Estimation of Moving Vehicle Position Based on GPS/CAN Data

Lenka Pavelková

Department of Adaptive Systems
Institute of Information Theory and Automation, Prague
e-mail: pavelkov@utia.cas.cz

6th June 2011

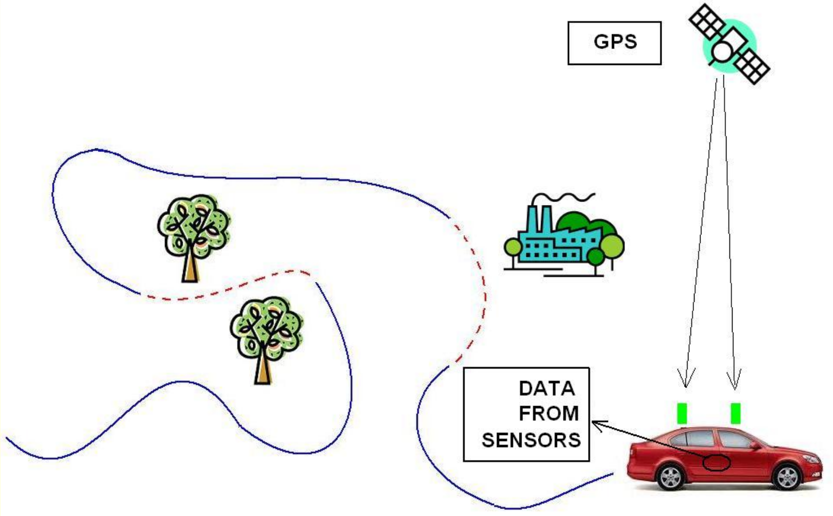
Presentation layout

1. Problem description
2. State of the art
3. Proposed model & Estimation
4. Algorithm & Conditions
5. Example
6. Conclusions

Problem description

- estimation of moving vehicle position
  - readily available data
  - reliable algorithm
  - simple initial setting and maintaining
  - precise estimation
  - off-line computation
- cooperation with Škoda Auto, a.s.

Problem description



Problem description

Available data

- GPS
  - differential GPS - precision in cm's
  - sampling frequency 0.1 s
  - signal outages
  - position, azimuth
- INS
  - MEMS sensors - accelerometers & gyroscopes
  - temperature dependence
  - acceleration, angular speed
- CAN
  - vehicle bus standard
  - mutual communication of devices within a vehicle
  - velocity, angular speed, distance moved, ...

Problem description

Available data

- GPS
  - differential GPS - precision in cm's
  - sampling frequency 0.1 s
  - signal outages
  - position, azimuth
- INS
  - MEMS sensors - accelerometers & gyroscopes
  - temperature dependence
  - acceleration, angular speed
- CAN
  - vehicle bus standard
  - mutual communication of devices within a vehicle
  - velocity, angular speed, distance moved, ...

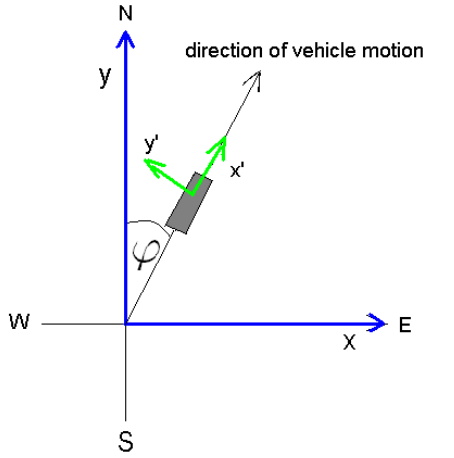
Problem description

Available data

- GPS
    - differential GPS - precision in cm's
    - sampling frequency 0.1 s
    - signal outages
    - position, azimuth
- INS
    - MEMS sensors - accelerometers & gyroscopes
    - temperature dependence
    - acceleration, angular speed
- CAN
    - vehicle bus standard
    - mutual communication of devices within a vehicle
    - velocity, angular speed, distance moved, ...

Problem description

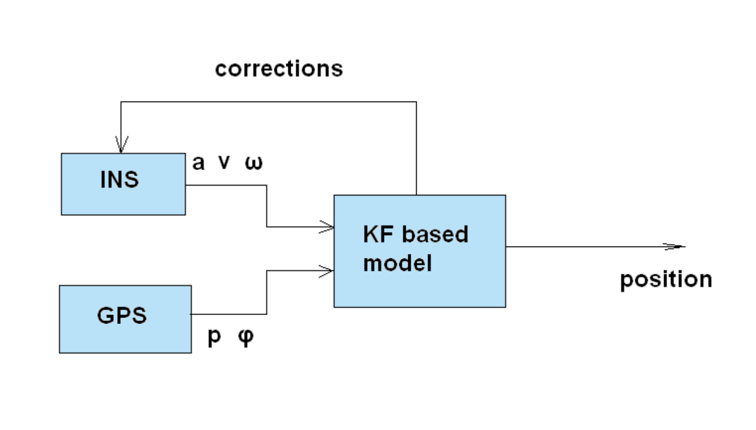
Coordinate systems



State of the art

- kinematic models (2D, 3D):
  - position - velocity - acceleration
  - azimuth - angular speed
- discrete-time state space model + normal noise
- Kalman based filters & smoothers
  - linear - KF
  - nonlinear - EKF, UKF
- errors correction
- coordinates transformation

GPS/INS integration via Kalman filter approach



Proposed model & Estimation

- kinematic state-space model
- GPS + CAN data
- 2D models
  - linear vs. non-linear
  - choice of involved data
  - definition of states, inputs, outputs
  - uniform noise
- algorithms
  - linear and non-linear programming
  - batch estimates

Proposed model & Estimation

- kinematic state-space model
- GPS + CAN data
- 2D models
  - linear vs. non-linear
  - choice of involved data
  - definition of states, inputs, outputs
  - uniform noise
- algorithms
  - linear and non-linear programming
  - batch estimates

Proposed model & Estimation

- kinematic state-space model
- GPS + CAN data
- 2D models
  - linear vs. non-linear
  - choice of involved data
  - definition of states, inputs, outputs
  - uniform noise
- algorithms
  - linear and non-linear programming
  - batch estimates

Proposed model & Estimation

PROS

- automatic initial setting
- without coordinate transformation
- simple treatment of missing data
- simple fitting of additional constraints

CONS

- long time computation

Proposed model & Estimation

PROS

- automatic initial setting
- without coordinate transformation
- simple treatment of missing data
- simple fitting of additional constraints

CONS

- long time computation

Proposed model & Estimation

State-space model with missing data

For $t \in t^* = 1, 2, \ldots, T$, $k \in k^* \subset t^*$

$$\begin{aligned} \mathbf{x}_t &= g(\mathbf{x}_{t-1}, \mathbf{u}_t) + \mathbf{w}_t; & f(\mathbf{w}_t) = \mathcal{U}(\mathbf{0}, \mathbf{p}) \\ \mathbf{y}_k &= h(\mathbf{x}_k) + \mathbf{e}_k; & f(\mathbf{e}_t) = \mathcal{U}(\mathbf{0}, \mathbf{r}) \end{aligned}$$

- m-dimensional state $\mathbf{x}_t$;
- l-dimensional input $\mathbf{u}_t$;
- n-dimensional output $\mathbf{y}_k$;
- real vector functions, $g : \mathcal{R}^m \to \mathcal{R}^m$; $h : \mathcal{R}^m \to \mathcal{R}^n$;
- state and output noises $\mathbf{w}_t$ and $\mathbf{e}_t$;

Proposed model & Estimation

Joint pdf

For $t \in t^* = 1, 2, \ldots, T$, $k \in k^* \subset t^*$

$$f(\mathbf{d}^{1:OUT:T}, \mathbf{x}^{0:T}, \Theta) \propto \left[\prod_{i=1}^{m} p_i\right]^{-T} \left[\prod_{j=1}^{n} r_j\right]^{-K} \chi(\mathcal{S})$$

$$\mathcal{S} : \begin{cases} \text{prior information} \\ \text{state model} \\ \text{restriction on states} \end{cases}$$

Proposed model & Estimation

The MAP estimate of $\mathbf{X} = \left[\ \left(\mathbf{x}^{0:t}\right)' \quad \mathbf{p}' \quad \mathbf{r}'\ \right]'$

$$\hat{\mathbf{X}} = \arg\,\min_{\mathbf{X}\in\mathcal{S}} \left(\sum_{i=1}^{m} \ln(p_i) + \frac{K}{T} \sum_{j=1}^{n} \ln(r_j) \right)$$

The MAP estimate $\rightarrow$ non-linear programming form

- missing data = missing inequality
- off-line estimate
- data at the beginning and at the end of data outage

Proposed model & Estimation

GPS/INS data – usage in the model

quantity (unit)	notation	source	in model as
true position No. 1 (m)	(p_{x1}, p_{y1})	-	state
true position No. 2 (m)	(p_{x2}, p_{y2})	-	state
true azimuth (rad)	φ	-	state
true velocity (m/s)	v	-	state
true yaw rate [rad/s]	ω	-	state
measured position No. 1 (m)	$(\tilde{p}_{x1}, \tilde{p}_{y1})$	GPS	output
measured position No. 2 (m)	$(\tilde{p}_{x2}, \tilde{p}_{y2})$	GPS	output
measured azimuth (rad)	$\tilde{\varphi}$	GPS	output
measured velocity (m/s)	$\tilde{v}$	INS	output
measured yaw rate [rad/s]	$\tilde{\omega}$	INS	output

State-space model of moving vehicle

State equation:

$$
\begin{aligned}
p_{x1;t} = &\quad p_{x1;t-1} + h v_{t-1} \sin \varphi_{t-1} & + w_{x1;t} \\
p_{y1;t} = &\quad p_{y1;t-1} + h v_{t-1} \cos \varphi_{t-1} & + w_{y1;t} \\
p_{x2;t} = &\quad p_{x2;t-1} + h v_{t-1} \sin \varphi_{t-1} & + w_{x2;t} \\
p_{y2;t} = &\quad p_{y2;t-1} + h v_{t-1} \cos \varphi_{t-1} & + w_{y2;t} \\
\varphi_{t} = &\quad \varphi_{t-1} - h\,\omega_{t-1} & + w_{\varphi;t} \\
v_{t} = &\quad v_{t-1} & + w_{v;t} \\
\omega_{t} = &\quad \omega_{t-1} & + w_{\omega;t},
\end{aligned}
$$

Proposed model & Estimation

State-space model of moving vehicle

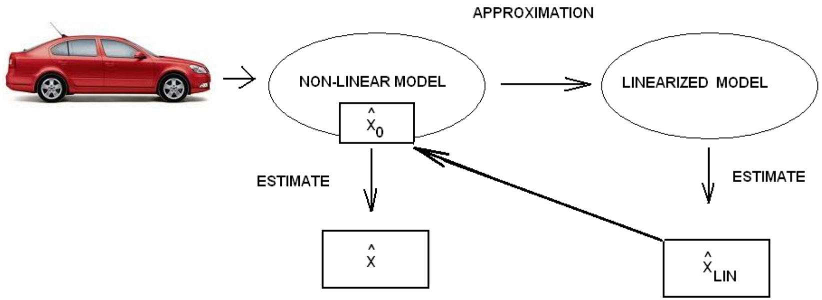
Output equation:

$$
\begin{aligned}
\mathcal{I}_t\ \tilde{p}_{x1;t} = &\ \mathcal{I}_t(p_{x1;t} + e_{x1;t}) \\
\mathcal{I}_t\ \tilde{p}_{y1;t} = &\ \mathcal{I}_t(p_{y1;t} + e_{y1;t}) \\
\mathcal{I}_t\ \tilde{p}_{x2;t} = &\ \mathcal{I}_t(p_{x2;t} + e_{x2;t}) \\
\mathcal{I}_t\ \tilde{p}_{y2;t} = &\ \mathcal{I}_t(p_{y2;t} + e_{y2;t}) \\
\mathcal{I}_t\ \tilde{\varphi}_t = &\ \mathcal{I}_t(\xi(p_{x1;t}, p_{y1;t}, p_{x2;t}, p_{y2;t}) + e_{\varphi;t}) \\
\tilde{v}_t = &\ v_t + e_{v;t} \\
\tilde{\omega}_t = &\ \omega + e_{\omega;t}
\end{aligned}
$$

$\mathcal{I}_t \in 0, 1$ - measurement indicator

Algorithm & Conditions

Problem of initial setting



Linearised model

$$\begin{aligned} \mathbf{p}_t &= \mathbf{p}_{t-1} + \mathbf{u}_t + \mathbf{w}_t \\ \mathcal{I}_t \tilde{\mathbf{p}}_t &= \mathcal{I}_t \left(\mathbf{p}_t + \mathbf{e}_t \right) \end{aligned}$$

- $\mathbf{u}_t = \phi(v_t, \omega_t, \hat{\varphi}_t)$
- $\hat{\varphi} = \begin{cases} \tilde{\varphi}_t & \ldots \quad \text{if GPS data are available} \\ \hat{\varphi}_{t-1} - h\omega_{t-1} & \ldots \quad \text{otherwise} \end{cases}$

Algorithm & Conditions

- **choice of data set**
  - one outage
  - availability of measurements at the beginning
  - availability of measurements at the end
- **estimation of linearised model**
  - computation of $\hat{\varphi}$
  - estimation of $\mathbf{X}_{\mathrm{LIN}}$
- **estimation of non-linear model**
  - initialization - $\hat{\mathbf{X}}_0 = \hat{\mathbf{X}}_{\mathrm{LIN}}$
  - estimation of $\hat{\mathbf{X}}$

Example

- real data - testing area, road
- sampling frequency $h = 0.1s$
- simulated outages in complete GPS data
- result evaluation - absolute error of estimates

Example

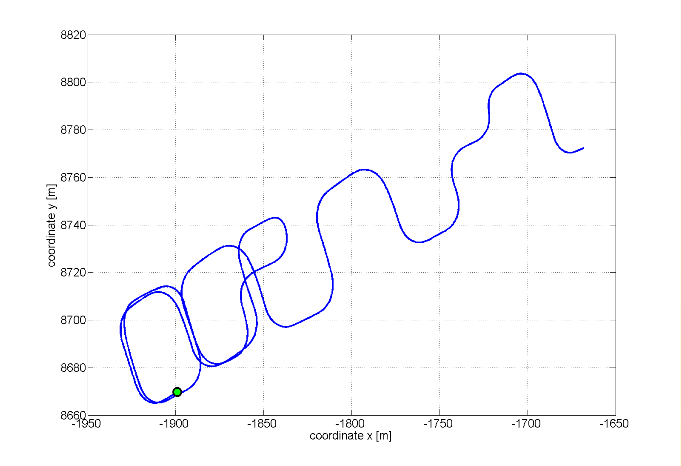


Figure: Testing area - complete vehicle trajectory

Example

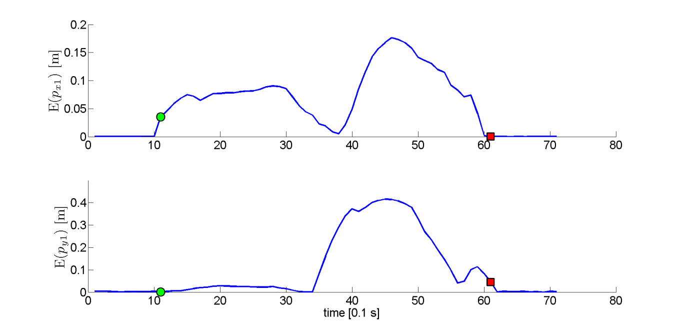


Figure: Course of absolute error for estimates $p_{x1;t}$, $p_{y1;t}$

Example

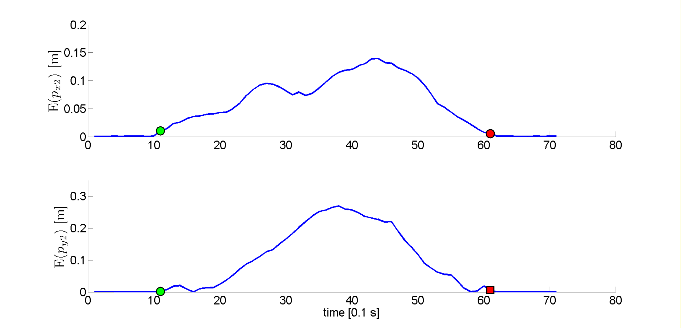
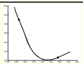
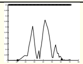
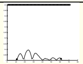
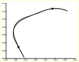
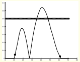
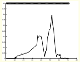
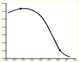
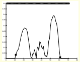
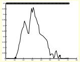
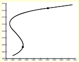
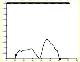
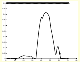
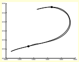
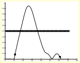
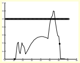


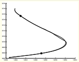
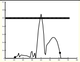
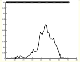
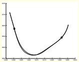
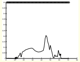
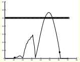
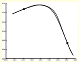
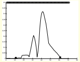
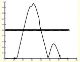
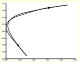
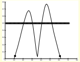
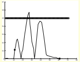
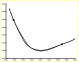
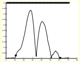
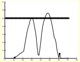
Figure: Course of absolute error for estimates $p_{x2;t}$, $p_{y2;t}$

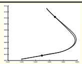
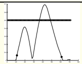
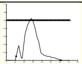
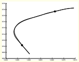
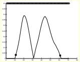
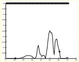
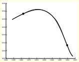
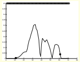
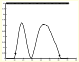
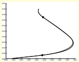
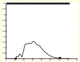
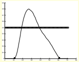
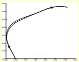
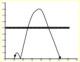
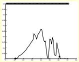
Example

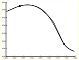
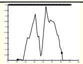
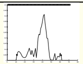
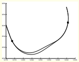
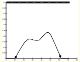
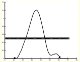
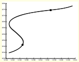
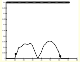
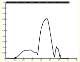
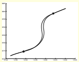
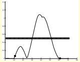
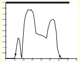
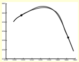
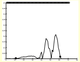
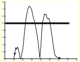
On the following slides:

Course of the absolute estimation errors Δ_{z_t}
with demarcated tolerance area 0.5 m
and values of $\max(\Delta_z)$ and $\mathrm{mean}(\Delta_z)$, $z \in \{p_{x1}, p_{y1}\}$
with respective part of trajectory (“TR”)

"TR"	$\Delta_{p_{x1;t}}$		$\Delta_{p_{y1;t}}$	
	course	max [m] (mean [m])	course	max [m] (mean [m])
		0.36 (0.08)		0.09 (0.02)
		0.64 (0.23)		0.39 (0.07)
		0.39 (0.11)		0.46 (0.13)
		0.18 (0.05)		0.41 (0.10)
		0.94 (0.28)		0.60 (0.18)

"TR"	$\Delta_{p_{x1;t}}$		$\Delta_{p_{y1;t}}$	
	course	max [m] (mean [m])	course	max [m] (mean [m])
		0.55 (0.10)		0.30 (0.07)
		0.20 (0.04)		0.56 (0.17)
		0.42 (0.08)		0.97 (0.30)
		0.77 (0.29)		0.57 (0.13)
		0.43 (0.12)		0.56 (0.19)

"TR"	$\Delta_{p_{x1;t}}$ course	$\Delta_{p_{x1;t}}$ max [m] (mean [m])	$\Delta_{p_{y1;t}}$ course	$\Delta_{p_{y1;t}}$ max [m] (mean [m])
		0.69 (0.24)		0.52 (0.12)
		0.39 (0.14)		0.25 (0.04)
		0.31 (0.08)		0.32 (0.13)
		0.16 (0.05)		0.80 (0.29)
		0.80 (0.30)		0.27 (0.08)

"TR"	$\Delta_{p_{x1;t}}$ course	$\Delta_{p_{x1;t}}$ max [m] (mean [m])	$\Delta_{p_{y1;t}}$ course	$\Delta_{p_{y1;t}}$ max [m] (mean [m])
		0.48 (0.15)		0.41 (0.07)
		0.23 (0.10)		1.20 (0.34)
		0.15 (0.07)		0.35 (0.08)
		1.08 (0.34)		0.43 (0.17)
		0.21 (0.03)		0.73 (0.24)

Conclusion

What is done:

- non-linear 2D state-space model of a moving vehicle
- algorithm for off-line position estimation

Future work:

- 3D model
- more precise data
- additional restrictions on states
- inclusion of a map
- estimates on the window

Conclusion

What is done:

- non-linear 2D state-space model of a moving vehicle
- algorithm for off-line position estimation

Future work:

- 3D model
- more precise data
- additional restrictions on states
- inclusion of a map
- estimates on the window

Thank you for your attention!

QUESTIONS ?