Prof. Dr. Hans van Lint
Traffic simulation & Computing, Civil Engineering & Geosciences, TU Delft
Director of the (Interfaculty) Msc Transport, Infrastructuur & Logistics

Many thanks to: Dr. Tamara Djukic, Dr. Thomas Schreiter, Prof. Dr. Serge Hoogendoorn

IPAM Tutorials

TRAFFIC STATE ESTIMATION BASICS

Contents two lectures

Today

- Basic ingredients State estimation
- Idiosyncrasies of traffic data
  - Bias due to time averaging
  - Cumulative drift
- Adaptive Smoothing Method
- Travel time estimation

Tomorrow

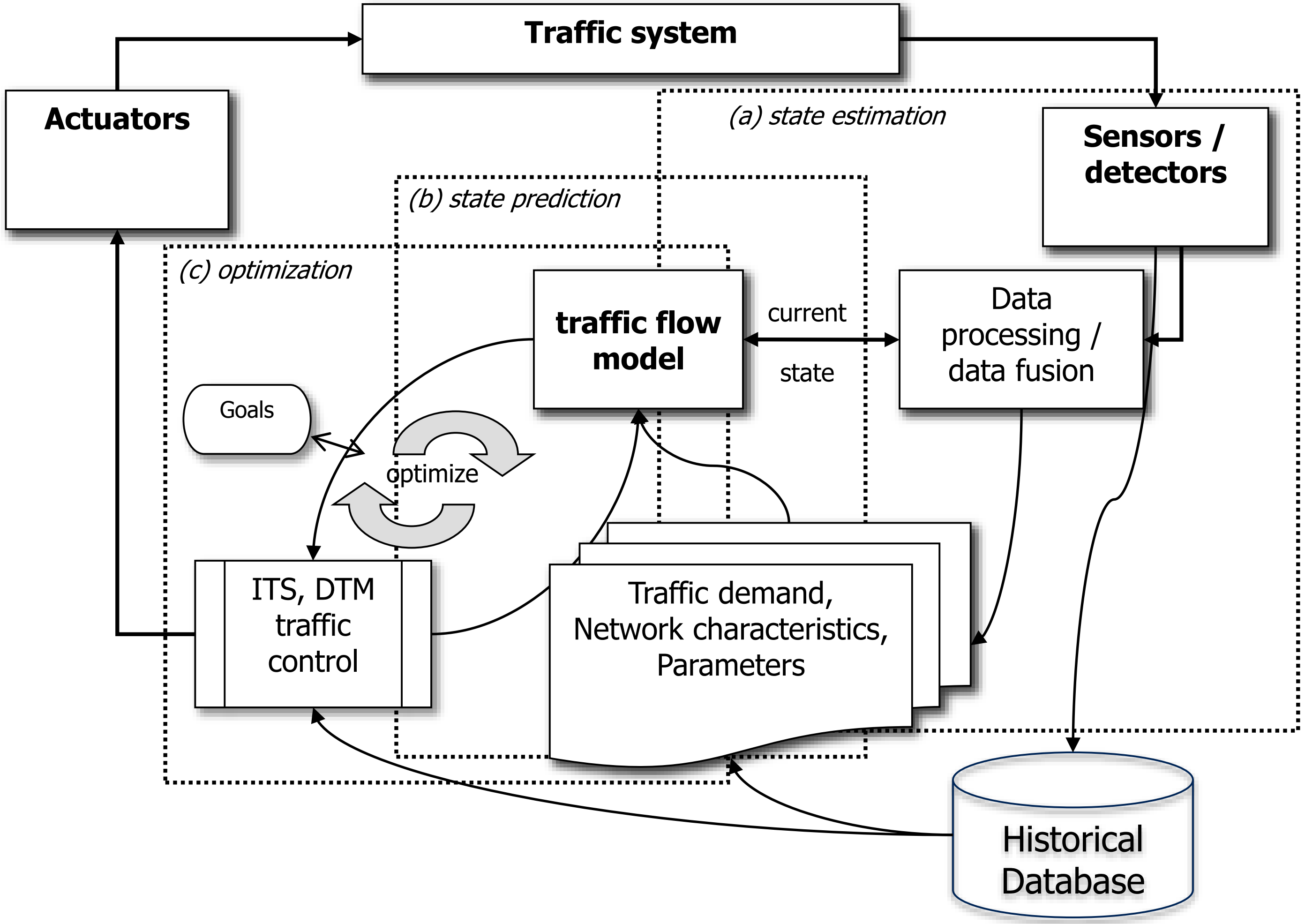
- The Kalman Filter
- All assumptions matter (through examples):
  - Tracking a bicycle trip
  - OD estimation
  - Traffic state estimation

Overall story

- State estimation refers to estimating (partially unobservable) state variables **x** from noisy, semantically different observations **y** using what ever theories and models we have relating these two
- Recipe:
  - Address structural bias in observations first! (due to time mean averaging, cumulative drift, etc). This can often be done using data-data consistency methods (use one variable to correct the next)
  - Optional (particularly for freeway data): use the ASM to smear / smooth the observations over time and space. This removes high frequency noise and gives you a rectangular grid of observations at a desired granularity to work with
  - Then use (E)KF to estimate the state variables, but be aware of the pittfals:
    - All assumptions matter and affect the estimation results
    - Resort to KF alternatives if the assumptions (linearizability, Gaussianity) do not match the problem: Ensemble KFs, Unscented KF, Particle filtering, etc

The (ideal) traffic control cycle

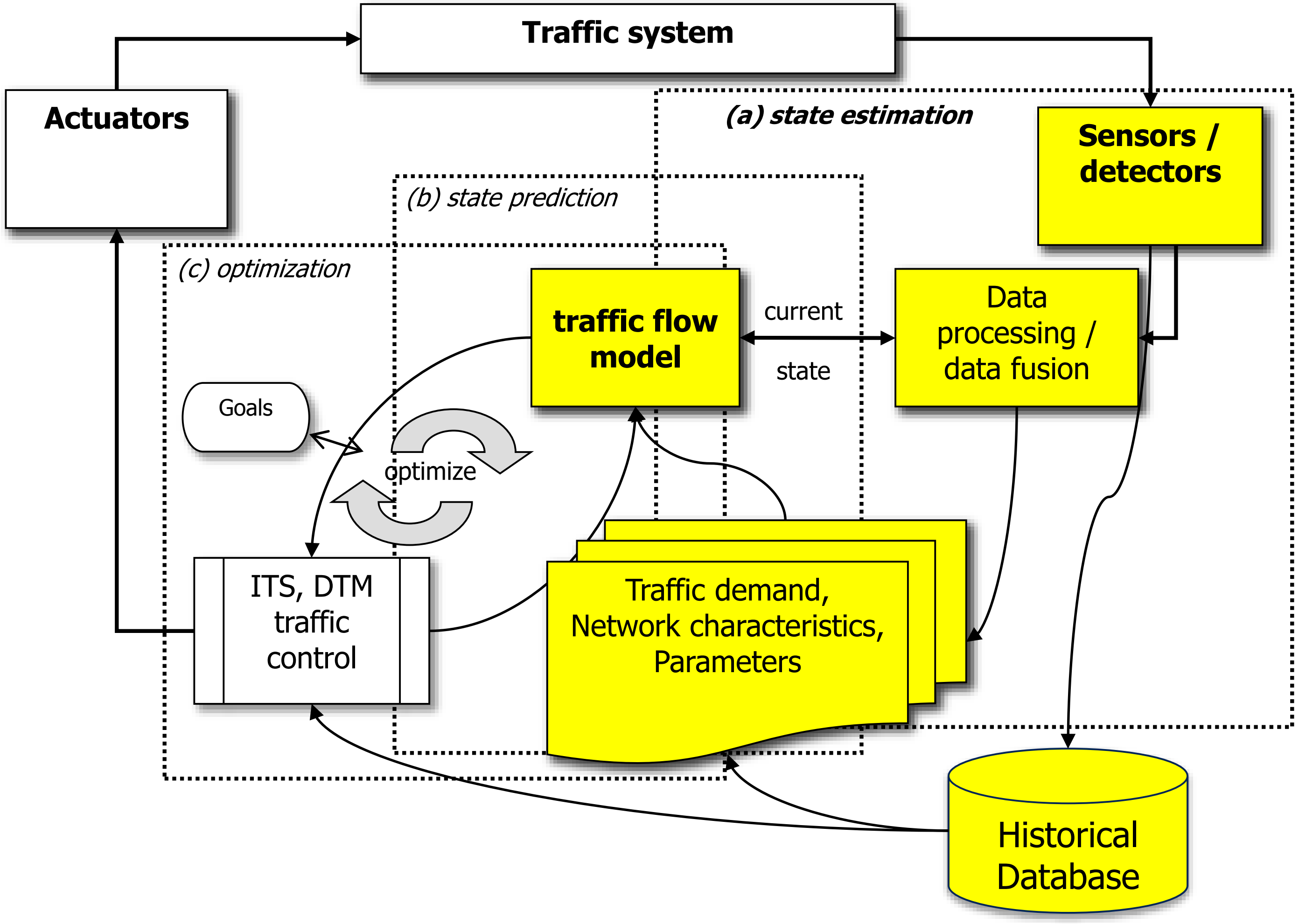





Crucial component: estimating state variables / inputs from data

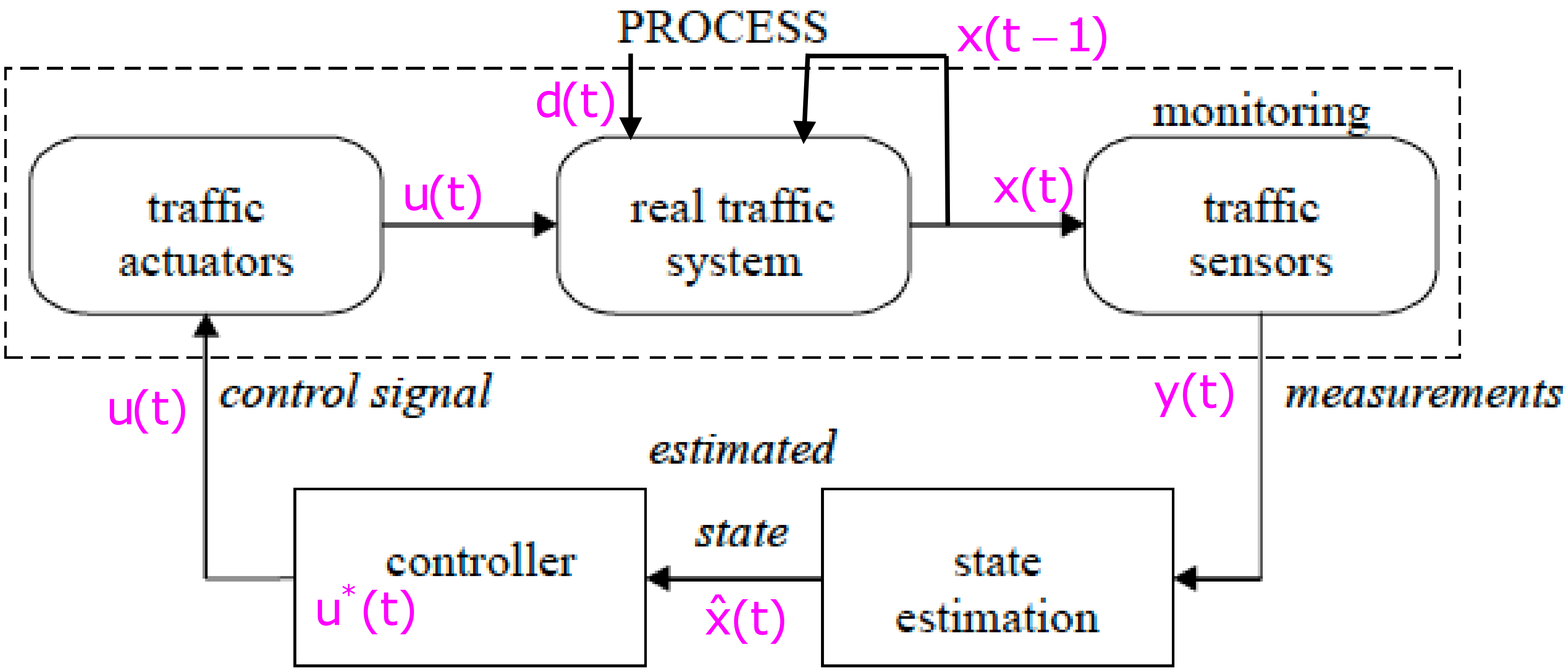





State estimation

The main idea is to translate the stuff that we measure (with sensors) into state information on the basis of which we can apply control or do other useful things



Different applications: different traffic state variables

	x(t)	y(t)	u(t)	d(t)
Coordinated ramp metering	density (+ speed, ramp queues)	Speeds, flows, occupancy	Greentimes (fraction ramp flow allowed in)	inflows, turns (or OD+splits), spillback from upstream, …
Route guidance	Nr Vehicles queuing (for bottlenecks)	queue lengths, travel times, Speeds, flows, occupancy	provided information / guidance	Inflows, compliance / behavior, captives, …
Network management	Accumulation, spatial variability Accumulation, OD flows	Aggregate speeds, flows, travel times	Perimeter control	Inflow, outflow constraints from other networks, OD flows

Ingredients traffic state estimation

Theories, assumptions, models which describe the dynamics of $x(t)$ and the relationship

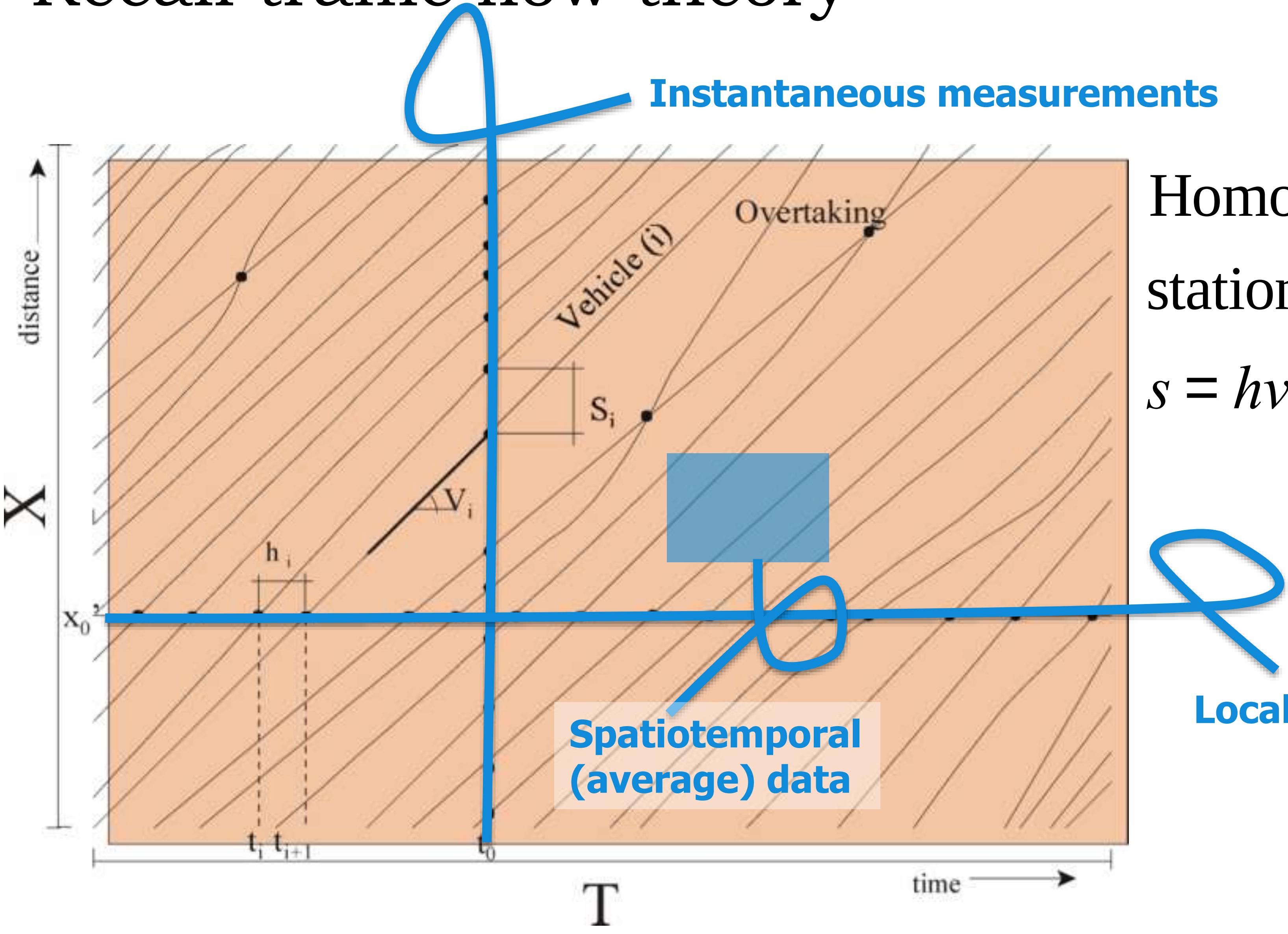
$$d\mathbf{x}/dt = g(\mathbf{x}(t), \ldots)$$

$$\mathbf{y}(t)=h(\mathbf{x}(t), \ldots)$$

data $y(t)$ (from whatever source) related to our state variables $x(t)$

Data Assimilation Tools & Techniques

Recall traffic flow theory



Homogeneous &

stationary conditions

$$s = hv \xrightarrow{q=1/\bar{h};\, r=1/\bar{s}} q = ru$$

$q = \rho u$, but which u ?

Local averaging

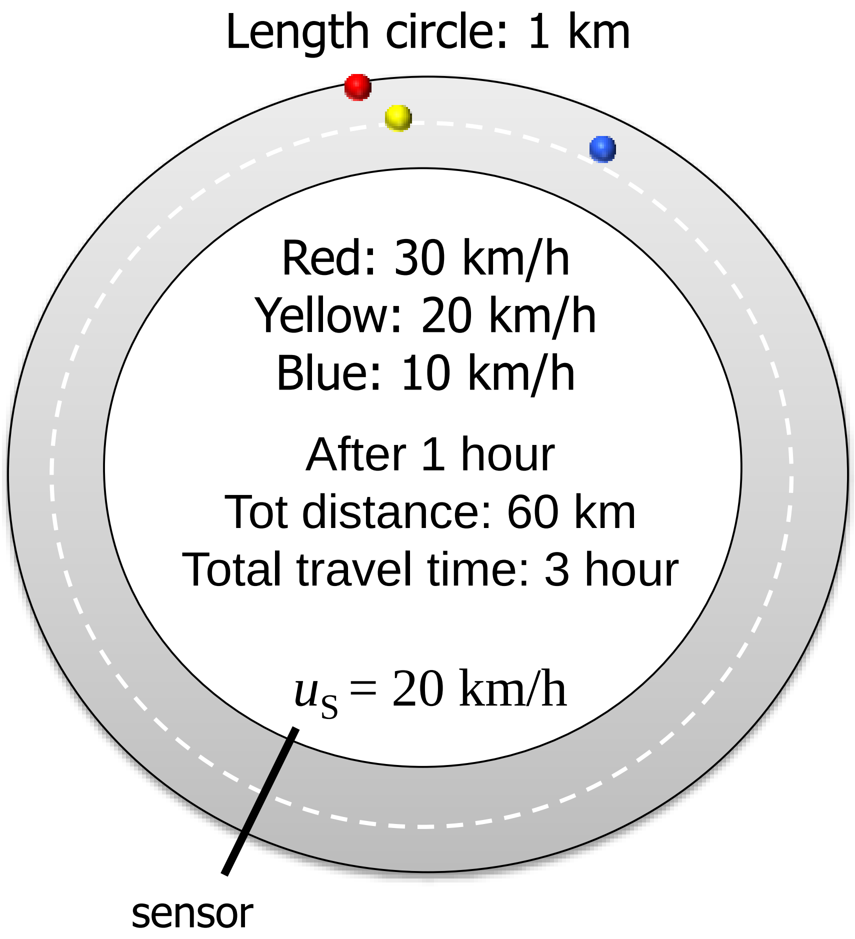
$$z_L = \frac{1}{\sum_i q_i} \sum_i q_i z_i \xrightarrow{q_i = 1, z_i = v_i, \forall i} u_L = \frac{1}{n} \sum_i v_i$$

Spatial averaging

$$z_S = \frac{1}{\sum_i \rho_i} \sum_i \rho_i z_i \xrightarrow{\text{q}=\rho\text{u (homogeneous \& stationary conditions)}}$$

$$z_S = \frac{1}{\sum_i q_i / u_i} \sum_i (q_i / u_i) z_i \xrightarrow{q_i = 1 \text{ (i.e.: } u_i \equiv v_i\text{)}, z_i = v_i, \forall i}$$

$$u_S = \frac{1}{\frac{1}{n} \sum_i \frac{1}{v_i}}$$



u_L= 23 km/h !
(17% overestimation)

Traffic variables

	Local measurements	**Instantaneous measurements**	**Spatiotemporal data (Edie)**
Variable	Cross-section *x* Period *T*	Road segment *X* Time *t*	Road segment *X* Period *T*
flow ***q*** **(veh/h)**	$q = \frac{n}{T} = \frac{1}{\bar{h}}$	$q = \rho u$	$q = \frac{\sum_i d_i}{XT}$
density ***ρ*** **(veh/km)**	$\rho = \frac{q}{u}$	$\rho = \frac{n}{X} = \frac{1}{\bar{s}}$	$\rho = \frac{\sum_j tt_j}{XT}$
Av. Speed ***u*** **(km/h)**	$u = \frac{n}{\sum_i (1/v_i)}$	$u = \frac{\sum_j v_j}{n}$	$u = \frac{q}{\rho}$

Ideal (the stuff we need)

	Local measurements	**Instantaneous measurements**	**Spatiotemporal data (Edie)**
Variable	Cross-section x Period T	Road segment X Time t	Road segment X Period T
flow ***q* (veh/h)**	$q = \frac{n}{T} = \frac{1}{\bar{h}}$	$q = \rho u$	$q = \frac{\sum_i d_i}{XT}$
density ***ρ* (veh/km)**	$\rho = \frac{q}{u}$	$\rho = \frac{n}{X} = \frac{1}{\bar{s}}$	$\rho = \frac{\sum_j tt_j}{XT}$
Av. Speed ***u* (km/h)**	$u_L = \frac{n}{\sum_i (1/v_i)}$	$u = \frac{\sum_j v_j}{n}$	$u = \frac{q}{\rho}$

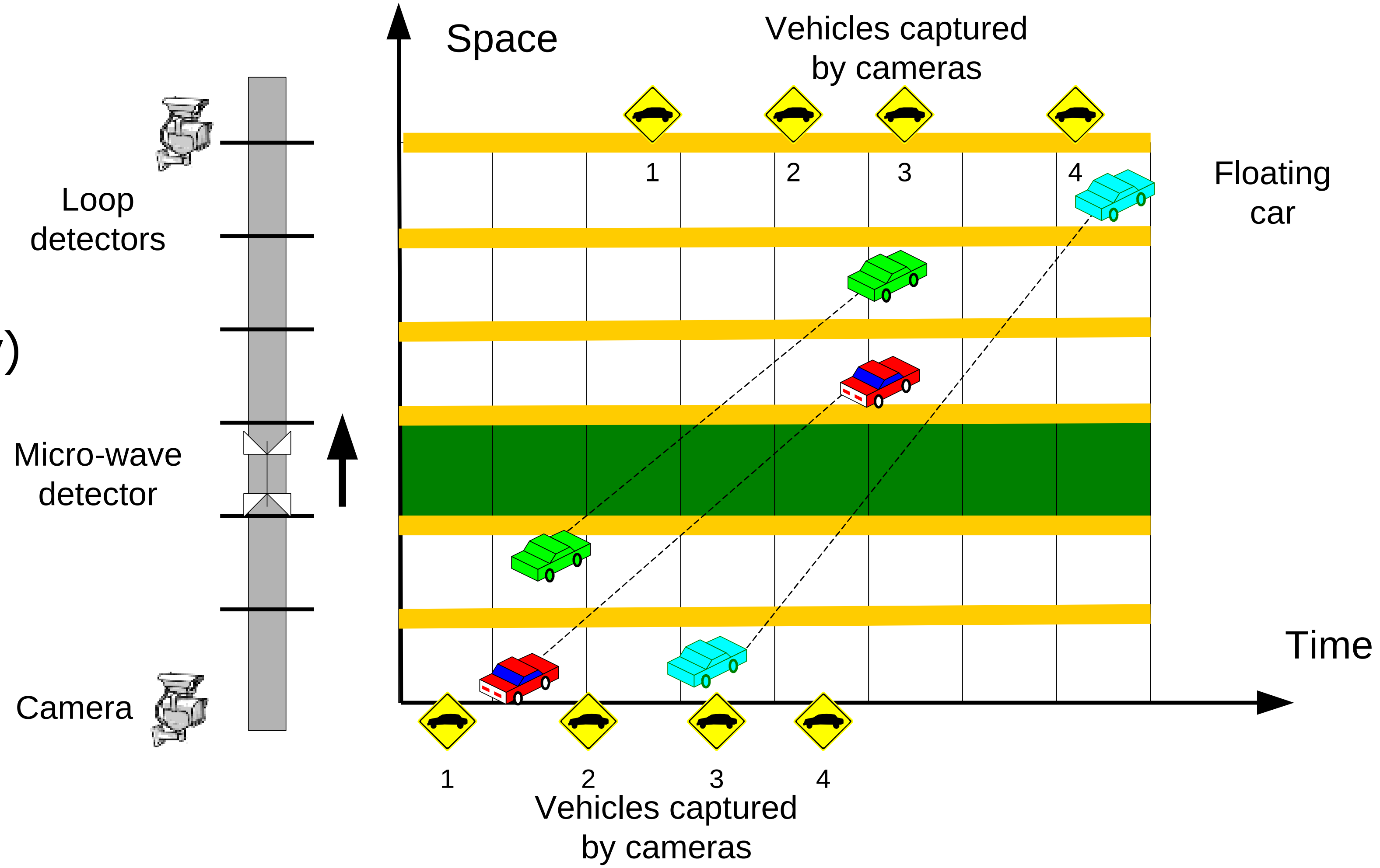
Practice (the stuff we have to work with)

	Local measurements	**Instantaneous measurements**	**Spatiotemporal data (Edie)**
Variable	Cross-section *x* Period *T*	Road segment *X* Time *t*	Road segment *X* Period *T*
flow ***q* (veh/h)**	$q = \frac{n}{T} = \frac{1}{\bar{h}}$	$q = \rho u$	$q = \frac{\sum_i d_i}{XT}$
density ***ρ* (veh/km)**	$\rho = \frac{q}{u}$	$\rho = \frac{n}{X} = \frac{1}{\bar{s}}$	$\rho = \frac{\sum_j tt_j}{XT}$
Av. Speed ***u* (km/h)**	$u_L = \frac{n}{\sum_i (1/v_i)}$	$u = \frac{\sum_j v_j}{n}$	$u = \frac{q}{\rho}$

Different traffic data sources tell (literally) different stories

Challenges for traffic state estimation

- Differences in spatiotemporal coverage, aggregation, semantics
  - e.g. travel time (journey) versus local speed
- Noise, Bias
- Observability
- Availability, completeness



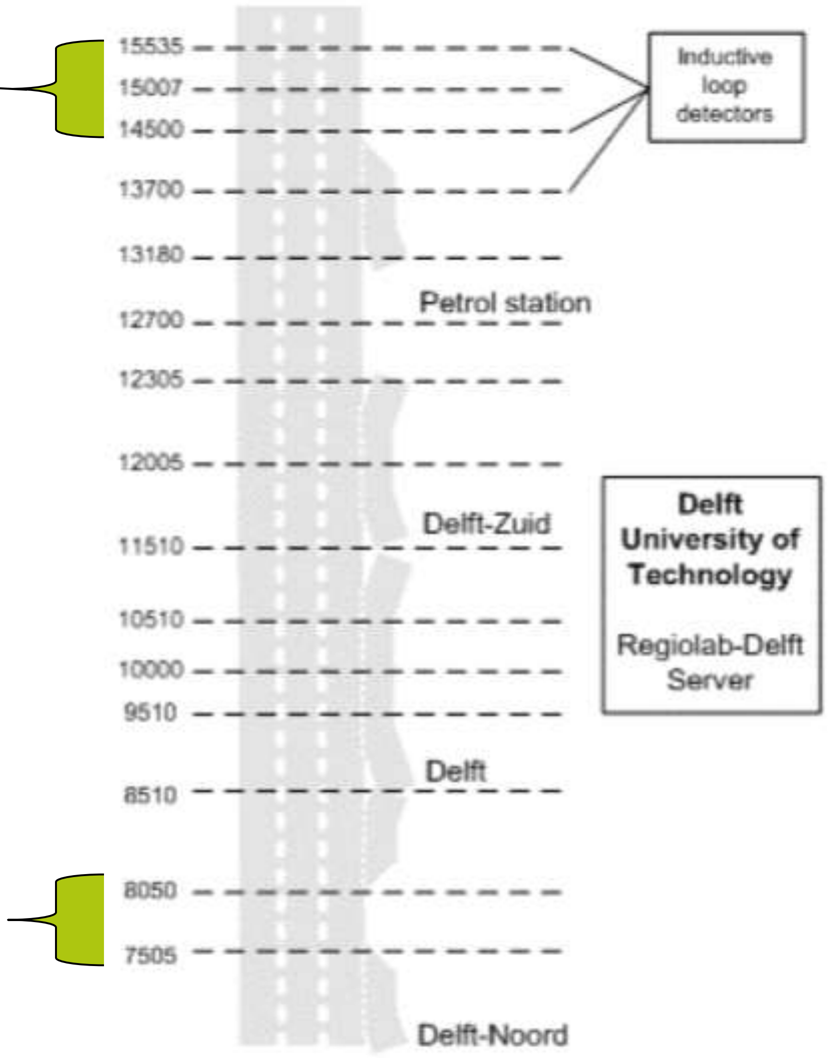
Example measurement noise with actual data

- Most important implication:
  - Drift in cumulative balance

	section 7505m – 8050m	section 14500m-15535m
Inflow [veh]	64823	68357
Outflow [veh]	71000	67368
Mass balance [veh]	-6177 ≈-9.5%	989 ≈1.5 %

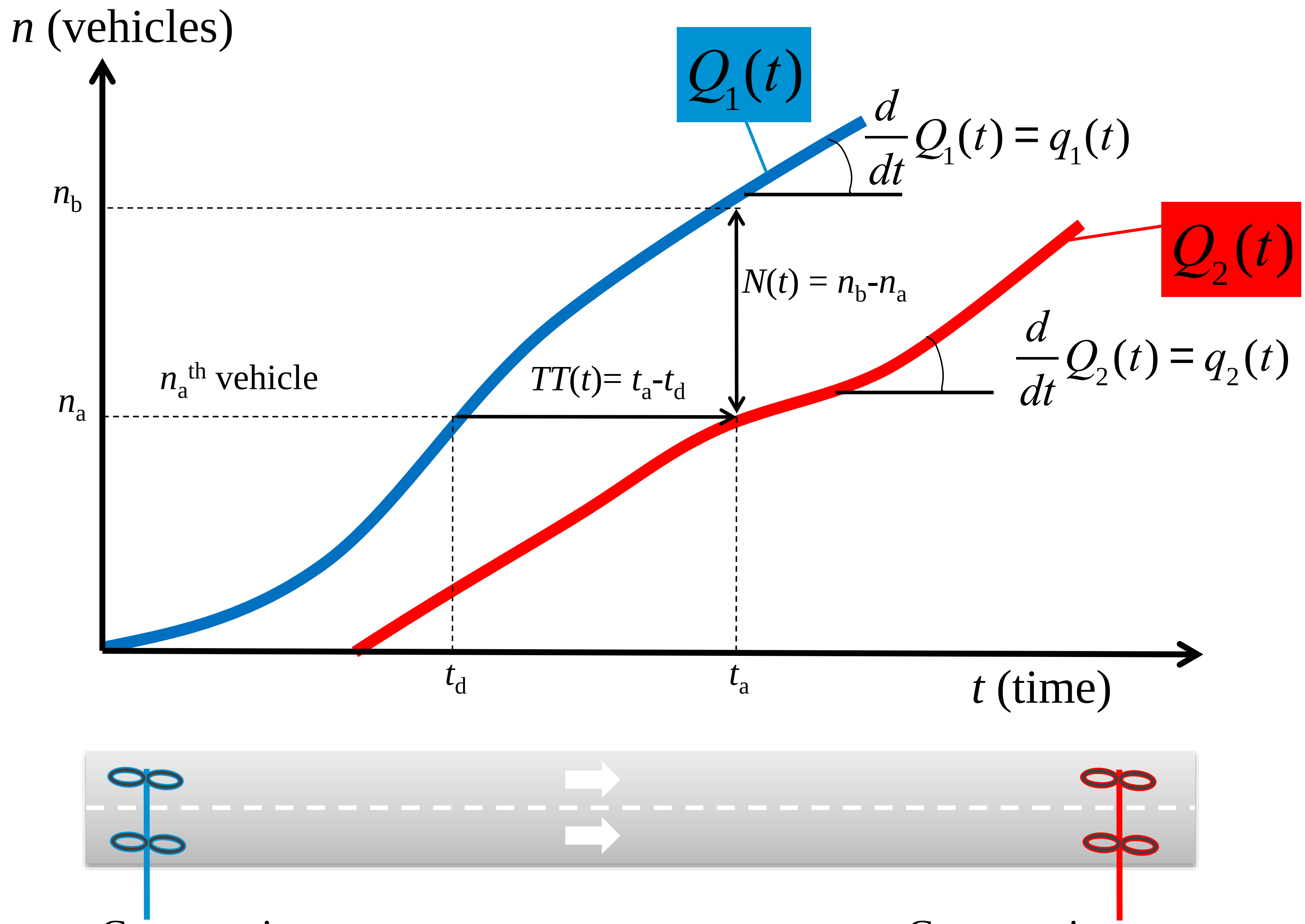
Mass balance between different detector stations along the A13 freeway over the period 6:00-20:00 (data from 2008)

- Causes:
  - (random) detector errors
  - Errors in **network description**
  - Errors in **timestamps** (did a vehicle arrive in period t or t+1?)



Example approach to solving structural errors in traffic data

- Van Lint, H. and S.P. Hoogendoorn. A Generic Methodology to Estimate Vehicle Accumulation on Urban Arterials by Fusing Vehicle Counts and Travel Times. in Transportation Research Board Annual Meeting. 2015. Washington D.C.: The National Academies.



Cross section x_1

- Flow: q_1 (veh/u)

Cross section x_2

- Flow: q_2 (veh/u)

The cumulative drift problem

- Source errors (miscounts, double counts):
  - lane changes, power failure, etc.
- Errors may be random or structural (bias)
- Consequence:

$$\left.\begin{aligned} N(t) &= \int_t q_1(s)ds - \int_t q_2(s)ds \\ q_i(t) &= \hat{q}_i(t) + \varepsilon_i(t) \end{aligned}\right\}$$

With e.g. $\varepsilon_i(t) \sim \mathcal{N}(\mu, \sigma)$

The cumulative drift problem

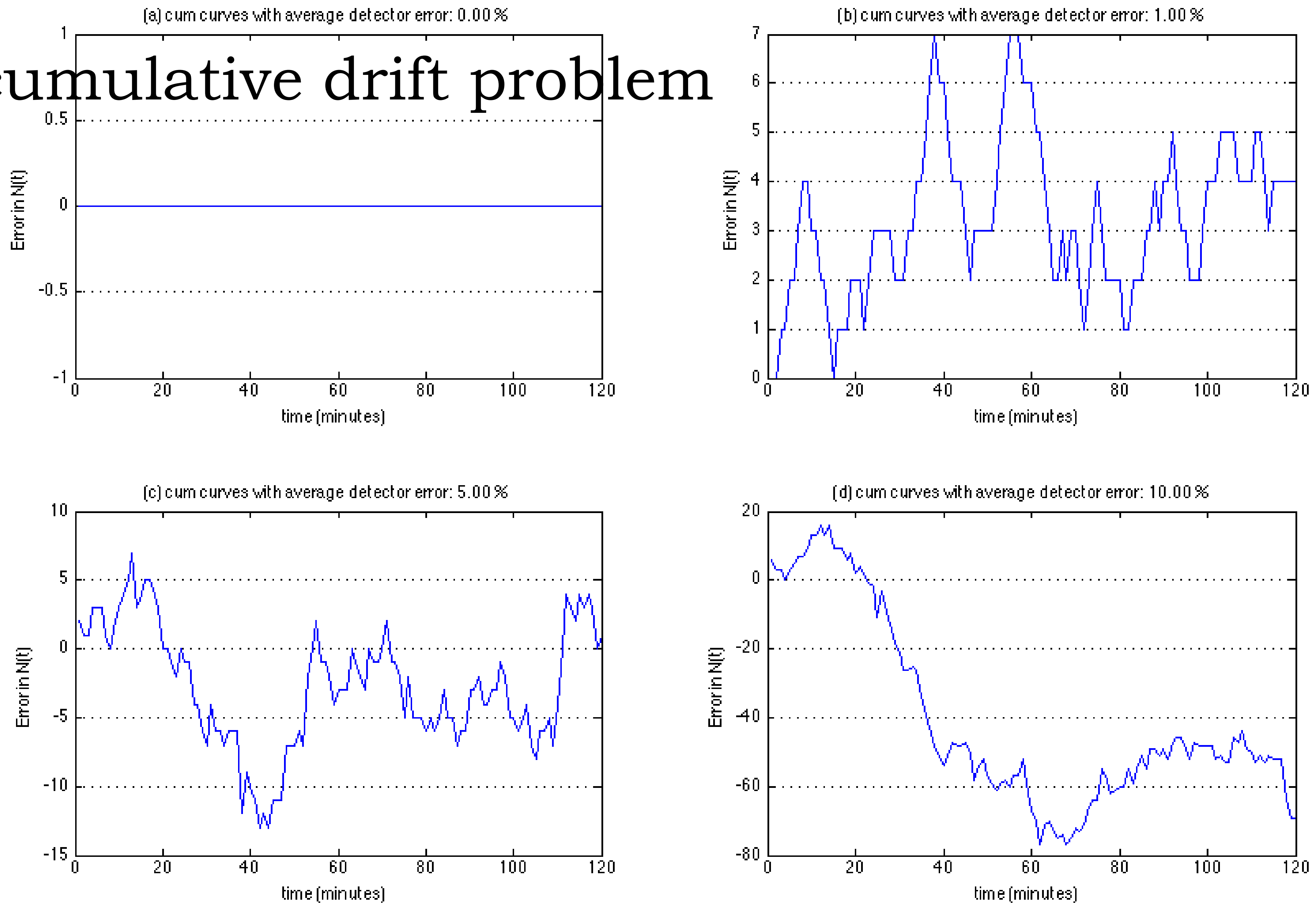
- Source errors (miscounts, double counts):
  - lane changes, power failure, etc.
- Errors may be random or structural (bias)
- Consequence:

$$\left.\begin{aligned} N(t) &= \int_t q_1(s)ds - \int_t q_2(s)ds \\ q_i(t) &= \hat{q}_i(t) + \varepsilon_i(t) \end{aligned}\right\} N(t) = \hat{N}(t) + \int_t \big(\varepsilon_1(s) - \varepsilon_2(s)\big)ds$$

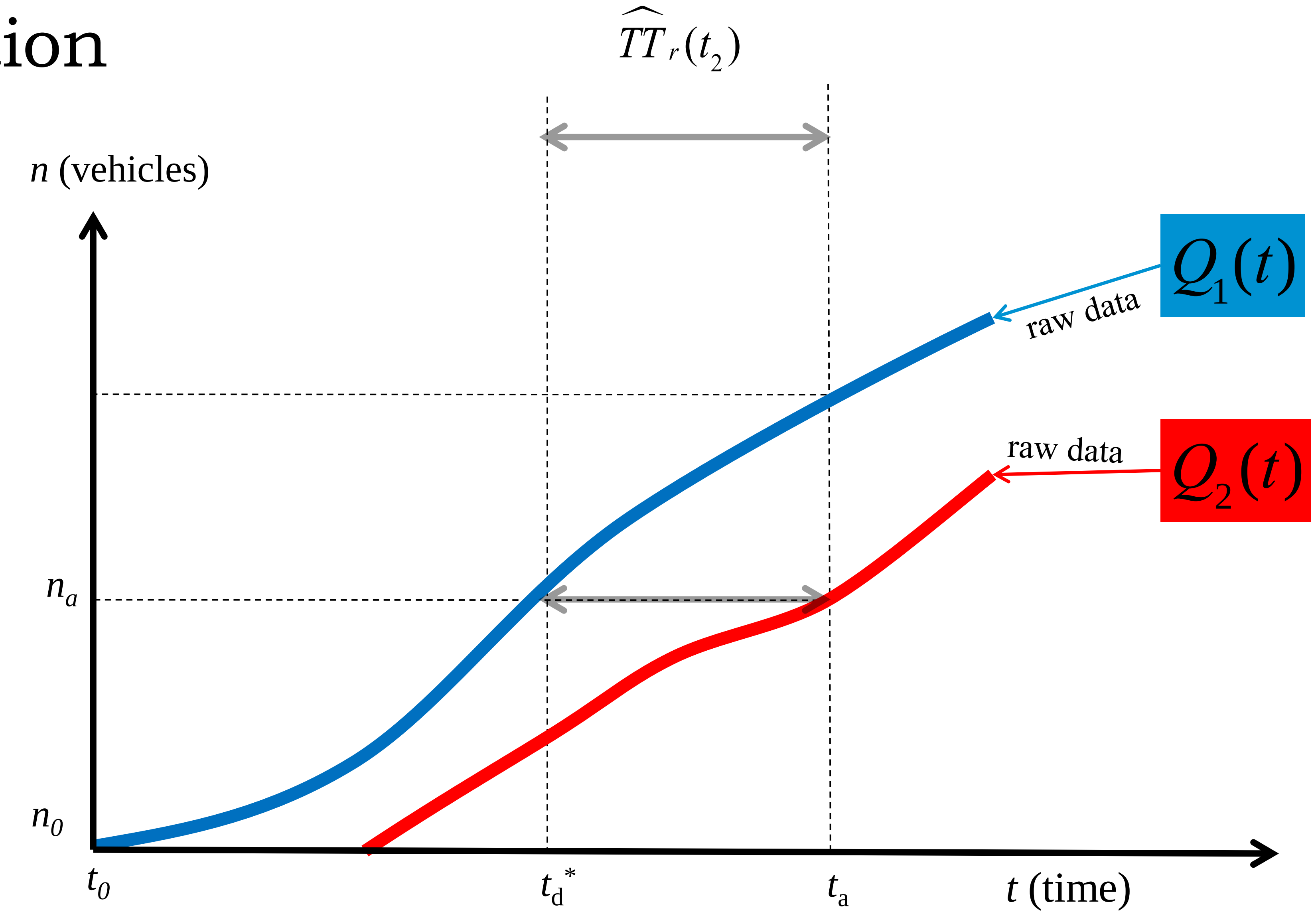
This is a random walk!

(which means vehicle accumulation is practically unobservable using counts)

The cumulative drift problem



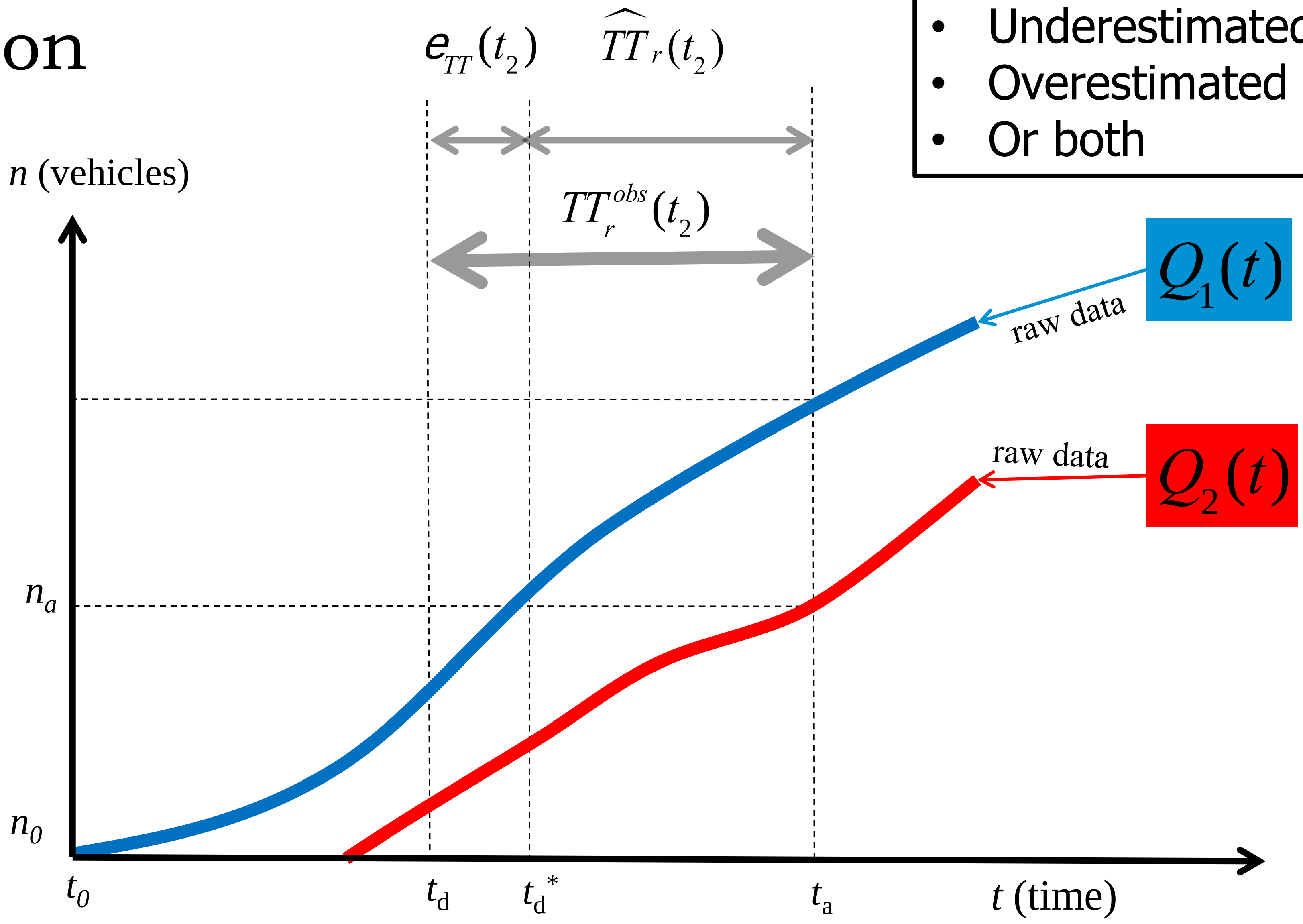
Solution



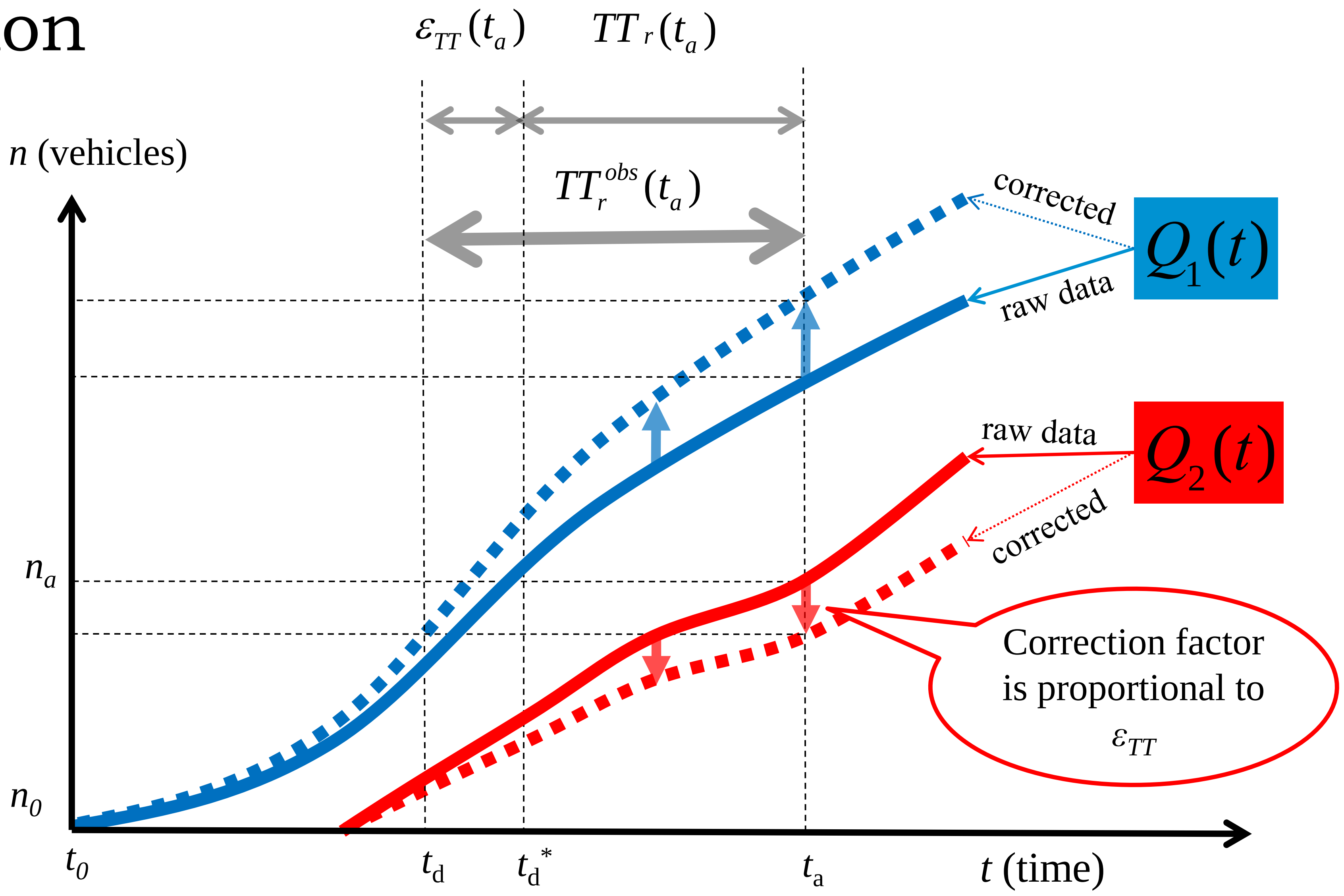
$\widehat{TT}_r(t_2)$
n (vehicles)
n_a
n_0
t_0
t_d^*
t_a
t (time)
raw data
$Q_1(t)$
raw data
$Q_2(t)$

Solution

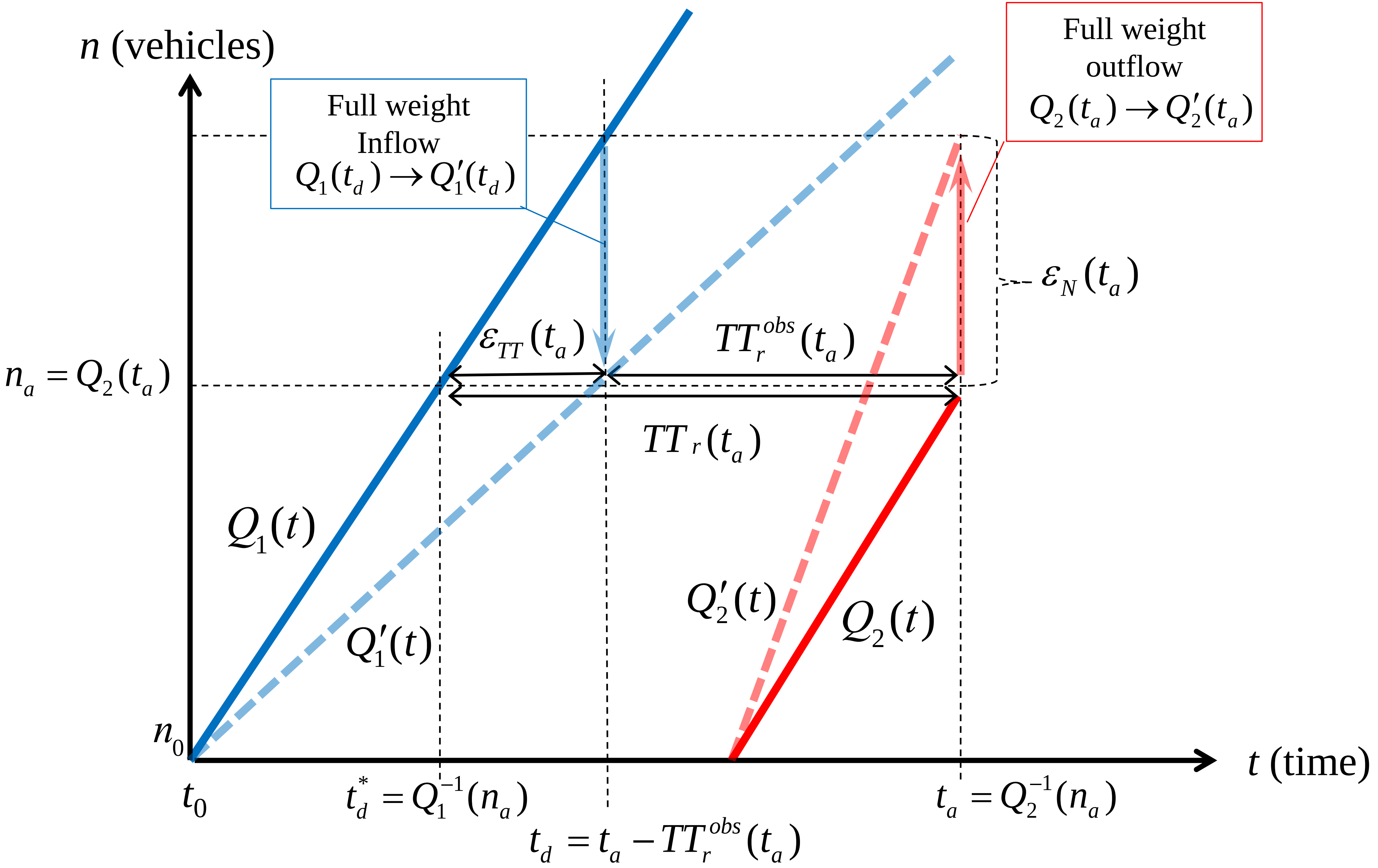
Implies we either
- Underestimated inflow
- Overestimated outflow
- Or both



Solution



$\varepsilon_{TT}(t_a)$
$TT_r(t_a)$
$TT_r^{obs}(t_a)$
n (vehicles)
corrected
raw data
$Q_1(t)$
raw data
$Q_2(t)$
corrected
Correction factor is proportional to ε_{TT}
n_a
n_0
t_0
t_d
t_d^*
t_a
t (time)



n (vehicles)
Full weight
Inflow
$Q_1(t_d) \to Q_1'(t_d)$
Full weight
outflow
$Q_2(t_a) \to Q_2'(t_a)$
$\varepsilon_N(t_a)$
$\varepsilon_{TT}(t_a)$
$TT_r^{obs}(t_a)$
$n_a = Q_2(t_a)$
$TT_r(t_a)$
$Q_1(t)$
$Q_2'(t)$
$Q_2(t)$
$Q_1'(t)$
n_0
t (time)
t_0
$t_d^* = Q_1^{-1}(n_a)$
$t_a = Q_2^{-1}(n_a)$
$t_d = t_a - TT_r^{obs}(t_a)$

Mathematically

- Correction factor can be expressed as function of known quantities

$$\frac{\varepsilon_N(t_a)}{\varepsilon_{TT}(t_a)} = \frac{n_a - n_0}{t_d - t_0} \Rightarrow \varepsilon_N(t_a) = \varepsilon_{TT}(t_a)\frac{Q_i(t_a) - n_0}{t_a - TT_r^{obs}(t_a) - t_0}$$

Cum count downstream

Correction cumulative curve

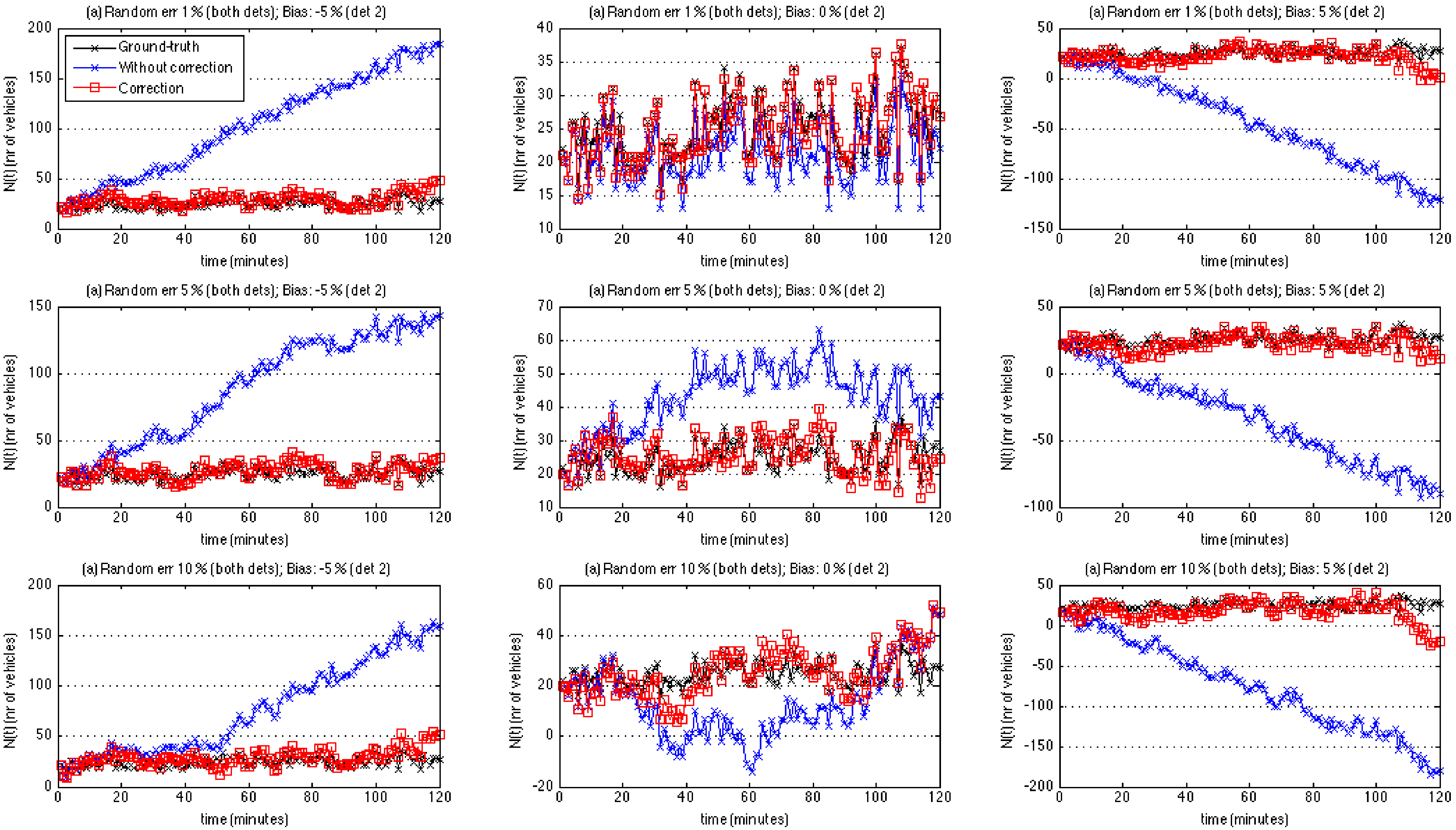
Error travel time

$\{t_0,n_0\}$ is the reference point

- You can choose which of the two curves you want to adjust by setting α

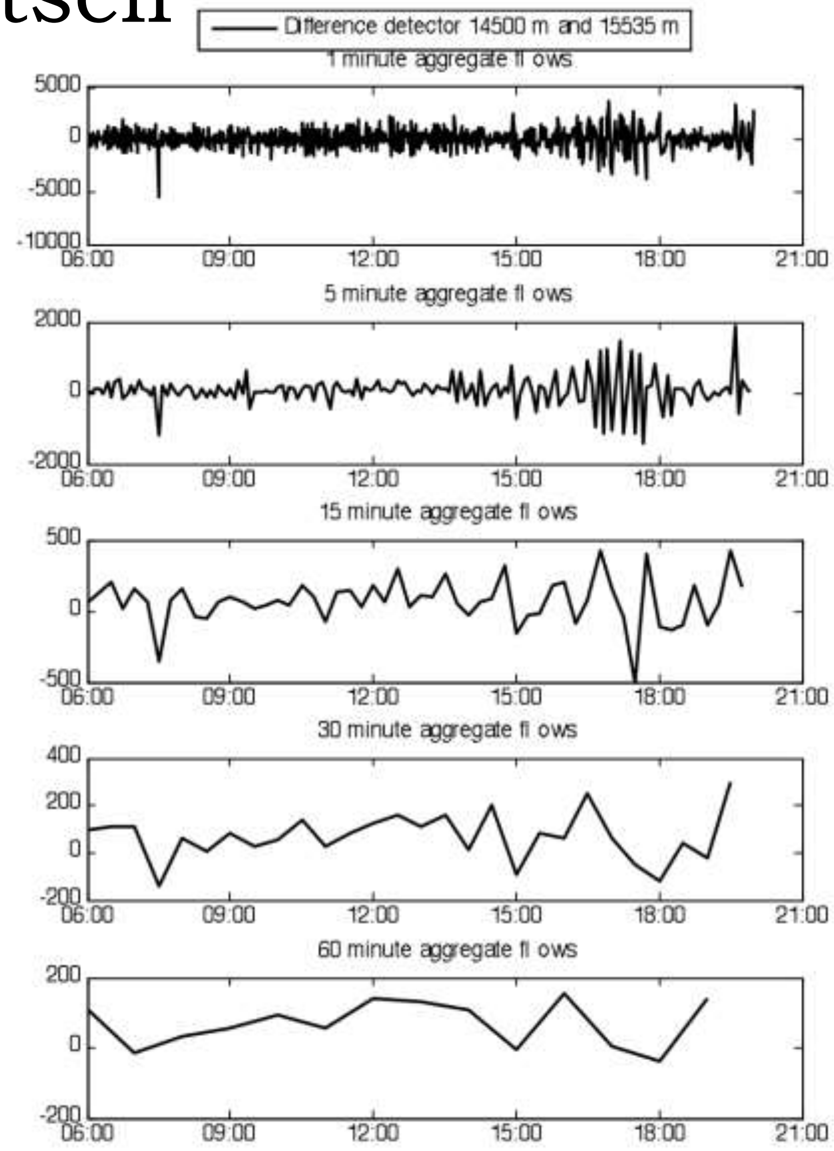
$$Q_1'(t_d) = Q_1(t_d) - (1-\alpha)\times\varepsilon_N(t_a)$$
$$Q_2'(t_a) = Q_2(t_a) + \alpha \times\varepsilon_N(t_a)$$

- You can also use it recursively on a series of detectors starting from the most upstream couple of detectors and work your way downstream (using α=1)



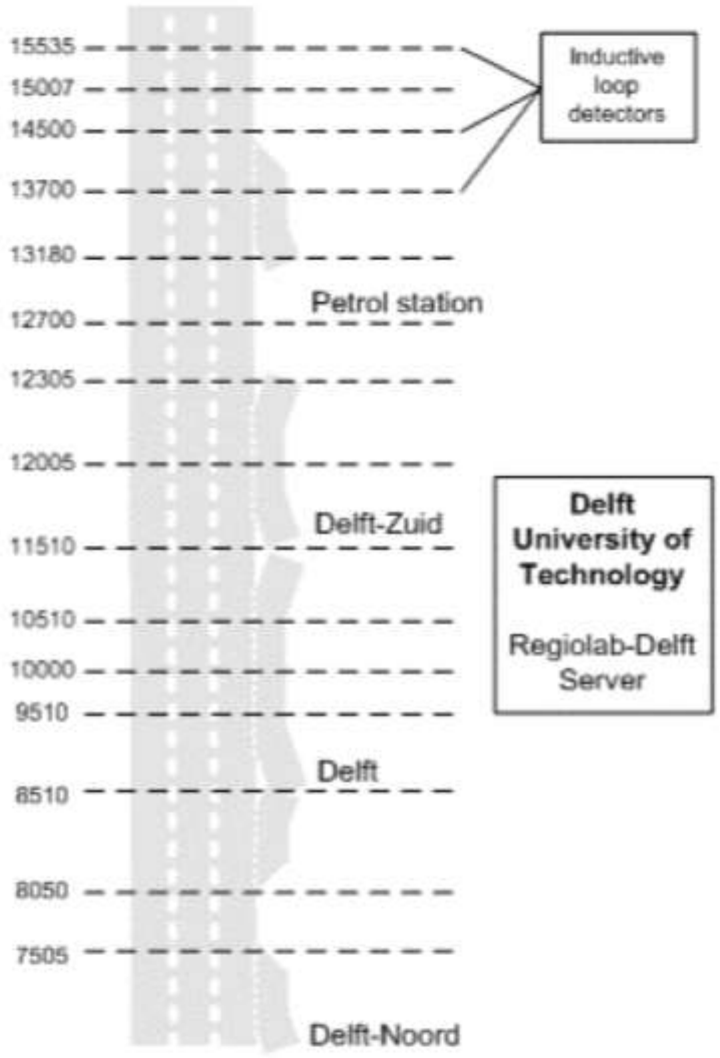
Rows (random errors): {1%, 5%, 10%}; Columns (bias): {-5%, 0, 5%};

Final note: there is also “noise” in the process itself



Mass balance between two consecutive detectors at different time aggregation levels

The smaller the time aggregation the larger variance of course …



Ingredients traffic state estimation

Theories, assumptions, models which describe the dynamics of $x(t)$ and the relationship

$$dx/dt = g(\mathbf{x}(t), \ldots)$$

$$\mathbf{y}(t)=h(\mathbf{x}(t), \ldots)$$

data $y(t)$ (from whatever source) related to our state variables $x(t)$

Data Assimilation Tools & Techniques

Usable theories / assumptions

Recall traffic flow theory

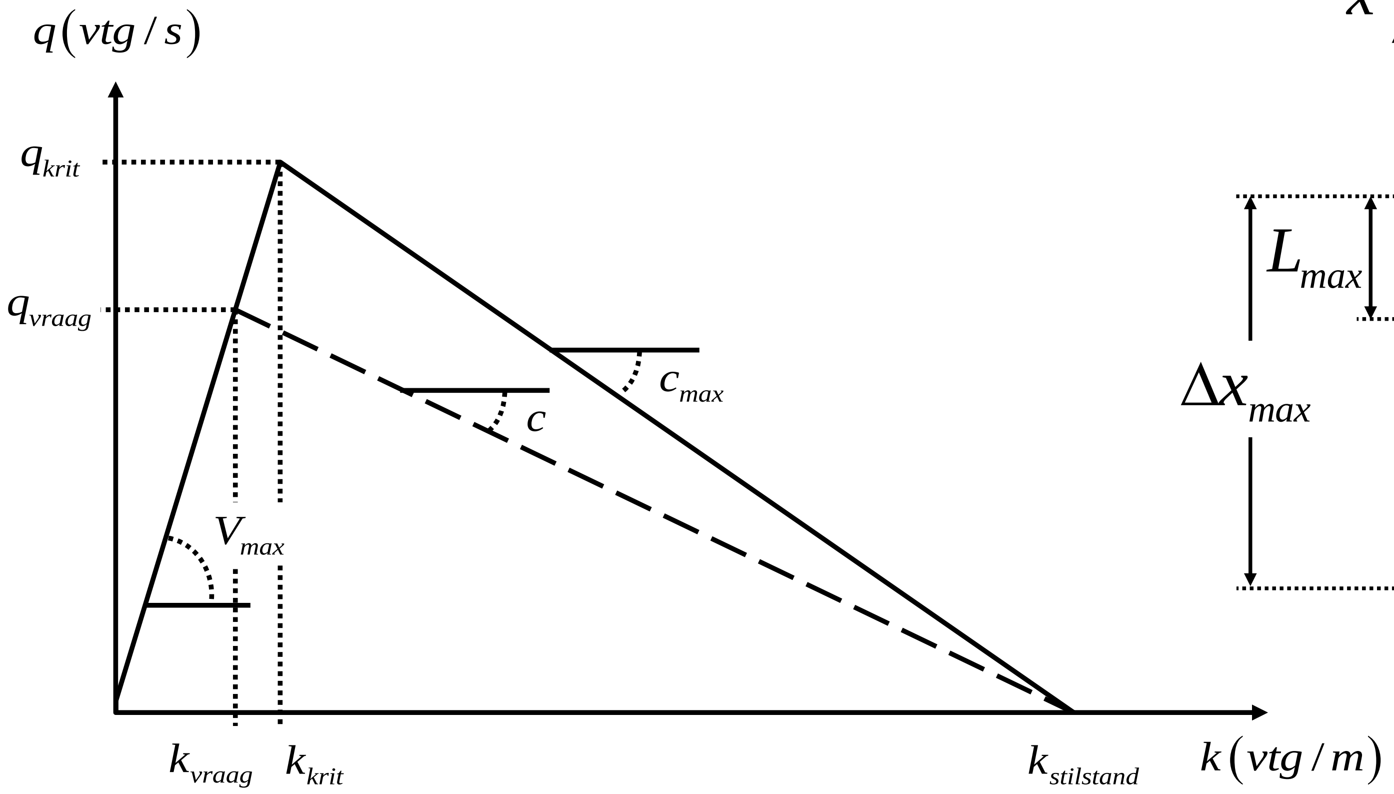
- Basic (“physical”) relationships between variables:
  - $\rho = q/u$
  - Travel time = L/u

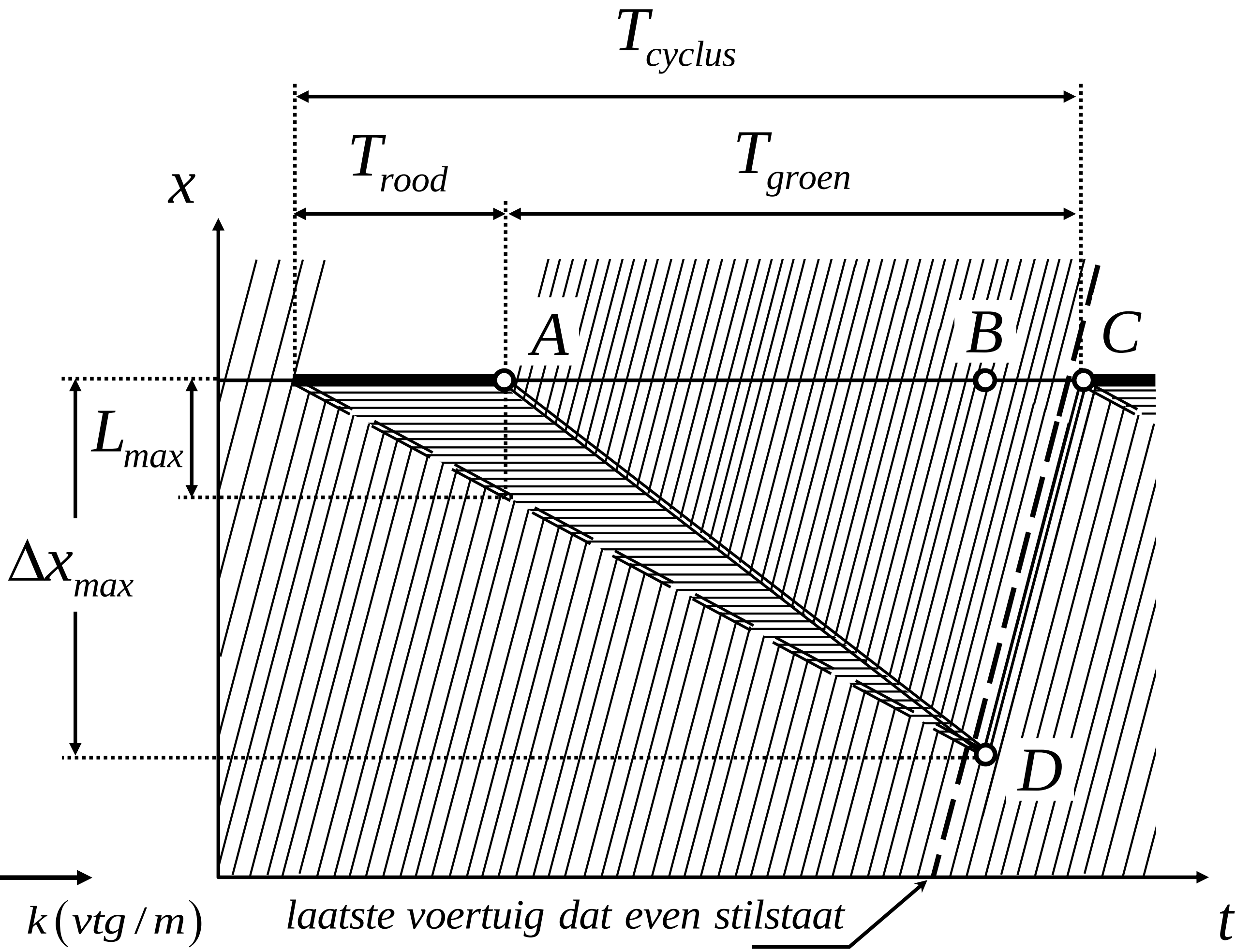
- The fundamental diagram = behavioral (but statistical) relationship
  - $q = Q^e(\rho)$
  - $u = U^e(\rho)$

- Shockwave / kinematic wave theory
- More complex/involved traffic flow simulation models – anything that encapsulates our knowledge about traffic flow operations

Queue at intersection = Queue on freeway?

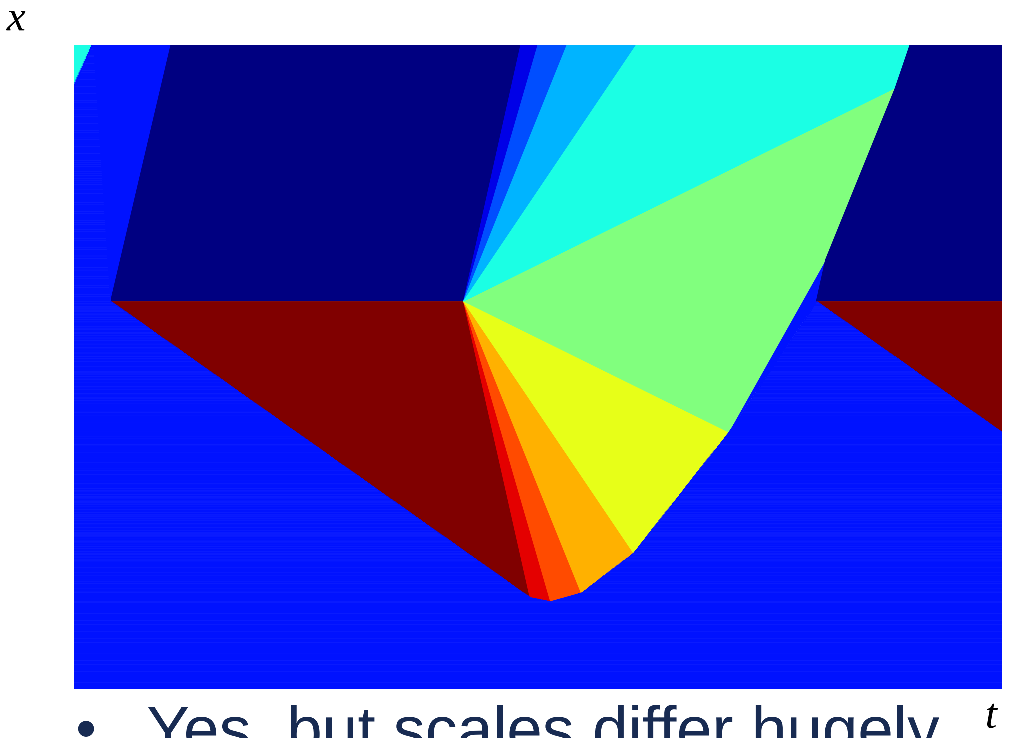
Recall traffic flow theory





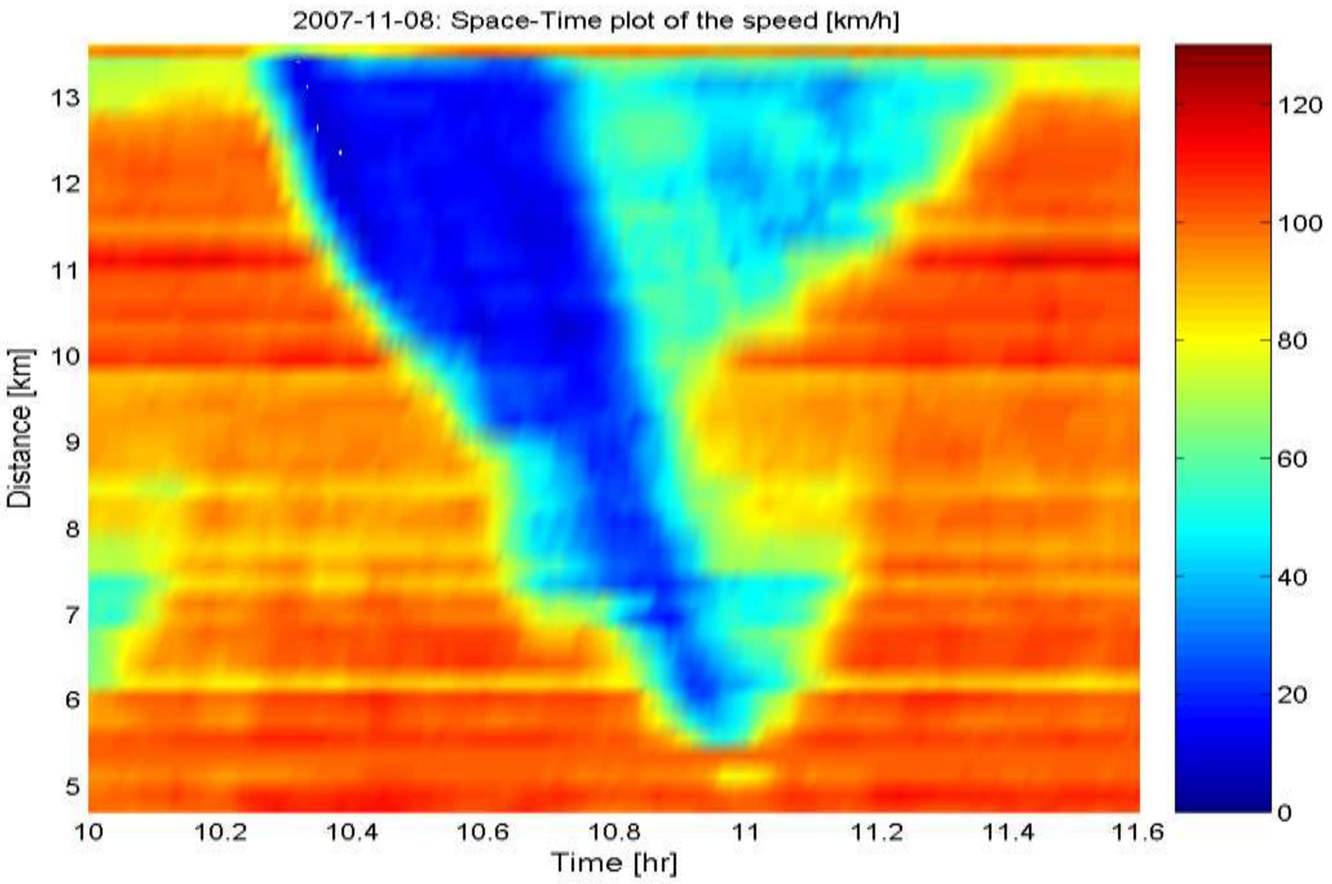
Queue at intersection = Queue on freeway?

Queue traffic light

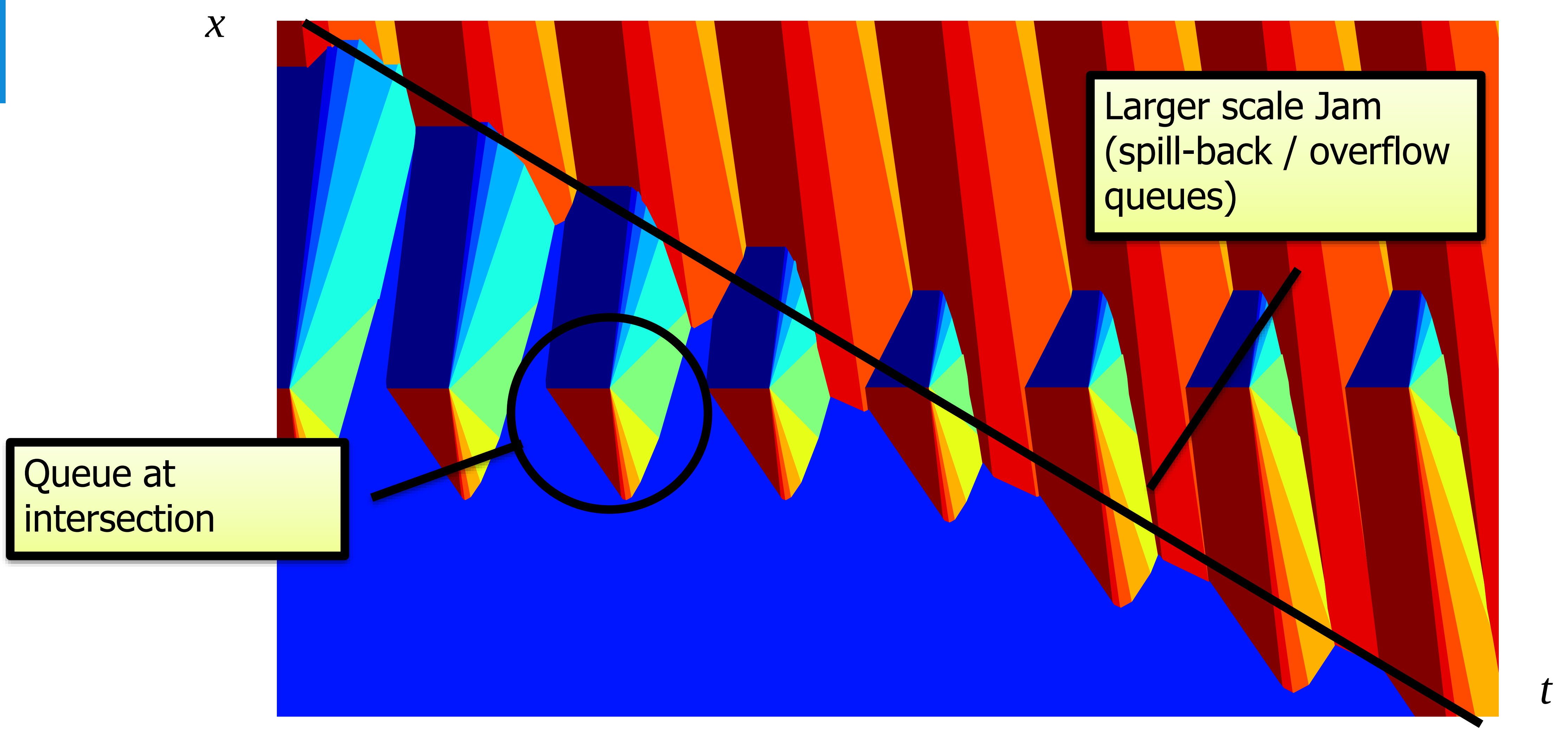


- Yes, but scales differ hugely
  – In space AND time
- Urban: more difficult!

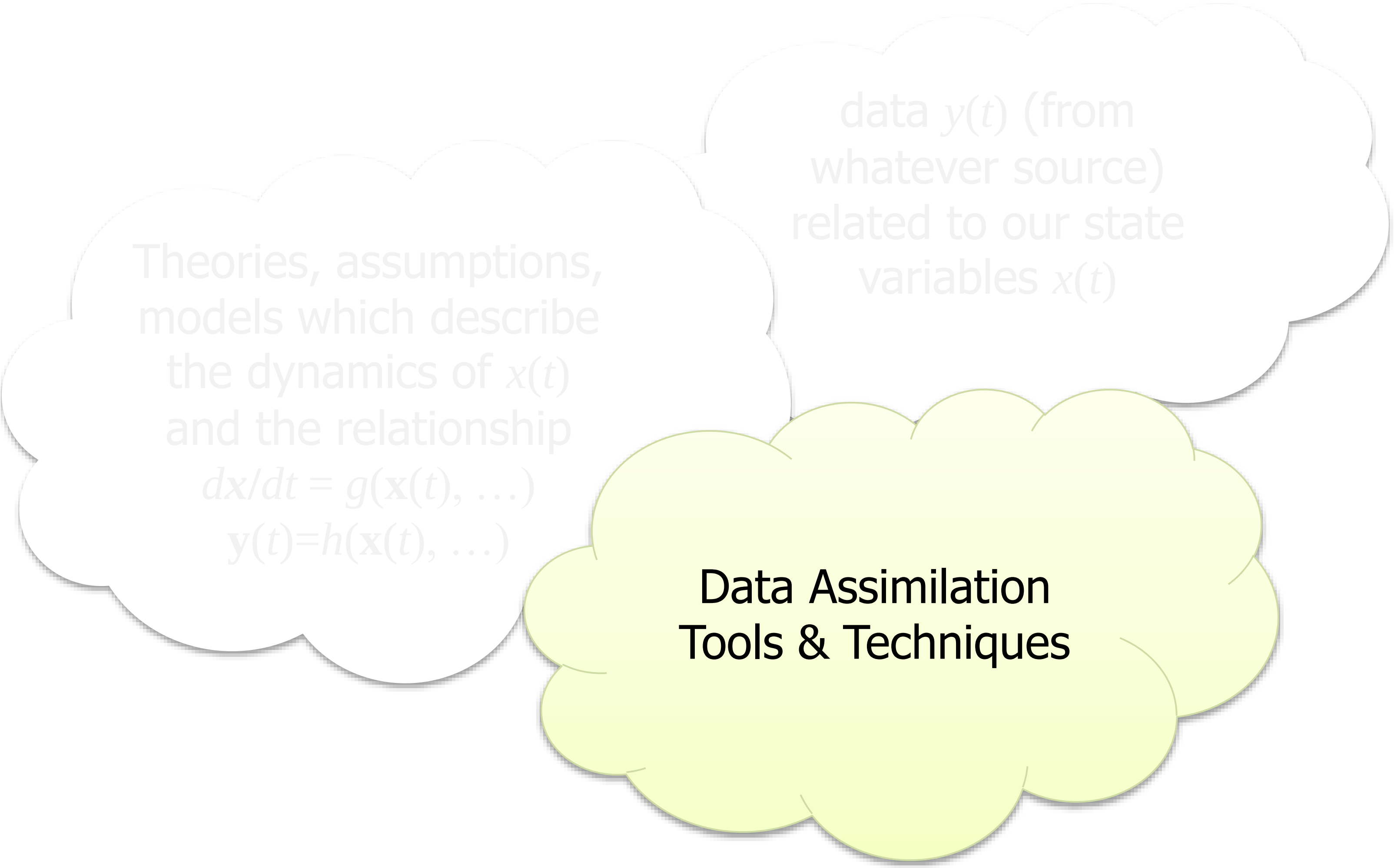
Queue due to accident on freeway



Freeway traffic jam versus intersection queuing



Ingredients traffic state estimation



The simplest but still (traffic flow theoretically) correct way of smoothing traffic data

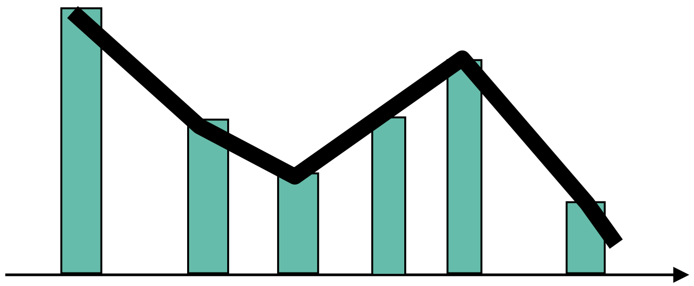
1. Interpolation and smoothing (quick but flawed: for many years the default method in practice)
2. Kinematic wave / shockwave theory: the adaptive smoothing method (ASM)

- Treiber, M. and D. Helbing, Reconstructing the Spatio-Temporal Traffic Dynamics from Stationary Detector Data. Cooper@tive Tr@nsport@tion Dyn@mics, 2002. **1: p. 3.1-3.24.**
- van Lint, J.W.C. and S.P. Hoogendoorn, A Robust and Efficient Method for Fusing Heterogeneous Data from Traffic Sensors on Freeways. Computer-Aided Civil and Infrastructure Engineering, 2010. **25(8): p. 596-612.**

Interpolation, smoothing, time series

- Interpolation (space or time)

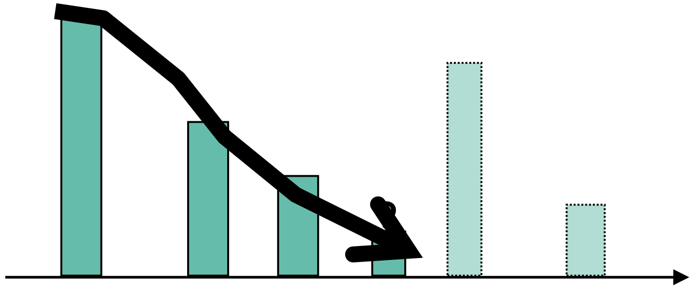
$$z(t_j, x_i) = \begin{cases} z(t_j, x_a) & x_i = x_1 \\ z(t_j, x_{i-1}) + \dfrac{x_i}{x_a - x_{i-1}} z(t_j, x_a) & x_1 < x_i < x_K \\ z(t_j, x_{i-1}) & x_i = x_K \end{cases}$$



- Smoothing, forecasting (space or time)

$$z(t_j, x_i) = f_z(t_j, x_i)$$

$$f_z(t_j, x_i) = f_z(t_{j-1}, x_i) + \alpha\left(z(t_{j-1}, x_i) - f_z(t_{j-1}, x_i)\right)$$



- OK for "isolated" data with no spatial dynamics involved (we know of)

Interpolation, smoothing, time series

- Interpolation (space or time)

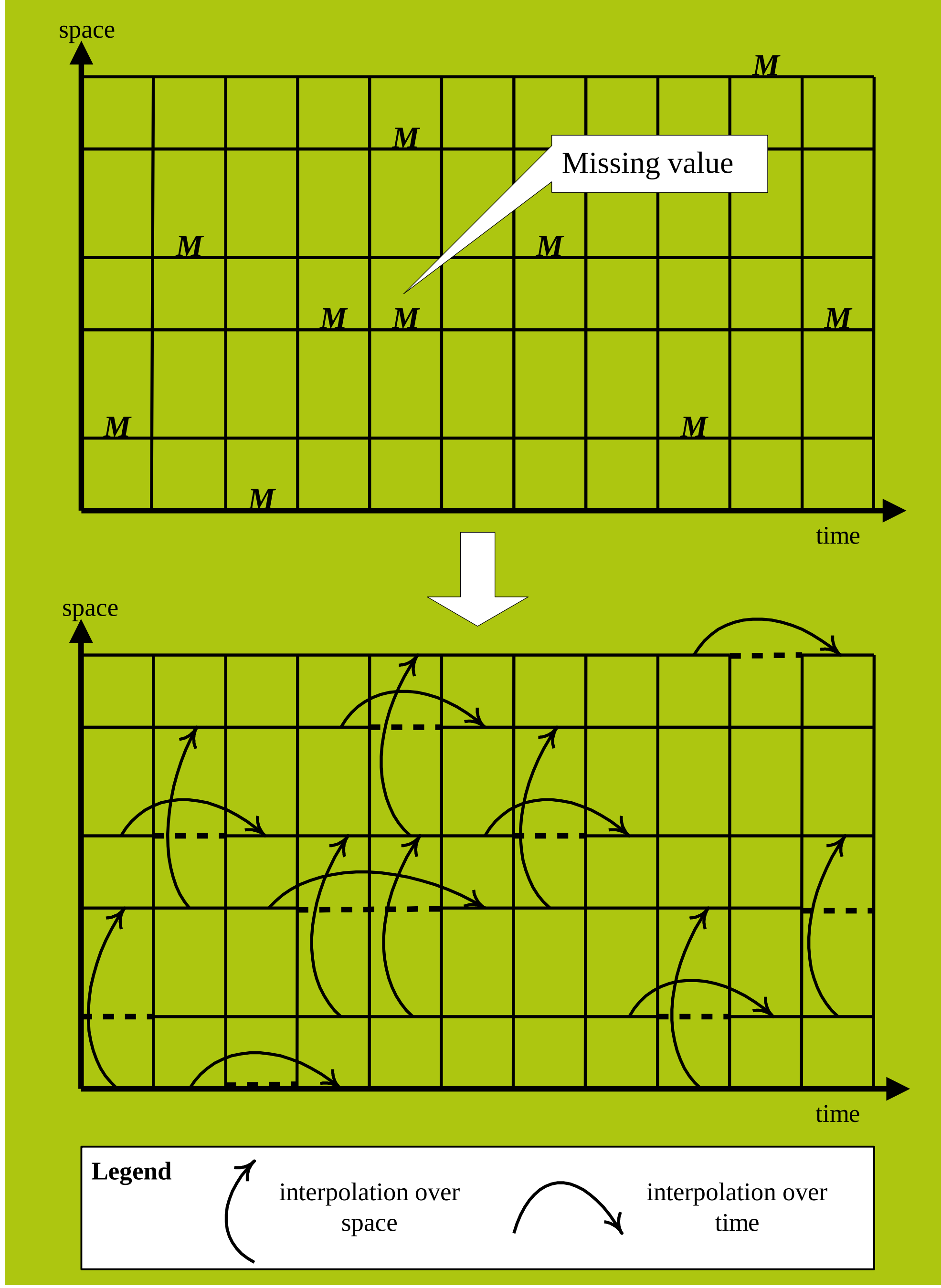
$$z(t_j, x_i) = \begin{cases} z(t_j, x_a) & x_i = x_1 \\ z(t_j, x_{i-1}) + \dfrac{x_i}{x_a - x_{i-1}} z(t_j, x_a) & x_1 < x_i < x_K \\ z(t_j, x_{i-1}) & x_i = x_K \end{cases}$$

- Smoothing, forecasting (space or time)

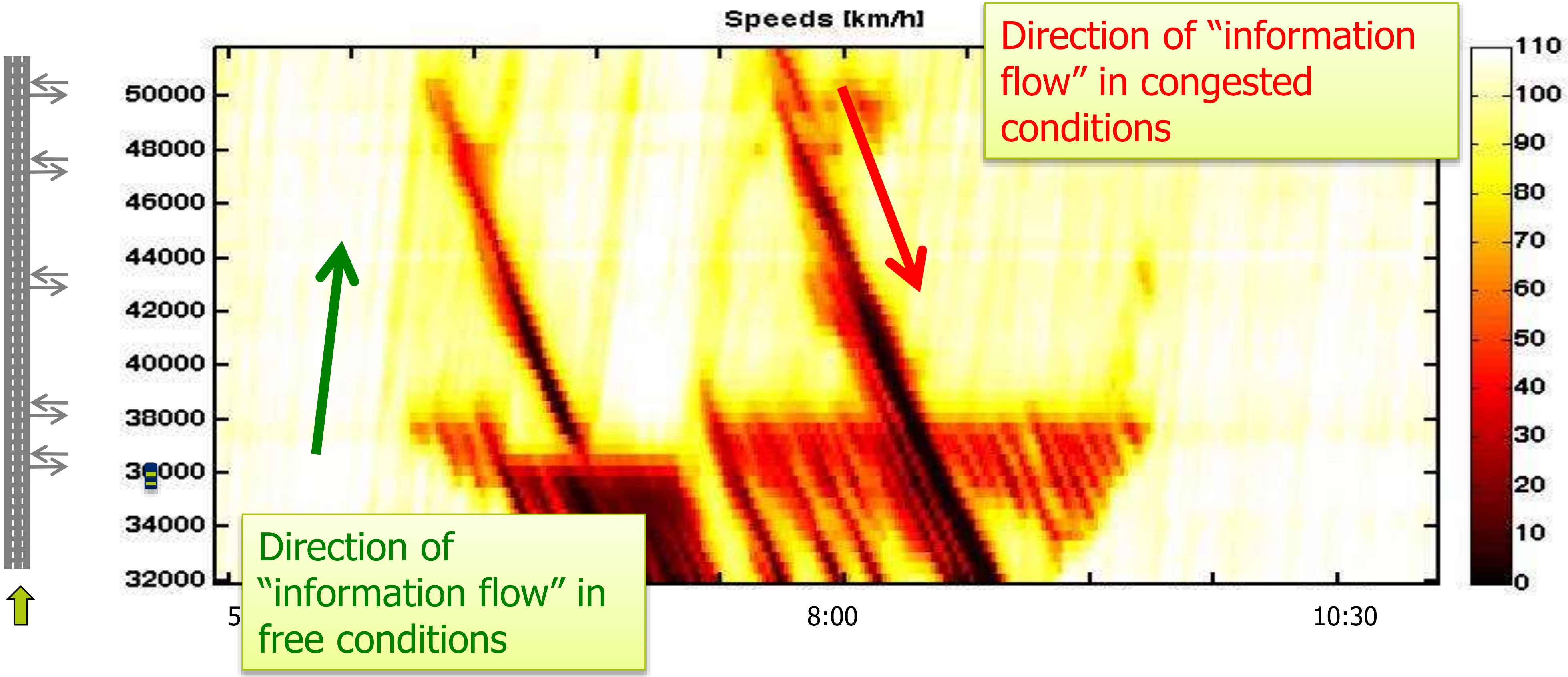
$$z(t_j, x_i) = f_z(t_j, x_i)$$

$$f_z(t_j, x_i) = f_z(t_{j-1}, x_i) + \alpha\left(z(t_{j-1}, x_i) - f_z(t_{j-1}, x_i)\right)$$

- OK for "isolated" data with no spatial dynamics involved (we know of)



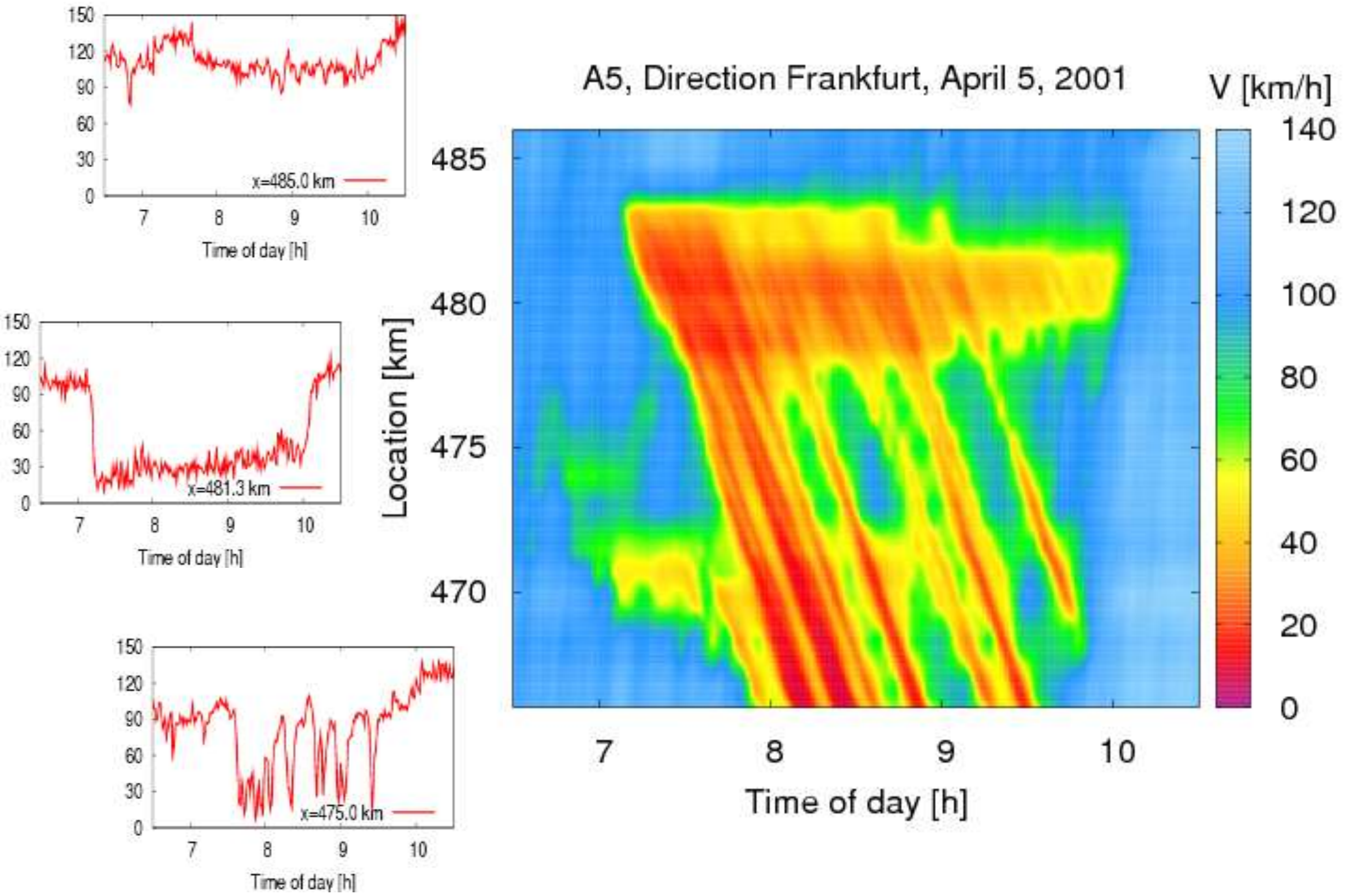
Empirical facts about traffic data



The adaptive smoothing method (ASM)

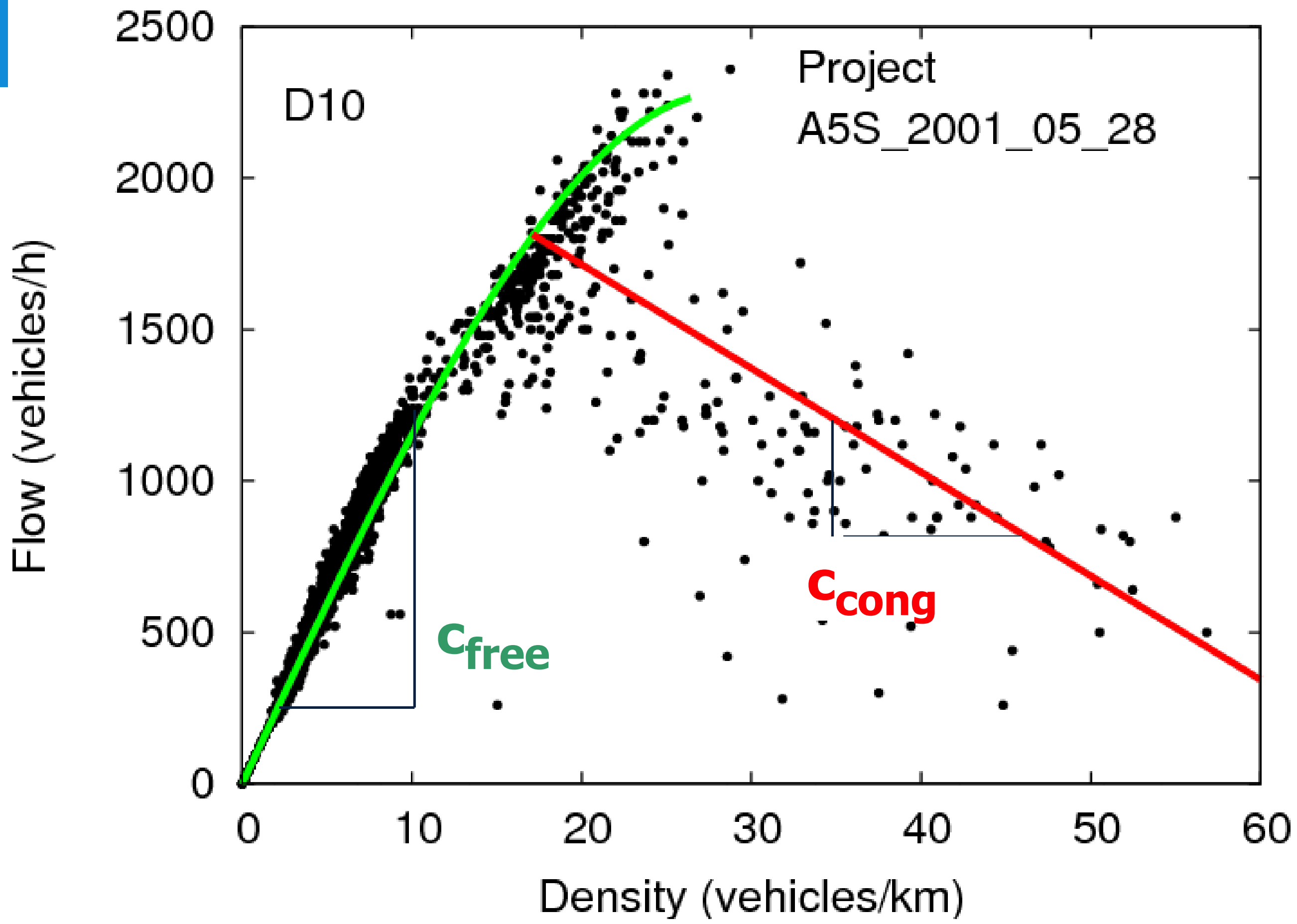
Slides adapted from Martin Treiber's lectures in Delft 2009

- INPUT:
  - Any data $z^{obs}(t_j,x_i)$ from fixed detectors, GPS enabled vehicles, etc
  - Associated speed observation $v^{obs}(t_j,x_i)$ (or another quantity with which the regime can be determined)
- OUTPUT
  - Discretised grid $z(t,x)$ in any desired granularity



The Three Components of the ASM

Slides adapted from Martin Treiber's lectures in Delft 2009



1. Linear kernel-based **spatiotemporal filters** (to smooth the data over space and time)
2. **The direction of information flow** of free and congested traffic is incorporated by **skewing the time axes** of the filters
3. **Nonlinear weighting** of the filters for free and congested traffic

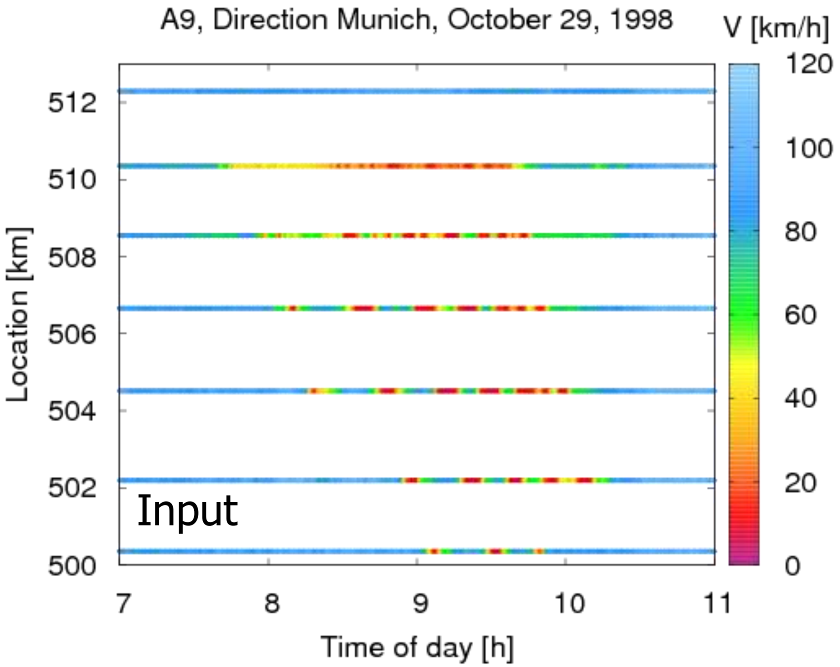
ASM (1): Linear Lowpass Filter

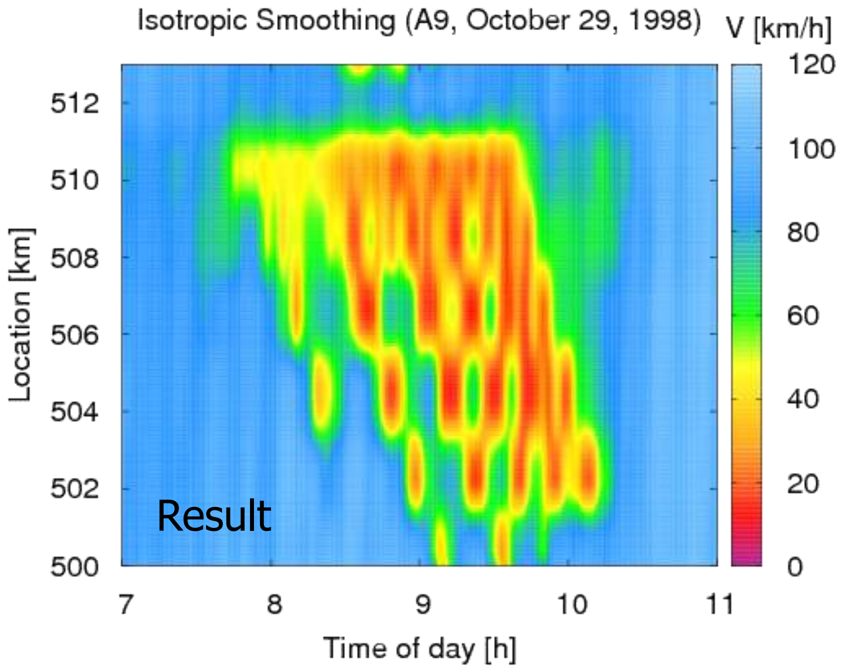
Slides adapted from Martin Treiber's lectures in Delft 2009

Orthogonal homogeneous kernel: $u(x,t) = \sum \varphi_0(x - x_i, t - t_j) u(x_i, t_j)$

Spatial smoothing σ ~ 400m,
temporal smoothing τ ~ 60s

$$\text{with } \varphi_0(s,r) = \exp\left[-\left(\frac{|s|}{\sigma} + \frac{|r|}{\tau} \right) \right]$$





ASM (2): Skewing the time axes

Slides adapted from Martin Treiber's lectures in Delft 2009

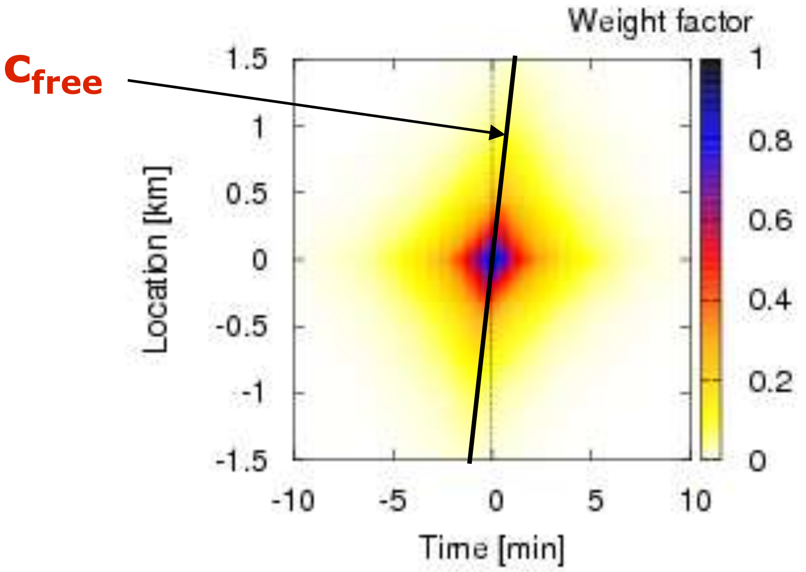
Two anisotropic kernels:

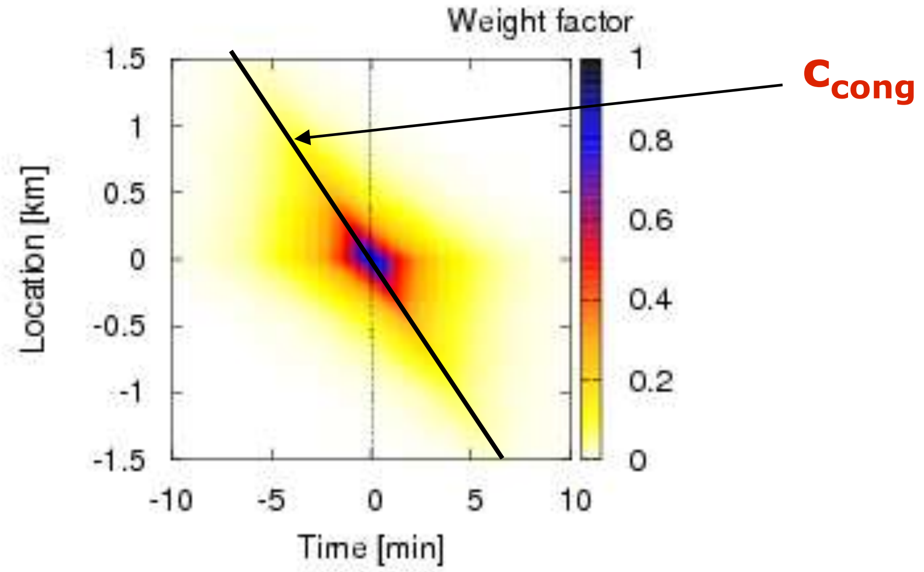
Spatial smoothing σ ~ 400m,
temporal smoothing τ ~ 60s

$$u_{free}(x,t) = \sum \varphi_0(x - x_i, t - t_j, c_{free}) u(x_i, t_j)$$

$$u_{cong}(x,t) = \sum \varphi_0(x - x_i, t - t_j, c_{cong}) u(x_i, t_j)$$

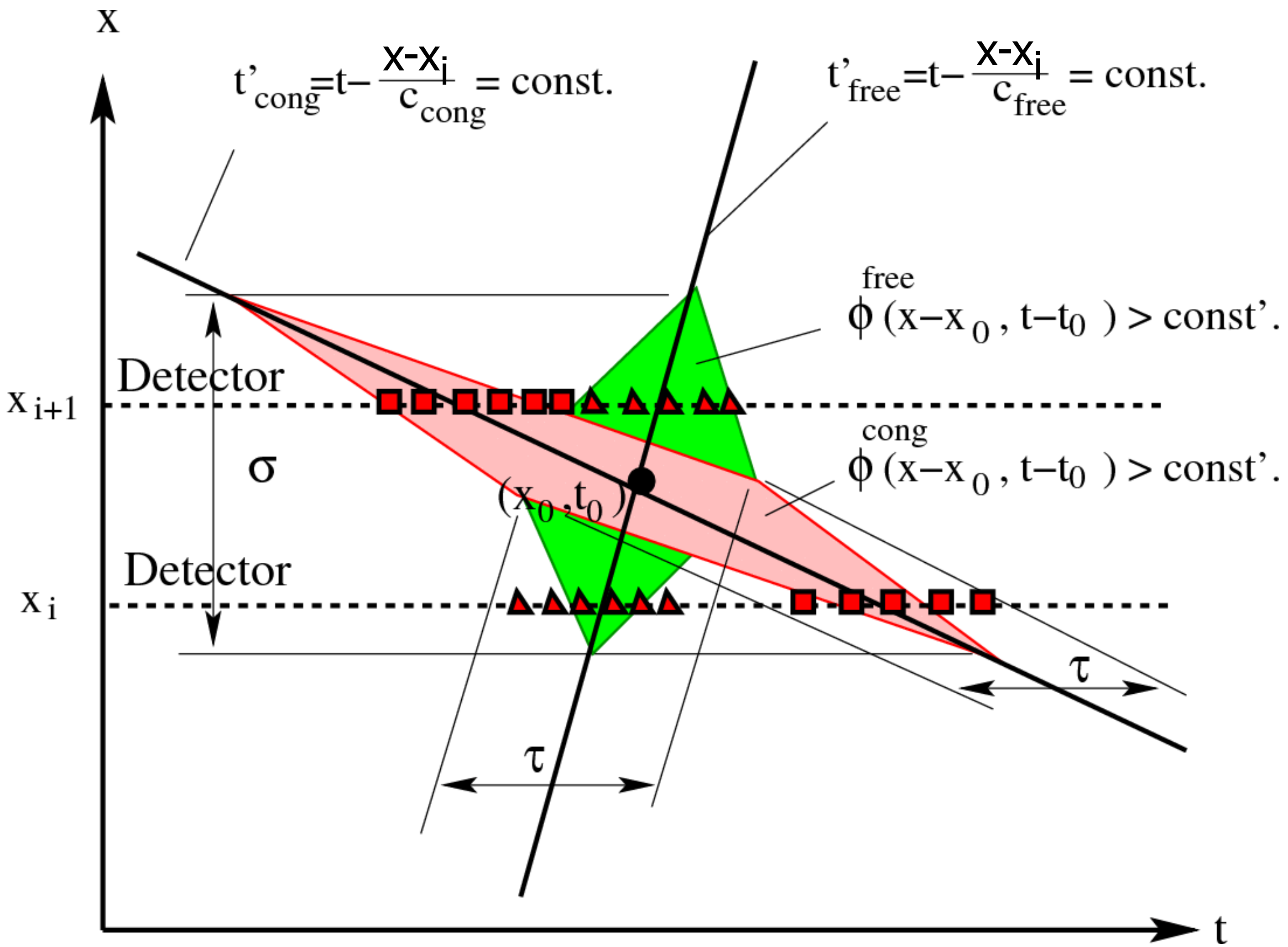
$$\text{with } \varphi_0(s,r,c) = \exp\left[-\left(\frac{|s|}{\sigma} + \frac{|r - s/c|}{\tau} \right) \right]$$





Effect on the Weighting of Data Points

Slides adapted from Martin Treiber's lectures in Delft 2009



ASM (3): Adaptive Nonlinear Speed Filter

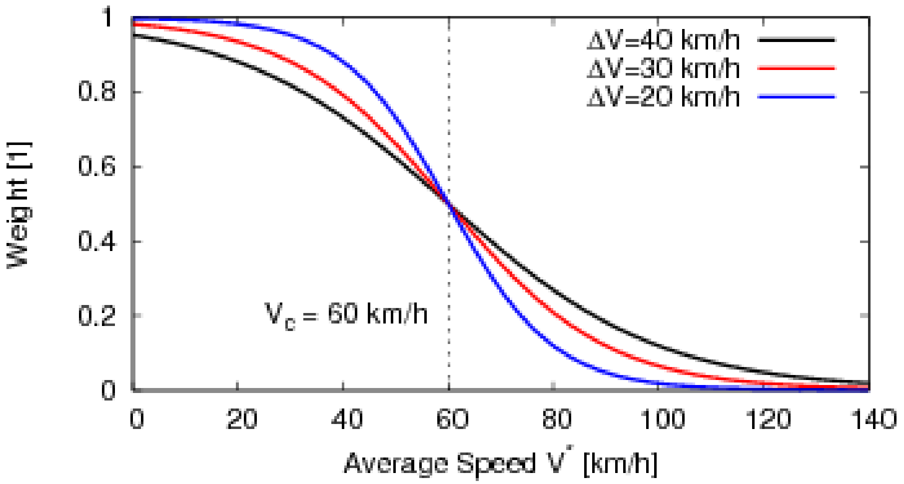
Slides adapted from Martin Treiber's lectures in Delft 2009

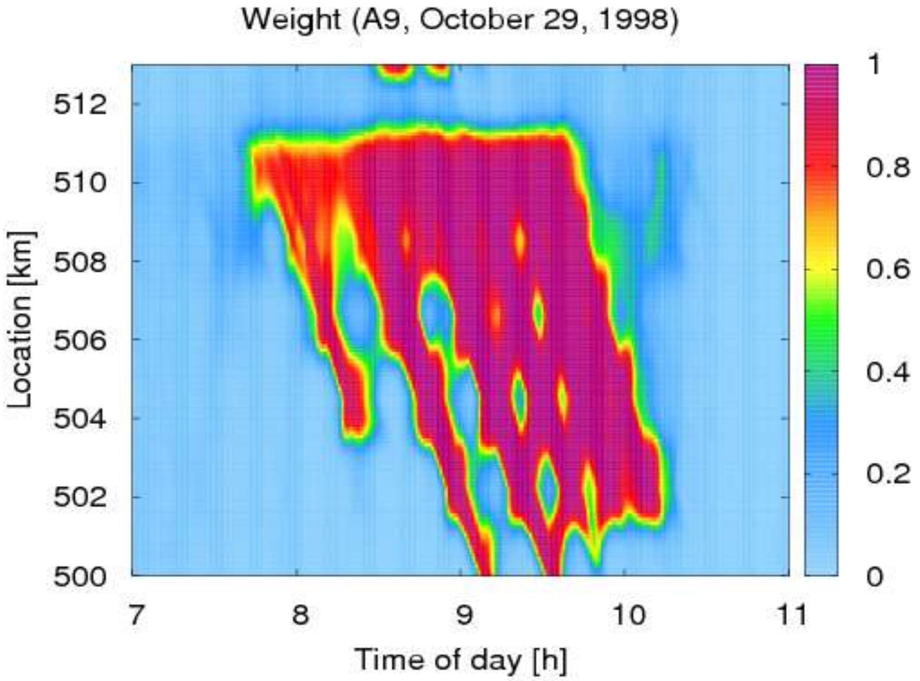
Superposition of free and congested speed regimes:

$$u(x,t) = w(x,t)u_{cong}(x,t) + \big(1 - w(x,t)\big)u_{cong}(x,t)$$

Nonlinear weight function (threshold V_c and width ΔV):

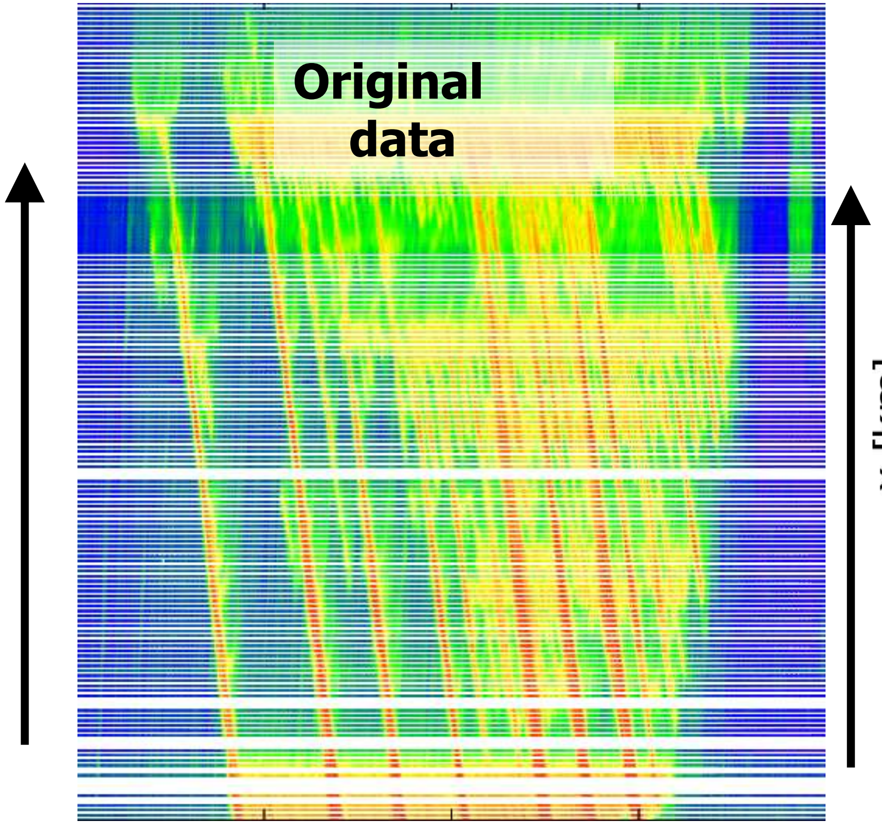
$$w(x,t) = \frac{1}{2}\left[1 + \tanh\left(\frac{V_c - \min[u_{free}, u_{cong}]}{\Delta V}\right)\right]$$

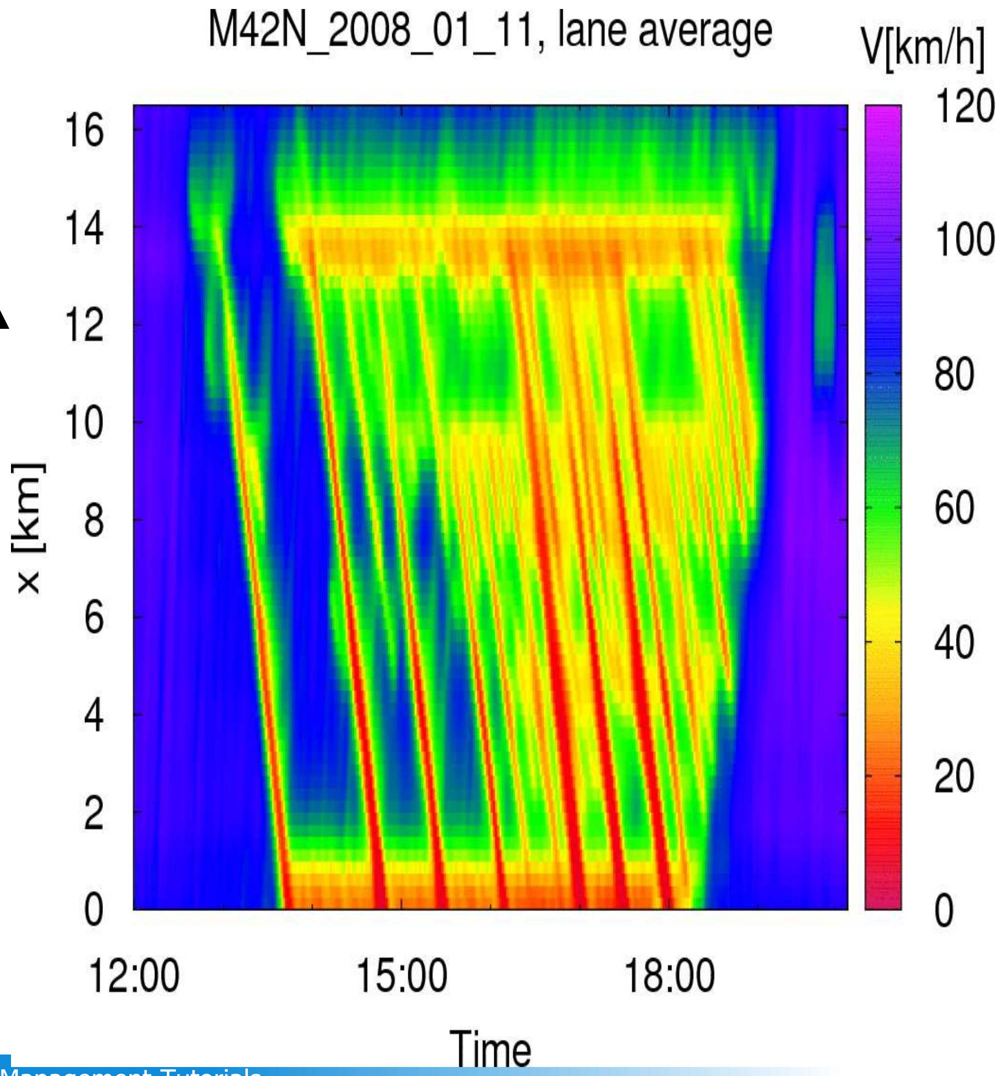




Validating the ASM: M 42 North (England)

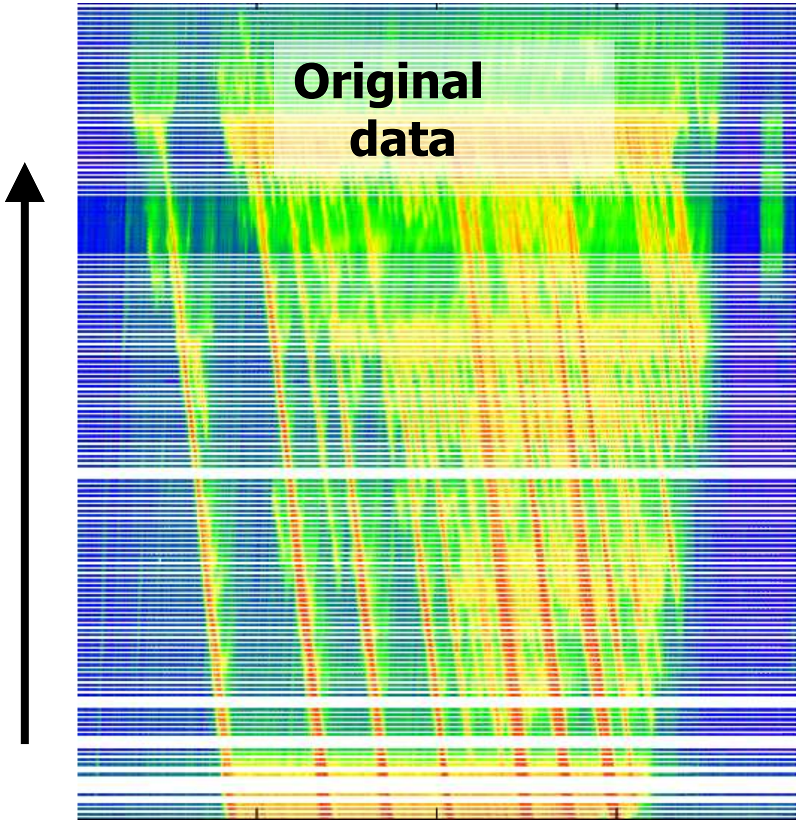
Slides adapted from Martin Treiber's lectures in Delft 2009

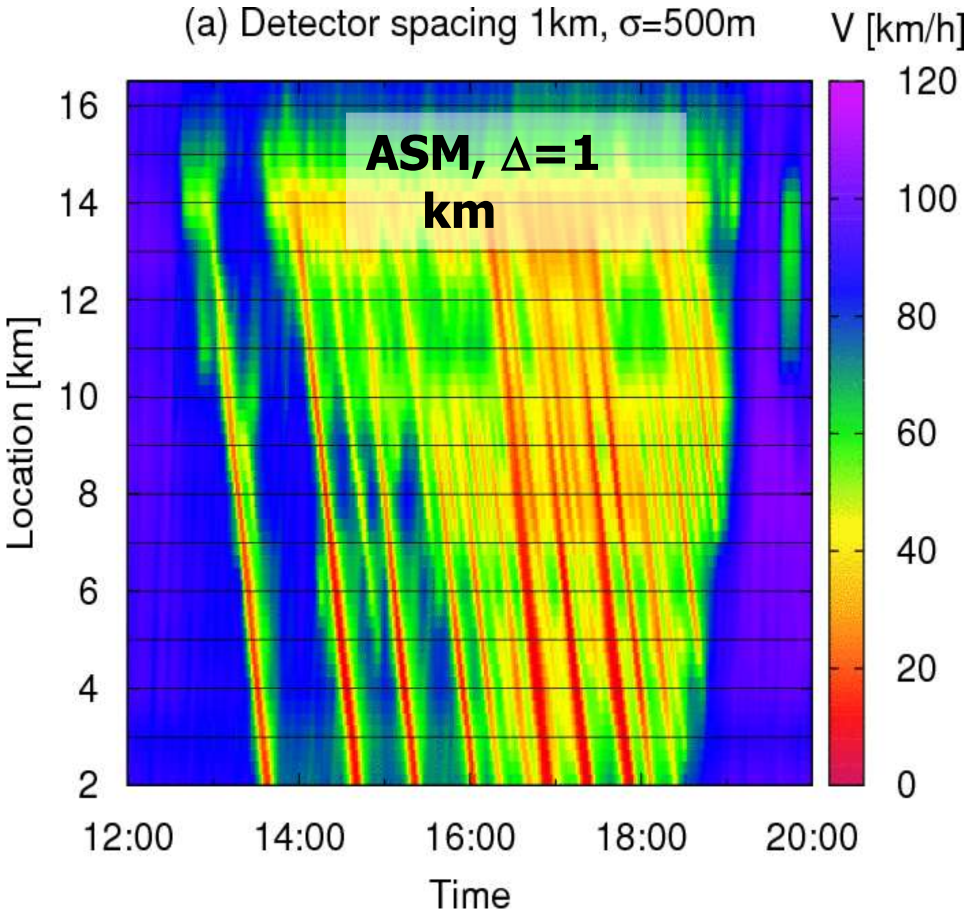




Validating the ASM: M 42 North (England)

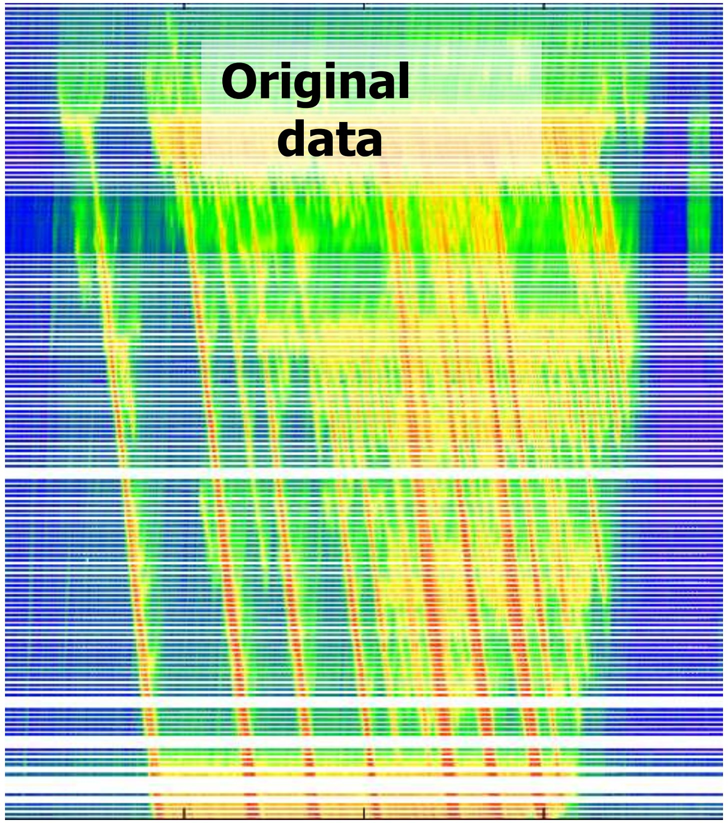
Slides adapted from Martin Treiber's lectures in Delft 2009

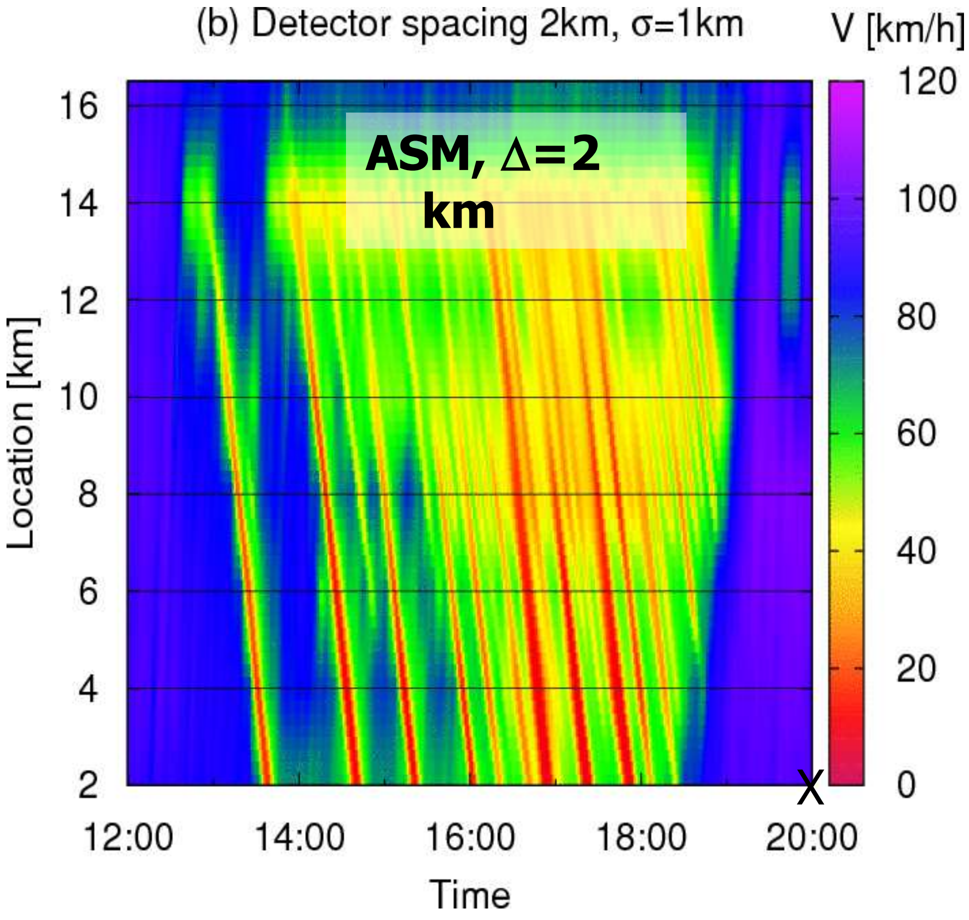




Validating the ASM: M 42 North (England)

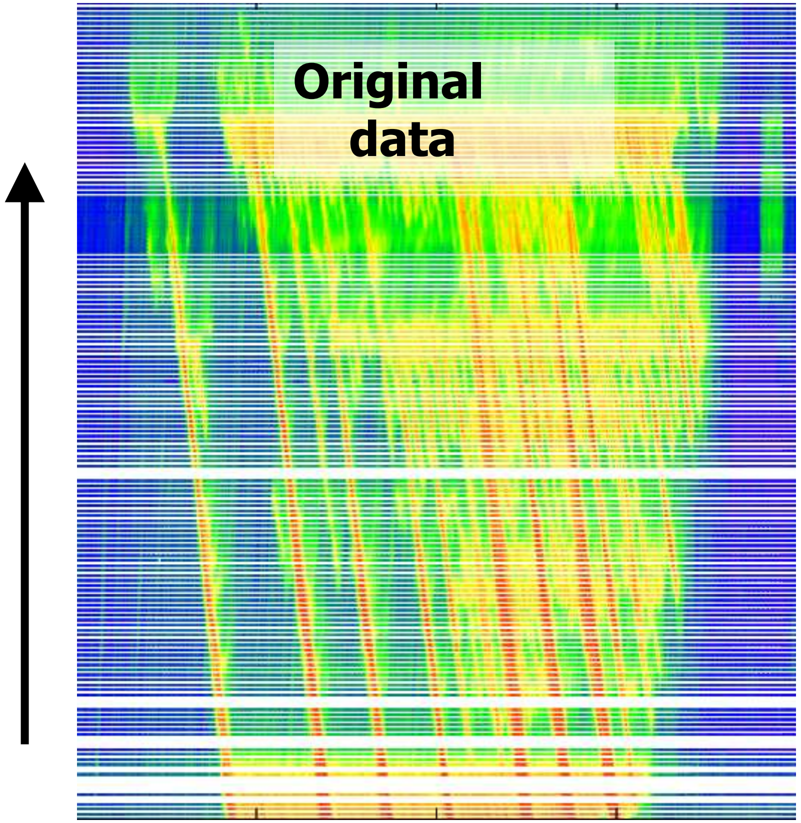
Slides adapted from Martin Treiber's lectures in Delft 2009

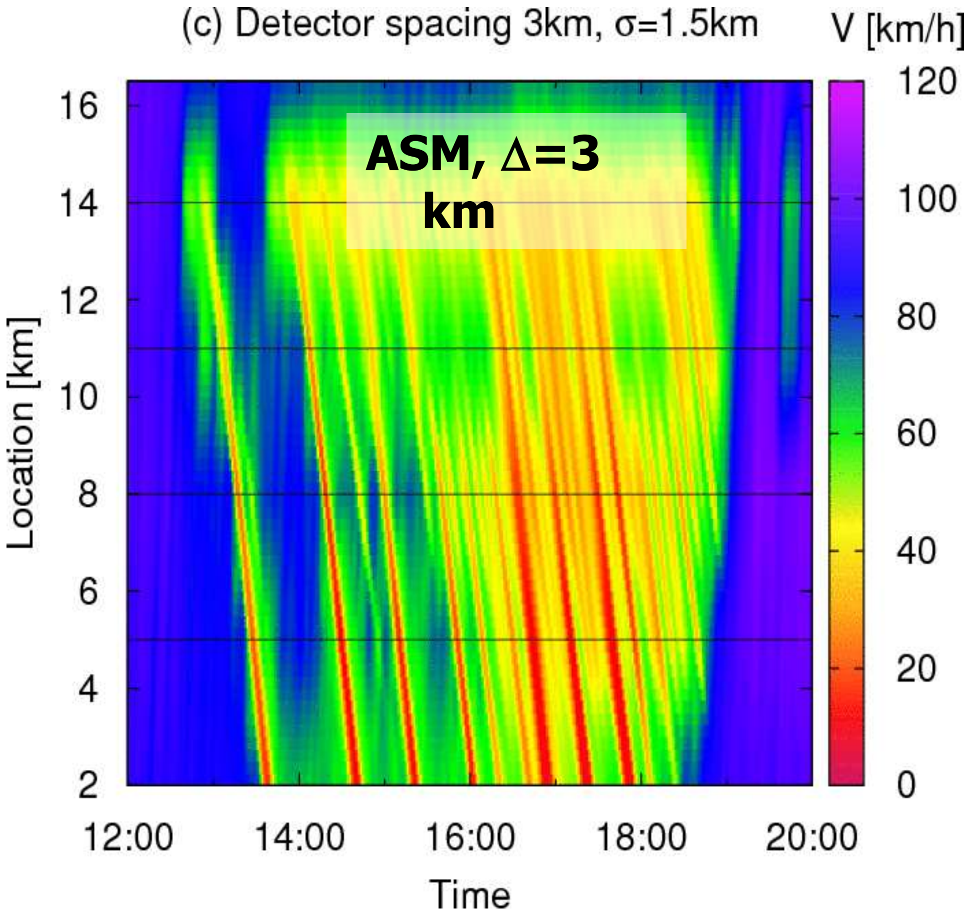




Validating the ASM: M 42 North (England)

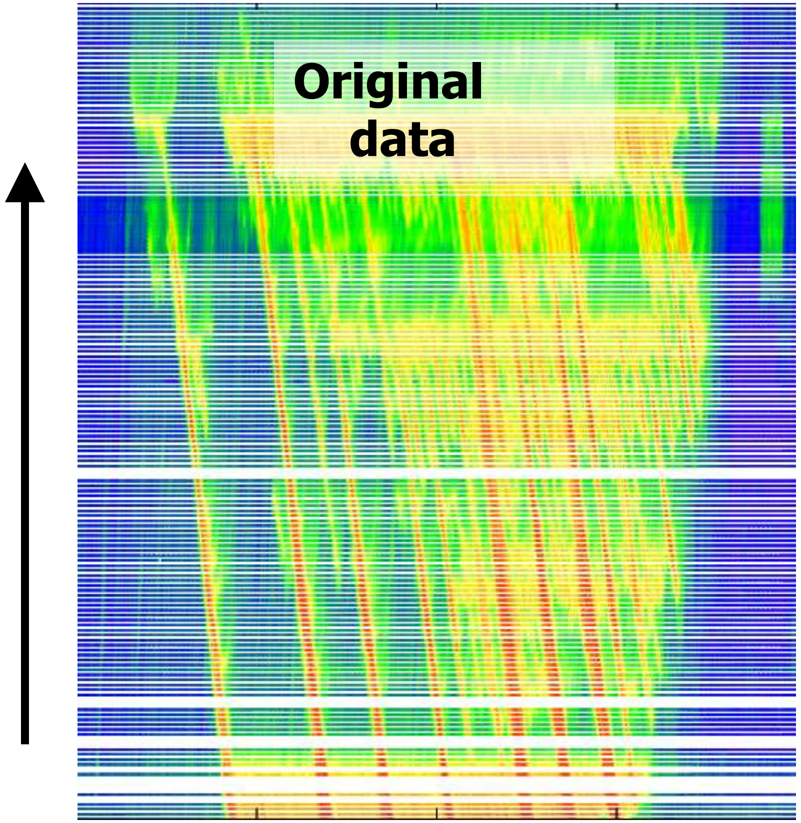
Slides adapted from Martin Treiber’s lectures in Delft 2009

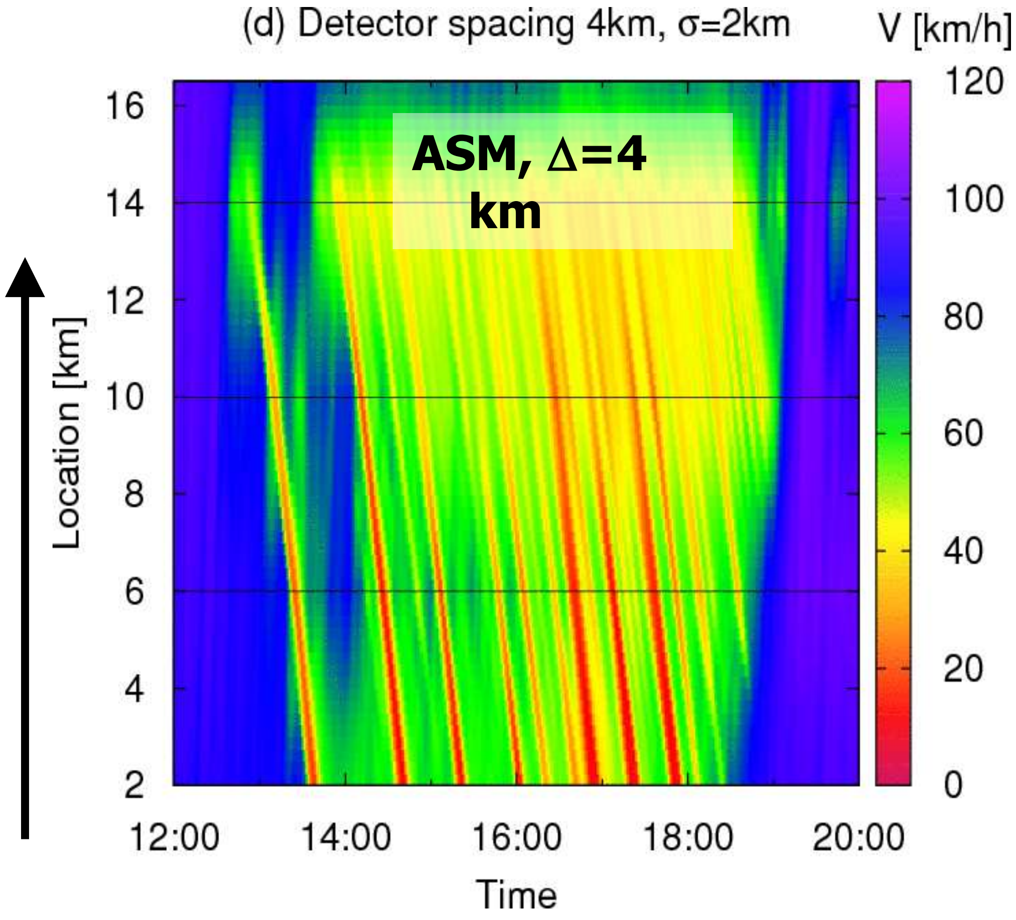




Validating the ASM: M 42 North (England)

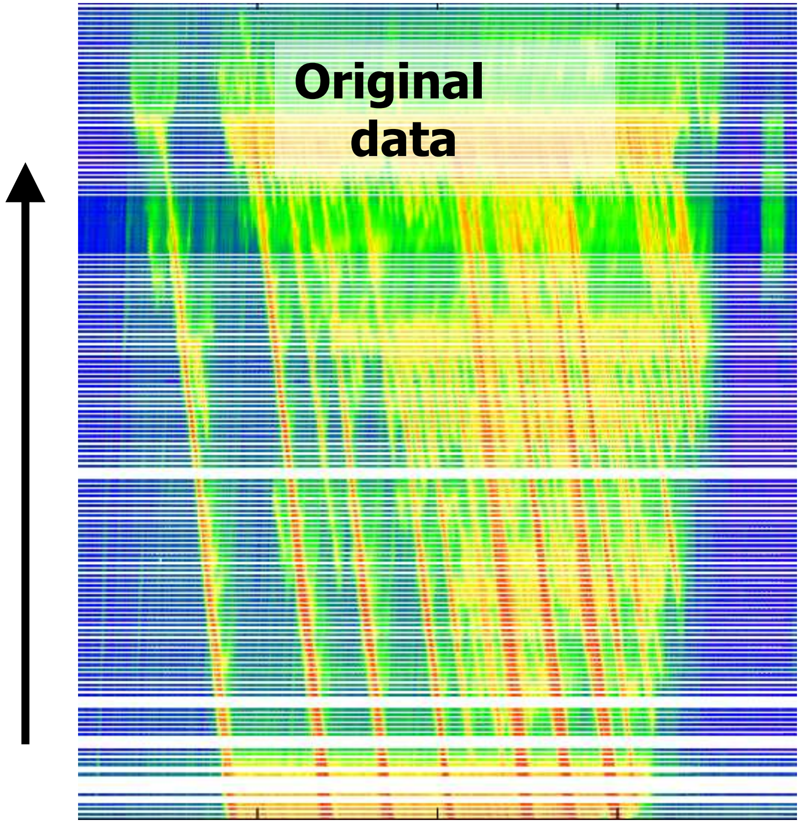
Slides adapted from Martin Treiber’s lectures in Delft 2009

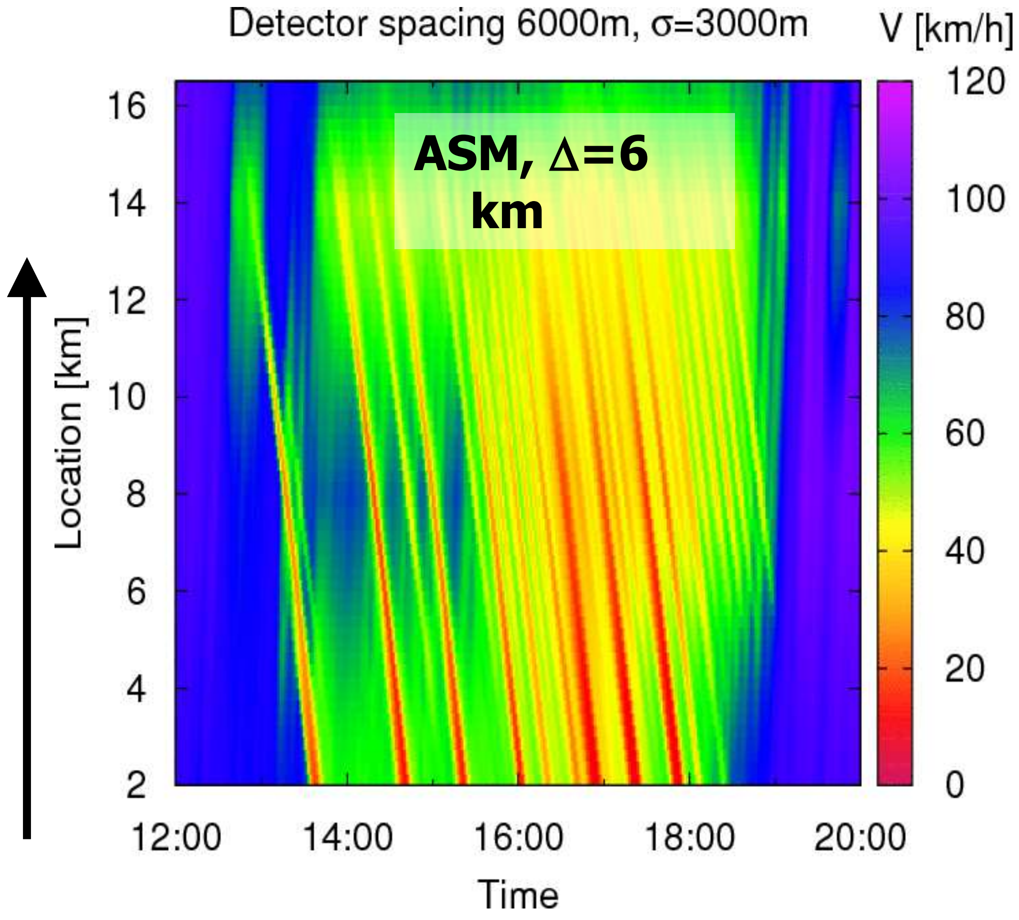




Validating the ASM: M 42 North (England)

Slides adapted from Martin Treiber's lectures in Delft 2009



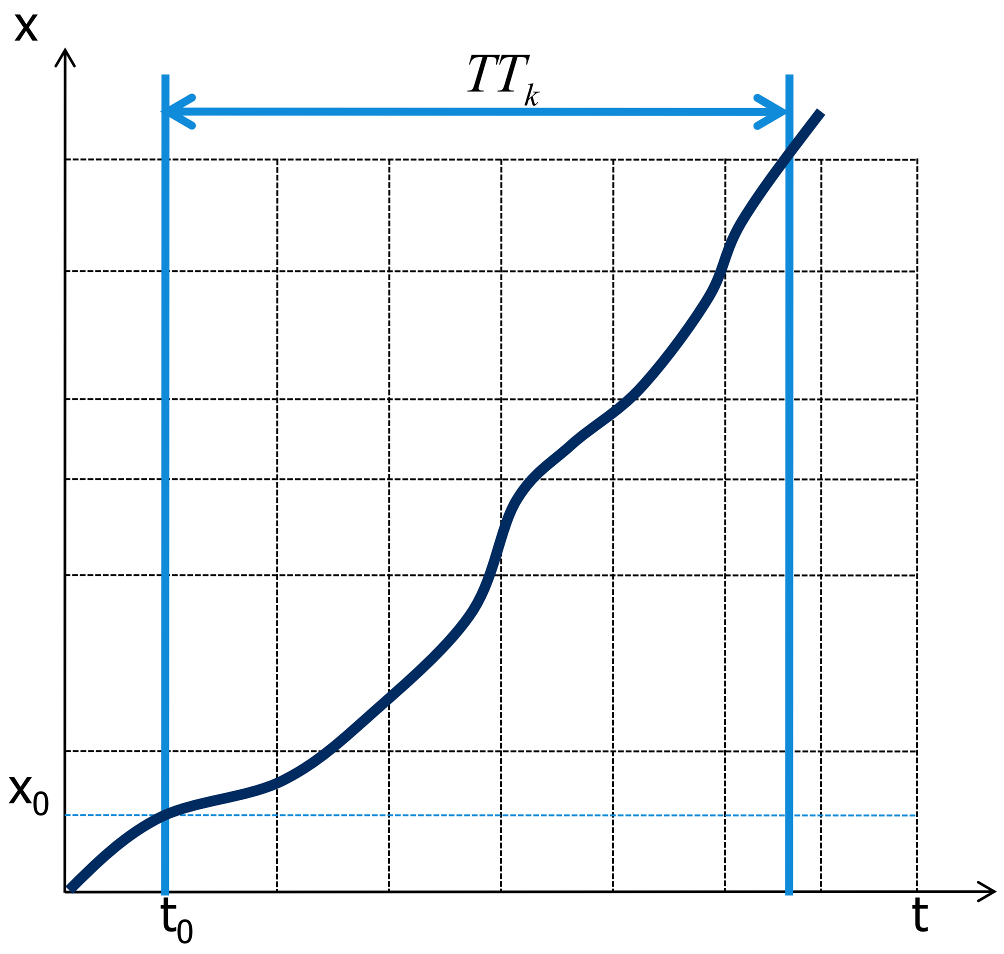


Application ASM: Travel time estimation

- Van Lint, J. W. C. (2010). "Empirical Evaluation of New Robust Travel Time Estimation Algorithms." Transportation Research Record 2160: 50-59.

- You'll find many refs in that paper to other travel time estimation methods

Trajectory methods



Solve this differential equation:

$$\frac{dx^{(n)}(t)}{dt} = u(t, x^{(n)}(t)), \text{with } x^{(n)}(t_0) = x_0$$

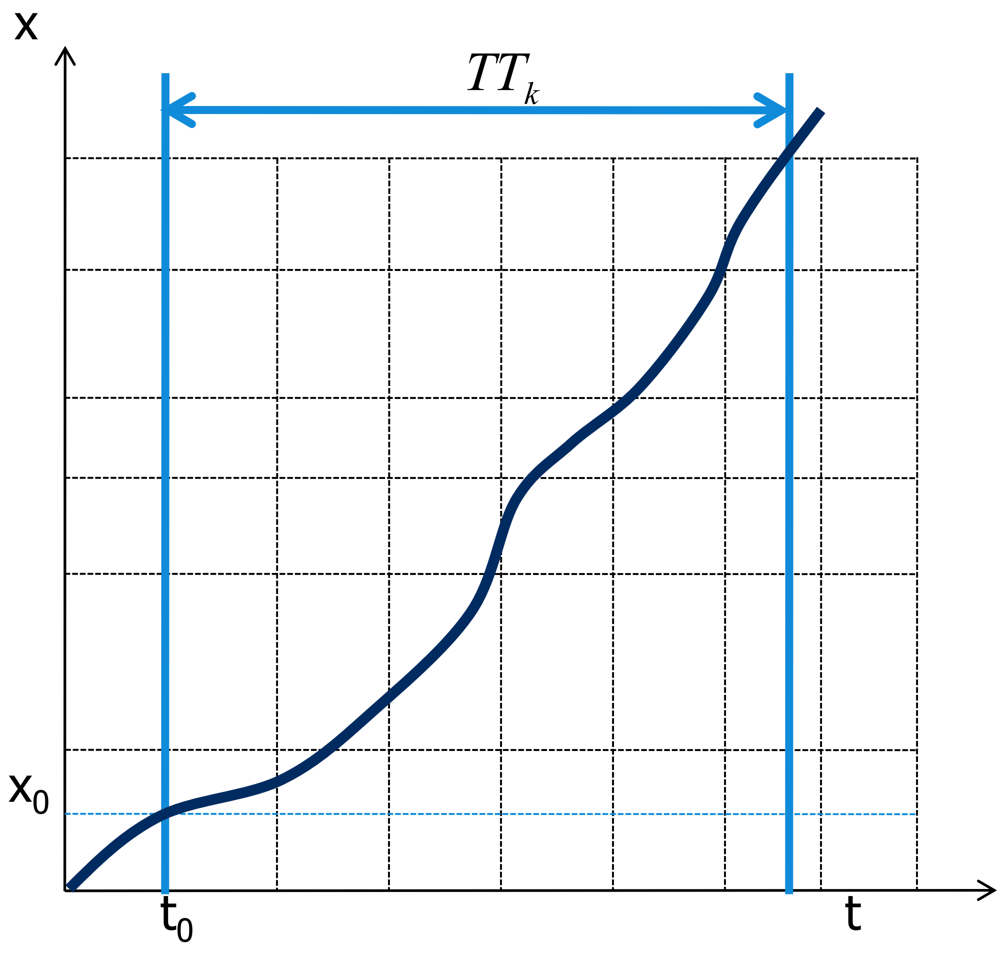
Trajectory methods

- use imaginary average vehicle trajectories
- to deduce average travel time

Trajectory methods differ

- in discretization of space and time (cells j,i)
- in formulation of u_{ji}

"Classic" Trajectory methods



Solve this differential equation:

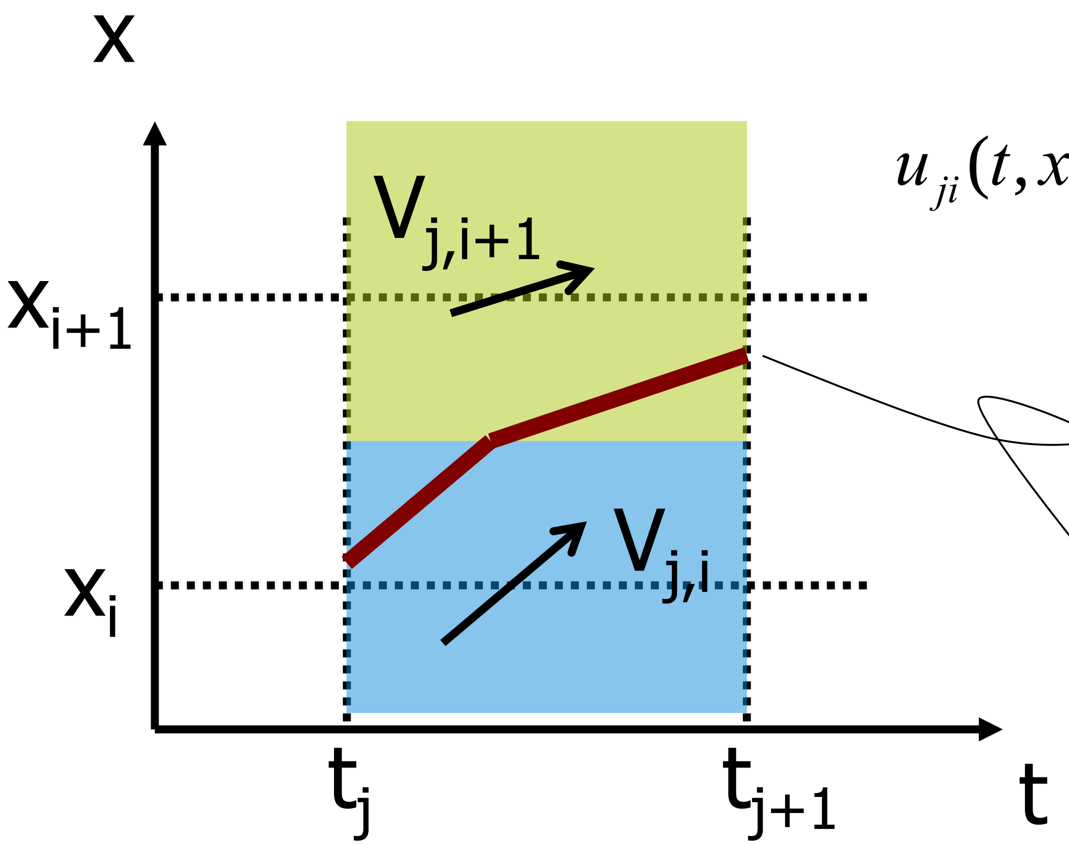
$$\frac{dx^{(n)}(t)}{dt} = u(t, x^{(n)}(t)), \text{with } x^{(n)}(t_0) = x_0$$

- use grid of detector locations × time periods
- in cells: speed is function of up- and downstream detector speeds

(which (unfortunately) is the equivalent of orthogonal interpolation)

“Classic” Trajectory methods ()

PCSB: piece-wise constant speed-based method

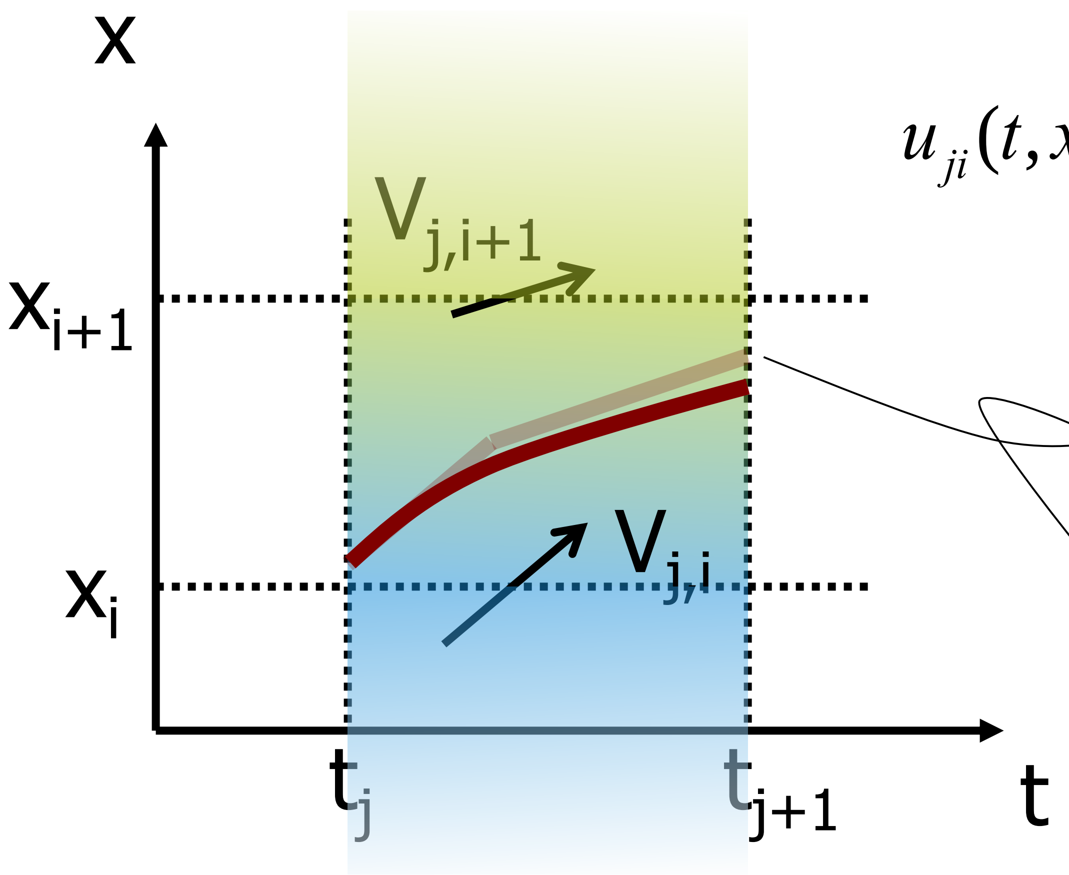


$$u_{ji}(t,x) = \begin{cases} V_{j,i} & x < \frac{1}{2}\left(x_i + x_{i+1}\right) \\ V_{j,i+1} & \text{otherwise} \end{cases}$$

calculate consecutive cell exit locations until destination is reached

"Classic" Trajectory methods

PLSB: piece-wise linear speed-based method



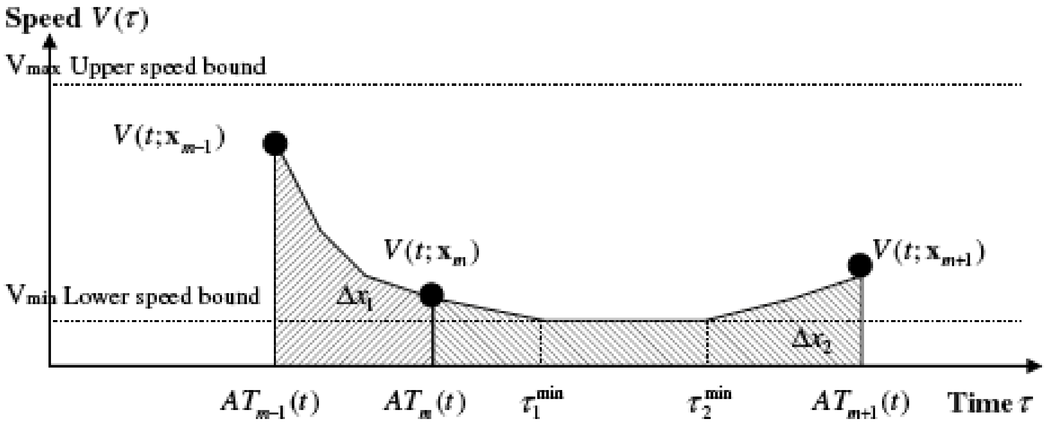
$$u_{ji}(t,x) = V_{j,i} + \frac{x - x_i}{x_{i+1} - x_i}\left(V_{j+1,i} - V_{j,i}\right)$$

calculate consecutive cell exit locations until destination is reached

"Classic" Trajectory methods

PQSB: piece-wise quadratic speed-based method, etc

$$u_{ji}(t,x) = f\left(V_{j-1,i}, V_{j,i}, V_{j+1,i}\right)$$

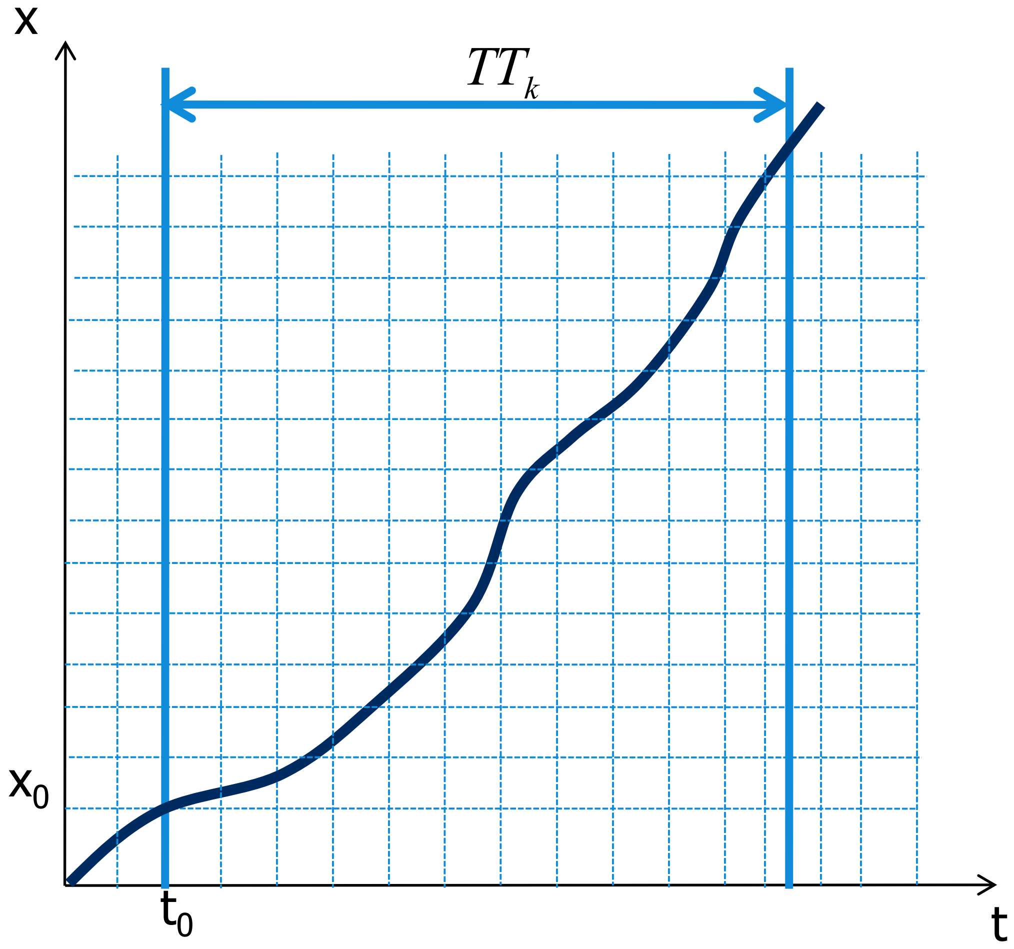


(c) Truncated quadratic speed trajectory (case 7)

L. Sun et al. / Transportation Research Part A 42 (2008) 173–186

or any other polynomial combination of detector speeds in the same time period

"Filtered" trajectory method

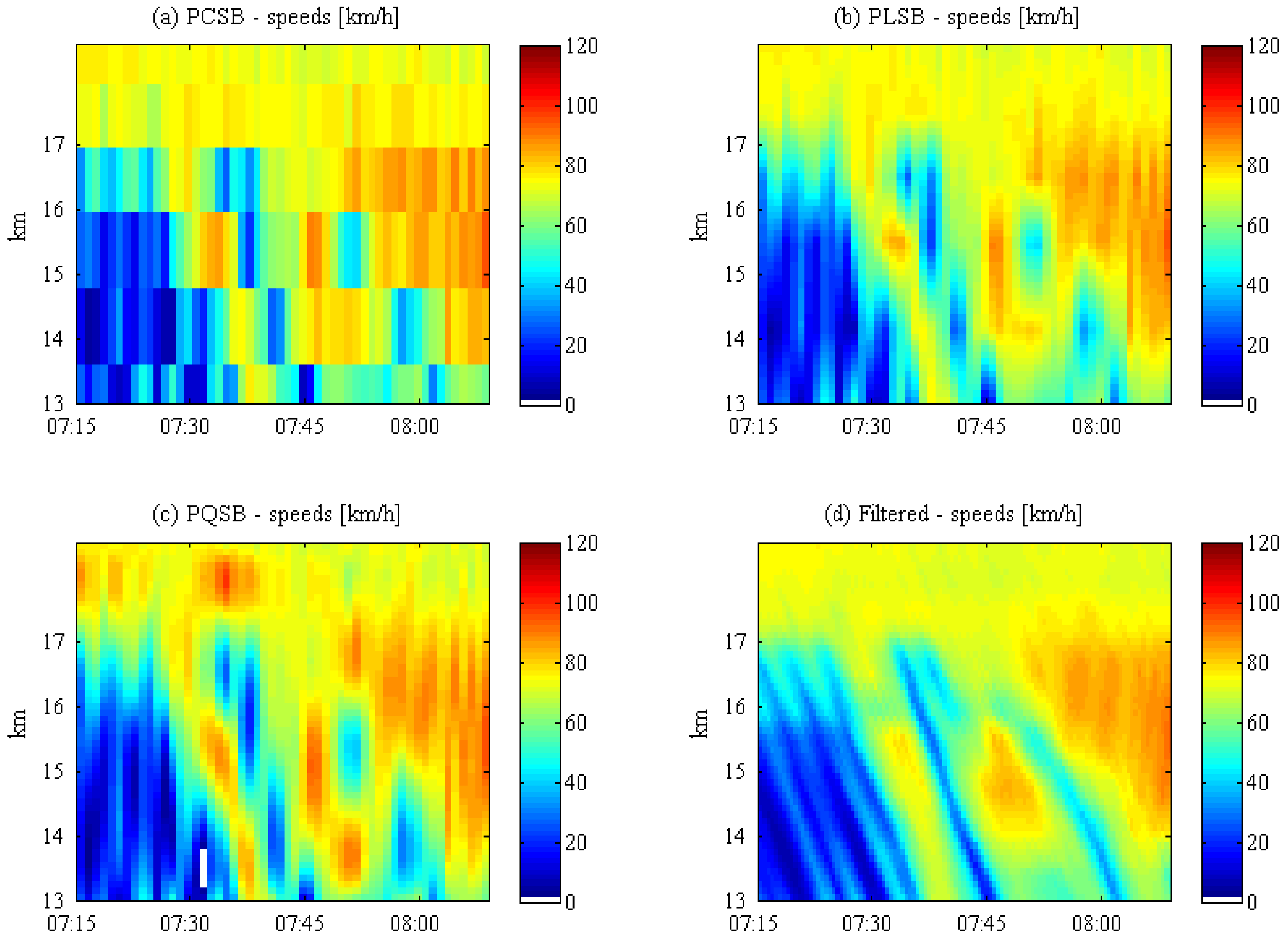


Solve this differential equation:

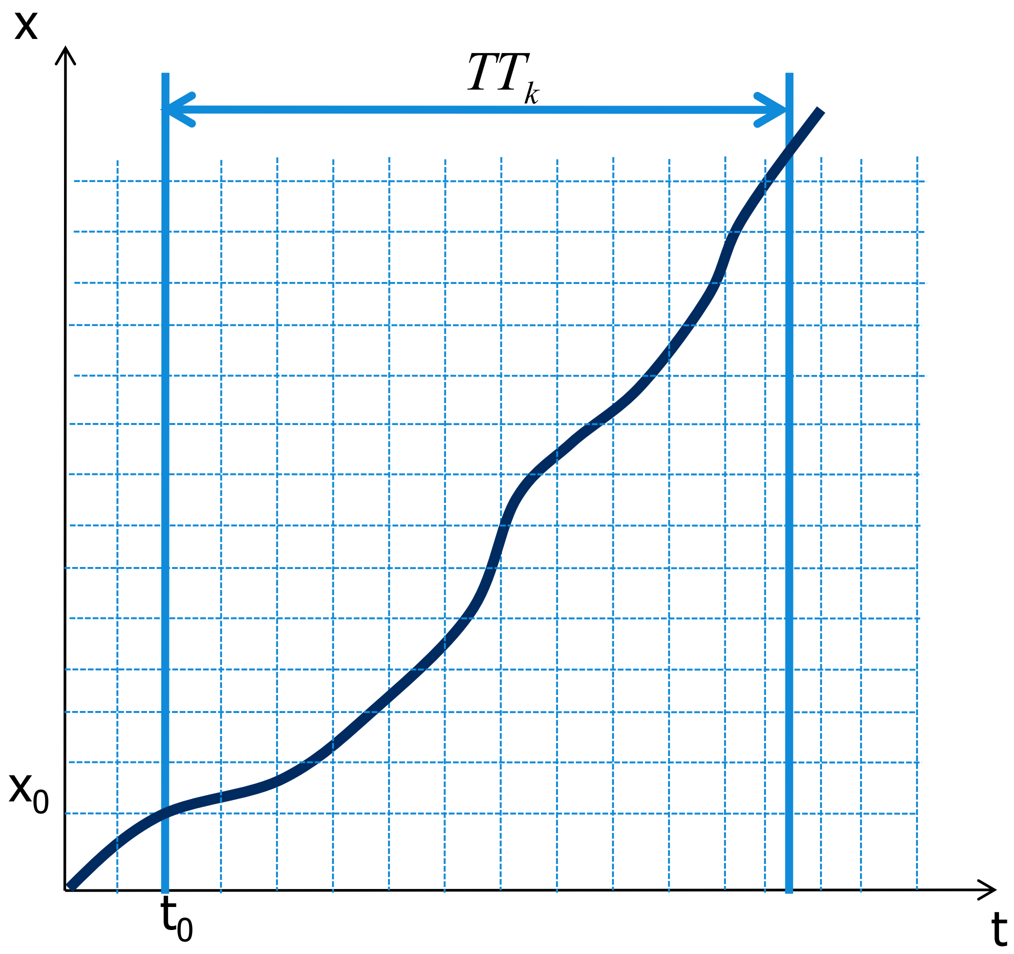
$$\frac{dx^{(n)}(t)}{dt} = u(t, x^{(n)}(t)), \text{with } x^{(n)}(t_0) = x_0$$

- filter (with ASM) speed ➔ create equidistant grid with estimates u_{ji} (as coarse or fine as needed)
- Use PCSB with these filtered speeds

Differences classic (orthogonal) vs filtered



“Inverse-Filtered” trajectory method



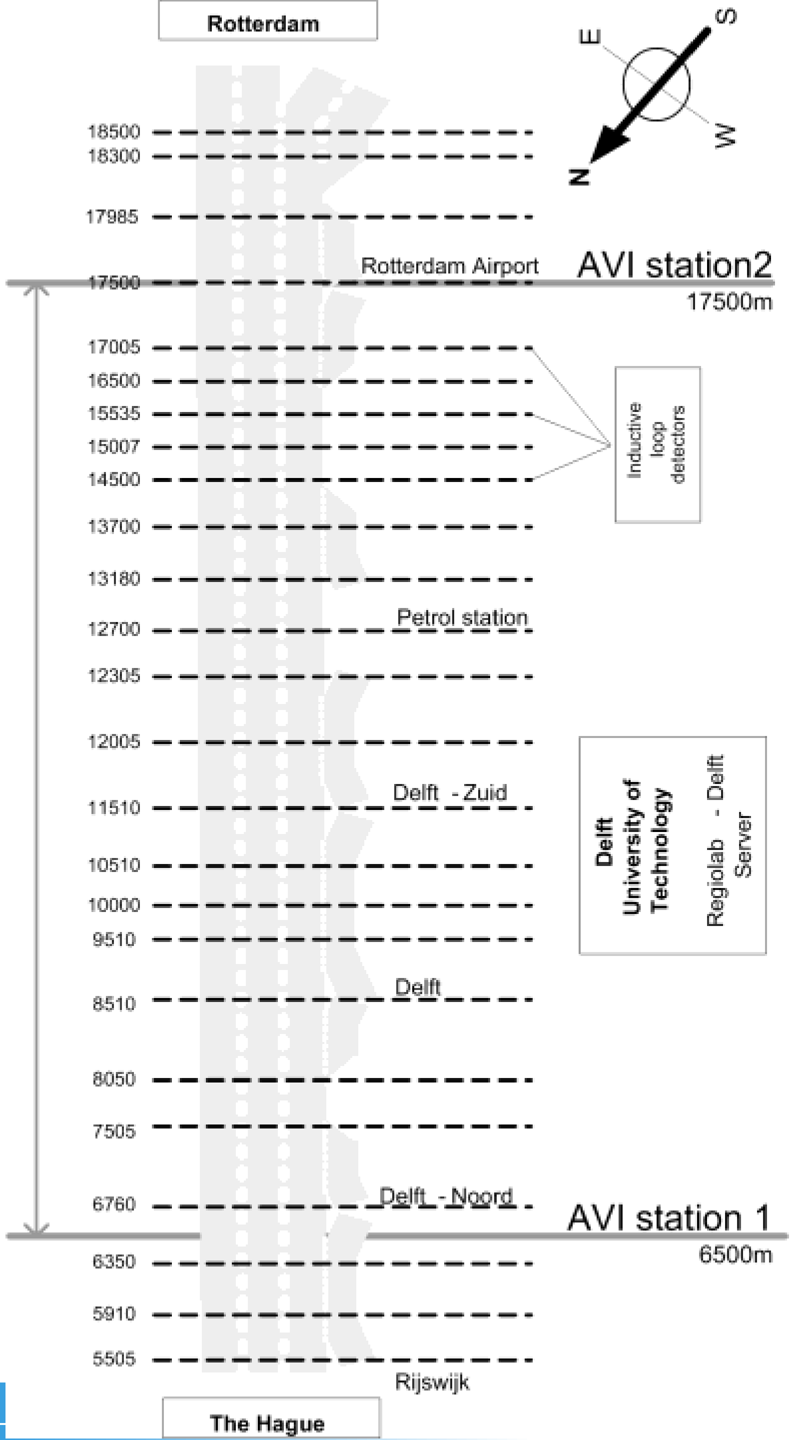
Solve this differential equation:

$$\frac{dx^{(n)}(t)}{dt} = 1/w(t, x^{(n)}(t)), \text{ with } x^{(n)}(t_0) = x_0$$

- filter (with ASM) 1/speed (**pace**) ➔ create equidistant grid with estimates w_{ji} (as coarse or fine as needed)
- Use adjusted PCSB with filtered pace
- Result is an **exponential (harmonically) averaged speed**

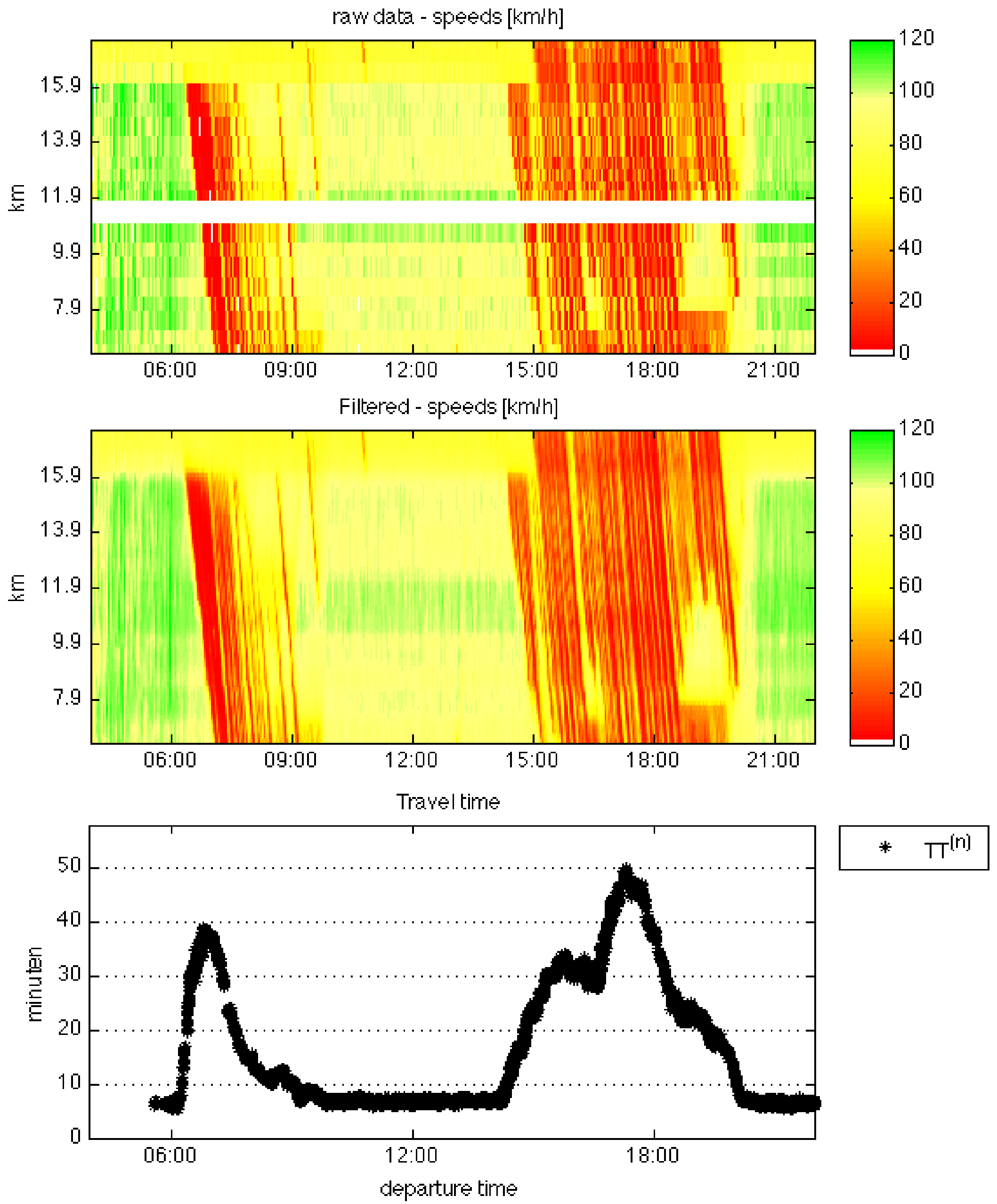
Case

- A13 Z The Hague – R'dam
- Evaluated PCSB, PLSB, FSB (filtered speed based) and FISB (filtered inverse-speed based) methods
- Three scenarios
  - A. all detectors
  - B. 1 out of 2 detectors
  - C. 1 out of 4 detectors

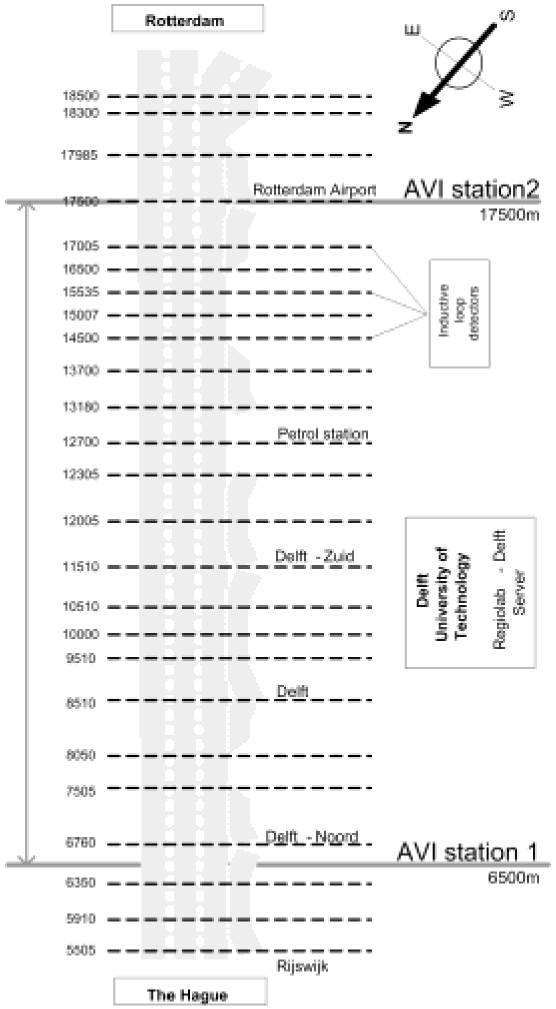


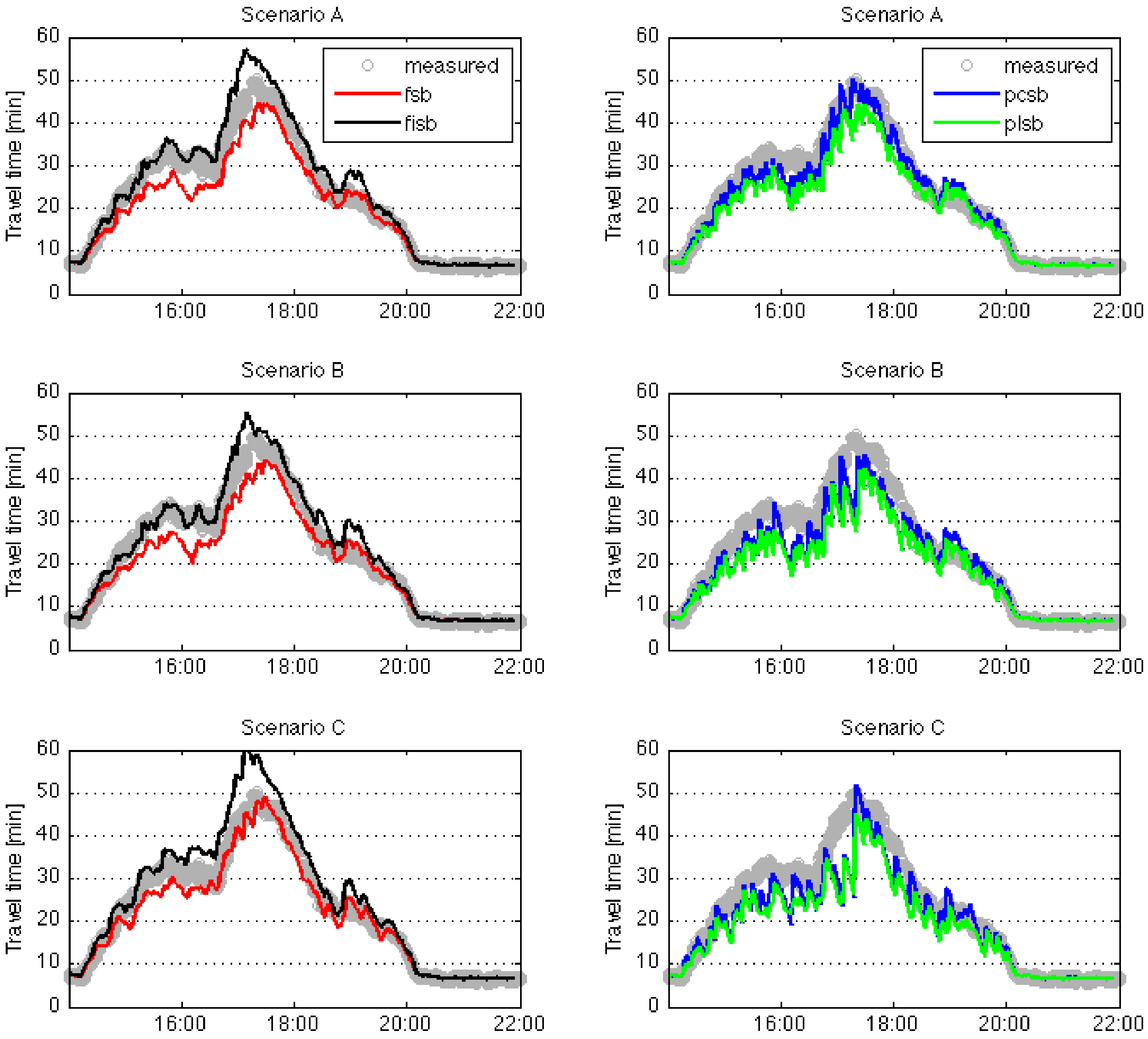
Case

- A13 Z The Hague – R'dam
- Evaluated PCSB, PLSB, FSB (filtered speed based) and FISB (filtered inverse speed based) methods
- Three scenarios
  - all detectors
  - 1 out of 2 detectors
  - 1 out of 4 detectors



Results


Rotterdam
E
S
W
N
18500
18300
17985
Rotterdam Airport
AVI station2
17500m
17500
17005
16500
15535
15007
14500
Inductive loop detectors
13700
13180
12700
Petrol station
12305
12005
Delft - Zuid
11510
Delft University of Technology
Regiolab - Delft Server
10510
10000
9510
8510
Delft
8050
7505
6760
Delft - Noord
AVI station 1
6500m
6350
5910
5505
Rijswijk
The Hague


Scenario A
measured
fsb
fisb
Scenario A
measured
pcsb
plsb
Scenario B
Scenario B
Scenario C
Scenario C
Travel time [min]
0
10
20
30
40
50
60
16:00
18:00
20:00
22:00

Results

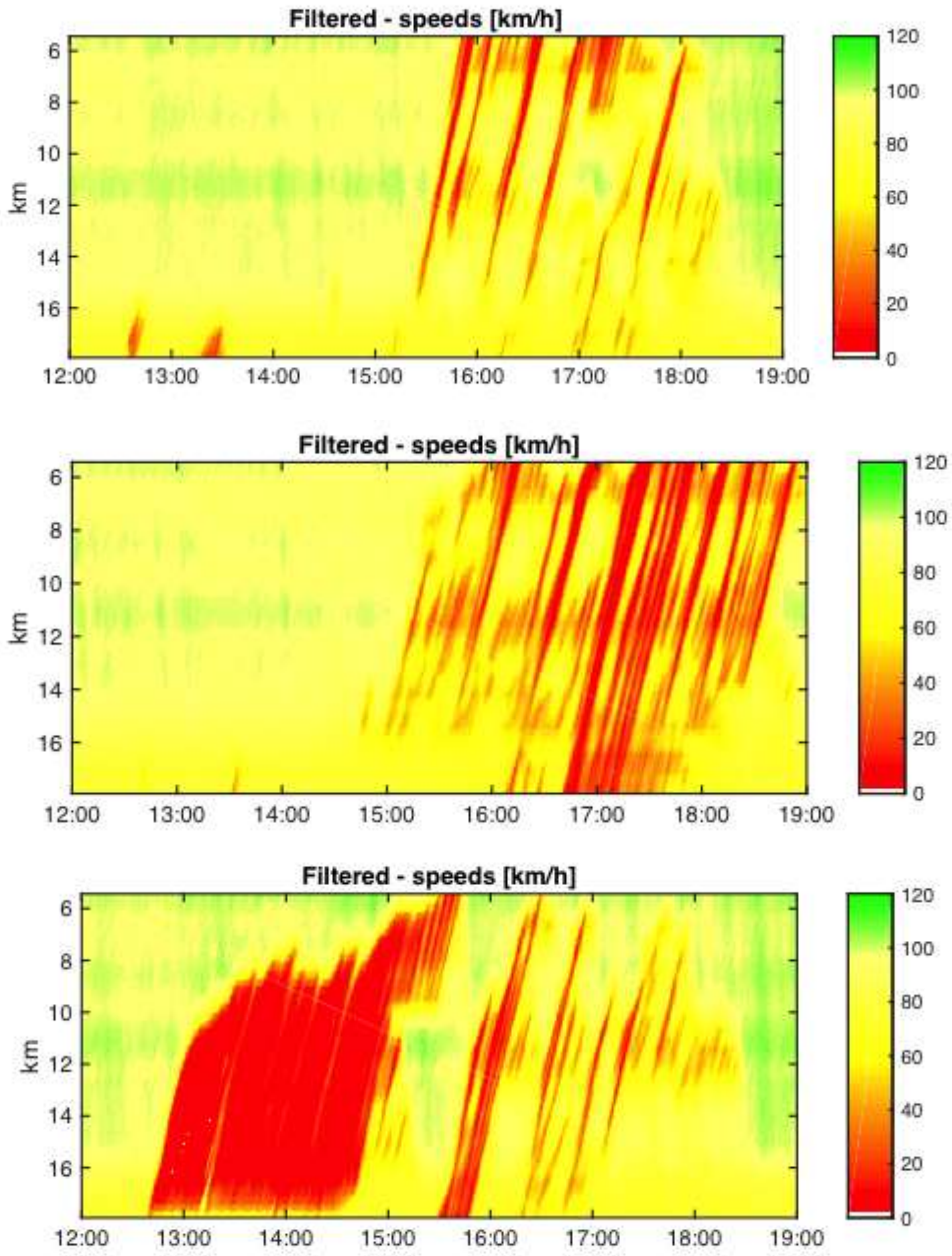
		RMSE (min)	MAPE (%)	MPE (%)	SPE (%)
Scenario A					
	FISB	3.2	7.9	7.0	7.9
	FSB	3.6	10.6	-9.4	7.8
	PLSB	4.0	11.3	-9.8	8.8
	PCSB	2.6	7.6	-3.3	8.9
Scenario B					
	FISB	2.9	8.1	6.1	9.8
	FSB	3.7	10.6	-8.5	9.1
	PLSB	5.8	15.5	-14.0	11.6
	PCSB	4.7	13.0	-7.2	14.2
Scenario C					
	FISB	4.3	11.4	10.4	11.0
	FSB	2.4	7.8	-5.1	8.1
	PLSB	5.9	16.6	-15.8	10.9
	PCSB	5.1	14.1	-7.8	16.7

Some lessons

- Orthogonal interpolation
  - Destroys shockwaves,
  - results in biased travel time estimates

use the ASM method

- Moreover
  - Real traffic data looks very different than predicted with first order traffic flow theory (CTM)
  - But that theory still is very powerful in reconstructing real traffic patterns



TUTORIALS IN
OPERATIONS RESEARCH
INFORMS—New Orleans 2005

informs
© 2005 INFORMS | ISBN 0000-0000
DOI 10.1287/educ.1053.0000

Applications of Kalman Filtering in Traffic Management and Control

Hans van Lint, PhD
Associate professor Traffic Management, Dept. Transport & Planning, Faculty of Civil Engineering Geosciences, Delft University of Technology, j.w.c.vanlint@tudelft.nl

Tamara Djukic
Dept. Transport & Planning, Faculty of Civil Engineering Geosciences, Delft University of Technology, t.djukic@tudelft.nl

Abstract In many areas of traffic management & control the variables that are of most interest are often the ones that are most difficult to measure and estimate. Take for example vehicular density (veh/km) and space-mean speed (km/h). A reliable real-time estimate of these quantities is critically important for real-time control of traffic networks. However, neither can be straightforwardly deduced from available sensor data.

Example 1: dynamic state space systems

INTRODUCTION KALMAN FILTER

Rudolf Kalman

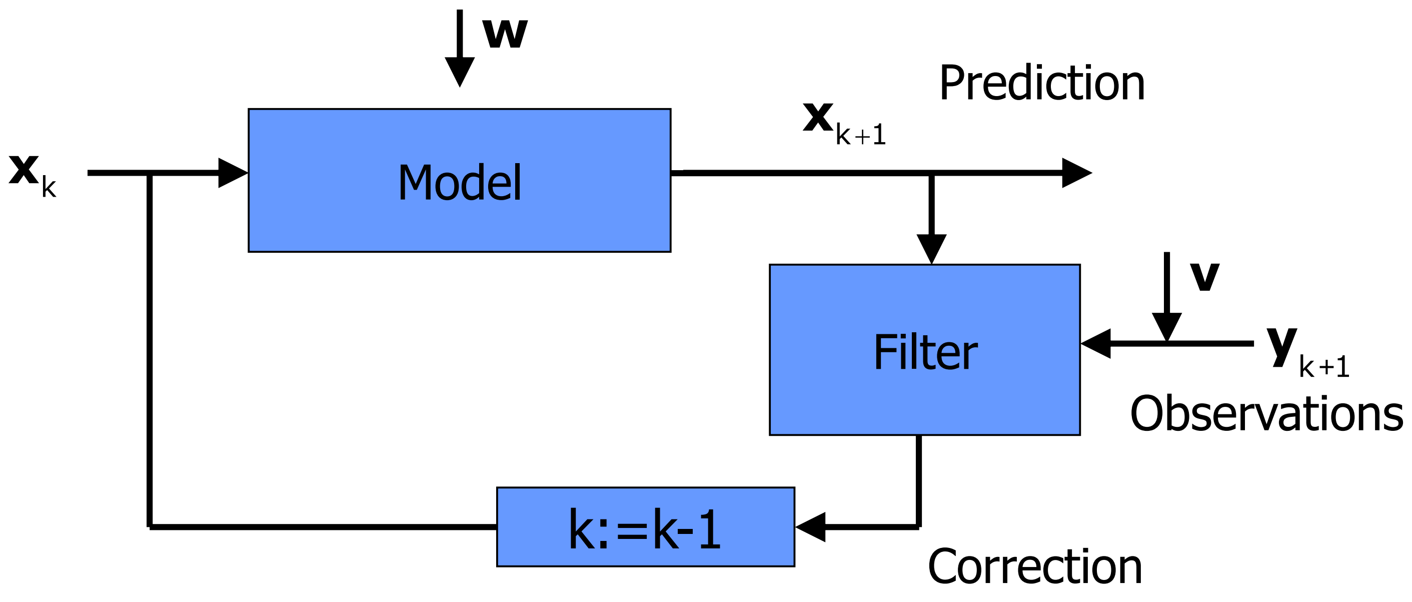
- Seminal paper:
  R.E. Kalman (1960). "A new approach to linear filtering and prediction problems". Journal of Basic Engineering **82** (1): 35–45

- Combine mathematical models with measurement information in a way that is efficient, optimal and elegant
- First important application: navigation of space-ships (Apollo program)
  - Data from radar, gyroscopes, visual observations
  - Knowledge regarding dynamic behavior of the space-ship
- Many other applications since then: location tracking, process control, oceanography, military applications, imaging, recursive parameter identification, etc
- Good resource / starting point: www.wikipedia.org/wiki/Kalman_filter

Kalman Filter Basics

Filter (blok) diagram

- It's a recursive method (memoryless)
- Intuitive prediction <–> correction structure
- Both are affected by noise (i.e. both are uncertain)



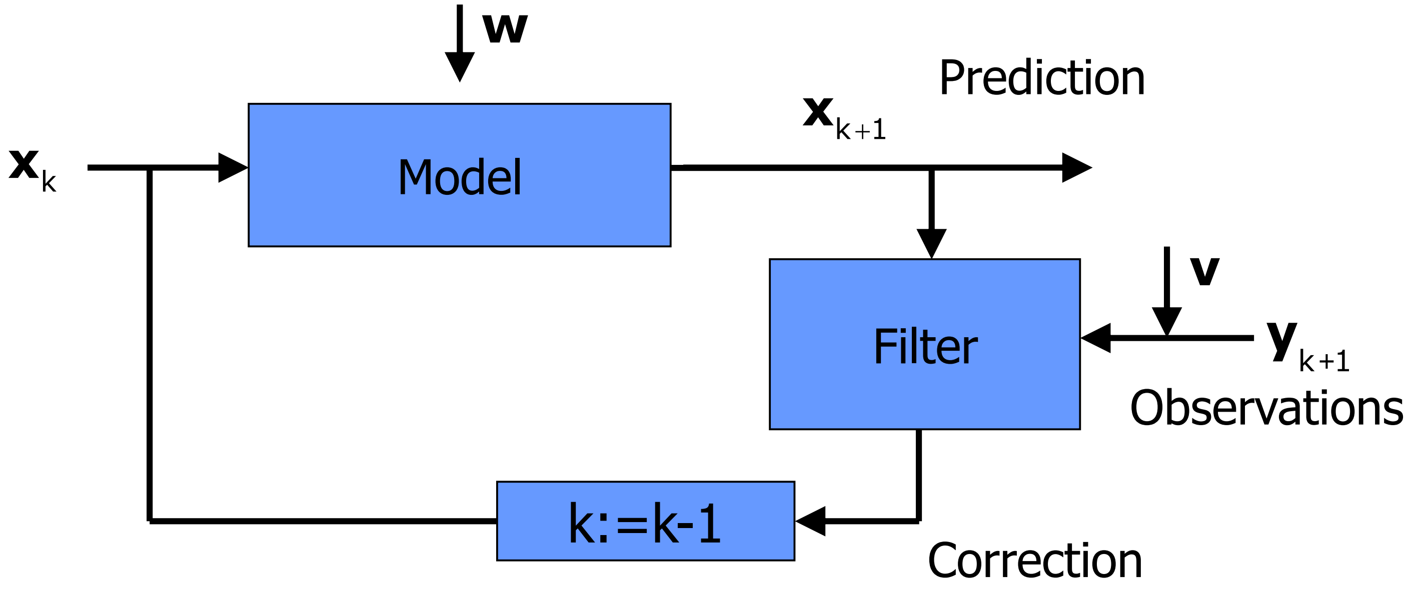
Kalman Filter Basics

Objectives and use

- Measurement information is used to find and eliminate modeling errors, errors in the input, and errors in the parameters

- Model prediction is used to eliminate outliers in the measurements and to smooth the measurement

- But how do you combine the two?



Example 1

Estimating the (CONSTANT) temperature in a room

- Suppose we have a model M for the temperature x:

$$x_M = x + w$$
$$E(w) = 0,\ E(w^2) = Q$$

- We have a thermostat that makes noisy observations y of x

$$y_T = x + v$$
$$E(v) = 0,\ E(v^2) = R$$

- Assume error terms are independent

$$E[wv] = 0$$

- Let's for now choose as an estimator for x the following linear combination of model and measurement:

$$\hat{x} = (1 - K)x_M + Ky_T$$

Example 1

Statistical properties (mean and variance) of this estimator

$$\hat{x} = (1-K)x_M + Ky_T$$

$$E(\hat{x}) = E(x)$$

Mean and variance:
(bit of algebra)

$$\text{var}(\hat{x}) = E\left[\left(\hat{x} - E[\hat{x}]\right)^2\right] = \ldots$$

$$= (1-K)^2 Q + K^2 R$$

For K=0 (no weight to measurement): $\text{var}(\hat{x}) = Q$

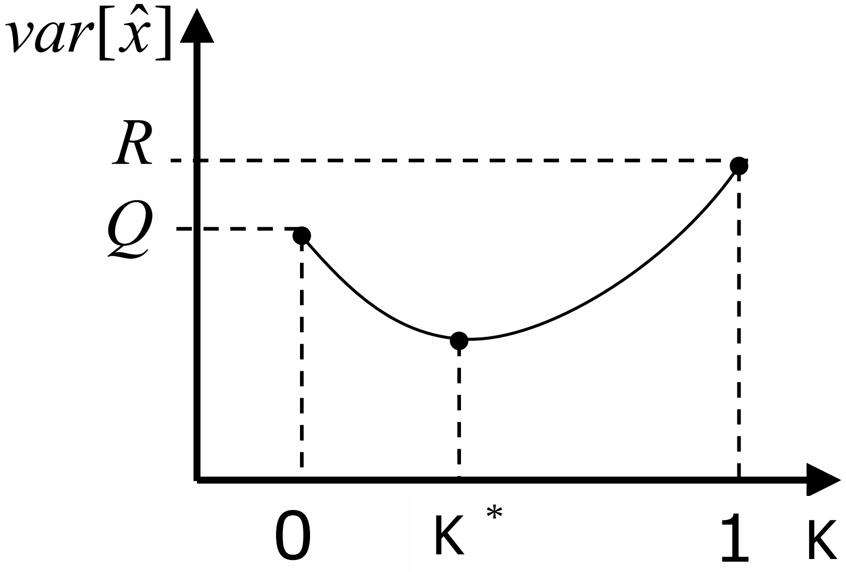
For K=1 (no weight to model): $\text{var}(\hat{x}) = R$

Example 1

Optimal G in terms of minimum error variance?

$$K = \arg\min \operatorname{var}(\hat{x})$$

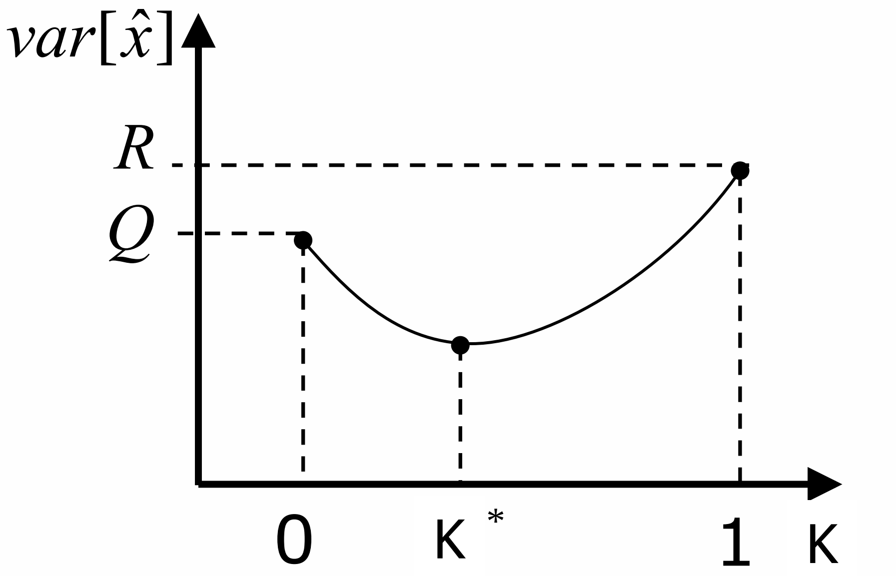
$$\frac{d}{dK}\operatorname{var}(\hat{x}) = -2(1-K)Q + 2KR = 0 \Rightarrow K^* = \frac{Q}{Q+R}$$



Example 1

Relation to the Kalman Filter

- Basics of the general Kalman filter are the same!
- The Kalman Filter is a multi-dimensional generalization based on a discrete-time dynamic system and measurement equation, i.e. models which can be formulated in **state-space** form:



$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{w}_k$$
$$\mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k$$

Discrete state space models

The good news: most processes can be cast in that form!

- We consider linear stochastic discrete time systems

$$
\begin{aligned}
\mathbf{x}_{k+1} &= F_k \mathbf{x}_k + \mathbf{w}_k \\
\mathbf{y}_k &= H_k \mathbf{x}_k + \mathbf{v}_k
\end{aligned}
$$

- Model and measurement noise are assumed Gaussian and mutually independent; the following holds

$$
E(\mathbf{w}_k) = 0, \quad E(\mathbf{w}_k \mathbf{w}_l') = \begin{cases} Q_k & k = l \\ 0 & k \neq l \end{cases}
$$

$$
E(\mathbf{v}_k) = 0, \quad E(\mathbf{v}_k \mathbf{v}_l') = \begin{cases} R_k & k = l \\ 0 & k \neq l \end{cases}
$$

Examples of state-space formulation

The good news: most processes can be cast in that form!

- This is an ARMA(1,1) model $$x_{k+1} = a x_k + w_k$$
- Now consider ARMA(2,1) model $$x_{k+1} = a x_k + b x_{k-1} + w_k$$
- Define new state $$\mathbf{z}_k = \begin{bmatrix} x_k & x_{k-1} \end{bmatrix}'$$
- then:

$$\mathbf{z}_{k+1} = \begin{bmatrix} x_{k+1} \\ x_k \end{bmatrix} = \begin{bmatrix} a x_k + b x_{k-1} + w_k \\ x_k \end{bmatrix} = \begin{bmatrix} a & b \\ 1 & 0 \end{bmatrix} \mathbf{z}_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} w_k$$

The Kalman filter

Initial conditions (discrete timestep k=0)

- Initial conditions for the state $\mathbf{x}_0$

$$E(\mathbf{x}_0) = \hat{\mathbf{x}}_0$$
$$\mathrm{var}(\mathbf{x}_0) = E([\mathbf{x}_0 - \hat{\mathbf{x}}_0][\mathbf{x}_0 - \hat{\mathbf{x}}_0]') = P_0$$

- KF is an estimator for the state that is **optimal in the following sense**
  - Maximum a-posteriori estimator

$$\hat{\mathbf{x}}_k = \arg\max_{\mathbf{x}_k} \Pr\left(\mathbf{x}_k \mid \mathbf{y}_1, \ldots, \mathbf{y}_k\right)$$

  - Conditional mean estimator

$$\hat{\mathbf{x}}_k = E\left[\mathbf{x}_k \mid \mathbf{y}_1, \ldots, \mathbf{y}_k\right]$$

  - Minimum variance estimator

$$\hat{\mathbf{x}}_k = \arg\min_{\mathbf{x}_k} E\left[\left\|\mathbf{x}_k - \hat{\mathbf{x}}_k\right\|^2\right]$$

Important property

If this thing is **Gaussian**, then these three estimators are identical !

The Kalman filter

Estimate of mean and error covariance at k:=k+1 …

$$
\begin{aligned}
\hat{\mathbf{x}}_{k+1|k} &= E\left(\mathbf{x}_{k+1|k}\right) \\
&= E\left(F_k \mathbf{x}_{k|k} + \mathbf{w}_k\right) \\
&= F_k E\left(\mathbf{x}_{k|k}\right) + E\left(\mathbf{w}_k\right) \\
&= F_k E\left(\mathbf{x}_{k|k}\right) \\
&= F_k \hat{\mathbf{x}}_{k|k}
\end{aligned}
$$

$$
\begin{aligned}
P_{k+1} &= \mathrm{var}\left[\mathbf{x}_{k+1}\right] \\
&= E\left[(\mathbf{x}_{k+1} - E[\mathbf{x}_{k+1}])(\mathbf{x}_{k+1} - E[\mathbf{x}_{k+1}])'\right] \\
&= E\left[(F_k \mathbf{x}_k + \mathbf{w}_k - F_k E[\mathbf{x}_k])(F_k \mathbf{x}_k + \mathbf{w}_k - F_k E[\mathbf{x}_k])'\right] \\
&= E\left[(F_k\left(\mathbf{x}_k - E[\mathbf{x}_k]\right) + \mathbf{w}_k)(F_k\left(\mathbf{x}_k - E[\mathbf{x}_k]\right) + \mathbf{w}_k)'\right] \\
&= F_k E\left[\left(\mathbf{x}_k - E[\mathbf{x}_k]\right)\left(\mathbf{x}_k - E[\mathbf{x}_k]\right)'\right] F_k' + E\left[\mathbf{w}_k \mathbf{w}_k'\right] \\
&= F_k P_k F_k' + Q_k
\end{aligned}
$$

Model prediction without noise!

Recursive Estimation: ADDITIVE errors

This describes the conditional mean estimator (equal to the minimum variance estimator, and the maximum a posteriori estimator!)

What about the measurements?

Remember we want to maximize $p(x_k | y_k, y_{k-1}, \ldots)$

- Reconsider our model in example 1:

$$x_M = x + w \quad \text{and} \quad y_T = x + v$$

$$w \sim N(0, Q) \quad \text{and} \quad v \sim N(0, R)$$

- Then obviously we have

$$x_M \sim N(x, Q) \quad \text{and} \quad y_T \sim N(x, R)$$

What about the measurements?

Remember we want to maximize $p(x_k | y_k, y_{k-1}, \ldots)$

- The conditional p.d.f. for x given y equals

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} = \frac{p(y|x)p(x)}{\int_{-\infty}^{\infty} p(y|x)p(x)\,dx}$$

$$\overset{y=x+v}{=} \frac{p(x+v|x)p(x)}{\int_{-\infty}^{\infty} p(x+v|x)p(x)\,dx} = \frac{p(v|x)p(x)}{\int_{-\infty}^{\infty} p(v|x)p(x)\,dx}$$

- Some tedious computation leaves us with:

$$p(x|y) \sim N\left(\frac{R}{Q+R}x_m + \frac{Q}{Q+R}y, \frac{QR}{Q+R}\right)$$

And the optimal filter is …

Remember we want to maximize $p(x_k | y_k, y_{k-1}, \ldots)$

- The maximum a-posteriori estimate in case of a Gaussian estimate is equal to the mean:

$$\hat{x} = \frac{R}{Q+R} x_m + \frac{Q}{Q+R} y_T \quad = (1 - K^*) x_m + K^* y_T \,!!!$$

- Note
  - this is equal to the minimum variance estimator we derived earlier
  - we did not make any prior assumptions regarding the linear structure of the estimator

- For general discrete linear processes with Gaussian noise:
  - Optimal filter is linear
  - Kalman filter is optimal in the sense of minimum variance

Recall: discrete state space models

Optimal filter estimate is indeed that nice linear combi of model predictions and measurements!

- We consider linear stochastic discrete time systems

$$\begin{aligned} \mathbf{x}_{k+1} &= F_k \mathbf{x}_k + G_k \mathbf{w}_k \\ \mathbf{y}_k &= H_k \mathbf{x}_k + \mathbf{v}_k \end{aligned}$$

- Model and measurement noise are assumed <u>Gaussian</u> and mutually independent; the following holds

$$E(\mathbf{w}_k) = 0, \quad E(\mathbf{w}_k \mathbf{w}_l') = \begin{cases} Q_k & k = l \\ 0 & k \neq l \end{cases}$$

$$E(\mathbf{v}_k) = 0, \quad E(\mathbf{v}_k \mathbf{v}_l') = \begin{cases} R_k & k = l \\ 0 & k \neq l \end{cases}$$

The Kalman Filter

The algorithm then becomes

1. Initial conditions $\hat{\mathbf{x}}_{0|0} = \hat{\mathbf{x}}_0, \quad P_{0|0} = P_0$

2. Time-propagation (prediction):

$$\hat{\mathbf{x}}_{k|k-1} = F_{k-1}\hat{\mathbf{x}}_{k-1|k-1}$$

$$P_{k|k-1} = F_{k-1}P_{k-1|k-1}F'_{k-1} + Q_{k-1}$$

3. Measurement adaptation (correction):

$$\hat{\mathbf{x}}_{k|k} = \left[I - K_k H_k\right]\hat{\mathbf{x}}_{k|k-1} + K_k \mathbf{y}_k$$

$$P_{k|k} = \left[I - K_k H_k\right]P_{k|k-1}\left[I - K_k H_k\right]' + K_k R_k K'_k$$

Kalman gain:

$$K_k = P_{k|k-1}H'_k\left[H_k P_{k|k-1} H'_k + R_k\right]^{-1}$$

The Kalman Filter

The algorithm then becomes (alternative formulation)

1. Initial conditions $$\hat{\mathbf{x}}_{0|0} = \hat{\mathbf{x}}_0, \quad P_{0|0} = P_0$$

2. Time-propagation (prediction): $$\hat{\mathbf{x}}_{k|k-1} = F_{k-1}\hat{\mathbf{x}}_{k-1|k-1}$$ $$P_{k|k-1} = F_{k-1}P_{k-1|k-1}F'_{k-1} + Q_{k-1}$$

3. Measurement adaptation (correction): $$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + K_k\left[\mathbf{y}_k - H_k\hat{\mathbf{x}}_{k|k-1}\right]$$ $$P_{k|k} = \left[I - K_kH_k\right]P_{k|k-1}\left[I - K_kH_k\right]' + K_kR_kK'_k$$

   Kalman gain: $$K_k = P_{k|k-1}H'_k\left[H_kP_{k|k-1}H'_k + R_k\right]^{-1}$$

The Kalman Filter

properties

- Note the clear predictor – corrector structure

- The filter produces besides the optimal estimate, the covariance matrix $P_{k|k}$ which is an important estimate for the accuracy of the estimate (error bars)

- The covariance $P_{k|k-1}$ and $P_{k|k}$ and filter-gain K_k do not depend on the measurements y_k (only on assumed error terms!) and can thus be computed off-line (drawbacks?)

A closer look at the filter gain

Looks familiar, right?

$$K_k = \frac{P_k H_k'}{H_k P_k H'_k + R_k}$$

Remember Example 1: $K^* = \frac{P}{P+R}$

A closer look at the filter gain

$$\text{Kalman Gain} = \frac{\text{Variance in process equation} \times \text{Sensitivity measurement equation}}{\text{Variance in measurement equation}}$$

So balance between uncertainty in process model and uncertainty in measurements / observation model!

All KF assumptions matter (largely):
Assumption 1. The process and observation models can be formulated as a linear discrete state space system of equations.

Assumption 2. All noise terms in this system are independent, zero-mean Gaussian noise processes of known covariance matrices.

Example 2

TRACKING A BICYCLE TRIP

Tracking a bicycle

Using noisy data

- Consider you're on a bicycle trip
- Your driving with 10 km/h (approx. constant speed) , due to wind, grades, i.e. some variation in the speeds are present.
- This is our process model:

$$s_k = s_{k-1} + \Delta t u_k$$
$$u_k = u_{k-1} + w_k.$$

$$\Rightarrow \quad \mathbf{x}_k = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \mathbf{x}_{k-1} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w_k$$

- You measure both location and speed using your knowledge of the terrain, physiology, celestial bodies. Per hour, we want to determine location *s* and speed *u* of bicycle. (and you have a GPS in your pocket to check how well you've done)

Tracking a bicycle

Process and observation models

$$s_k = s_{k-1} + \Delta t u_k$$
$$u_k = u_{k-1} + w_k.$$

$\Rightarrow$

$$\mathbf{x}_k = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \mathbf{x}_{k-1} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w_k$$

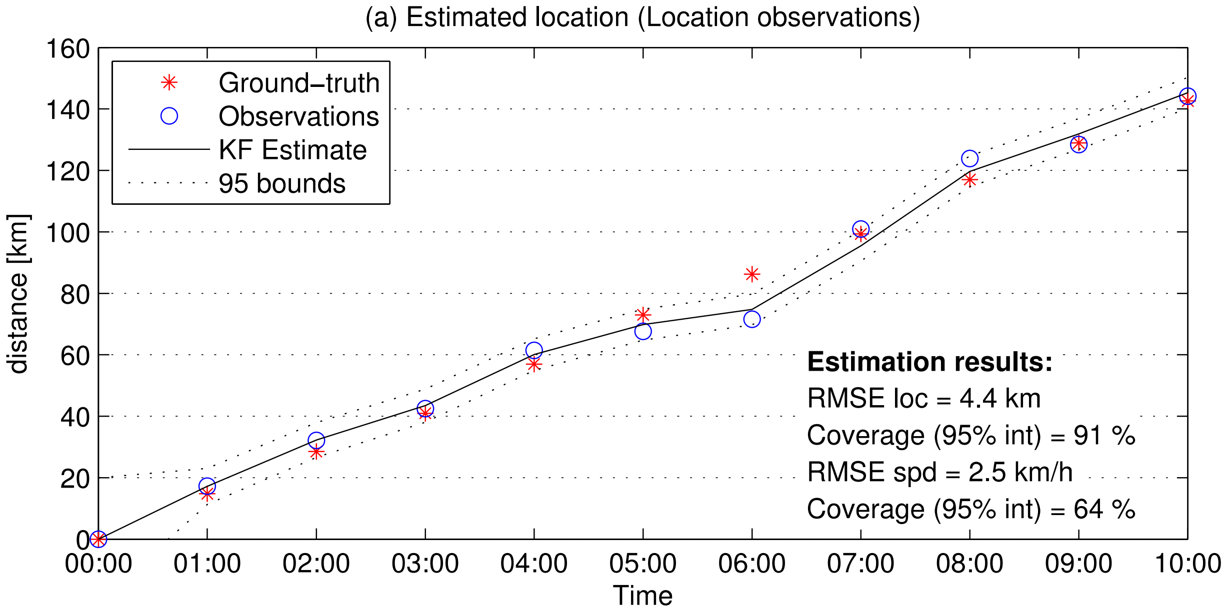
Per hour, we want to determine location *s* and speed *u* of bicycle

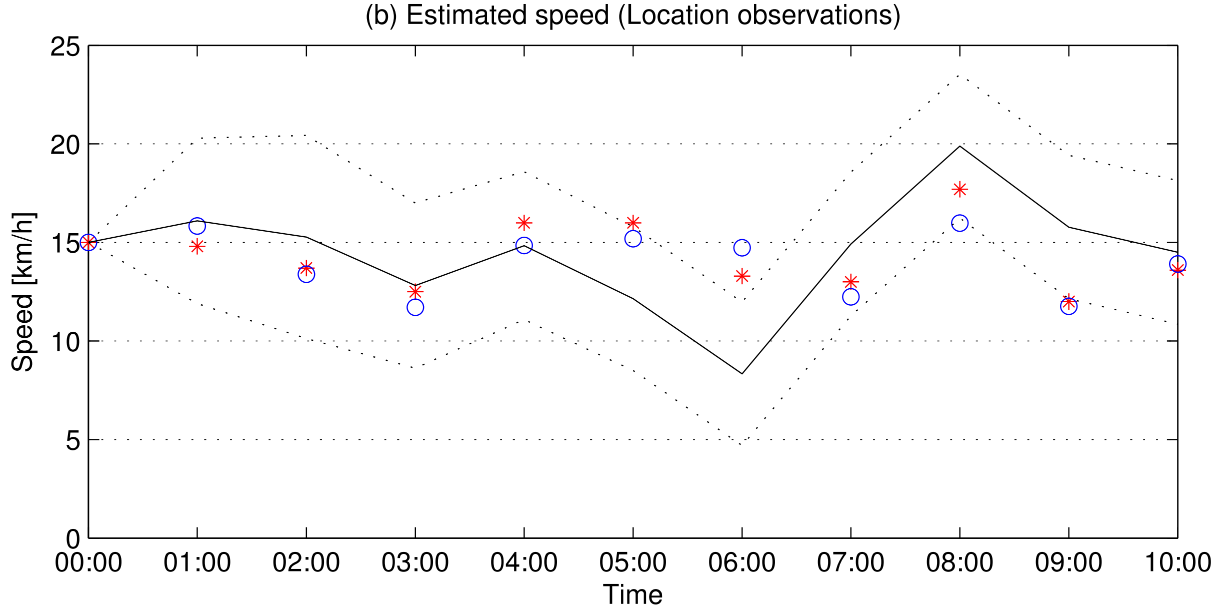
1. Using observations of the location $y_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}_k + v_k^s$
2. Using your obs on the speeds $y_k = \begin{bmatrix} 0 & 1 \end{bmatrix} \mathbf{x}_k + v_k^u$
3. Using both! $\mathbf{y}_k = \mathbf{x}_k + \mathbf{v}_k$
4. Same as 3, now adaptive $R_k = \gamma R_{k-1} + (1-\gamma)\, \mathbf{e}_k \mathbf{e}_k^T$

Tracking a bicycle

Additional scenarios: initial assumptions

Variable	Value	Remark
$\hat{\mathbf{x}}_0$	$[0, 15]^T$	Start location and speed
P_0	$\begin{bmatrix} 10^2 & 0 \\ 0 & 10^2 \end{bmatrix}$	Initial error covariance large
Q_k	$\begin{bmatrix} 0 & 0 \\ 0 & 2^2 \end{bmatrix}, \forall k$	No error on the law of motion, an assumed variance 4 (km^2/h^2) for the speed dynamics
R_k^A	$\begin{bmatrix} 3^2 & 0 \\ 0 & 1^2 \end{bmatrix}, \forall k$	With this observation error covariance matrix we (correctly!) assume larger errors in the location observations than in the speed observations
R_k^B	$\begin{bmatrix} 1^2 & 0 \\ 0 & 3^2 \end{bmatrix}, \forall k$	With this observation error covariance matrix we (incorrectly) assume larger errors in the speed observations than in the location observations

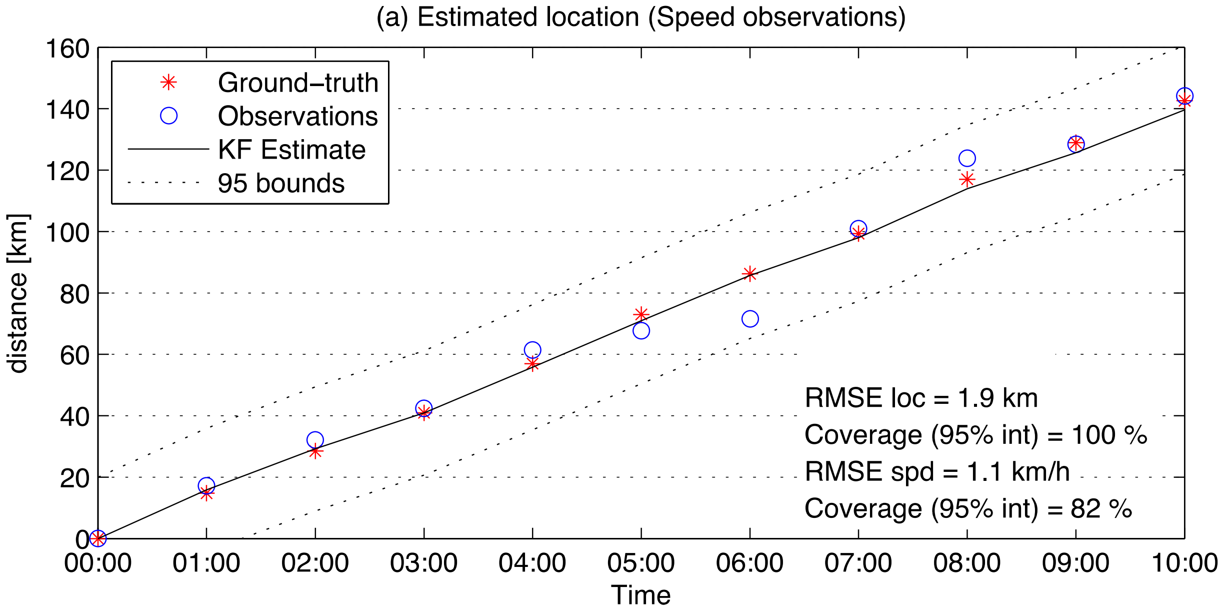


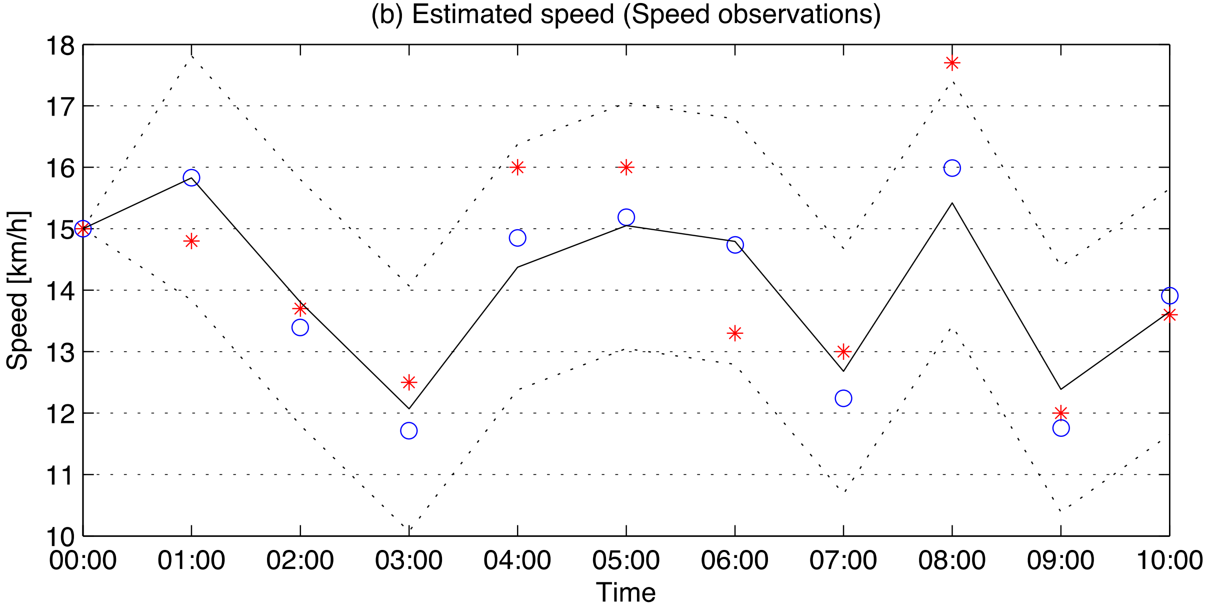


KF estimate improves RMSE locations but worsens speed estimates

Just locations

Time (h)	Ground truth s^{GPS}	u^{GPS}	Observations s^{obs}	u^{obs}
0	0.0	15.0	0.0	15.0
1	14.8	14.8	17.2	15.8
2	28.5	13.7	32.1	13.4
3	41.0	12.5	42.5	11.7
4	57.0	16.0	61.4	14.9
5	73.0	16.0	67.7	15.2
6	86.3	13.3	71.6	14.7
7	99.3	13.0	100.9	12.2
8	117.0	17.7	123.9	16.0
9	129.0	12.0	128.5	11.8
10	142.6	13.6	144.2	13.9
RMSE			5.5	0.9
CIC				



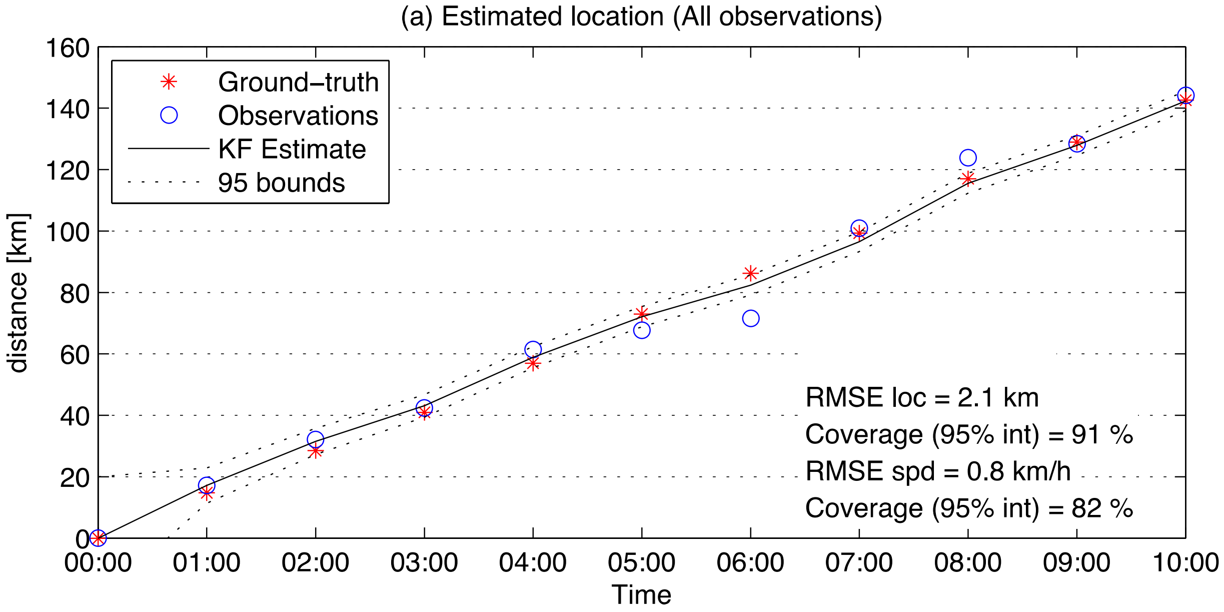


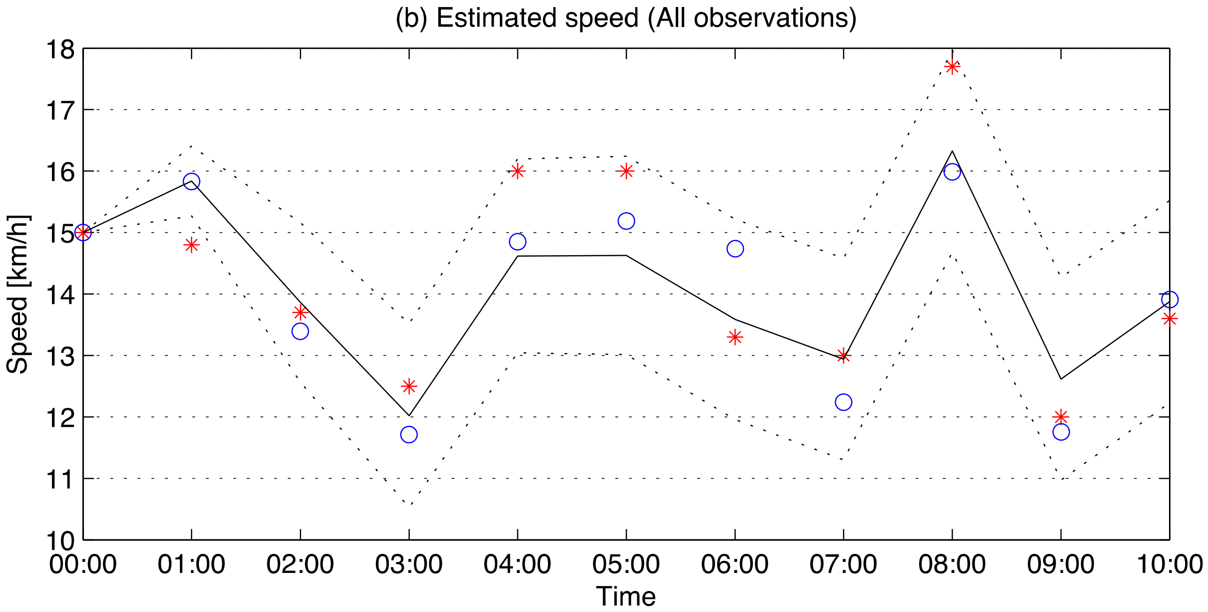
KF estimate improves RMSE locations and *slightly* worsens speed estimates

Just speeds

Time (h)	Ground truth s^{GPS}	u^{GPS}	Observations s^{obs}	u^{obs}
0	0.0	15.0	0.0	15.0
1	14.8	14.8	17.2	15.8
2	28.5	13.7	32.1	13.4
3	41.0	12.5	42.5	11.7
4	57.0	16.0	61.4	14.9
5	73.0	16.0	67.7	15.2
6	86.3	13.3	71.6	14.7
7	99.3	13.0	100.9	12.2
8	117.0	17.7	123.9	16.0
9	129.0	12.0	128.5	11.8
10	142.6	13.6	144.2	13.9
RMSE			5.5	0.9
CIC				

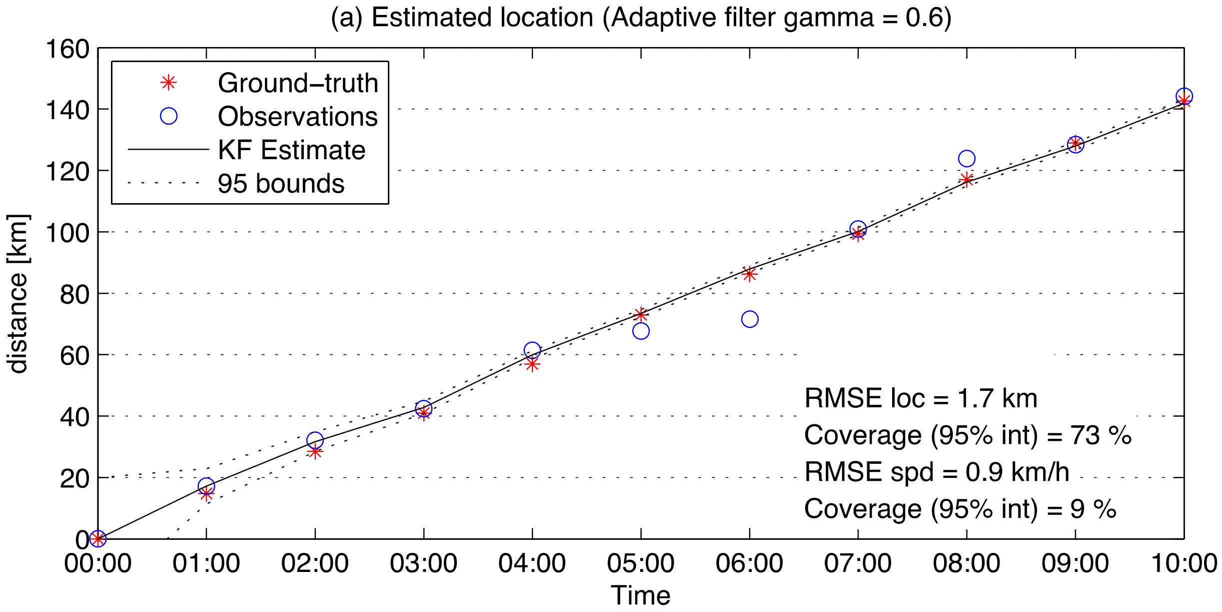
KF estimate improves RMSE locations AND speed estimates

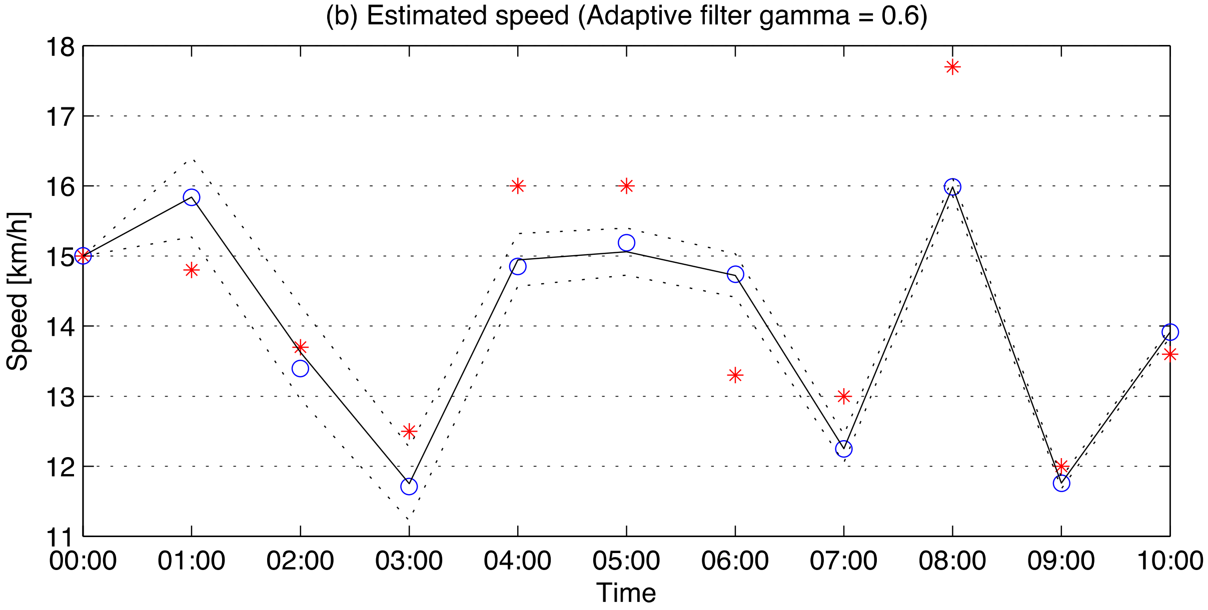




Both location and speed data

Time (h)	Ground truth s^{GPS}	u^{GPS}	Observations s^{obs}	u^{obs}
0	0.0	15.0	0.0	15.0
1	14.8	14.8	17.2	15.8
2	28.5	13.7	32.1	13.4
3	41.0	12.5	42.5	11.7
4	57.0	16.0	61.4	14.9
5	73.0	16.0	67.7	15.2
6	86.3	13.3	71.6	14.7
7	99.3	13.0	100.9	12.2
8	117.0	17.7	123.9	16.0
9	129.0	12.0	128.5	11.8
10	142.6	13.6	144.2	13.9
RMSE			5.5	0.9
CIC				





KF estimate improves RMSE locations AND speed estimates (but CIC deteriorates)

location and speed data + adaptive filter

	Ground truth		Observations	
Time (h)	s^{GPS}	u^{GPS}	s^{obs}	u^{obs}
0	0.0	15.0	0.0	15.0
1	14.8	14.8	17.2	15.8
2	28.5	13.7	32.1	13.4
3	41.0	12.5	42.5	11.7
4	57.0	16.0	61.4	14.9
5	73.0	16.0	67.7	15.2
6	86.3	13.3	71.6	14.7
7	99.3	13.0	100.9	12.2
8	117.0	17.7	123.9	16.0
9	129.0	12.0	128.5	11.8
10	142.6	13.6	144.2	13.9
RMSE			5.5	0.9
CIC				

Time (h)	Ground truth s^{GPS}	u^{GPS}	Observations s^{obs}	u^{obs}	Scenario I-A $\hat{s}_k$	$\hat{u}_k$	Scenario II-A $\hat{s}_k$	$\hat{u}_k$	Scenario III-A $\hat{s}_k$	$\hat{u}_k$	Scenario IV-A $\hat{s}_k$	$\hat{u}_k$
0	0.0	15.0	0.0	15.0	0.0	15.0	0.0	15.0	0.0	15.0	0.0	15.0
1	14.8	14.8	17.2	15.8	17.1	16.1	15.8	15.8	17.1	15.8	17.1	15.8
2	28.5	13.7	32.1	13.4	32.3	15.3	29.2	13.8	31.5	13.9	31.7	13.6
3	41.0	12.5	42.5	11.7	43.5	12.8	40.9	12.1	43.1	12.0	42.8	11.8
4	57.0	16.0	61.4	14.9	60.1	14.8	55.8	14.4	58.9	14.6	60.0	14.9
5	73.0	16.0	67.7	15.2	69.8	12.2	71.0	15.1	72.1	14.6	73.5	15.1
6	86.3	13.3	71.6	14.7	74.7	8.3	85.7	14.8	82.4	13.6	87.9	14.7
7	99.3	13.0	100.9	12.2	95.5	14.9	98.0	12.7	96.5	12.9	100.2	12.3
8	117.0	17.7	123.9	16.0	119.7	19.9	114.0	15.4	115.6	16.3	116.2	16.0
9	129.0	12.0	128.5	11.8	131.9	15.8	125.7	12.4	127.9	12.6	127.9	11.8
10	142.6	13.6	144.2	13.9	145.3	14.5	139.6	13.7	142.4	13.9	141.9	13.9
RMSE			5.5	0.9	4.4	2.5	1.9	1.1	2.1	0.8	1.7	0.9
CIC					90.9	63.6	100.0	81.8	90.9	81.8	72.7	9.1

Scenario I-B $\hat{s}_k$	$\hat{u}_k$	Scenario II-B $\hat{s}_k$	$\hat{u}_k$	Scenario III-B $\hat{s}_k$	$\hat{u}_k$	Scenario IV-B $\hat{s}_k$	$\hat{u}_k$
0.0	15.0	0.0	15.0	0.0	15.0	0.0	15.0
17.2	16.1	15.8	15.8	17.2	15.9	17.2	15.9
32.2	15.0	29.2	14.4	32.1	14.8	32.0	14.6
42.9	11.7	41.0	13.0	43.0	11.7	42.6	11.1
60.6	16.4	55.8	13.9	60.5	16.2	61.1	17.7
68.8	9.9	71.0	14.5	69.1	10.5	68.0	7.3
72.4	5.0	85.8	14.6	73.3	5.9	71.7	3.9
98.1	21.3	98.0	13.5	97.6	20.1	100.6	27.7
123.3	24.4	114.0	14.7	122.5	23.5	123.9	23.4
130.8	11.1	125.8	13.3	130.7	11.4	128.6	5.3
143.9	12.7	139.7	13.6	144.0	13.0	144.1	15.4
5.2	4.5	1.8	1.4	4.8	4.0	5.5	6.5
36.4	54.5	100.0	90.9	36.4	45.5	9.1	9.1

All scenarios with the wrong assumptions on Q and R perform worse than the "A" scenarios (even worse than no KF at all)

Don't put your money on the wrong data …

All KF assumptions matter (largely):

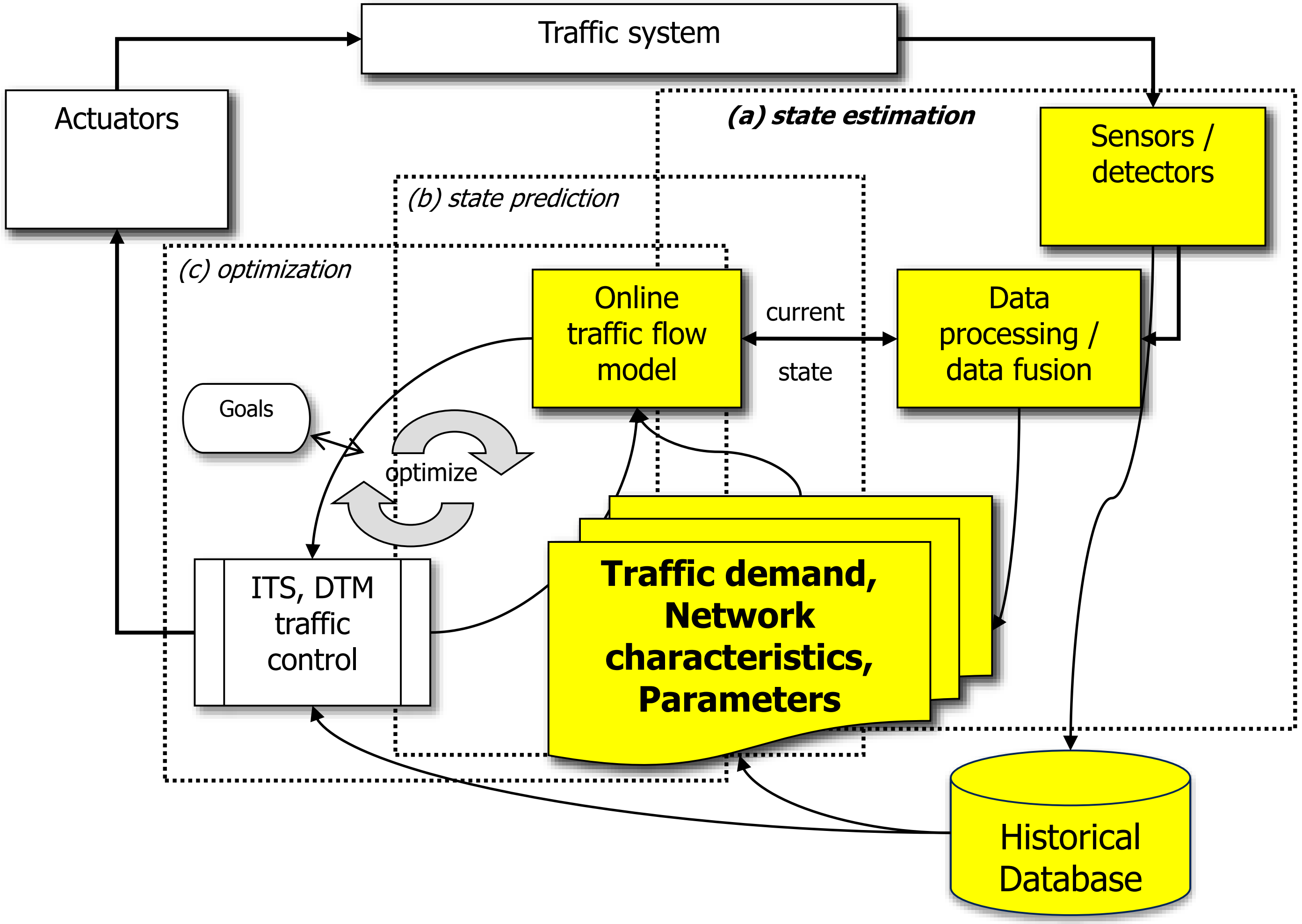
Assumption 1. The process and observation models can be formulated as a linear discrete state space system of equations.

Assumption 2. All noise terms in this system are independent, zero-mean Gaussian noise processes of known covariance matrices.

Example 3

ESTIMATING OD FLOWS

Crucial component: estimating inputs (off and online) from data



Estimating OD matrices

General problem formulation

OD flows

Prior OD flows

Observations

$$\hat{x} = \arg\min_{x} f_1(x, \tilde{x}) + f_2(y(x), \tilde{y})$$

Observations predicted using OD flows

subject to

$$x \geq 0 \quad , \quad y \geq 0$$
$$\ldots$$

Estimating OD matrices

General problem formulation

$$\hat{\mathrm{x}} = \arg\min_{\mathrm{x}} f_1(\mathrm{x}, \tilde{\mathrm{x}}) + f_2(\mathrm{y}(\mathrm{x}), \tilde{\mathrm{y}})$$

This term (gently) forces solution in direction of prior, but no garantuees

(alternative is hard constraints)

Estimating OD matrices

General problem formulation

$$\hat{\mathrm{x}} = \arg\min_{\mathrm{x}} f_1(\mathrm{x}, \tilde{\mathrm{x}}) + f_2(\mathrm{y}(\mathrm{x}), \tilde{\mathrm{y}})$$

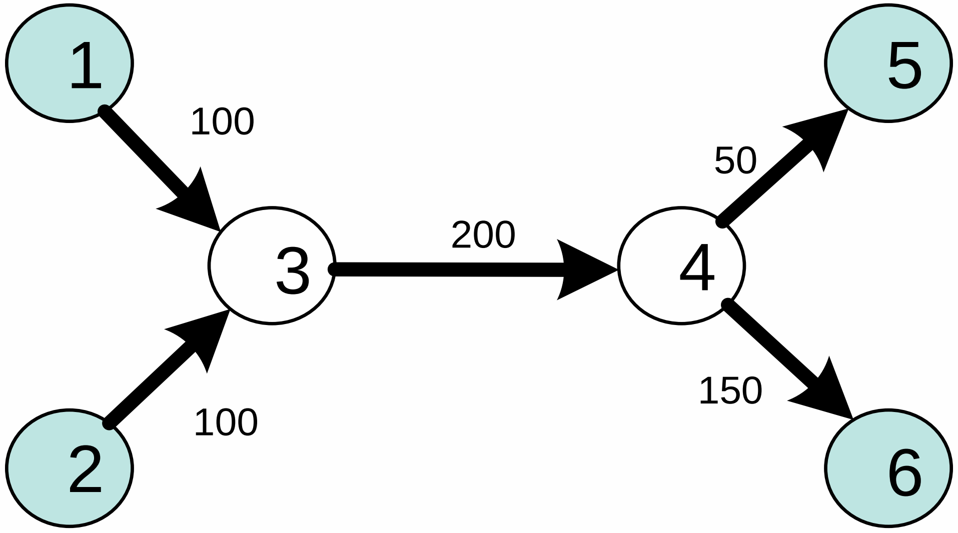
State-space model: Solvable with a Kalman Filter!

$$x_{r,k+1} = \sum_{p=k-q'}^{k} f_{rp} x_{rp} + w_{rk}$$

$$y_{lc} = \sum_{p=1}^{k} \sum_{r=1}^{N_r} a_{lk}^{rp} x_{rp} + v_{lk}$$

Estimating OD matrices

1st key problem: under-determinedness



$$
\begin{aligned}
L1:&\quad 100 = x_{15} + x_{16} \\
L2:&\quad 100 = x_{25} + x_{26} \\
L3:&\quad 200 = x_{15} + x_{16} + x_{25} + x_{26} \\
L4:&\quad 50 = x_{15} + x_{25} \\
L5:&\quad 150 = x_{16} + x_{26}
\end{aligned}
\quad \xrightarrow[L4' = L4 - L1]{L3' = L3 - L1 = L2}
$$

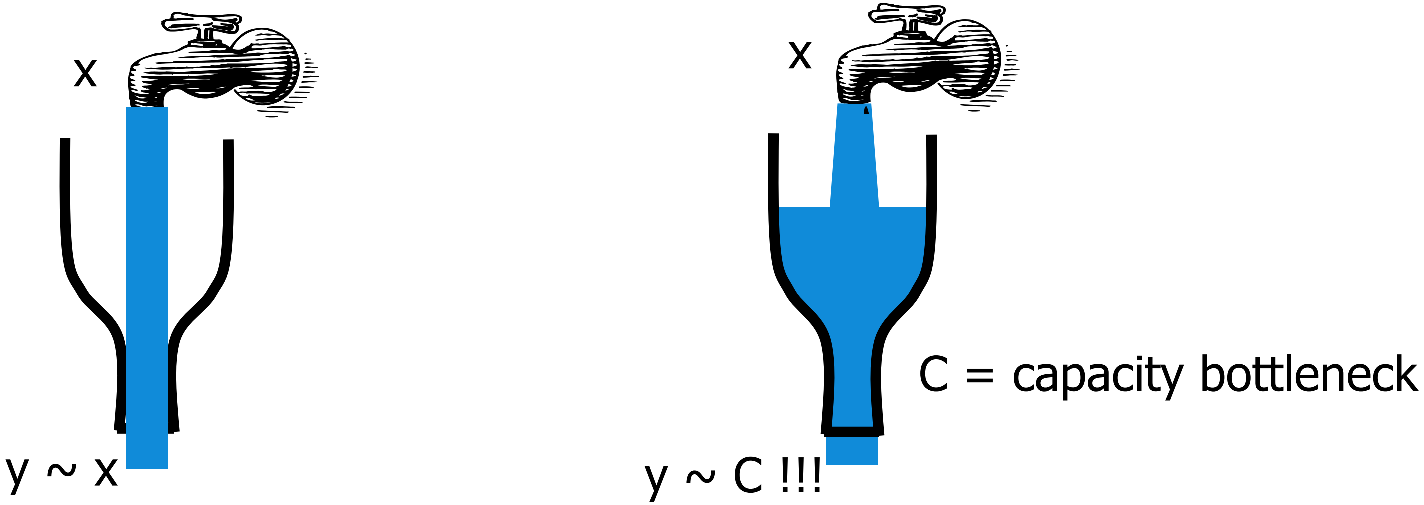
$$
\begin{aligned}
L1:&\quad 100 = x_{15} + x_{16} \\
L2:&\quad 100 = x_{25} + x_{26} \\
L4':&\quad -50 = -x_{16} + x_{25} \\
L5:&\quad 150 = x_{16} + x_{26}
\end{aligned}
\quad \xrightarrow[L4'' = -L4']{L5' = L5 + L3' = L2}
$$

$$
\begin{aligned}
L1:&\quad 100 = x_{15} + x_{16} \\
L2:&\quad 100 = x_{25} + x_{26} \\
L4'':&\quad 50 = x_{16} - x_{25}
\end{aligned}
$$

Estimating OD matrices

But there are more (fundamental) problems

- The simple observation model $y = Ax$ is **flawed** in congestion



- There is mutual dependence between the assignment and the OD matrices: $A = A(x)$, but also $x = x(A)$: **fixed point problem**
  - Route and departure time choice depend on prevailing conditions
- Gaussian assumption very unrealistic (OD flows < 0 ????)

Estimating OD Matrix

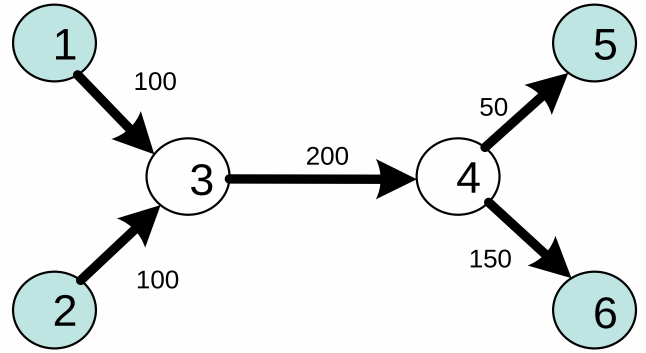
Ashok & Ben-Akiva (2000)

- By reformulating the problem it becomes (a) much more linear and (b) the noise terms become much more Gaussian

$$\partial \mathbf{x}_{k+1} = \sum_{p=k-q'}^{k} \mathbf{f}_k^p \partial \mathbf{x}_p + \mathbf{w}_k$$

$$\partial \mathbf{y}_k = \sum_{p=k-p'}^{k} \mathbf{a}_k^p \partial \mathbf{x}_p + \mathbf{v}_k$$

- Given
  - **Uncongested** network
  - **Sufficient** data
  - **Good** prior assumptions



- The Kalman filter does a pretty good job in reconstructing OD matrices …

We now only consider 3 unknowns: x_{15}, x_{16} and x_{26}

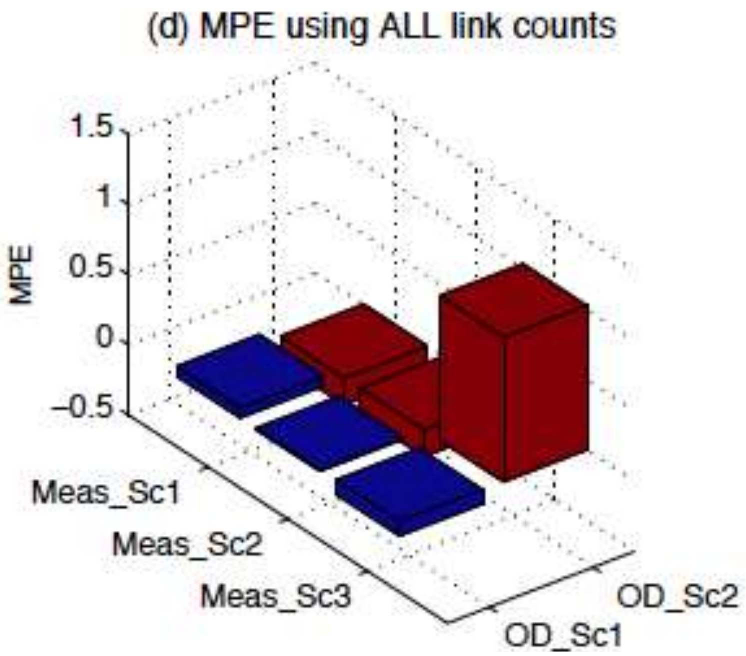
Example

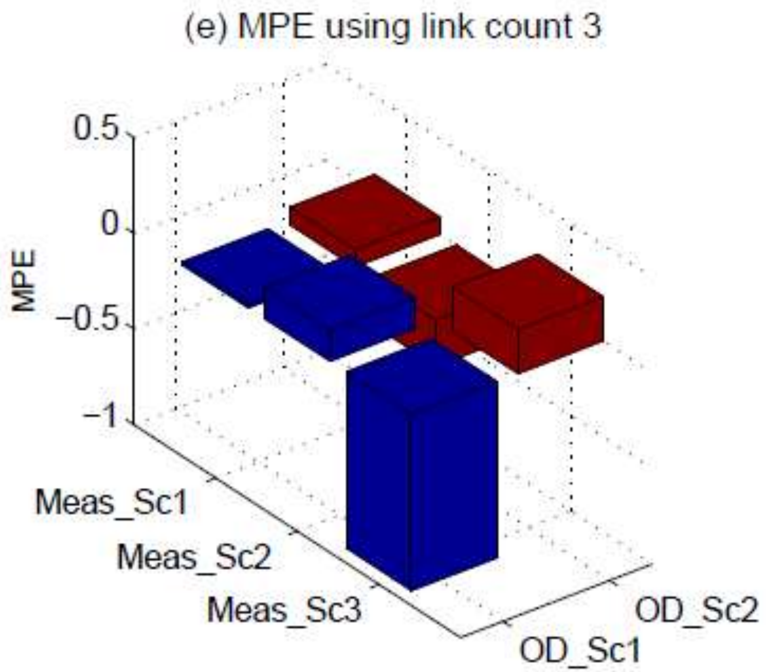
Scenario’s and results

- We perturb the prior OD
  - With random errors (OD scenario 1)
  - With structural errors (OD scenario 2)

- We perturb the measurements
  - With 20% random errors
  - With 40% random erros
  - With 60% random errors

- We restrict measurement availability
  - ALL data
  - Only link 3
  - Link 1 and 4





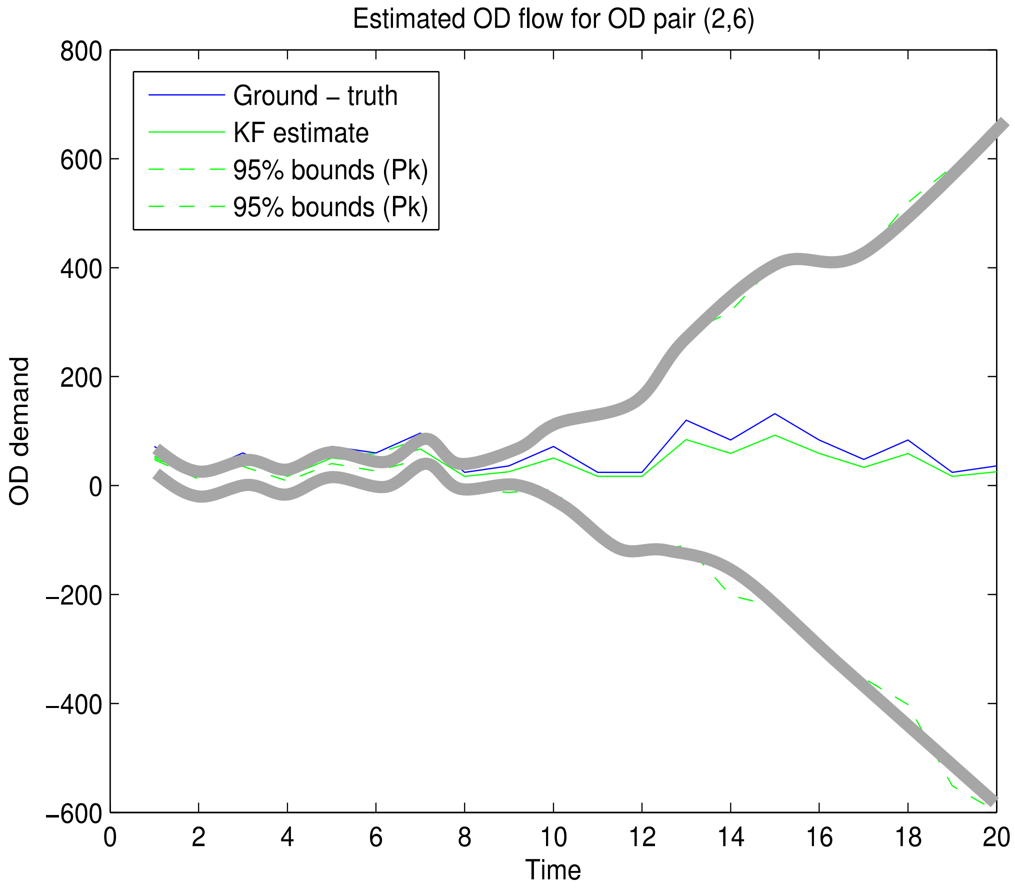
Example shows

(Based on work of Ashok and Ben-Akiva)

- By reformulating the problem it does become (a) much more linear and (b) the noise terms become much more Gaussian

- Notes:

- Underdeterminedness is not solved, BUT
- Filter divergence tell-tale sign



Non-linear KF approach: extended KF

Assumption 1. The process and observation models can be formulated as a linear discrete state space system of equations.

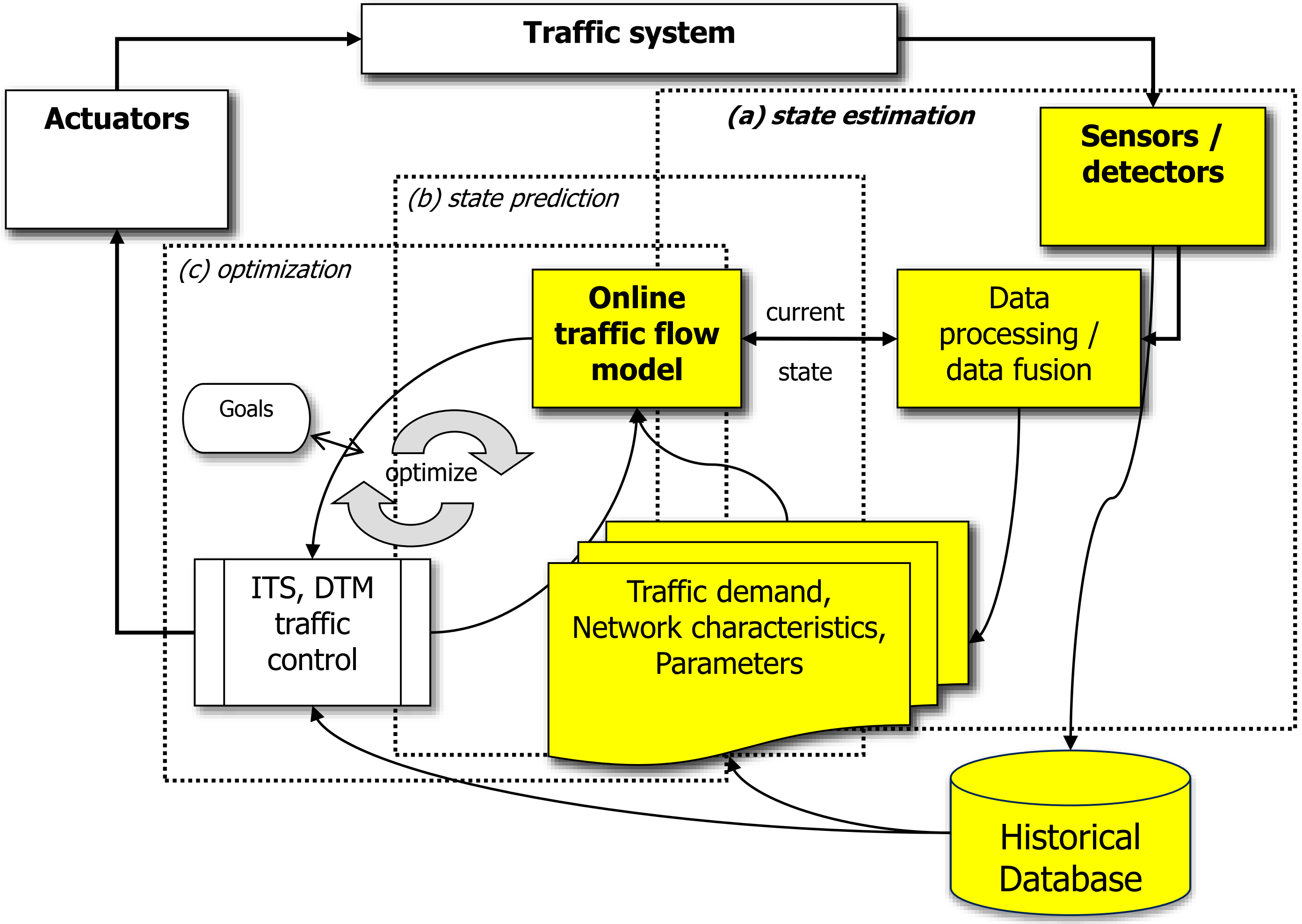
Assumption 2. All noise terms in this system are independent, zero-mean Gaussian noise processes of known covariance matrices.

Assumption 3: process & observation models can be sufficiently accurately approximated through first order linearization around current state

Example 4

TRAFFIC STATE ESTIMATION

Traffic state estimation: traffic flow models + **Extended** KF



Recall traffic flow theory

Process model: discrete solution pdf traffic flow
Observation model: fundamental diagram $Q^e(\rho)$

$$\mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k, \psi_k, \mathbf{d}_k\right) + \eta_k$$
$$\mathbf{y}_k = \mathbf{h}\left(\mathbf{x}_k, \psi_k\right) + \zeta_k.$$

$$\frac{\partial \rho}{\partial t} + c(\rho)\frac{\partial \rho}{\partial x} = 0, \qquad c(\rho) = \frac{\partial}{\partial \rho} Q^e(\rho)$$

Change in density (veh/km) over time

Change in flow (km/h) over space

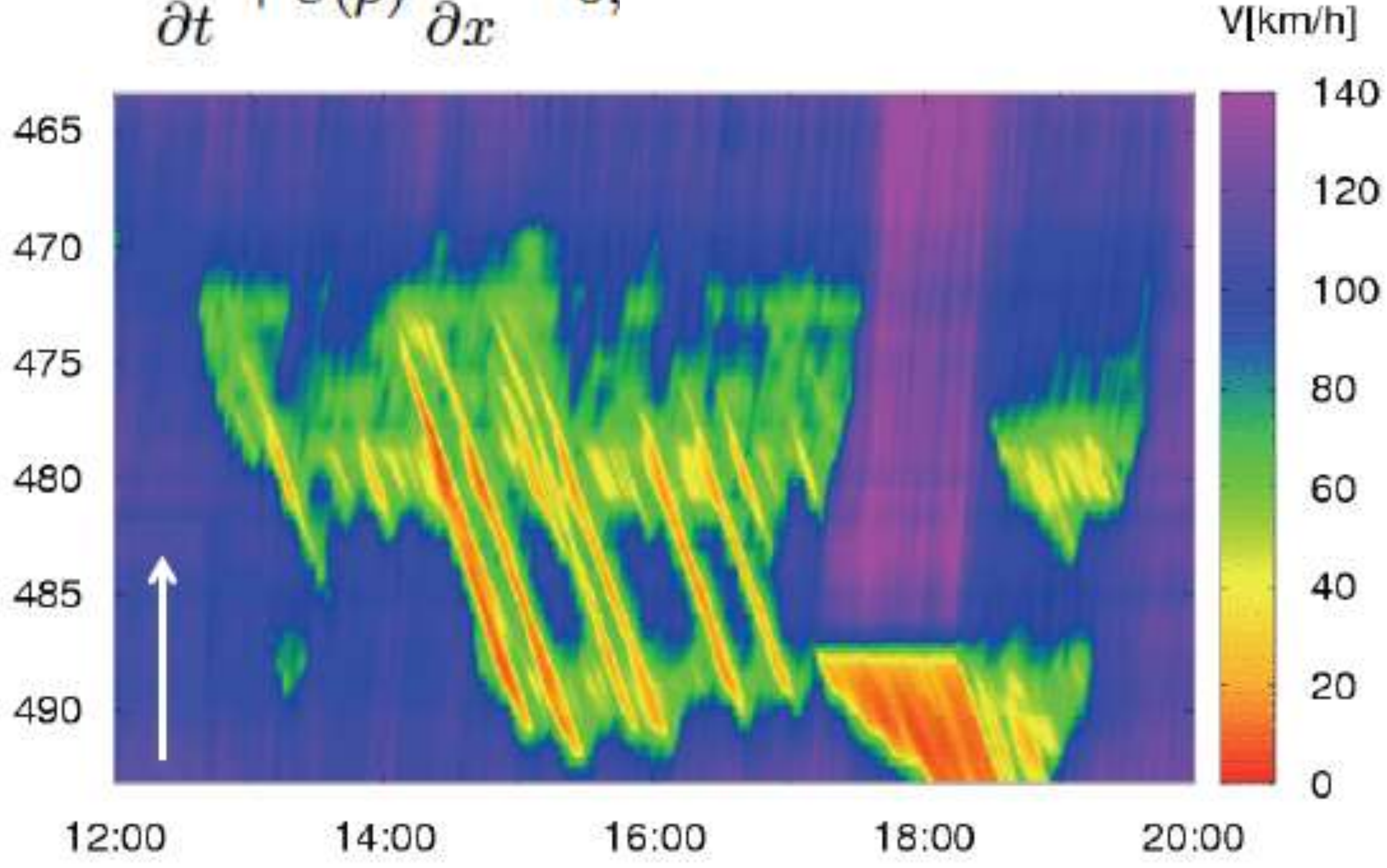
Governed by the speed c (function of density) with which perturbations move over space and time

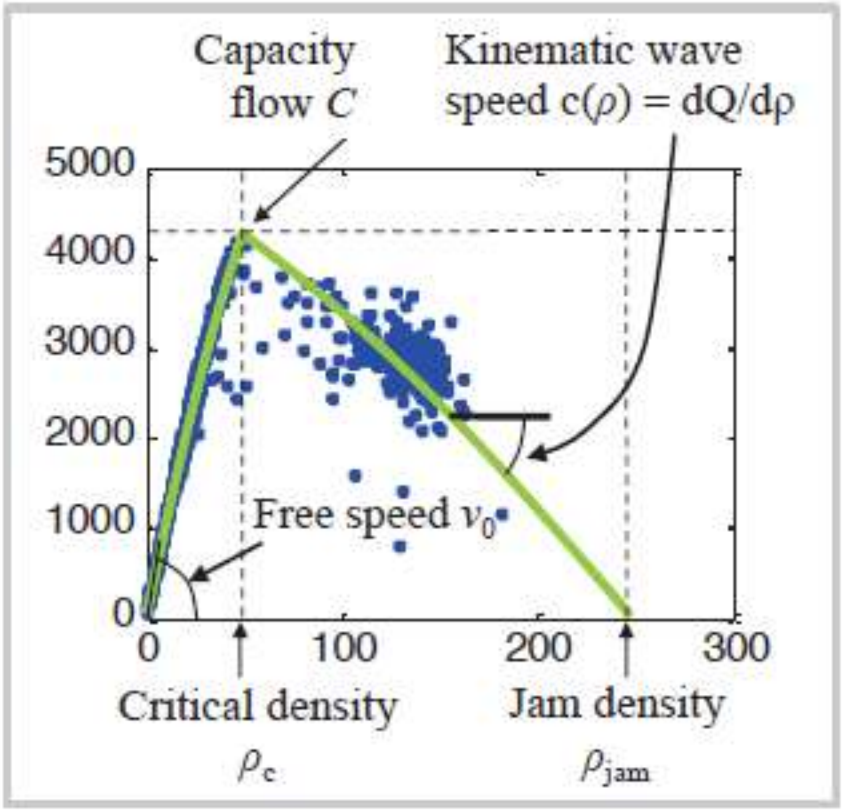
Recall traffic flow theory

Reality check … only a first order approximation

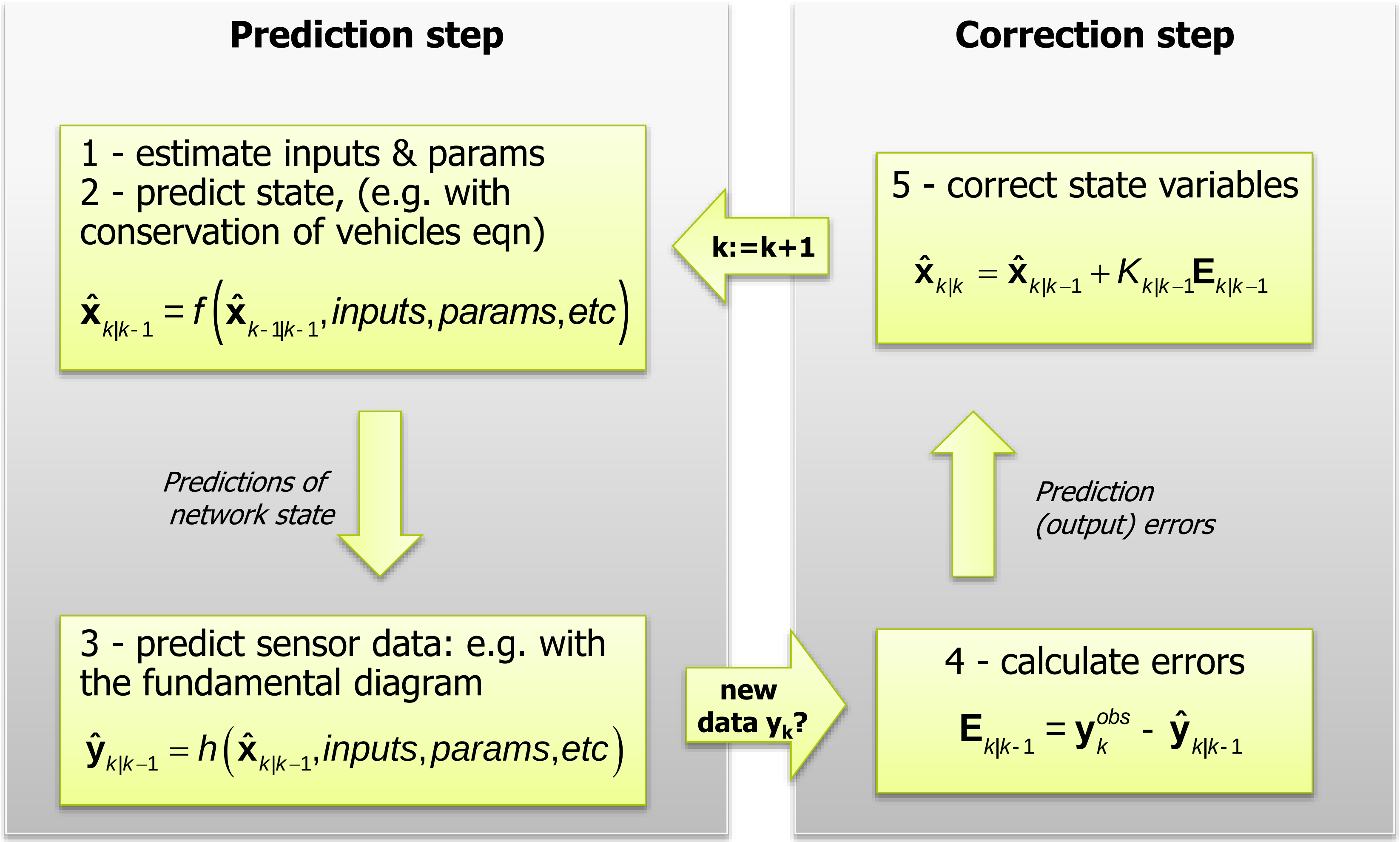
$$\frac{\partial\rho}{\partial t}+c(\rho)\frac{\partial\rho}{\partial x}=0,$$

$$c(\rho)=\frac{\partial}{\partial\rho}Q^{e}(\rho)$$





Discretized Traffic flow model + (Extended) Kalman Filter



Here's the EKF math

For $k=1,2,\ldots$ **do:**

Predict mean and variance of state variables:

$$\hat{\mathbf{x}}_k^- = \mathbf{f}\left(\mathbf{x}_k, \psi_k, \mathbf{d}_k\right)$$
$$P_k^- = F_{k|k-1} P_{k-1} F_{k|k-1}^T + Q_{k-1}$$

in which

$$F_{k|k-1} = \left.\frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_k^-}$$

Predict output variables:

$$\hat{\mathbf{y}}_k^- = \mathbf{h}\left(\mathbf{x}_k, \psi_k\right)$$

Compute the Kalman Gain:

$$G_k = \frac{P_k^- H_{k|k-1}^T}{H_{k|k-1} P_k^- H_{k|k-1}^T + R_{k-1}}$$

in which

$$H_{k|k-1} = \left.\frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_k^-}$$

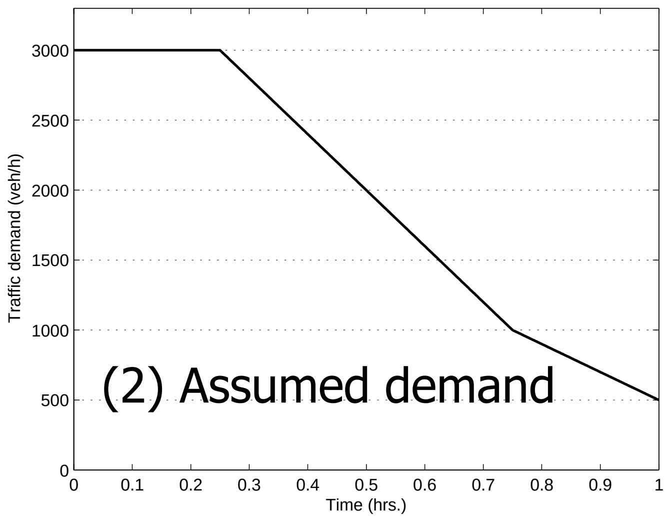
Assumption nr 3

Update mean and covariance:

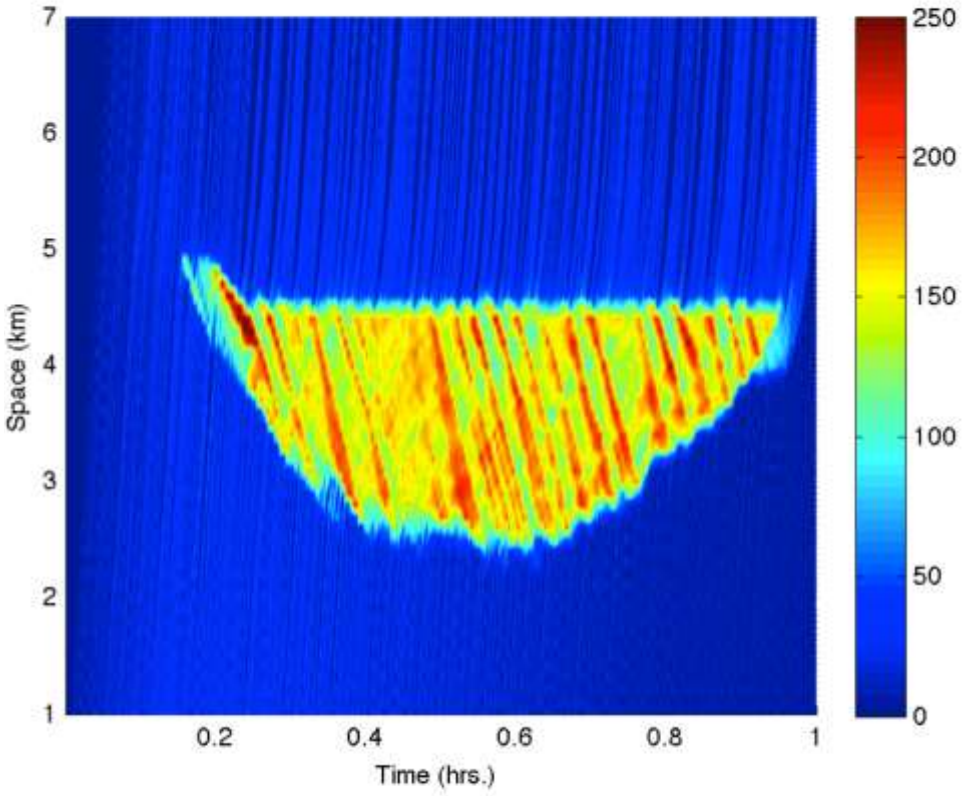
$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + G_k\left(\mathbf{y}_k - \hat{\mathbf{y}}_k^-\right)$$
$$P_k = \left(1 - G_k H_{k|k-1}\right) P_k^-$$

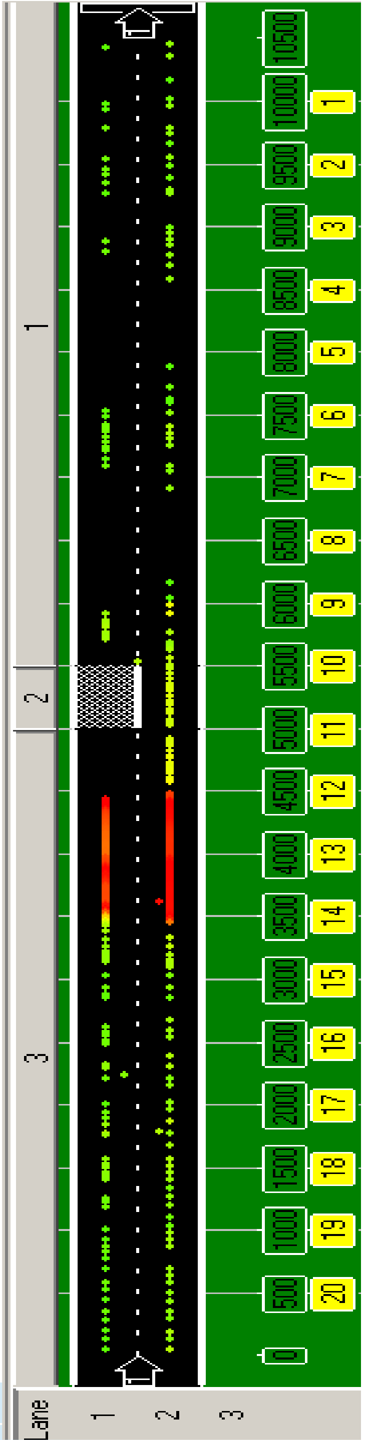
Traffic state estimation example

(1) freeway stretch with bottleneck and sensors available



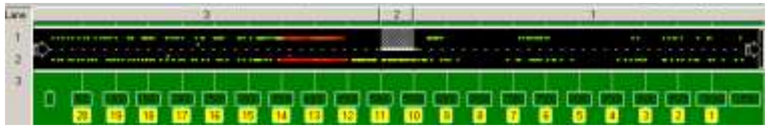
(3) Resulting traffic state





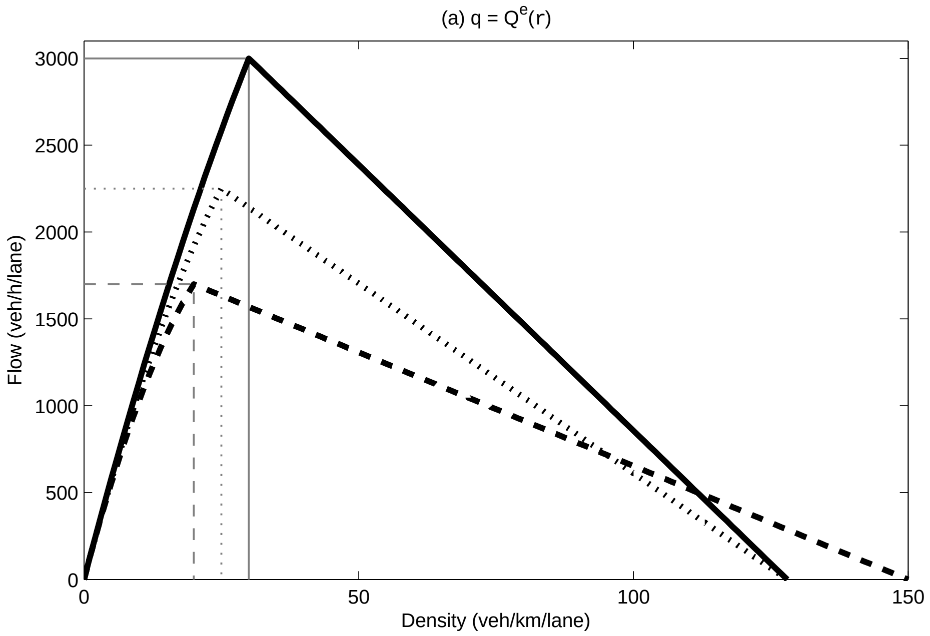
TU Delft

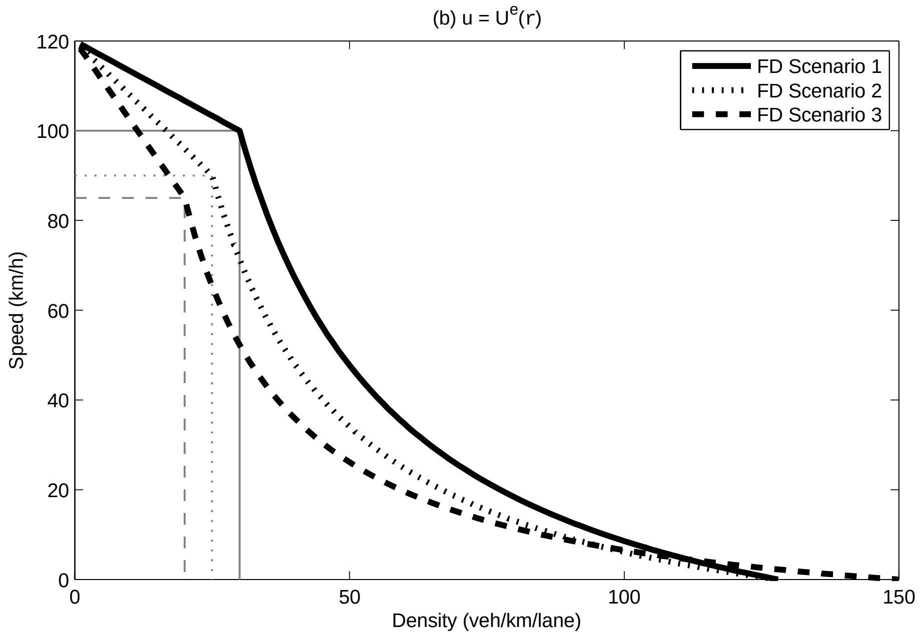
Traffic state estimation example



Scenarios: mess with the observation models (add bias)

1. Distort the observation models (I: ok; II: wrong; III very wrong)





2. 3 scenario's detector availability
   - ALL detectors; 1 out of 2 detectors; 1 out of 6 detectors

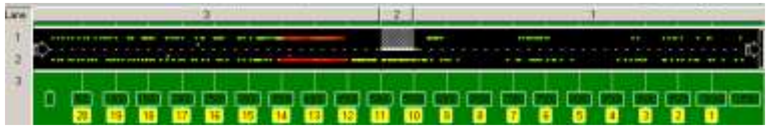
Traffic state estimation example

Results (first numbers)

- These results may surprise you a bit (or not at all)
  - More data does not mean better estimation (particularly not with biased observation models). Redundancy in some cases leads to larger errors
  - Neither RMSE nor MAPE are very informative measures

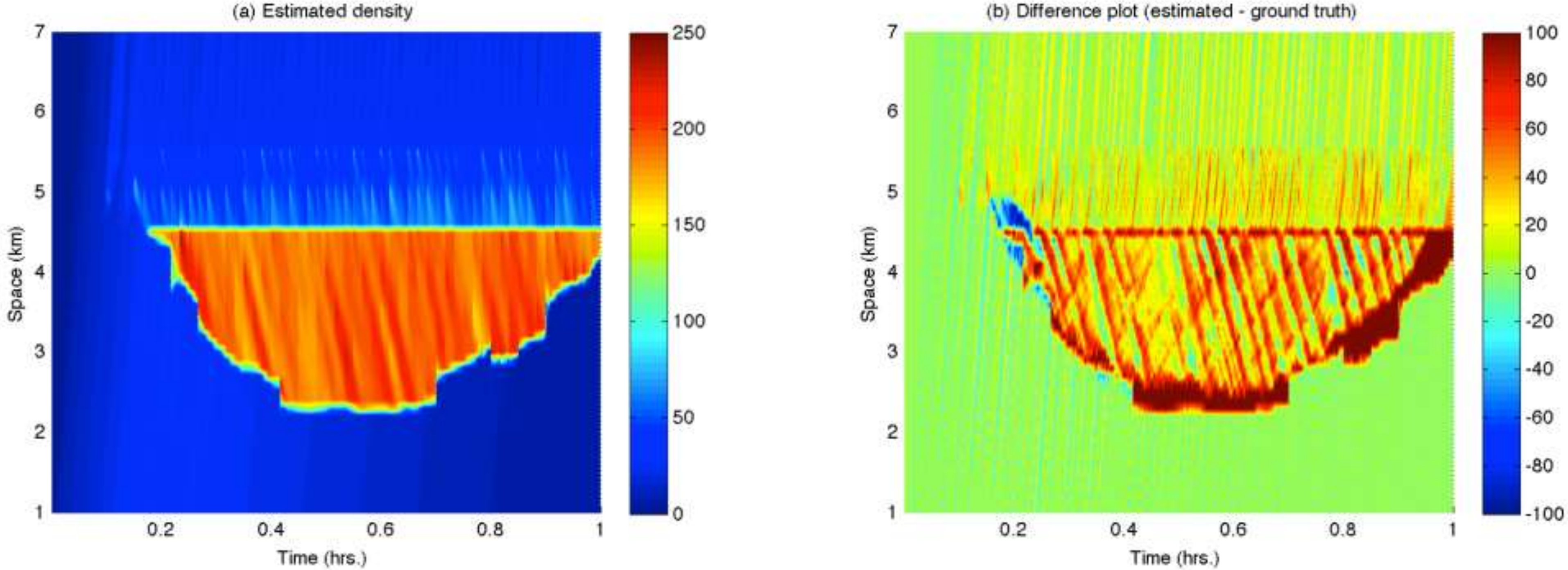
Detector spacing	FD-Scenario	RMSE (veh/km)	MPE (%)	SPE (%)	MAPE (%)
A (500 m)	I	31	18	108	46
	II	26	13	100	45
	III	29	0	103	45
B (1000 m)	I	27	13	80	39
	II	26	9	96	43
	III	27	-2	89	43
C (3000 m)	I	40	-2	53	38
	II	23	6	75	38
	III	31	9	135	51

Traffic state estimation example

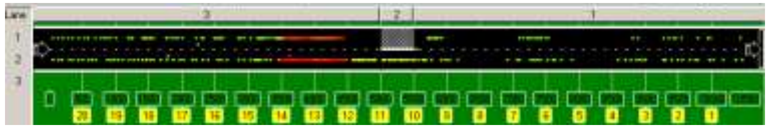


Results (qualitative)

- All detectors and reasonable observation model:
  - Congestion right place and time
  - Largest errors at the interface free <-> congestion
  - But also large errors within congestion (first order approximation cannot handle these moving jams)

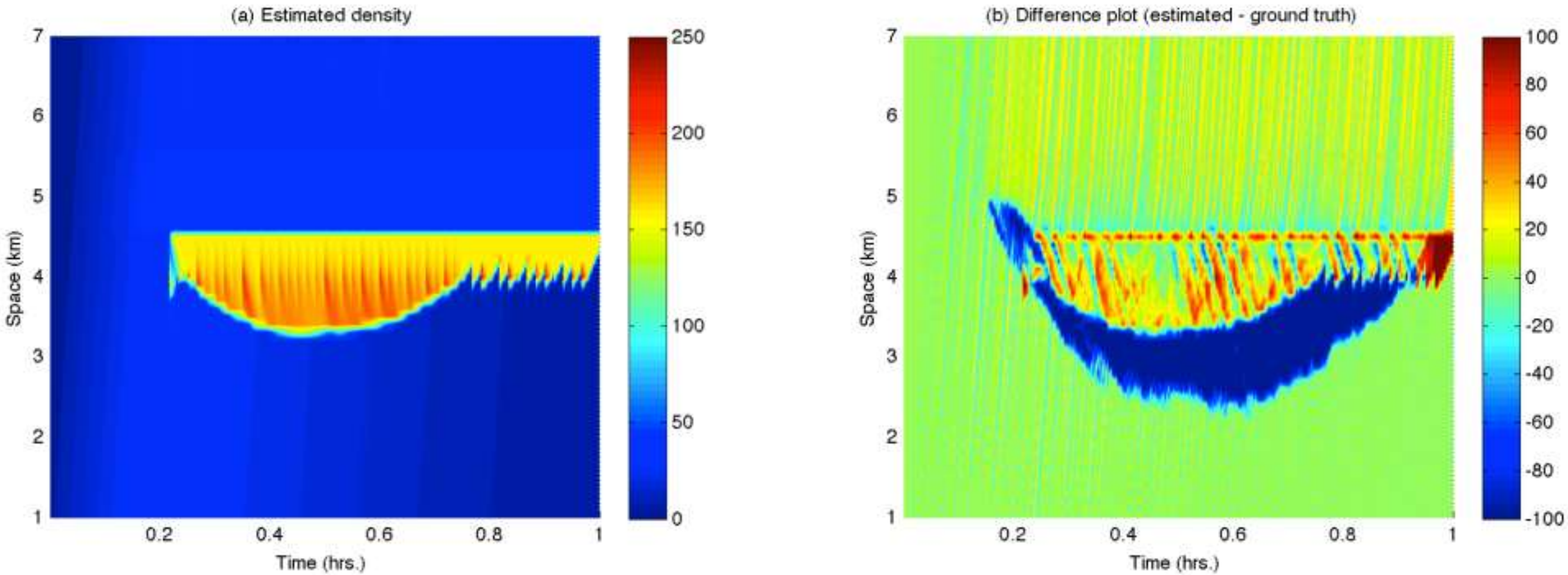


Traffic state estimation example

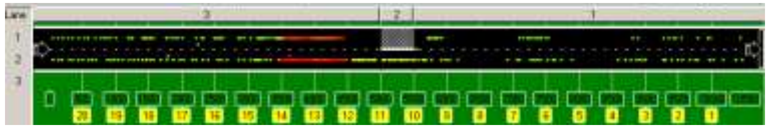


Results (qualitative)

- 1/6 detectors and reasonable observation model:
  - Underestimation congestion (no capacity drop)
  - Largest errors at the interface free <-> congestion
  - But also large errors within congestion (first order approximation)

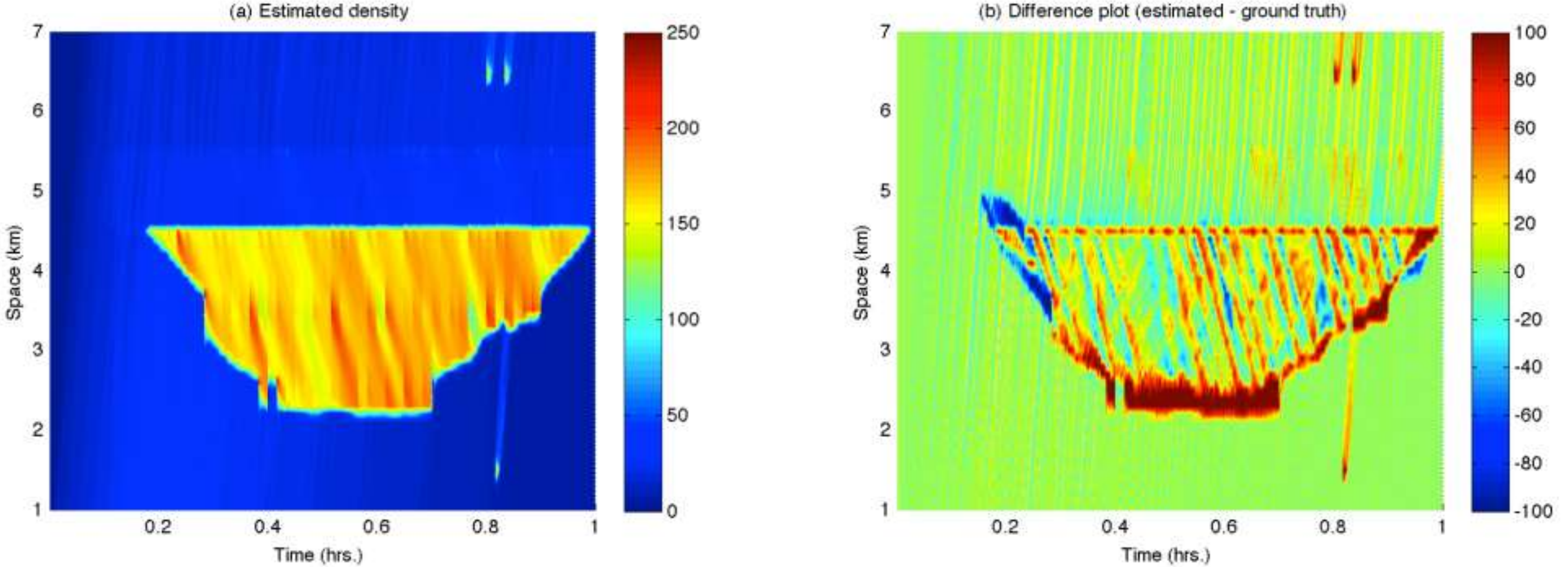


Traffic state estimation example

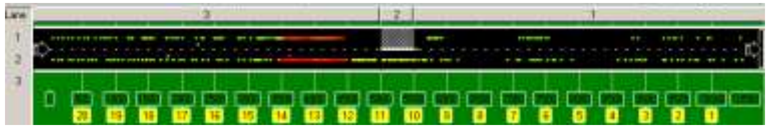


Results (qualitative)

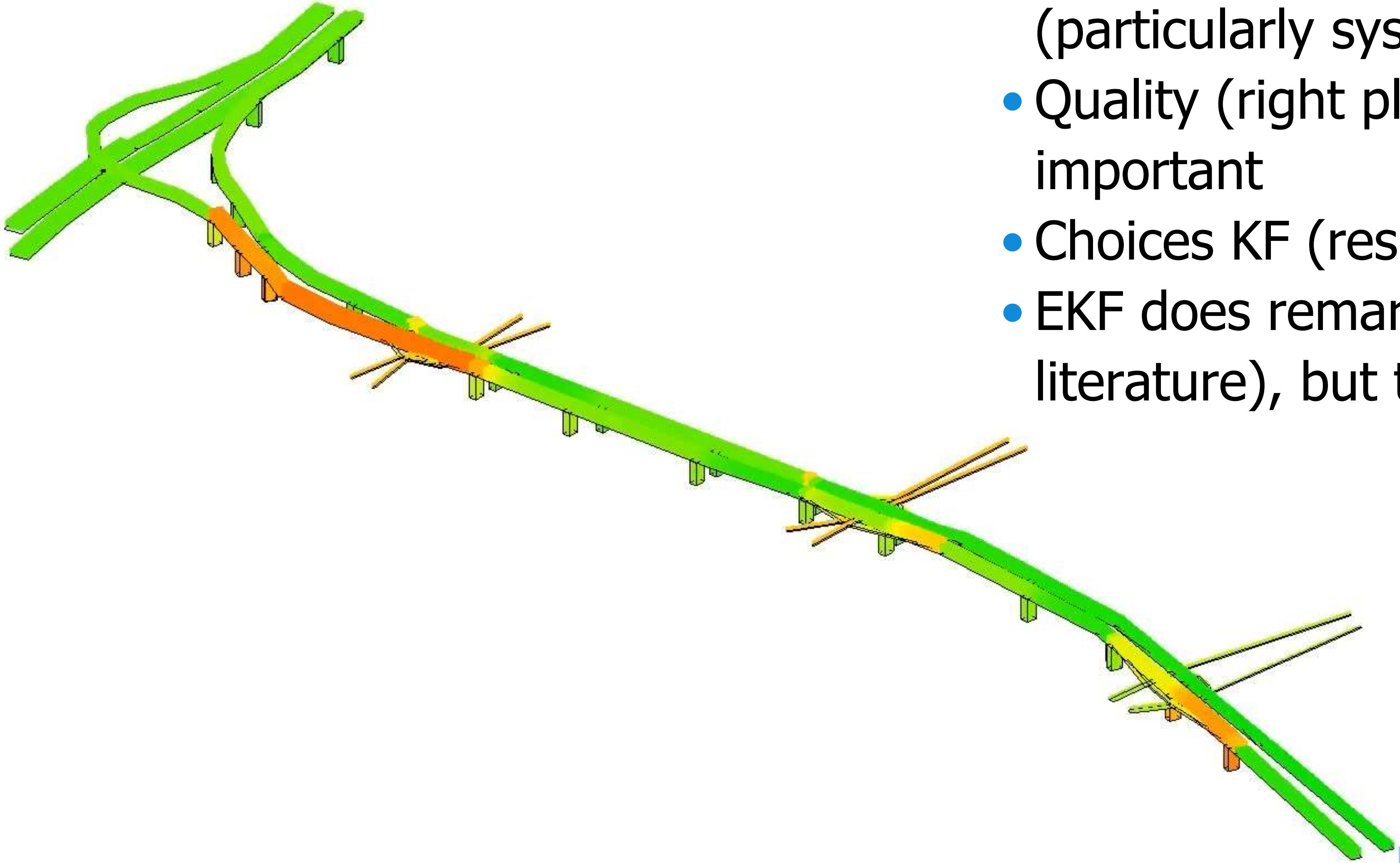
- 1/2 detectors and very biased observation model:
  - Congestion right place and time
  - Largest errors at the interface free <-> congestion
  - slightly smaller errors within congestion (biased observation model is actually of good use here)



Traffic state estimation example



Lessons learned

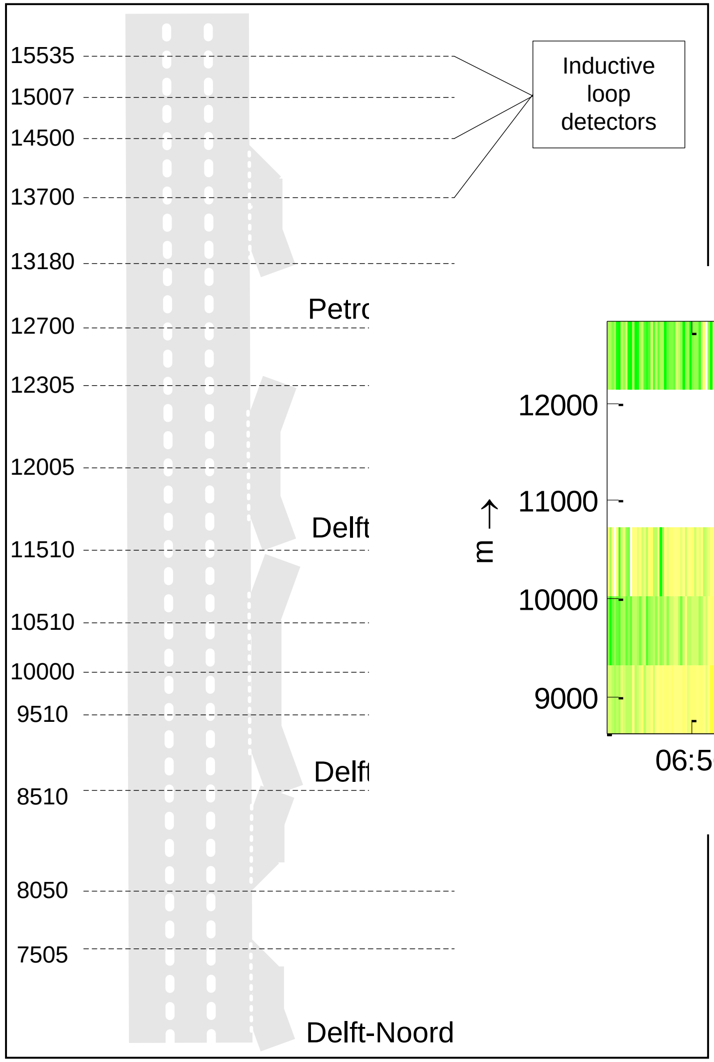


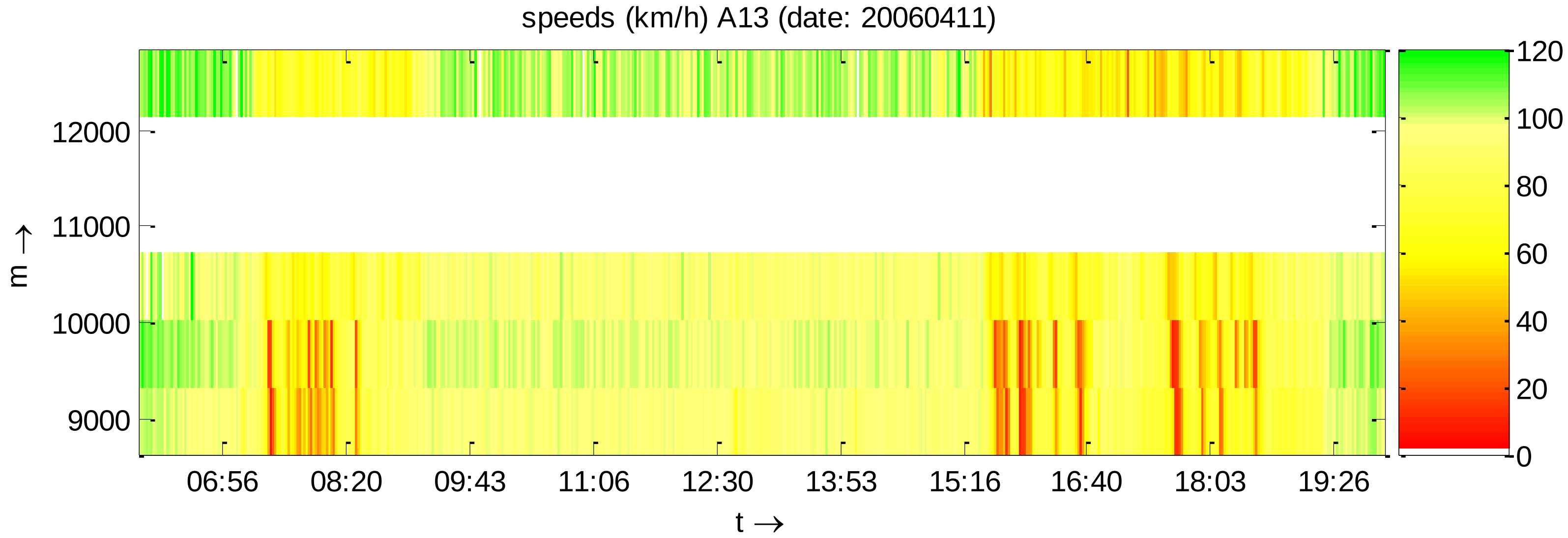
- Quality of the models critically important (particularly systematic errors)
- Quality (right place, no bias) of the data is important
- Choices KF (responsiveness) is important
- EKF does remarkable well (check the literature), but there are better options

Final Examples

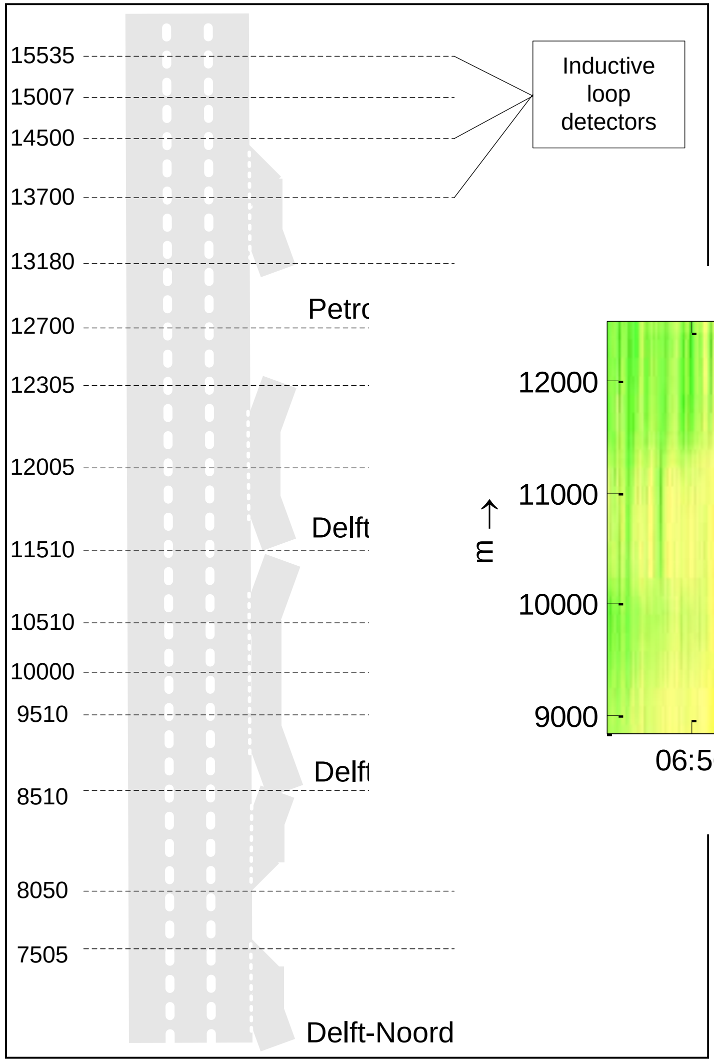
TRAFFIC STATE ESTIMATION

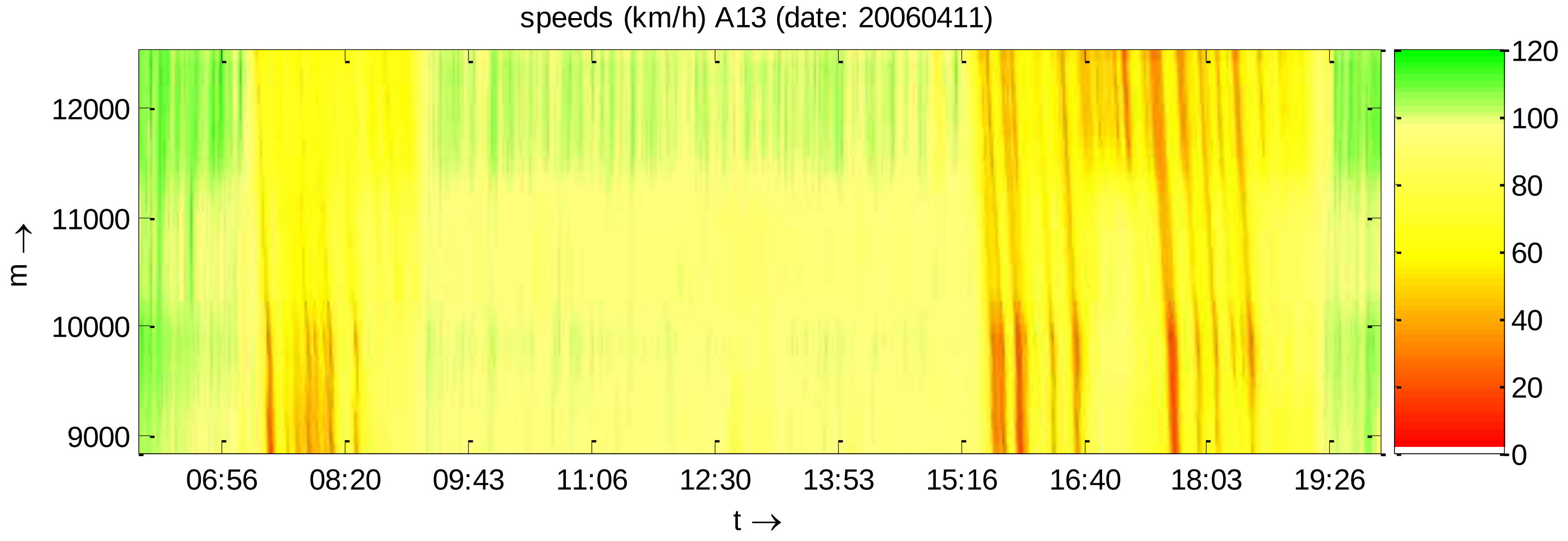
Final example 1: ASM vs EKF



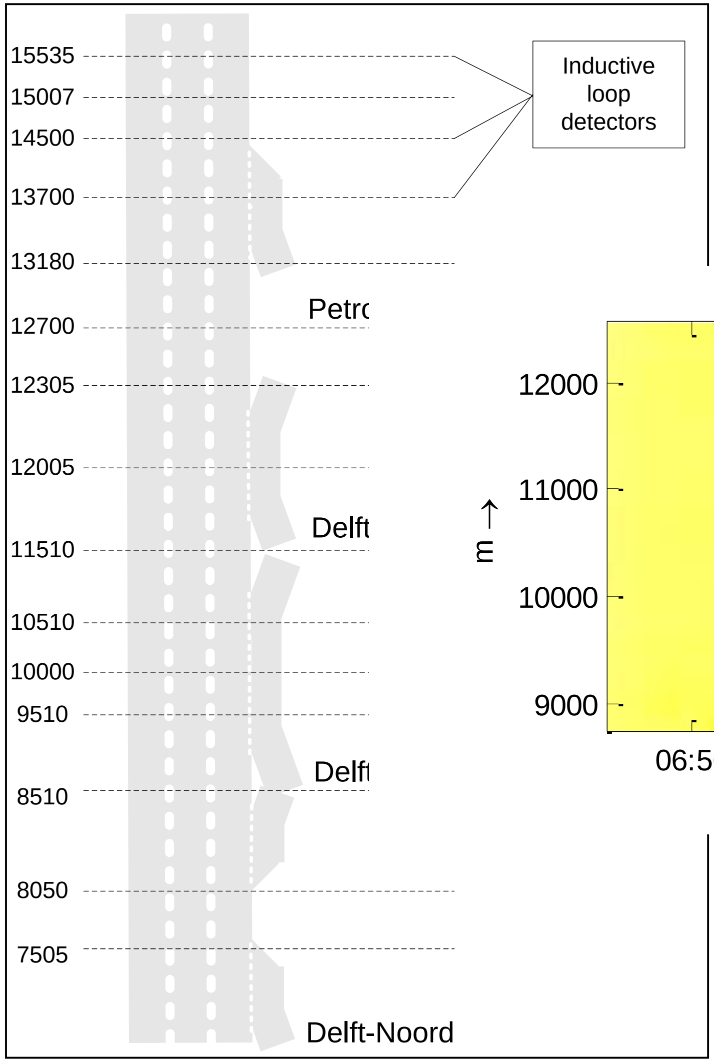


State estimate ASM



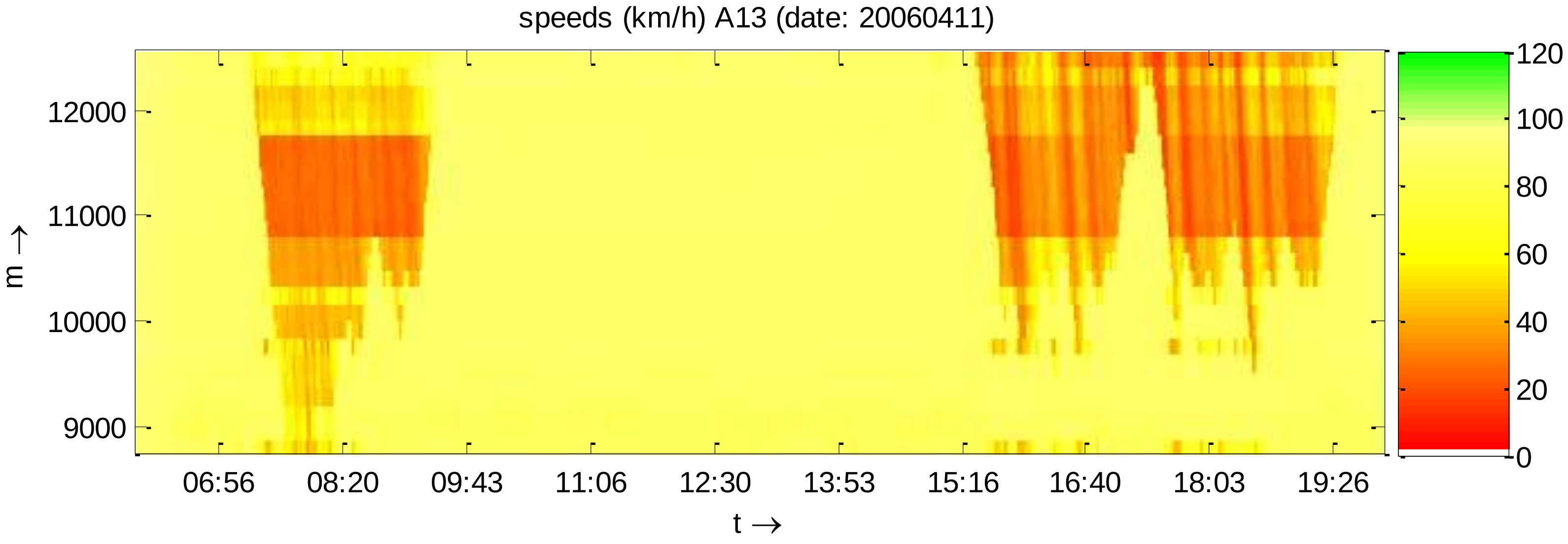


State estimate LWR model + Extended Kalman Filter



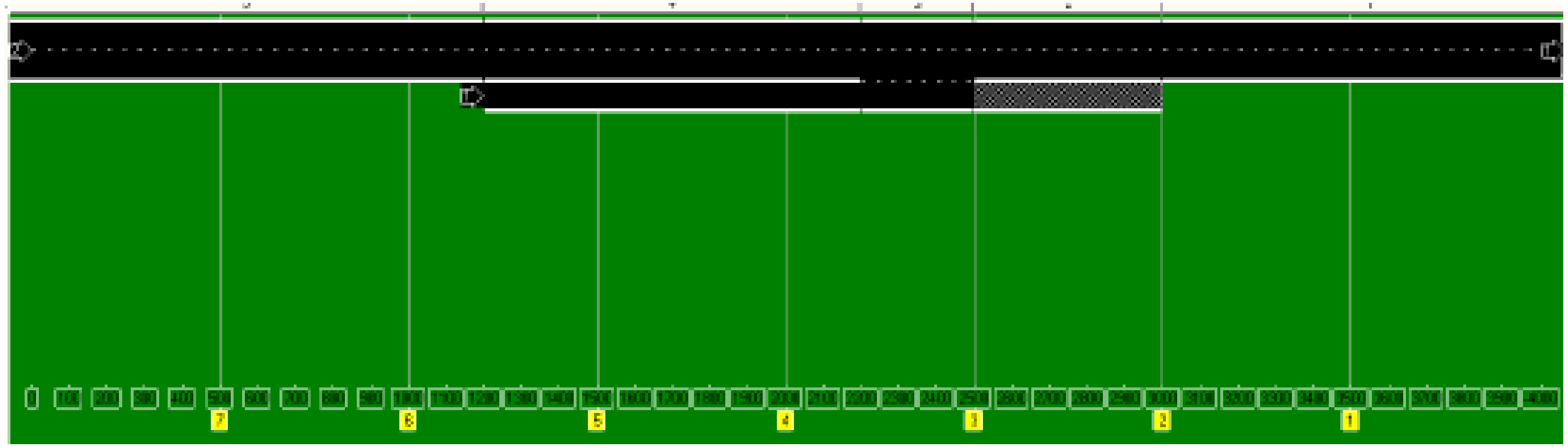
Think about this question: which of the two estimates is “right” and why?

HINT: I used quotation marks with a purpose



Final example 2: Data fusion

- Suppose a traffic manager wishes to install a monitoring system on this freeway



- Van Lint, J.W.C. and S.P. Hoogendoorn. The technical and economic benefits of data fusion for real-time monitoring of freeway traffic : preliminary results and implications of a study with simulated data. in Proceedings of the 11th World Conference on Transport Research. 2007. Berkeley: University of California.

Data fusion: an example

- First question: Information needs, for example:
  - Speeds on every place and time (for incident detection)

- But there can be many other information needs:
  - Head and tale of queue
  - Travel times, delay times
  - Traffic volumes
  - Total number of vehicles, total delay time
  - Etcetera …

Data fusion: an example

The (15) options:

A. 1, 3 or 7 loops
   costs: 20 k€ + 5k€/year/loop

- 0, 2, 6, 8, 10% probe vehicles (FCD)
  costs: 6 k€/1%

Extra difficulty: 20% of all measurements are missing or unreliable

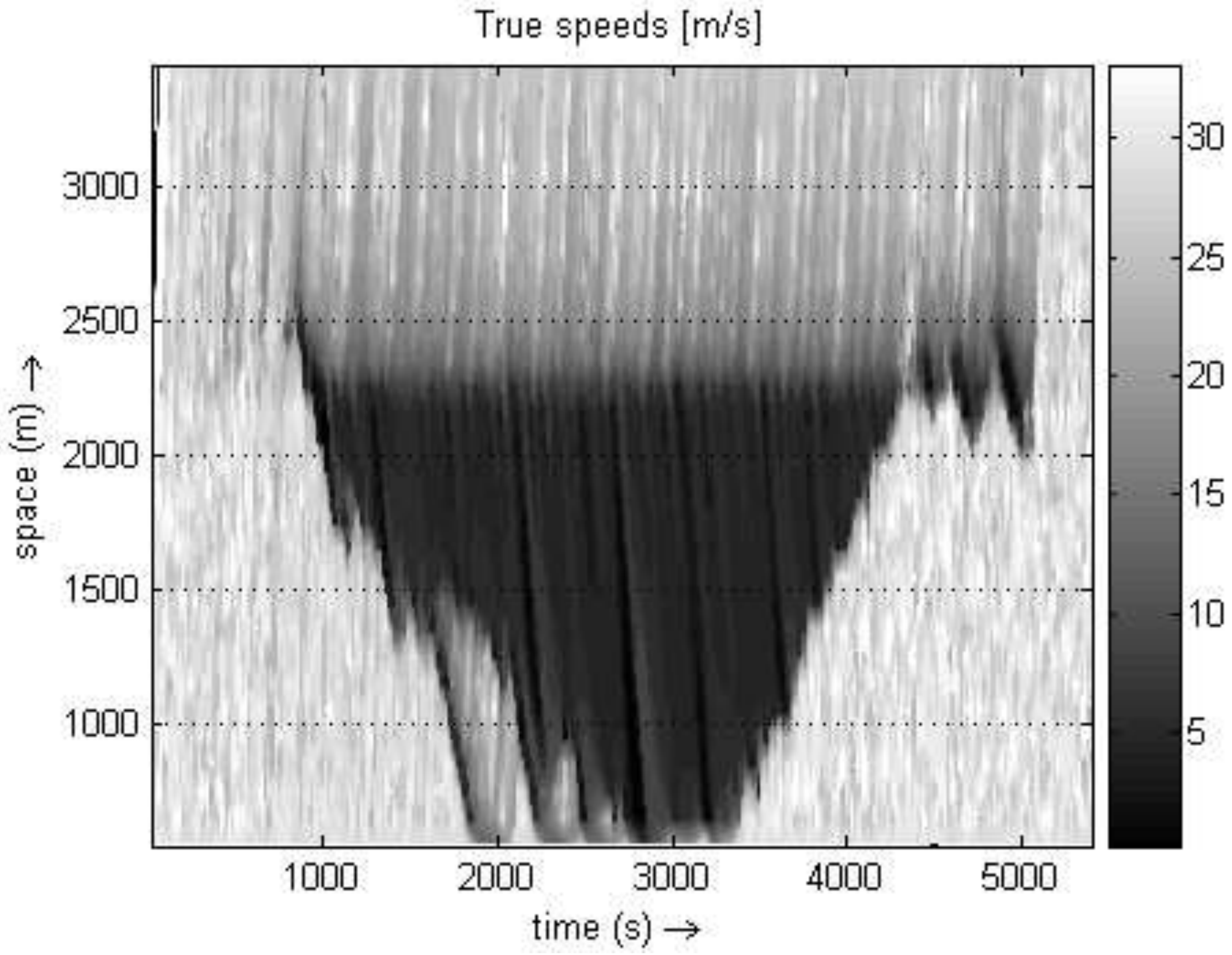
Data fusion: an example

Different approaches

A. Simple approach: averaging + some heuristics
B. LWR Model + Extended Kalman Filter

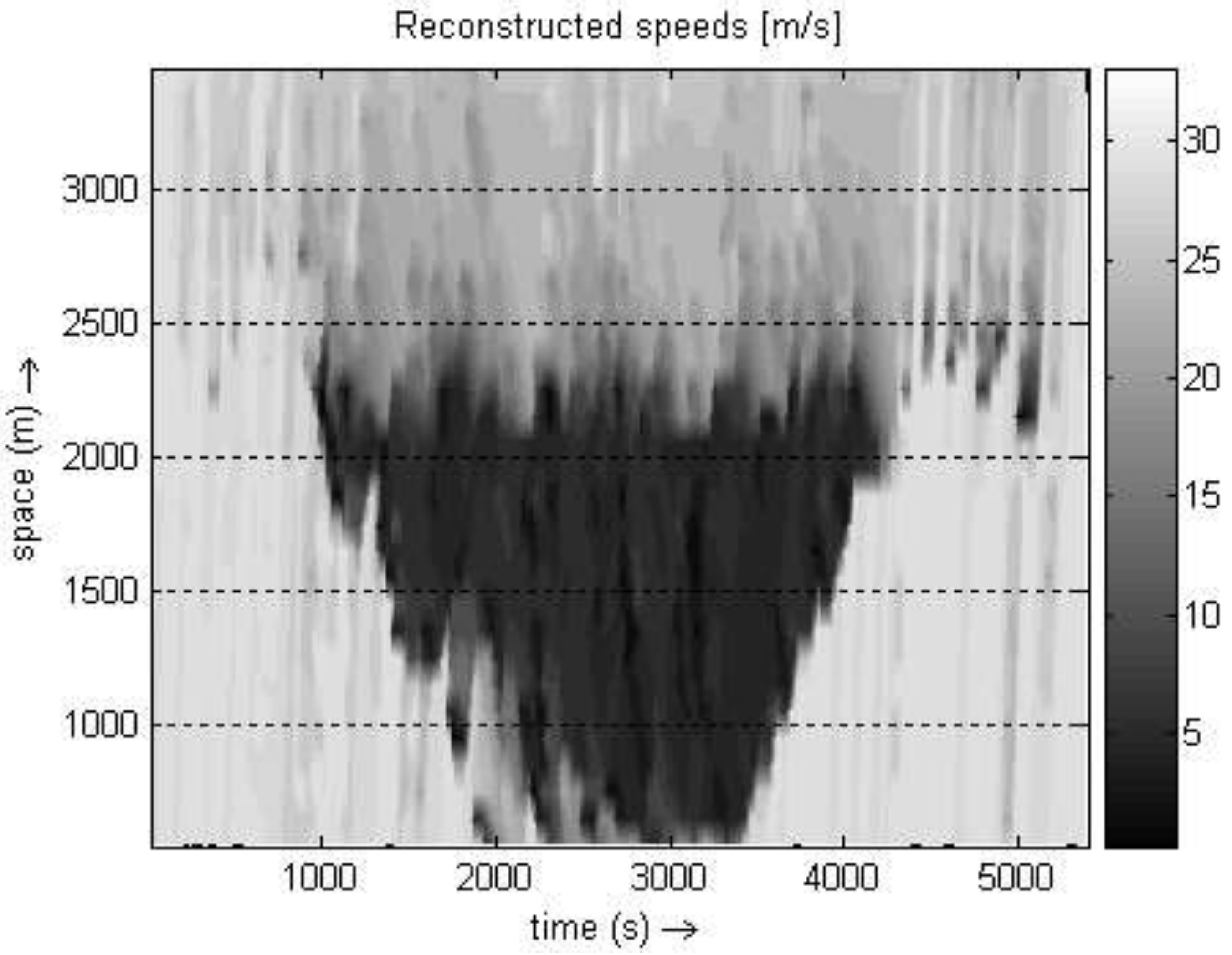
Data fusion: an example

This is the
Ground-truth



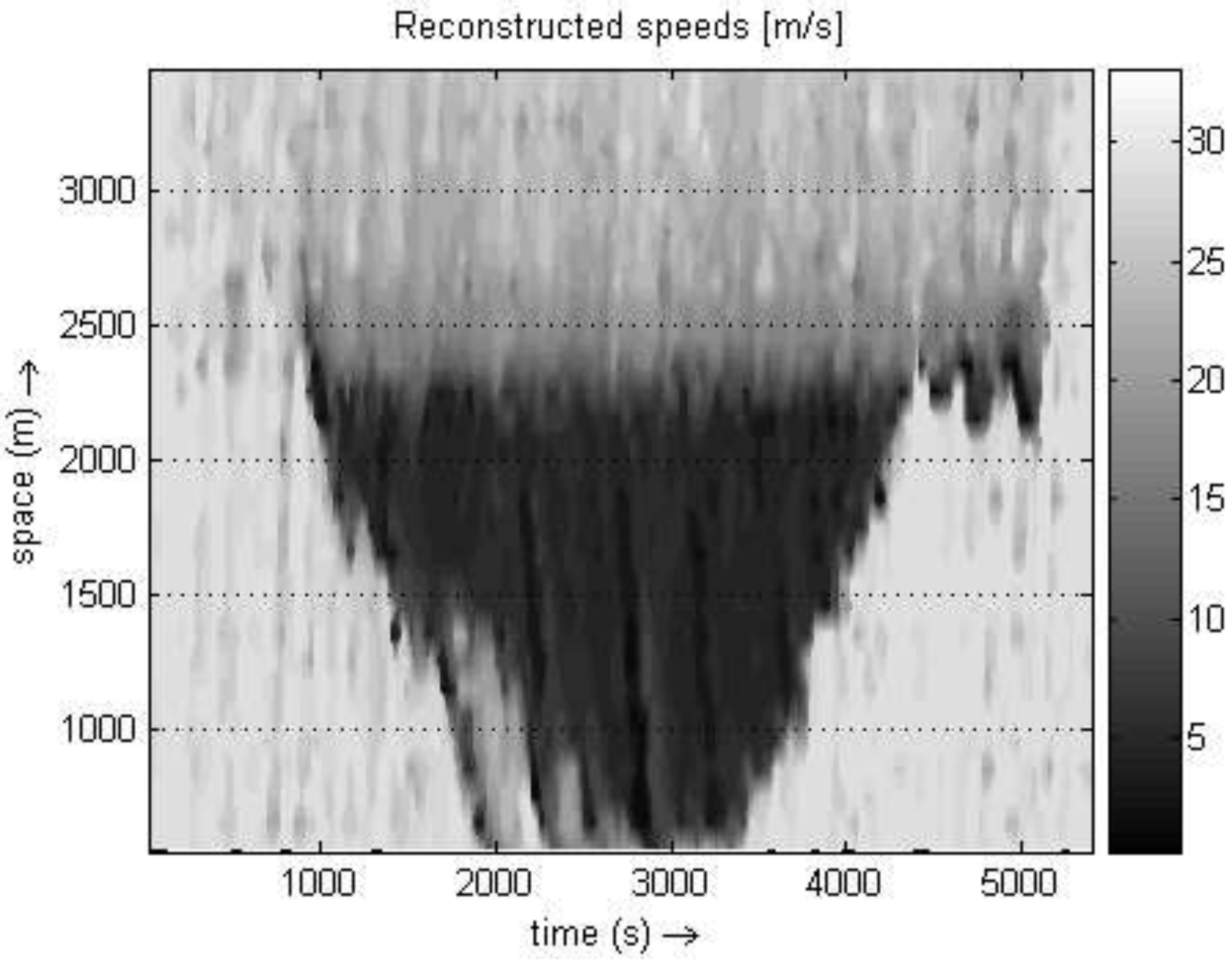
Data fusion: an example

2% FCD
+ 3 loops

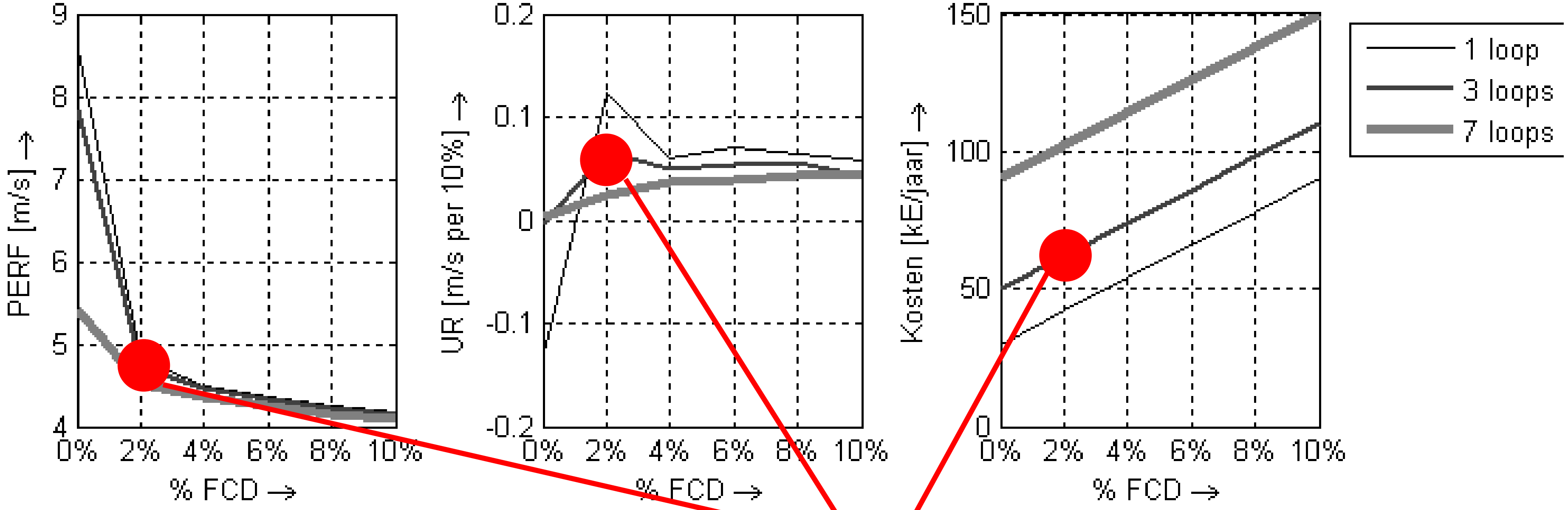


Data fusion: an example

10% FCD +
7 loops



Data fusion: Costs and Benefits



"winning" combination: 3 loops and 2% FCD

Performance

How accurate is each data fusion combination?

How we

combination handle missing data?

A more elaborate evaluation

(total 90 combinations scenario + state/parameters estimated + type of sensor data used)

Tested monitoring/data fusion scenarios:

1. Speeds + interpolation/smoothing
2. Speeds + LWR/EKF state estimation
3. Speeds + LWR/EKF state+params estimation
4. Speeds & Flows + LWR/EKF state estimation
5. Speeds & Flows + LWR/EKF state+params estimation

Each with all possible combinations of 1 -7 loops and 0-10% FCD

A more elaborate evaluation

(total 90 combinations scenario + state/parameters estimated + sensor data used)

- Top 10 most and least efficient systems (compared to lowest cost and best performance possible)

Rank	Scenario	% FCD	No. Loops	CB	RMSE	Total Cost
least efficient data fusion scenarios						
81	2	10	7	76%	4,10	€ 150.000
82	4	10	7	76%	4,10	€ 150.000
83	1	0	3	75%	12,40	€ 50.000
84	1	10	3	75%	5,66	€ 110.000
85	5	10	7	75%	4,20	€ 150.000
86	3	10	7	75%	4,26	€ 150.000
87	1	6	7	74%	5,12	€ 126.000
88	1	8	7	70%	5,29	€ 138.000
89	1	10	7	66%	5,44	€ 150.000
90	1	0	7	52%	12,08	€ 90.000

Interesting: current Dutch practice …

Conclusions

- Use of a model (assumptions, knowledge) to enrich data IS ALWAYS A GOOD IDEA

- Model-based data fusion offers benefits in terms of:
  - Accuracy
  - Reliability
  - Costs

- A few loops (at the right places) + FCD (+ cameras + … + …) = reliable and accurate monitoring system

BUT …

(alternative / complementary data fusion and state estimation approaches are needed)

- Not all traffic data can be fused with such a recursive state estimation approach (e.g. traffic flow model + EKF)

- **Bias** (in speed, flows, etc) : **better to deal with this first**
  - filtering will “smear out” bias over space and time
- **Underdetermined and or ill-posed relationships**
  - speed-density relationship under free flow conditions
  - travel times ➔ segment speeds
- **Spatiotemporal alignment problem**
  - observations **y**(t) (e.g. travel time) relate to different state variables **x**(t), depending on the traffic conditions:

$$TT_k = h\left(\mathbf{u}_k, \mathbf{u}_{k-1}, \ldots, \mathbf{u}_{k-TT_k}\right)$$