A localized deterministic Ensemble Kalman Filter

LARGE-SCALE TRAFFIC STATE ESTIMATION
• Intro: need for large-scale traffic state estimation
• Some Kalman Filter basics
• The Ensemble Kalman Filter
  ➤ getting rid of matrix inversions
  ➤ deterministic formulation
  ➤ localisation
• Simulations, Results and Discussions
• Conclusions & outlook (big) plans
INTRODUCTION

Dutch National Data Warehouse Traffic Information

- **Network**
  - ✓ 2500 km freeways
  - ✓ 3500 km prov/urban
  - ✓ 24,000 measurement sites

- **Real-time Data**
  - ✓ Dynamic: Flow, Speed, Occupancy, Travel time (lane / usr class specific), FCD
  - ✓ Status: roadworks, incidents, events
  - ✓ 150,000 measurements/min

- **Historical database**
  - ✓ 200 TB
  - ✓ Raw data
INTRODUCTION

Dutch National Data Warehouse Traffic Information

• Ambition
  ✓ Build national traffic observatory (for RT and archived traffic info)

• From raw data to real information:
  ✓ Intelligent database (search terms related to traffic patterns and explanatory factors)
  ✓ Data-based routable network graphs
  ✓ Densities, space mean speeds
  ✓ Capacities, other FD params
  ✓ MFDs and other agg state vars
  ✓ Inflows, turns, OD flows
  ✓ Opensource tools to dive into all this
TotFlowRoute = 26426.00 [veh]
AvgSpeedRoute = 69.05 [km/h]
PercMissingSpeedDetectors = 7.47 %
PercMissingFlowDetectors = 7.47 %
PercMissingSpeedCrossSections = 7.49 %
PercMissingFlowCrossSections = 7.47 %
TotalVehicleLossHours = 473.10 [veh-h]
TravelTimePercentile10 = 6.64 [min]
TravelTimePercentile25 = 7.24 [min]
TravelTimePercentile50 = 8.97 [min]
TravelTimePercentile75 = 9.70 [min]
TravelTimePercentile90 = 10.65 [min]
AvgTravelTime = 8.71 [min]
StdTravelTime = 1.45 [min]
INTRODUCTION

Dutch National Data Warehouse Traffic Information

- Ambition
  ✓ Build national traffic observatory (for RT and archived traffic info)

- Next step:
  ✓ Scalable efficient model-based traffic state estimation: densities (inflows, turns)

- TRADE OFF
  ✓ Estimation accuracy vs computational efficiency
THE KALMAN FILTER

For any system cast in discrete state space form

- the KF is an efficient recursive solution for estimating state vector $x$;

- that is optimal in three ways
  - Minimum error (co)variance
  - Maximal posterior probability
  - Conditional mean estimation

- given a some (fairly strict) assumptions
  - the validity of which strongly affect the quality of the result

\[
x_{k+1} = F_k x_k + G_k u_k + w_k \\
y_k = H_k x_k + v_k
\]

\[
E(w_k) = 0, \quad E(w_k w'_l) = \begin{cases} Q_k & k = l \\ 0 & k \neq l \end{cases} \\
E(v_k) = 0, \quad E(v_k v'_l) = \begin{cases} R_k & k = l \\ 0 & k \neq l \end{cases}
\]
THE KALMAN FILTER

• Initial conditions (reasonable choices for $Q$ and $R$ must also be made)

\[
\hat{x}_{0|0} = \hat{x}_0, P_{0|0} = P_0
\]

• Time-propagation (prediction):

\[
\begin{align*}
\hat{x}_{k|k-1} &= F_{k-1} \hat{x}_{k-1|k-1} + G_{k-1} u_{k-1} \\
P_{k|k-1} &= F_{k-1} P_{k-1|k-1} F_{k-1}^T + Q_{k-1}
\end{align*}
\]

• Measurement adaptation (correction):

\[
\begin{align*}
e_k &= y_k - H_k \hat{x}_{k|k-1} \\
K_k &= P_{k|k-1} H_k / (H_k P_{k|k-1} H_k + R_k) \\
\hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k e_k \\
P_{k|k} &= [I - K_k H_k] P_{k|k-1}
\end{align*}
\]
A CLOSER LOOK AT THE FILTER GAIN

\[ K_k = \frac{P_{k|k-1}H_k}{H_k P_{k|k-1}H_k + R_k} \]

sensitivity observation model to changes in the state

error covariance of the innovations (i.e. uncertainty in the new information we get from observations).
A CLOSER LOOK AT THE FILTER GAIN

Kalman Gain = \frac{\text{uncertainty in (process) model} \times \text{Sensitivity} \text{ observation model}}{\text{uncertainty in observations}}

So balance between uncertainty in process model and uncertainty in measurements / observation model!
TRAFFIC FLOW SIMULATION MODELS

Are naturally cast in discrete state-space form

- Conservation of vehicles (+ possible additional pde for speed)
  \[ x_{k+1} = f(x_k, u_k) + w_k \]
  \[ \text{typically } f : \rho_{k+1} = \rho_k + \frac{\Delta L}{\Delta t} (q_k^{in} - q_k^{out} + r - s) \]

- Observation model typically fundamental diagram
  \[ y_k = h_k(x_k) + v_k \]
  \[ \text{typically } h : q_k = Q^e_k(\rho_k) \]

\[ Q^e_k(\rho_k) \]
\[ V^e_k(\rho_k) = \frac{Q^e_k(\rho_k)}{\rho_k} \]
Extended Kalman filtering

- Nonlinear extension KF
- Assumptions
  - Process model can be approximated by 1st order Taylor expansion
  - Obs model can be approximated with 1st order Taylor expansion
  - All other KF assumptions (iid Gaussian WN, etc)

\[
\begin{align*}
x_{k+1} &= f_k(x_k) + w_k \\
y_k &= h_k(x_k) + v_k \\
F_k &= \frac{df_k(x)}{dx} \bigg|_{x=\hat{x}_k} \\
H_k &= \frac{dh_k(x)}{dx} \bigg|_{x=\hat{x}_k}
\end{align*}
\]
EXTENDED KALMAN FILTER

• Many challenges
  ➡ linearisation may cause adaptation in the wrong direction
  ➡ Computing derivatives = pain in the @!$$ …
  ➡ Scalability: matrix manipulations & inversions expensive
  ➡ (E)KF assumptions on violated all the time

\[
\hat{x}_{0|0} = \hat{x}_0, P_{0|0} = P_0
\]

\[
\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_k)
\]

\[
P_{k|k-1} = F_{k-1} P_{k-1|k-1} F_{k-1}^T + Q_{k-1}
\]

\[
e_k = y_k - h(\hat{x}_{k|k-1})
\]

\[
K_k = P_{k|k-1} H_k \left( H_k P_{k|k-1} H_k + R_k \right)^{-1}
\]

\[
\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k e_k
\]

\[
P_{k|k} = \left[ I - K_k H_k \right] P_{k|k-1}
\]
ALTERNATIVE: ENSEMBLE KALMAN FILTER

- Use ensemble of $N$ state vectors and noisy observation replications

\[ X_{0/0} = [x_0^1, \ldots, x_0^N], \]

\[ X_{k|k-1} = f(X_{k-1|k-1}, U_k) \]

\[ x_{k|k-1} = E(X_{k|k-1}) = \frac{1}{N} \sum_{i=1}^{N} x_{k|k-1}^i \]

\[ P_{k|k-1} = A_{k|k-1}A_{k|k-1}^T/(N-1), \text{ with } A_x = X_x - E(X_x) \]

\[ D_k = [d_k^1, \ldots, d_k^N], \text{ with } d_k^i = y_k + e_k^i \]

\[ E_k = D_k - h(X_{k|k-1}) \]

\[ K_k = P_{k|k-1} \big( H_k P_{k|k-1} H_k + R_k \big)^{-1} \]

\[ X_{k|k} = X_{k|k-1} + K_k E_k \]

- Predicting ensemble mean and variance is simple ...

- Update step similar to EKF

⇒ Still requires linearisation?
ALTERNATIVE: ENSEMBLE KALMAN FILTER

• Use ensemble of $N$ state vectors and noisy observation replications

$X_{00} = [x_0^1, ..., x_0^N]$

$X_{klk-1} = f(X_{klk-1}, U_k)$

$x_{klk-1} = E(X_{klk-1}) = \frac{1}{N} \sum_{i=1:N} x_{klk-1}^i$

$P_{klk-1} = A_{klk-1} A_{klk-1}^T / (N - 1)$, with $A_x = X_x - E(X_x)$

$D_k = [d_k^1, ..., d_k^N]$, with $d_k^i = y_k + e_k^i$

$E_k = D_k - h(X_{klk-1})$

$X_{klk} = X_{klk-1} + K_k E_k$

• Predicting ensemble mean and variance is simple …

$K_k = \frac{1}{N-1} A_{klk-1} (HA)^T / P_{klk}^*$

with $P_{klk}^* = \frac{1}{N-1} HA (HA)^T + R_k$

and $[HA^i] = h(x_{klk-1}^i) - \frac{1}{N} \sum_{j=1:N} h(x_{klk-1}^j)$
A CLOSER LOOK AT THIS FILTER GAIN

\[ K_k = A_{k|k-1} (HA)^T \]

\[ \frac{HA(HA)^T + R_k}{HA(HA)^T + R_k} \]

distance prior state to mean ensemble state

distance predicted observations to mean ensemble prediction

total error variance (uncertainty)

measurement equation
A CLOSER LOOK AT THIS FILTER GAIN

Kalman Gain = \frac{\text{uncertainty in model predictions (sign?)}}{\text{uncertainty in observations}}

So again balance between uncertainty in model and uncertainty in measurements!
EKF VS ENKF

EnKF Improves upon the ‘wrong sign’ adaptation problem

density change in positive direction of derivative

Observation  △Prior mean state  ◊Prior/posterior Ensemble  □Posterior mean state
EKF VS ENKF

EnKF Improves upon the ‘wrong sign’ adaptation problem

density change in positive direction of derivative

right direction

Observation △ Prior mean state ◇ Prior/posterior Ensemble □ Posterior mean state
EKF VS ENKF

EnKF Improves upon the ‘wrong sign’ adaptation problem

density change in negative direction of derivative

right direction

Observation △Prior mean state ◇Prior/posterior Ensemble □Posterior mean state
EKF VS ENKF

EnKF Improves upon the ‘wrong sign’ adaptation problem

Given proper spread, the sign is (most of the time) correct

Observation  △ Prior mean state  ◈ Prior/posterior Ensemble  □ Posterior mean state
EnKF Improves upon the ‘wrong sign’ adaptation problem

\[ X_{k|k} = X_{k|k-1} + K_k E_k \]

In this case most ensemble members move in the correct direction: so does the ensemble mean

\[
K_k = \frac{1}{N-1} A_{k|k-1} (HA)^T / P_{kk}^*,
\]

with \( P_{kk}^* = \frac{1}{N-1} HA (HA)^T + R_k \)

and \([HA]^i = h(x_{k|k-1}^i) - \frac{1}{N} \sum_{j=1:N} h(x_{k|k-1}^j)\)
Three further adaptations

- Getting rid of expensive matrix inversion
- Deterministic instead of stochastic resampling of $D$
- Localisation of the filter
Three further adaptations

• Getting rid of expensive matrix inversion
  ➔ Computational cost traditional EnKF: $O(m^3 + m^2N + mN^2 + nN^2)$
    - cubic in nr measurements: (2 as much ⇒ 8 times more cost)
    - quadratic in state variables and ensemble members
  ➔ Computational cost SMW alternative: $O(N^3 + mN^2 + nN^2)$
    - linear in measurements and state size
    - quadratic in ensemble members: 8

Sherman-Morrison-Woodbury (SMW) formula, e.g. W. Hager, SIAM Rev., 31(2), 221–239

\[
K_k = \frac{1}{N-1} A_{k|k-1} (HA)^T \left( \frac{1}{N-1} HA (HA)^T + R_k \right)^{-1}
\]

\[
R^{-1} \left[ I - \frac{1}{N-1} (HA) \left( I + (HA)^T R^{-1} \frac{1}{N-1} (HA) \right)^{-1} (HA)^T R^{-1} \right]
\]
ENKF FOR LARGE-SCALE ESTIMATION

Three further adaptations

- Getting rid of expensive matrix inversion
- Deterministic instead of stochastic resampling of \( D \)
  - Correct: ensemble spans about the same (model) subspace as the forecasting errors (else: filter divergence looms …)
  - Stochastic resampling may lead to clustering ensemble

\[
D_k = [d_k^1, \ldots, d_k^N], \text{with } d_k^i = y_k + e_k^i
\]

\[
X_{klk} = X_{klk-1} + K_k \left( D_k - h \left( X_{klk-1} \right) \right)
\]

\[
x_{klk} = x_{klk-1} + K_k \left( d_k - h \left( x_{klk-1} \right) \right)
\]

\[
A_{klk} = A_{klk-1} + K_k^A \left( D_k - AH_k \right)
\]

- Deterministic approach manipulates \( K_k^A \) so that innovation / residuals \( AH \) ALWAYS spread entire measurement space (with cov \( R \))

Three further adaptations

- Getting rid of expensive matrix inversion
- Deterministic instead of stochastic resampling of $D$
- Localisation of the filter
  - Further reduces computational costs (chops up Cov matrix => update equations in small bits)
  - Avoids spurious correlations
  - Increases effective ensemble size (and spread!)
  - *Note: localisation does minimise effect SMW formulation*

Different techniques:
- Least efficient (state based): $O(m^3n + m^2nN + mnN^2)$
- Most efficient (obs based): $O(mnN^2 + nN)$
LOCALISATION INCREASES EFFECTIVE ENSEMBLE SIZE
LOCALISATION INCREASES EFFECTIVE ENSEMBLE SIZE
LOCALISATION INCREASES EFFECTIVE ENSEMBLE SIZE
LOCALISATION INCREASES EFFECTIVE ENSEMBLE SIZE

![Graph showing velocity vs point on route]
LOCALISATION INCREASES EFFECTIVE ENSEMBLE SIZE
LOCALISATION INCREASES EFFECTIVE ENSEMBLE SIZE

Note: Localisation may induce “jumps” in the state, which may not be realistic.
SIMULATION NETWORK

- 264 km
- 4656 cells
- 2 seconds time step
- 592 (dual) loop detectors

Introduction
KF Basics
EnKF
Simulations
Conclusions
EXPERIMENTAL SETUP
EXPERIMENTAL SETUP
EXPERIMENTAL SETUP
EXPERIMENTAL SETUP
**RESULTS**

Ensemble size: 10, 20, 30, 40, 50 respectively
## RESULTS

<table>
<thead>
<tr>
<th></th>
<th>Localized EKF</th>
<th>Localized (D)EnKF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>Van Hinsbergen et al. (2012)</td>
<td><em>This research</em></td>
</tr>
<tr>
<td>Hardware</td>
<td>3.0 GHz dual-core, 2 GB RAM</td>
<td>2.6 GHz quad-core i5-3230M, 8 GB RAM</td>
</tr>
<tr>
<td>Length network</td>
<td>272 km</td>
<td>264 km</td>
</tr>
<tr>
<td>Model time step</td>
<td>5 s</td>
<td>2 s</td>
</tr>
<tr>
<td>Number of cells</td>
<td>1911</td>
<td>4656</td>
</tr>
<tr>
<td>Number of detectors in network</td>
<td>531</td>
<td>592</td>
</tr>
<tr>
<td>Number of used measure-</td>
<td>398</td>
<td>1184</td>
</tr>
<tr>
<td>ments</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Computational speed</td>
<td>$\approx 20 \times$ real-time</td>
<td>$\approx 40 \times$ real-time</td>
</tr>
</tbody>
</table>
CONCLUSIONS

• Localized DEnKF overall winner in this study …
  ➡ Faster than Localised EKF on similar case
  ➡ More accurate and faster than all other EnKF alternatives
  ➡ No need (on the contrary) for SWM reformulation

• Localization of EnKF is absolutely crucial:
  ➡ Higher estimation accuracy (larger effective ensemble)
  ➡ Lower computational burden

• Deterministic sampling beneficial
  ➡ Outperforms stochastic slightly
  ➡ More robust wrt smaller ensemble sizes
**NEXT STEPS**

Dutch National Data Warehouse Traffic Information

- In the coming few years we are going to work on implementing and further developing these ideas on real-data

- Many additional puzzles:
  - Network graph updating
  - Parameter estimation (dual filtering?)
  - Inflows / turns / ODs / Paths
  - Urban / provincial (different data, different state variables)