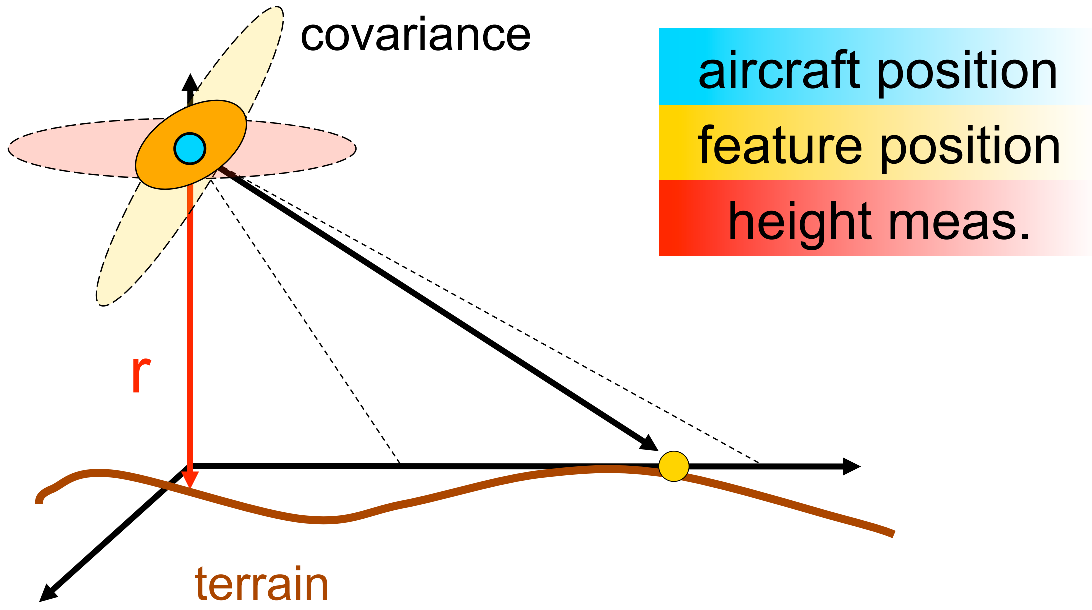
SPKF		EKF
$$\left.\begin{aligned} \mathbf{K} &= \mathbf{P_{xy}}\mathbf{P_{yy}}^{-1} \\ \hat{\vec{\mathbf{x}}}^{a,+} &= \hat{\vec{\mathbf{x}}}^{a,-} + \mathbf{K}(\delta - \hat{\delta}) \\ \mathbf{P}^{+} &= \mathbf{P}^{-} - \mathbf{K}\mathbf{P_{yy}}\mathbf{K}^{T} \end{aligned}\right\}$$	$\hat{=}$	$$\left\{\begin{aligned} \mathbf{K} &= \mathbf{P}\mathbf{H}^{T} \cdot (\mathbf{H}\mathbf{P}\mathbf{H}^{T} + \mathbf{R})^{-1} \\ \hat{\vec{x}}^{+} &= \hat{\vec{x}}^{-} + \mathbf{K} \cdot (\delta - \hat{\delta}) \\ \mathbf{P}^{+} &= \mathbf{P}^{-} - \mathbf{K}\mathbf{H}\mathbf{P}^{-} \end{aligned}\right.$$

Update step

- Update by SAR measurement
  - 2 measurement equations
- Independent measurements
  - Change in depression angle

- Different sensor characteristics
  - TRN update in regions of rough terrain
  - SAR update during flight over smooth terrain



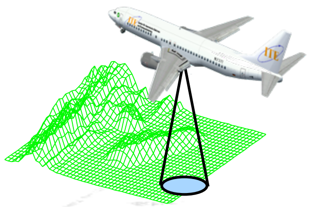


Simulations

Sensor accuracies

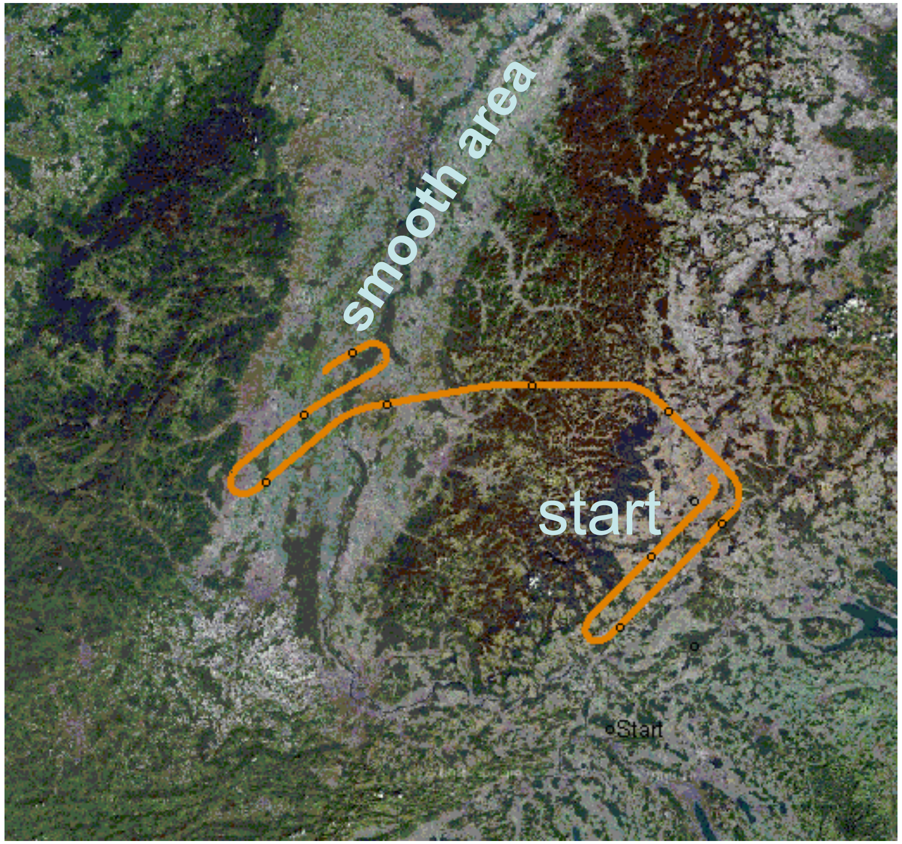
Navigation grade IMU

Map error standard deviation: 3m
Matching error standard deviation: 7m



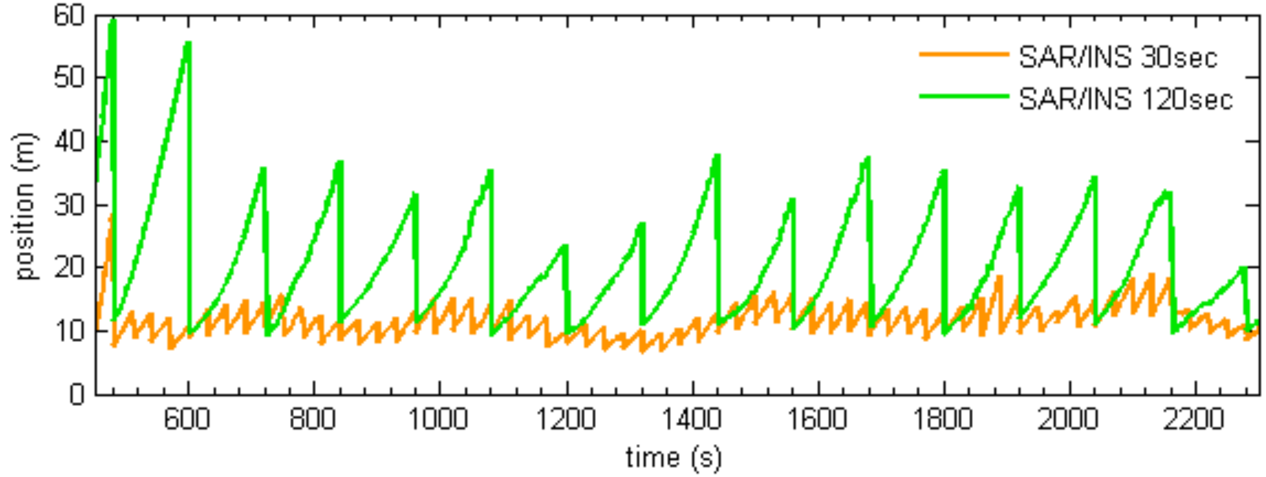
Measurement noise 2%
DTED level 1, 100m spacing

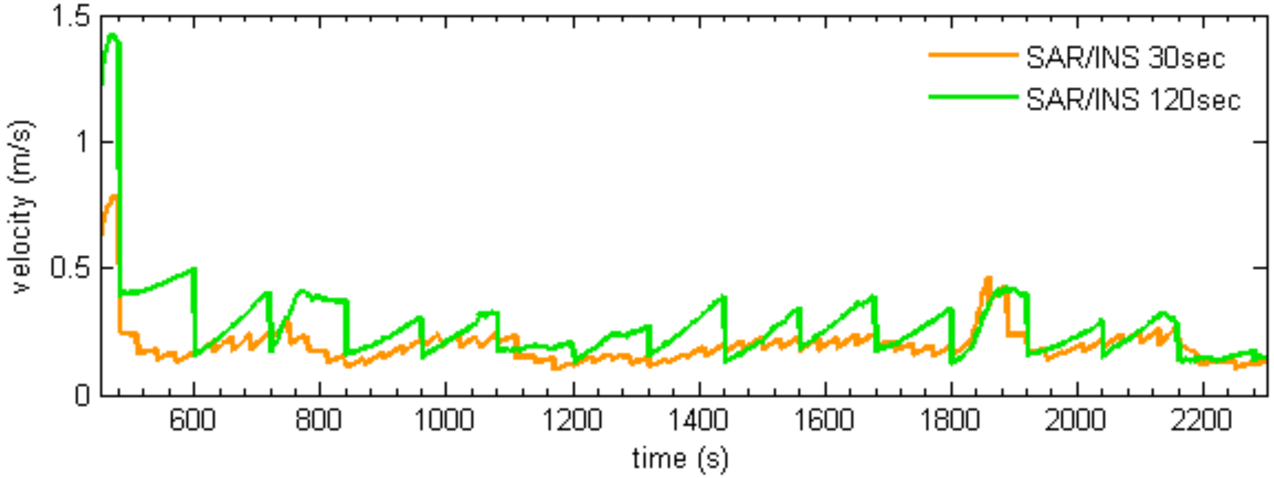
Trajectory



SAR/INS

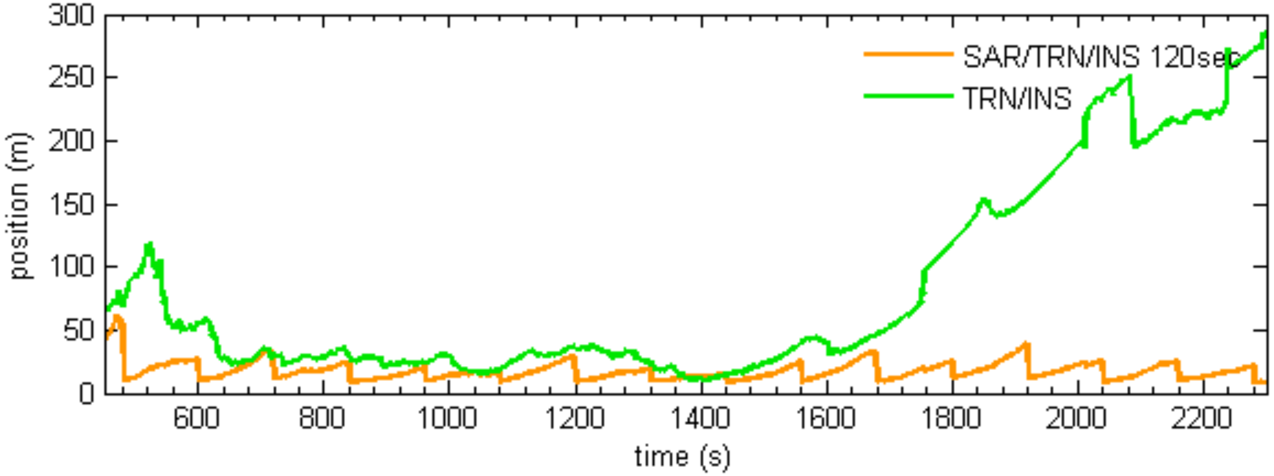
- Barometric altimeter aids height estimation
- Navigation error depends on feature update rate
- Feature updates in the scale of a few minutes needed

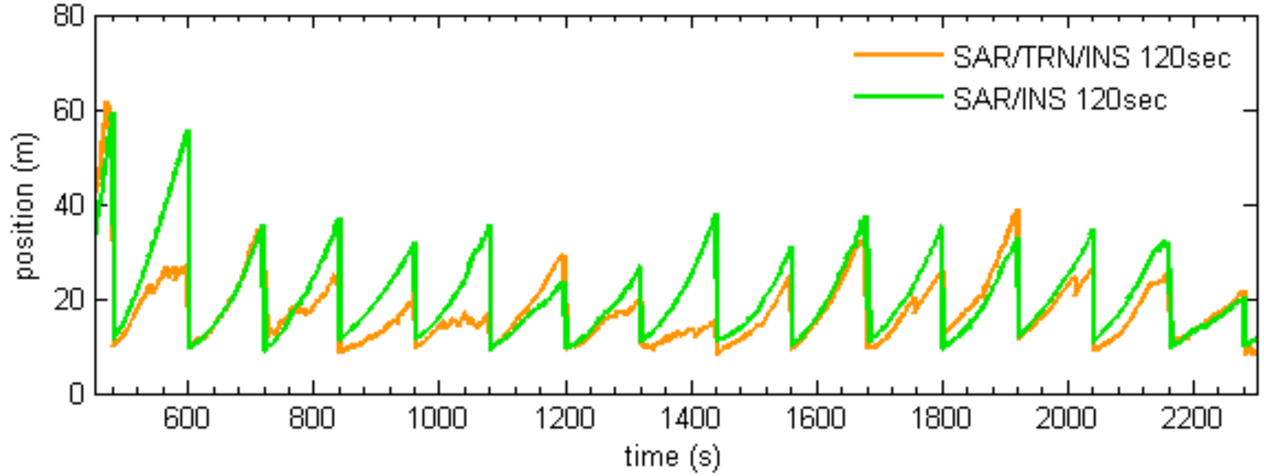




SAR/TRN/INS

- TRN: No accurate estimation over flat area
- SAR prevents increasing position errors over smooth terrain
- TRN leads to reliable navigation information even if no SAR-features are available
- TRN and SAR show different characteristics





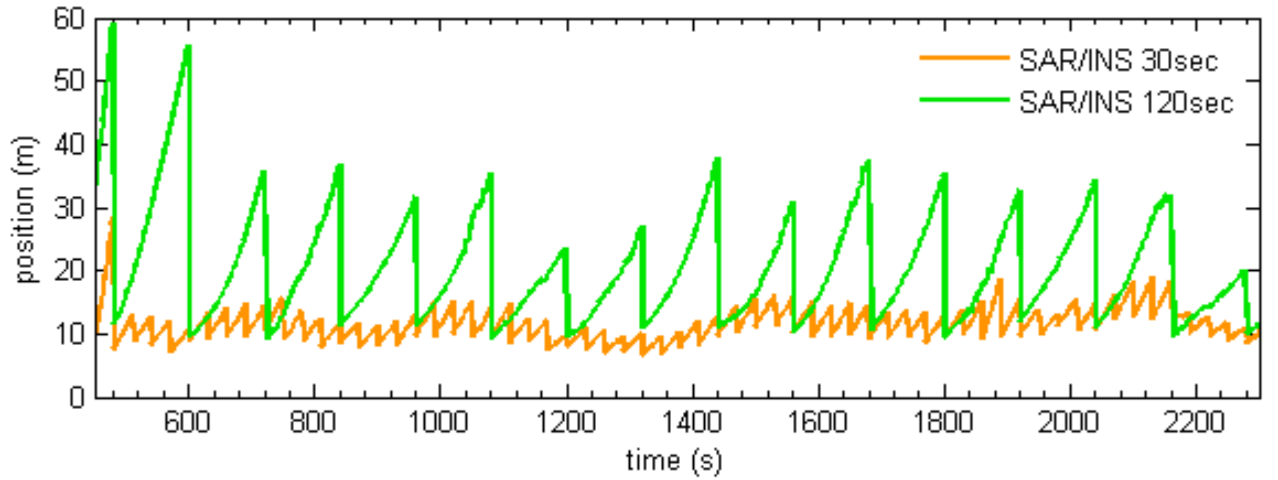
Conclusion

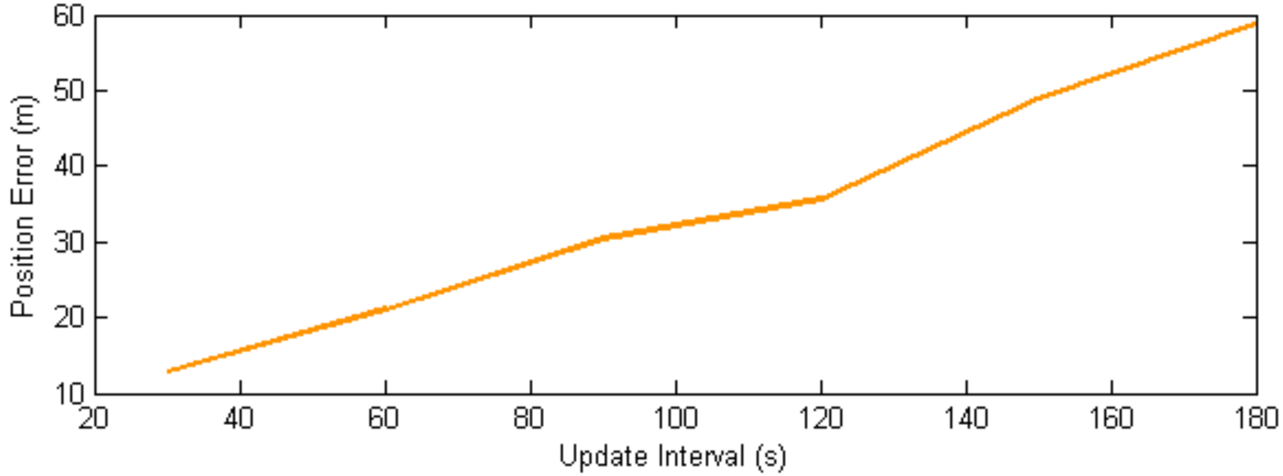
- SAR/INS is able to provide 3-dimensonal navigation information
- Autonomous navigation achievable by SAR/INS
- Feature updates in the scale of a few minutes required
- SAR in combination with low cost TRN is optimal due to complementary sensor characteristics.

Thank you for your attention

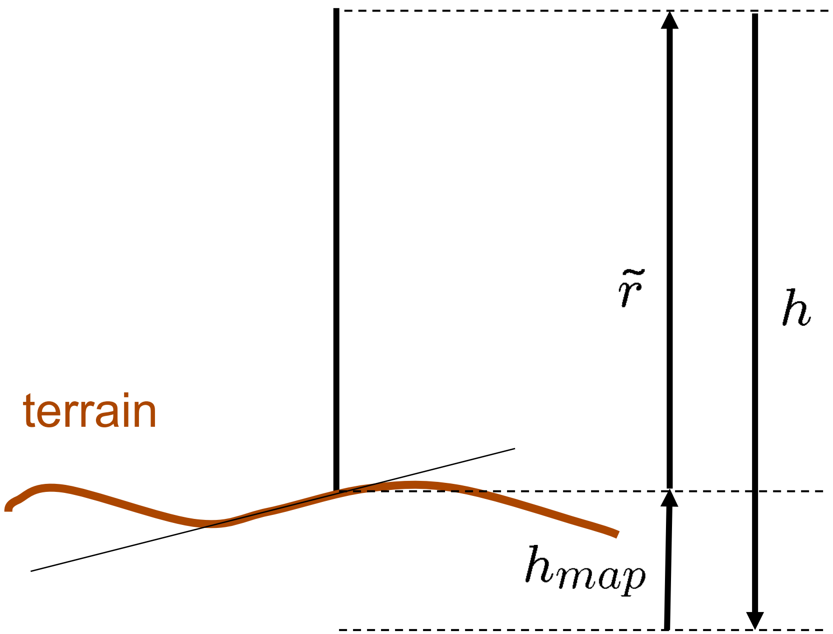
SAR/INS

- Barometric altimeter aids height estimation
- Navigation error depends on feature update rate
- Feature updates in the scale of a few minutes needed





Terrain Referenced Navigation



Reasonable terrain roughness required

Measurement equation

$$\tilde{r} = -h - h_{map}(\phi, \lambda) + \nu$$

Measurement matrix

$$\mathbf{H} = \left(\frac{\partial h_{map}}{\partial x_n}, \frac{\partial h_{map}}{\partial x_e}, -1, \ldots \right)$$

Update step

$$\begin{aligned} \mathbf{K} &= \mathbf{P}\mathbf{H}^T \cdot (\mathbf{H}\mathbf{P}\mathbf{H}^T + \mathbf{R})^{-1} \\ \hat{\vec{x}}^+ &= \hat{\vec{x}}^- + \mathbf{K} \cdot (\tilde{r} - \hat{r}) \\ \mathbf{P}^+ &= \mathbf{P}^- - \mathbf{K}\mathbf{H}\mathbf{P}^- \end{aligned}$$