ROYAL INSTITUTE OF TECHNOLOGY

Smoothing for ZUPT-aided INSs

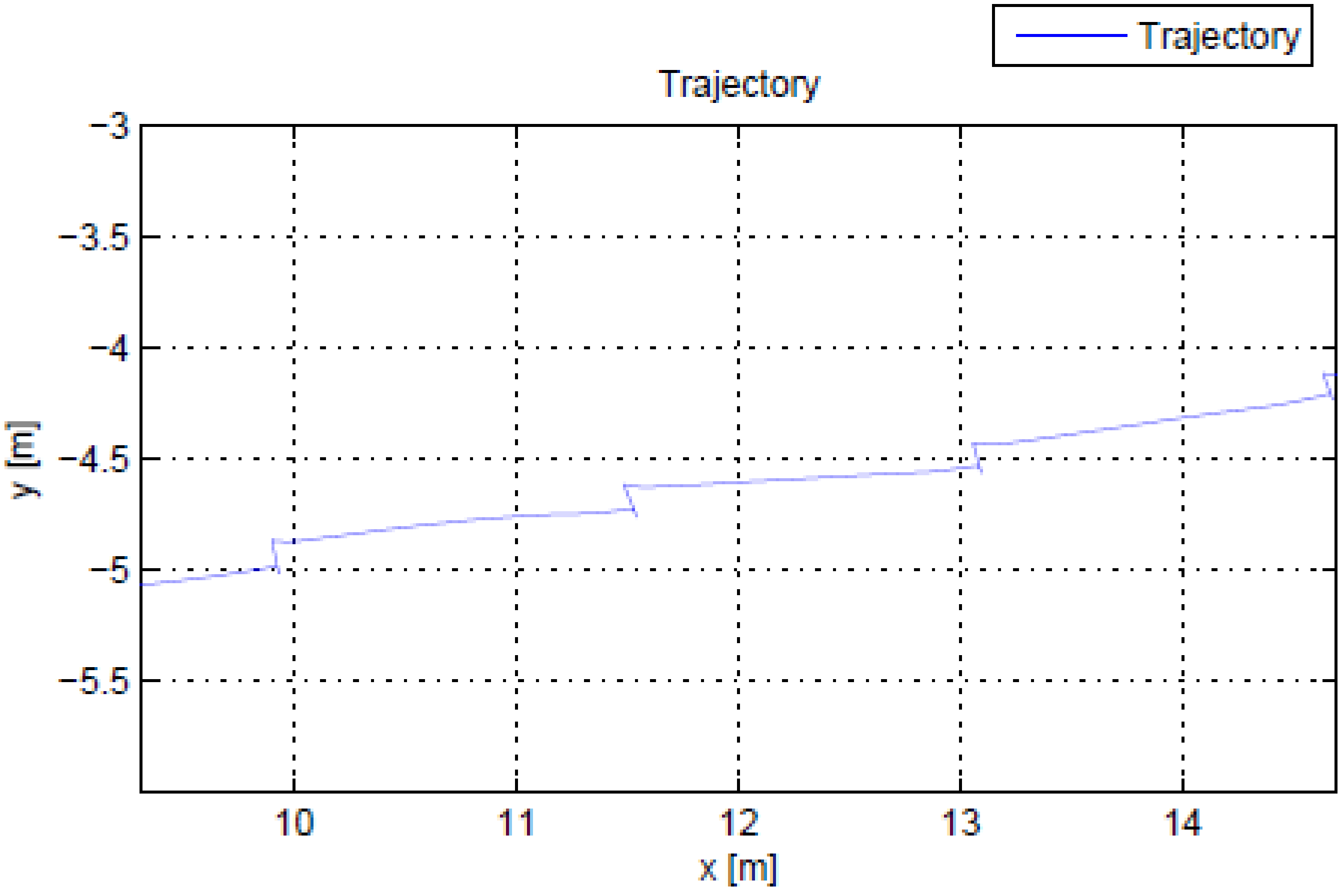
Presentation at IPIN 2012 (121114)

By John-Olof Nilsson
KTH Royal Institute of Technology

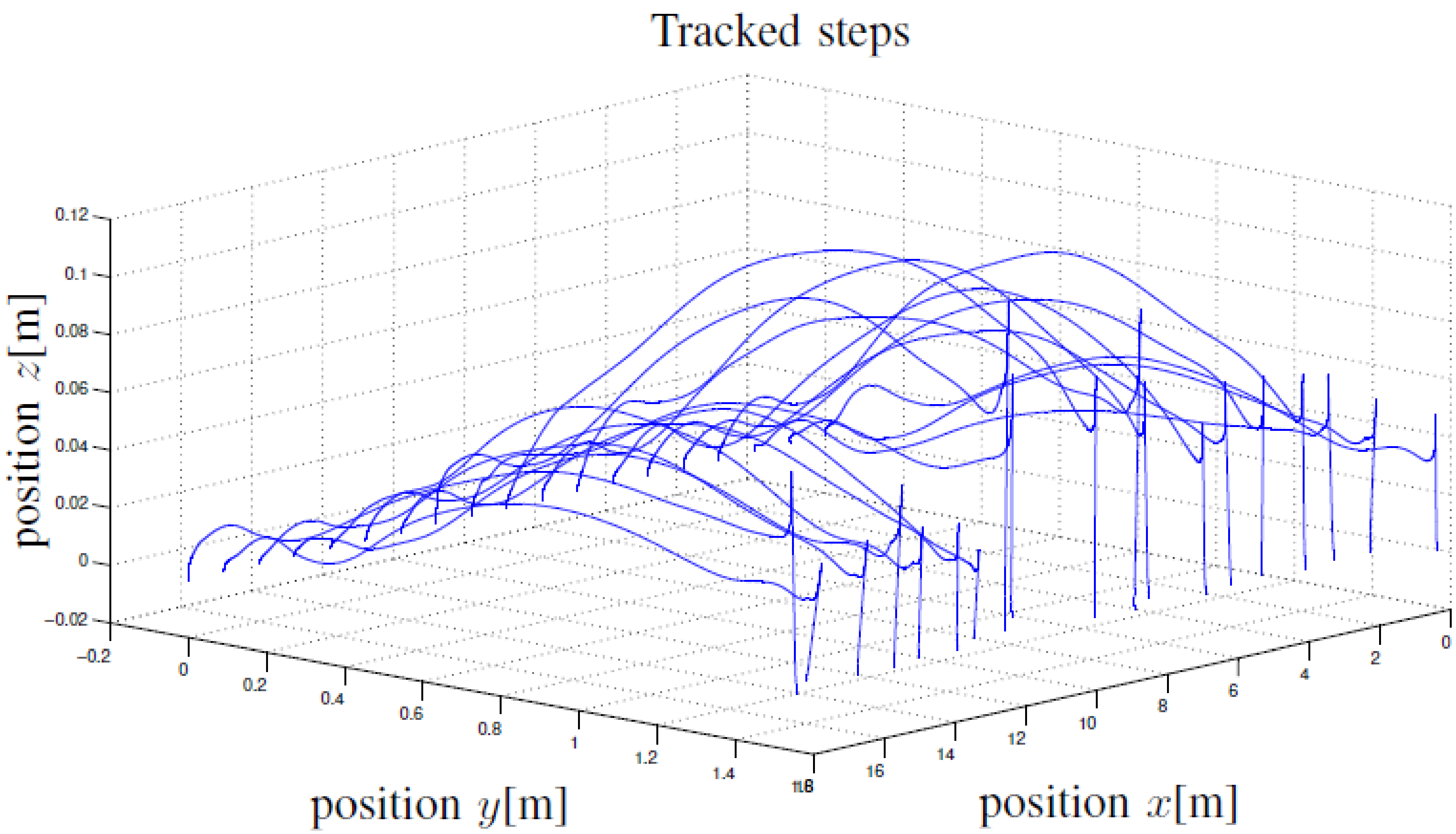
Acknowledgement

- David Simón Colomar

Why smoothing?



Why smoothing?



Outline

- How to do the smoothing
- Recursive step segmentation
- Results

Reproducible research: A Matlab implementation of the suggest algorithm and data sets are available at http://www.openshoe.org

Off-the-shelf algorithms

Initialization: $\hat{\mathrm{x}}_0 \leftarrow \mathrm{E}[\mathrm{x}_0]$, $\mathrm{P}_0 \leftarrow \mathrm{cov}(\mathrm{x}_0)$

Loop: n = 1 **to end of data**

% Time update

$$\hat{\mathbf{x}}_n = f_{\text{mech}}(\hat{\mathbf{x}}_{n-1}, \mathbf{f}_n, \omega_n)$$

$$\mathbf{P}_n = \mathbf{F}_n \mathbf{P}_{n-1} \mathbf{F}_n^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T$$

% Measurement update

if $T(\{\boldsymbol{\omega}^i, \mathbf{f}^i\}_{W_n}) < \gamma$

$$\mathbf{K}_n = \mathbf{P}_n \mathbf{H}^T (\mathbf{H}\mathbf{P}_n\mathbf{H}^T + \mathbf{R})^{-1}$$

$$\delta\hat{\mathbf{x}}_n = \mathbf{K}_n \hat{\mathbf{v}}_n$$

$$\mathbf{P}_n \leftarrow \mathbf{P}_n(\mathbf{I} - \mathbf{K}_n\mathbf{H})$$

% Compensate internal states

$$\begin{bmatrix} \hat{\mathbf{p}}_n \\ \hat{\mathbf{v}}_n \end{bmatrix} \leftarrow \begin{bmatrix} \hat{\mathbf{p}}_n \\ \hat{\mathbf{v}}_n \end{bmatrix} + \begin{bmatrix} \delta\hat{\mathbf{p}}_n \\ \delta\hat{\mathbf{v}}_n \end{bmatrix}$$

$$\hat{\mathbf{R}}_n \leftarrow (\mathbf{I}_3 - \boldsymbol{\Delta}_n)\hat{\mathbf{R}}_n$$

$$\delta\hat{\mathbf{x}}_n \leftarrow \mathbf{0}$$

Loop: $n = s_{\text{end}} - 1$ **to** s_{start}

$$\mathbf{A}_n = \mathbf{P}_{n|n} \boldsymbol{\Gamma}_n^T \mathbf{P}_{n+1|n}^{-1}$$

$$\hat{\chi}_{n|s_{\text{end}}} = \hat{\chi}_{n|n} + \mathbf{A}_n(\hat{\chi}_{n+1|s_{\text{end}}} - \hat{\chi}_{n+1|n})$$

$$\mathbf{P}_{n|s_{\text{end}}} = \mathbf{P}_{n|n} + \mathbf{A}_n(\mathbf{P}_{n+1|s_{\text{end}}} - \mathbf{P}_{n+1|n})\mathbf{A}_n^T$$

Initialization: $\hat{\mathrm{x}}_0 \leftarrow \mathrm{E}[\mathrm{x}_0]$, $\mathrm{P}_0 \leftarrow \mathrm{cov}(\mathrm{x}_0)$

Loop: n = 1 **to end of data**

% Time update

$$\hat{\mathbf{x}}_n = f_{\text{mech}}(\hat{\mathbf{x}}_{n-1}, \mathbf{f}_n, \omega_n)$$

$$\delta\hat{\mathbf{x}}_{n|n-1} = \mathbf{F}_n \delta\hat{\mathbf{x}}_{n-1|n-1}$$

$$\mathbf{P}_n = \mathbf{F}_n \mathbf{P}_{n-1} \mathbf{F}_n^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T$$

% Measurement update

if $T(\{\boldsymbol{\omega}^i, \mathbf{f}^i\}_{W_n}) < \gamma$

$$\mathbf{K}_n = \mathbf{P}_n \mathbf{H}^T (\mathbf{H}\mathbf{P}_n\mathbf{H}^T + \mathbf{R})^{-1}$$

$$\delta\hat{\mathbf{x}}_n = \mathbf{K}_n \hat{\mathbf{v}}_n$$

$$\mathbf{P}_n \leftarrow \mathbf{P}_n (\mathbf{I} - \mathbf{K}_n \mathbf{H})$$

Loop: $n = s_{\text{end}} - 1$ **to** s_{start}

$$\mathbf{A}_n = \mathbf{P}_{n|n} \mathbf{F}^T \mathbf{P}_{n+1|n}^{-1}$$

$$\delta\hat{\mathbf{x}}_{n|s_{\text{end}}} = \delta\hat{\mathbf{x}}_{n|n} + \mathbf{A}_n(\delta\hat{\mathbf{x}}_{n+1|s_{\text{end}}} - \delta\hat{\mathbf{x}}_{n+1|n})$$

$$\mathbf{P}_{n|s_{\text{end}}} = \mathbf{P}_{n|n} + \mathbf{A}_n(\mathbf{P}_{n+1|s_{\text{end}}} - \mathbf{P}_{n+1|n})\mathbf{A}_n^T.$$

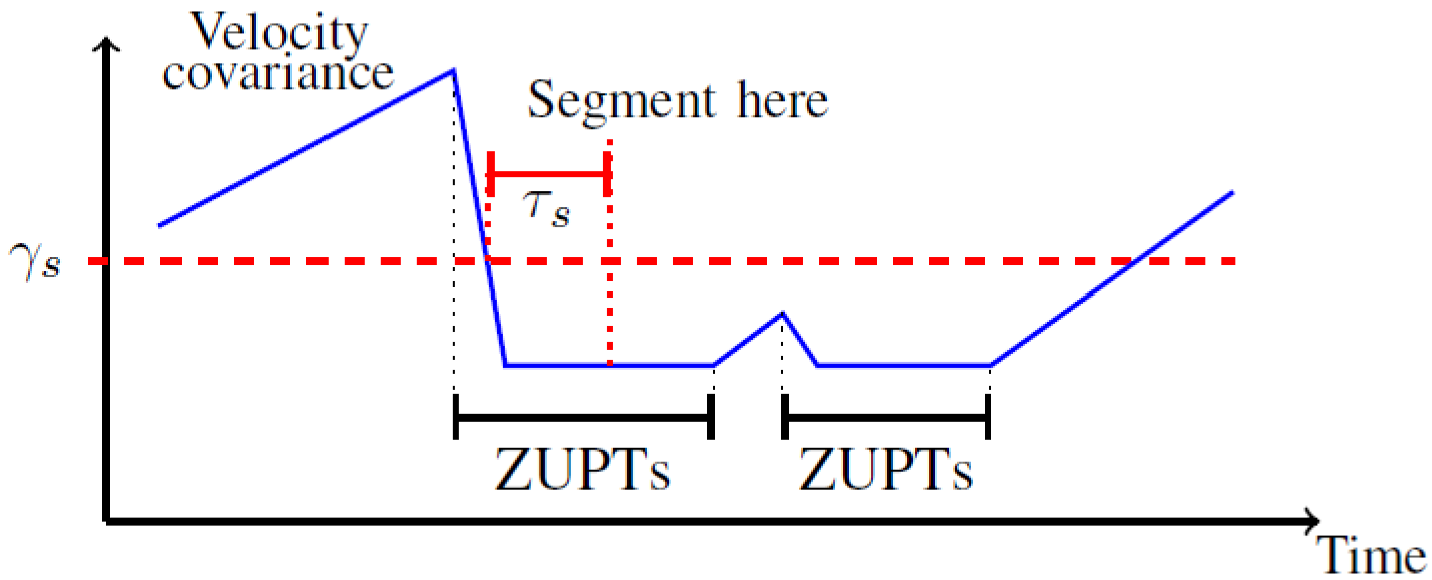
% Compensate internal states

$$\begin{bmatrix} \hat{\mathbf{p}}_n \\ \hat{\mathbf{v}}_n \end{bmatrix} \leftarrow \begin{bmatrix} \hat{\mathbf{p}}_n \\ \hat{\mathbf{v}}_n \end{bmatrix} + \begin{bmatrix} \delta\hat{\mathbf{p}}_n \\ \delta\hat{\mathbf{v}}_n \end{bmatrix}$$

$$\hat{\mathbf{R}}_n \leftarrow (\mathbf{I}_3 - \boldsymbol{\Delta}_n)\hat{\mathbf{R}}_n$$

$$\delta\hat{\mathbf{x}}_n \leftarrow \mathbf{0}$$

Recursive segmentation



ROYAL INSTITUTE OF TECHNOLOGY

3-pass algorithm

Loop while $s_{\text{start}} < s_{\text{end}}$

% Forward Kalman filter

Loop: $n = s_{\text{start}}$ **to** s_{end}

% Time update

$\hat{\mathbf{x}}_n = f_{\text{mech}}(\hat{\mathbf{x}}_{n-1}, \mathbf{f}_n, \omega_n)$

$\delta\hat{\mathbf{x}}_{n|n-1} = \mathbf{F}_n \delta\hat{\mathbf{x}}_{n-1|n-1}$

$\mathbf{P}_{n|n-1} = \mathbf{F}_n \mathbf{P}_{n-1|n-1} \mathbf{F}_n^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T$

% Measurement update

if $T(\{\boldsymbol{\omega}^i, \mathbf{f}^i\}_{W_n}) < \gamma$

$\mathbf{K}_n = \mathbf{P}_{n|n-1}\mathbf{H}^T(\mathbf{H}\mathbf{P}_{n|n-1}\mathbf{H}^T + \mathbf{R})^{-1}$

$\delta\hat{\mathbf{x}}_{n|n} = \delta\hat{\mathbf{x}}_{n|n-1} - \mathbf{K}_n(\delta\hat{\mathbf{v}}_{n|n-1} - \hat{\mathbf{v}}_n)$

$\mathbf{P}_{n|n} = \mathbf{P}_{n|n-1}(\mathbf{I} - \mathbf{K}_n\mathbf{H})$

% Segmentation rule eval.

if $c > 0$

$c = c + T_s$

if $\|\text{diag}(\mathbf{P}^v_{n-1})\| > \gamma_s \ \wedge \ \|\text{diag}(\mathbf{P}^{vel}_n)\| \leq \gamma_s \ \wedge c = 0$

$c = T_s$

if $c > \tau_s$

$s_{\text{end}} \leftarrow n$

break loop

% Smoothing

Loop: $n = s_{\text{end}} - 1$ **to** s_{start}

$\mathbf{A}_n = \mathbf{P}_{n|n}\mathbf{F}^T\mathbf{P}^{-1}_{n+1|n}$

$\delta\hat{\mathbf{x}}_{n|s_{\text{end}}} = \delta\hat{\mathbf{x}}_{n|n} + \mathbf{A}_n(\delta\hat{\mathbf{x}}_{n+1|s_{\text{end}}} - \delta\hat{\mathbf{x}}_{n+1|n})$

$\mathbf{P}_{n|s_{\text{end}}} = \mathbf{P}_{n|n} + \mathbf{A}_n(\mathbf{P}_{n+1|s_{\text{end}}} - \mathbf{P}_{n+1|n})\mathbf{A}_n^T$

% Internal state compensation

Loop: $n = s_{\text{start}}$ **to** s_{end}

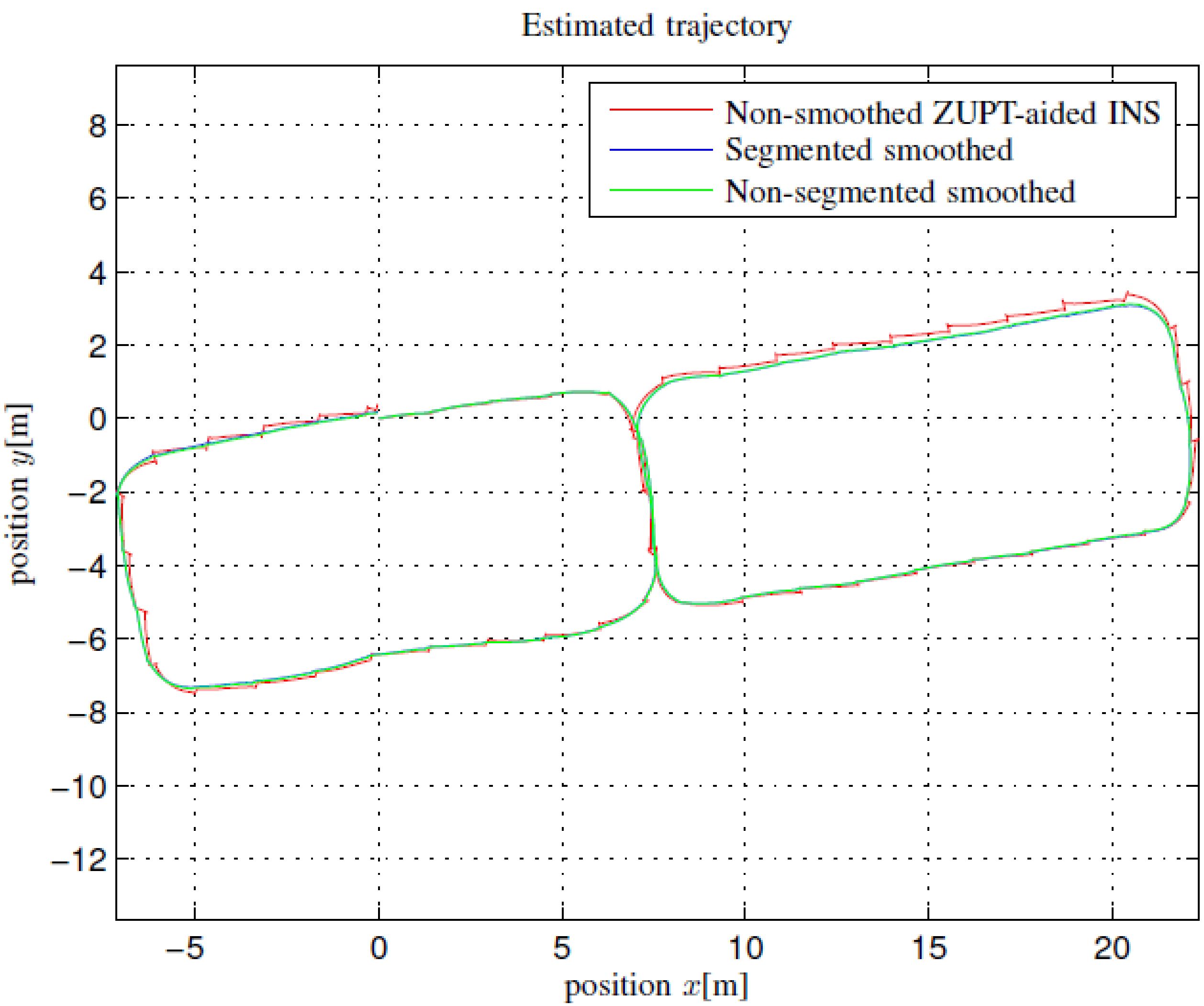
$$\begin{bmatrix} \hat{\mathbf{p}}_n \\ \hat{\mathbf{v}}_n \end{bmatrix} \leftarrow \begin{bmatrix} \hat{\mathbf{p}}_n \\ \hat{\mathbf{v}}_n \end{bmatrix} + \begin{bmatrix} \delta\hat{\mathbf{p}}_{n|s_{\text{end}}} \\ \delta\hat{\mathbf{v}}_{n|s_{\text{end}}} \end{bmatrix}$$

$\hat{\mathbf{R}}_n \leftarrow (\mathbf{I}_3 - \boldsymbol{\Delta}_{n|s_{\text{end}}})(\hat{\mathbf{R}}_n)$

$\delta\hat{\mathbf{x}}_n \leftarrow \mathbf{0}$

$s_{\text{start}} = s_{\text{end}} + 1$, $s_{\text{end}} =$ "end of data", $c = 0$

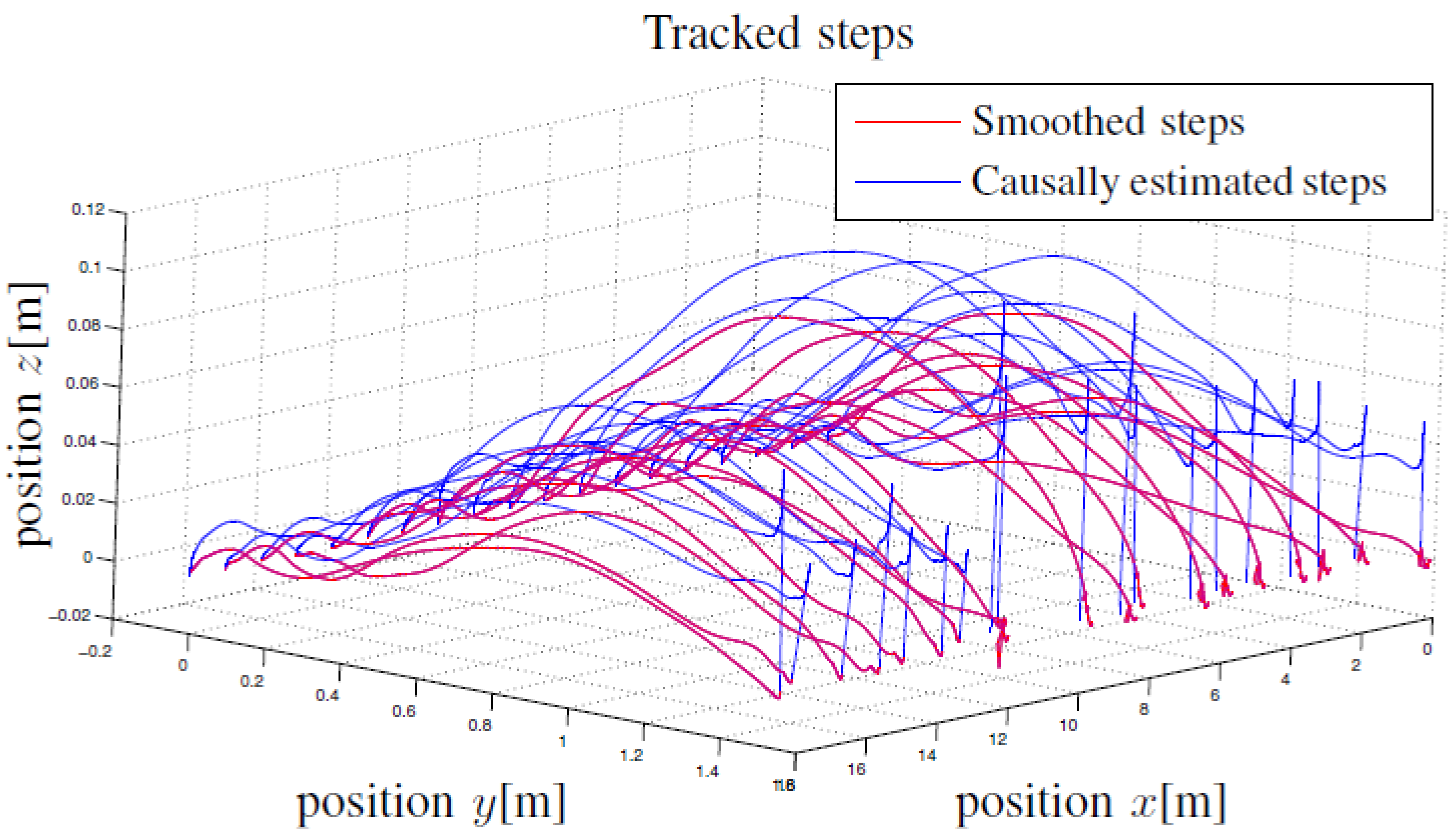
Results



Estimated trajectory
Non-smoothed ZUPT-aided INS
Segmented smoothed
Non-segmented smoothed
8
6
4
2
0
−2
−4
−6
−8
−10
−12
position y[m]
−5
0
5
10
15
20
position x[m]

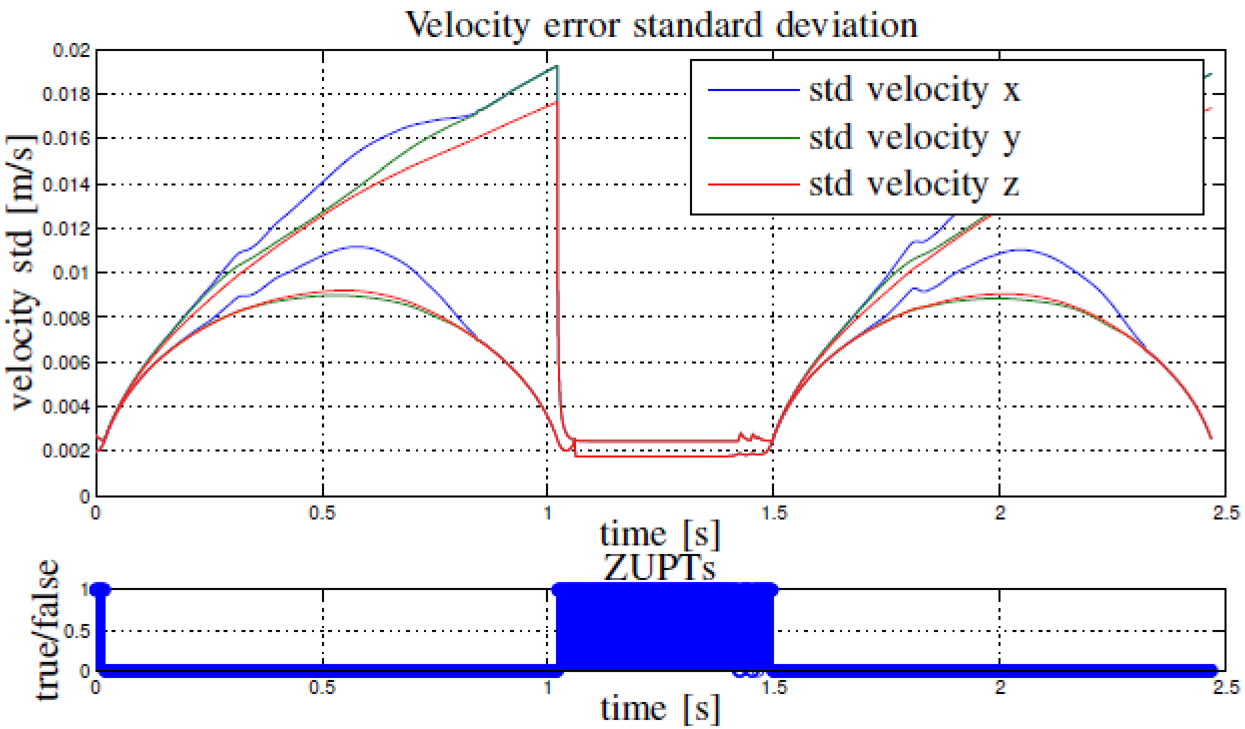
ROYAL INSTITUTE OF TECHNOLOGY

Results



ROYAL INSTITUTE OF TECHNOLOGY

Results



Conclusion

- A 3-pass smoothing algorithm for ZUPT-aided INS has been suggested.
- A recursive step segmentation for near realtime implementation has been suggested.
- Shown that step-wise smoothing give the same results as smoothing of the whole dataset
- The system has been shown to be insensitive to the linearization point

ROYAL INSTITUTE
OF TECHNOLOGY

The end