



Aalborg Universitet

AALBORG UNIVERSITY
DENMARK

Redskaber til monitorering af trafikken (REMOTE)

Lahrmann, Harry Spaabæk; Torp, Kristian; Runge Madsen, Jesper; Heide, Poul; Raguse, Jørgen; Juhl, Jens; Harms, Lisbeth; Hage, Christian

Published in:
Trafikdage på Aalborg Universitet [Online]

Publication date:
2004

Document Version
Også kaldet Forlagets PDF

[Link to publication from Aalborg University](#)

Citation for published version (APA):

Lahrmann, H. S., Torp, K., Runge Madsen, J., Heide, P., Raguse, J., Juhl, J., ... Hage, C. (2004). Redskaber til monitorering af trafikken (REMOTE). I Trafikdage på Aalborg Universitet [Online] Trafikforskningsgruppen, Aalborg Universitet.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- ? Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- ? You may not further distribute the material or use it for any profit-making activity or commercial gain
- ? You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

Redskaber til monitorering af trafikken

[REMOTE]

Forfattere:

Harry Lahrmann¹ lahrmann@plan.aau.dk, Kristian Torp² torp@cs.aau.dk, Jesper Runge Madsen⁷ jpm@cowi.dk, Poul Heide³ heide@m-tec.dk, Jørgen Raguse³ raguse@m-tec.dk, Jens Juhl⁴ jensjuhl@stofanet.dk, Lisbeth Harms⁵ Lisbeth.Harms@psy.ku.dk, Christian Hage⁶, ch@euman.com

¹Trafikforskningsgruppen, Institut for Samfundsudvikling og Planlægning, Aalborg Universitet,

²Institut for Datalogi, Aalborg Universitet

³M-tec

⁴Laboratoriet for Geoinformatik, Institut for Samfundsudvikling og Planlægning, Aalborg Universitet

⁵ Institut for psykologi, Københavns Universitet

⁶Euman

⁷COWI, tidligere Trafikforskningsgruppen, Institut for Samfundsudvikling og Planlægning, Aalborg Universitet,

Baggrund og formål

Detektorer i vejbanen på 100 km. motorvej i hovedstadsområdet giver præcise informationer om den øjeblikkelige trafikafvikling – som kan følges via Internettet og formidles til Københavns Radios trafikmeldinger. Med informationerne kan trafikanterne undgå eventuelle kødannelser og finde alternative ruter. Systemet i hovedstadsområdet kaldes ”Trim”. Et tilsvarende system findes i Aalborgområdet, med det formål at give trafikanterne informationer om kødannelser ved passage Limfjorden. Begge systemer findes på <http://www.trafikken.dk>. Etableringen og vedligeholdelse af detektorer i vejbanen er imidlertid en bekostelig affære, og derfor er det interessant at finde alternative løsninger.

Aalborg Universitet har i samarbejde med M-tec A/S i 2001 gennemført et projekt omkring Intelligent Farttilpasning (<http://www.infati.dk>). Projektet havde til formål at udvikle og teste et GPS-system, som kunne informere trafikanten om eventuelle hastighedsoverskridelser i forhold til de gældende hastighedsgrænser. Igennem projektforløbet blev det fastlagt, at GPS-systemets registreringer omkring bilernes hastighed ville være i stand til at tegne et billede af trafikafviklingen for et geografisk område – såfremt at et tilstrækkeligt antal biler i et område havde udstyret installeret og indsendte informationerne til en central server.

REMOTE projektet har haft til formål at bidrage med viden til at afklare, hvorvidt sådanne systemer baseret på Floating Car Data (FCD) kan erstatte eller supplere dyrt "vejudstyr" til monitorering af trafikken med lavere drifts- og vedligeholdelsesomkostninger til følge. Systemet består kortfattet af udstyr til montering i køretøjer og en server som behandler data og præsenterer trafikinformationerne til brugeren på enten Internettet eller som mobil service. Udstyret i bilerne består af en boks med en GPS modtager, GSM/GPRS-modem og et tastatur med få udvalgte funktionsknapper. Udstyret har både en automatisk registrering af køproblemer samt en manuel registrering baseret på vej reporterens brug af funktionsknapperne.

Projektet er finansieret med tilskud fra EU programmet VIKING som er et euroregional udviklingsprojekt for ITS på veje og omfatter landene Nordtyskland, Finland Sverige, Norge og Danmark

Oprindeligt var REMOTE projektet delt i et forprojekt og et hovedprojekt, og forprojektet skulle have karakter af et "prove of concept" med 10 biler, hvor de forskellige teknologier blev udviklet, afprøvet og dokumenteret. Det gælder både udstyret til bilerne og den bagvedliggende server som håndterer alle data samt skulle være motoren for flere services. I hovedprojektet skulle den udviklede teknologi videreudvikles og afprøves i et længerevarende testforsøg med 150-200 biler.

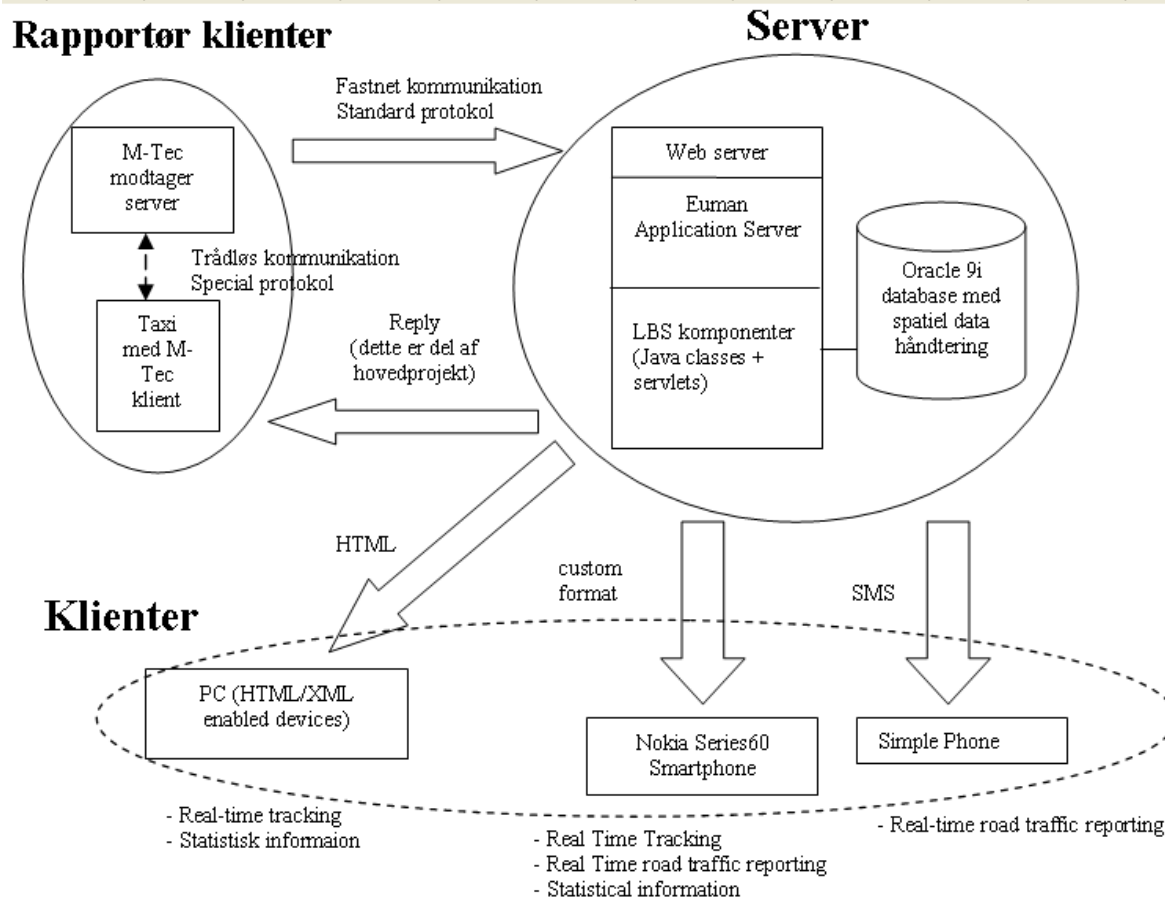
Projektet blev imidlertid stoppet efter forprojektet fordi der indenfor rammerne af VIKING ikke kunne findes midler til finansiering af hovedprojektet. I denne artikel præsenteret derfor alene forprojektet, som blev gennemført i efteråret 2003 og i foråret 2004.

Projektets organisering

Projektet har været et samarbejdsprojekt, hvor Institut for datalogi og Trafikforskningsgruppen begge Aalborg Universitet og firmaerne M-tec og Euman har været hovedkræfterne. Herudover har Laboratoriet for Geoinformatik, Aalborg Universitet og Institut for Psykologi, Københavns Universitet bidraget til projektet. Endelig har projektet været fulgt af en følgegruppe med repræsentanter fra Vejdirektoratet, Færdselsstyrelsen, Danmarks Transportforskning, Nordjyllands Amt og Aalborg Kommune.

Den overordnede arkitektur

Den overordnede arkitektur for IT-systemerne i REMOTE projektet er vist i Figur 1 nedenfor. De store cirkler indikerer hovedkomponenterne i arkitekturen, pil viser kommunikation og rektangler viser hardware. Vi vil først beskrive hovedkomponenterne i arkitekturen, dernæst beskrives kommunikation mellem disse.



Figur 1 Den overordnede arkitektur i REMOTE projektet

I øverste venstre hjørne ses *rapportør klienterne* dette er i projektet taxier, der kører rundt med den special udviklet hardware og som kan indrapportere trafikøer enten manuelt via et tastatur eller automatisk via GPS målinger. I øverste højre hjørne ses den *server*, hvor alt information fra rapportør klienterne gemmes, behandles og eventuelt videresendes. Denne server består af en web server, der er serverens grænseflade til alle de øvrige komponenter i arkitekturen. Dette er en standard web server, som f.eks. anvendes af mange organisationer til hjemmesider og lignende. Serveren består også af en række bagvedliggende databaser. Disse databaser er i Figur 1 simplificeret og vist som en enkelt database, der kan håndtere spatiel data. Den nederste store stiplede cirkel viser de *klienter* (eller *modtager klienter*), der anvender de services, der stilles til rådighed af projektet. Disse modtager klienter kan være forskellige. I figuren er angivet tre typiske modtager klienter. Den første er en standard PC, den anden er en såkaldt smartphone dvs. en avanceret telefon med mulighed for at vise billeder og den tredje er en simpel telefon, der i modsætning til smartphonen kun kan ringe og sende/modtage SMSer.

Når vi retter blikket på kommunikationen mellem hovedkomponenterne ses det, at rapportør klienterne ikke kommunikerer direkte med serveren, men går igennem en hardwarekomponent, der er benævnt *M-tec modtagerserver*. Dette mellemlid er indskudt, da taxierne så kan kommunikere med M-tec modtagerserveren via af en speciel kompakt protokol. Som det fremgår af figuren sker kommunikationen mellem taxierne og M-tec modtagerserveren trådløst. Da trådløs kommunikation er dyrere end fastnet kommunikation er der et stort

økonomisk argument for at bruge et meget kompakt format, da der betales per megabyte information, der sendes. Derfor er modtagerserveren indskudt som et mellemlid.

Modtagerserveren pakker de meget kompakte data den modtager fra taxierne ud. Data ud pakkes til et standard format, der er velegnet til kommunikation med de andre komponenter i arkitekturen. Oplysninger fra taxierne i et standard format sendes så videre til den anden server via en fast Internet forbindelse. Denne kommunikation sker via standard HTTP protokollen. Når data er modtaget på serveren bearbejdes og lagres de. Data kan så sendes videre til modtager klienterne. Dette kan f.eks. være til en standard pc, hvor HTML eller XML data sendes. Dette kan f.eks. være information om den nuværende trafiksituation eller statistik om trafikken. De samme data er sendt til en pc kan også sendes til en smartphone og vises på en sådan. Endelige kan information sendes ud til simple mobiltelefoner som en SMS.

De tre services

I forprojektet er der udviklet prototyper på tre typer af services til modtager klienterne. Disse services er de følgende.

1. En real-time tracking service, hvor klienter kan gå på Internettet via en pc eller en smartphone og se hvor rapportør klienterne er henne.
2. En real-time road traffic reporting service, hvor modtagere klienter kan få et overblik af den nuværende trafiksituation mht. kødannelse.
3. En web side med statistisk information modtagere klienter kan se information om hvornår der typisk opstår køer osv.

Tracking servicen vises på et kort og formatet, som klienterne modtager i, kan være HTML eller XML. Disse formater giver mulighed for at servicen kan vises på en HTML/XML enabled smartphone som Nokia's Series60 (modellerne) 3650 og 7650 eller Siemens SX1. Yderligere kan servicen ses på en PC laptop, en PDA eller en standard pc med en web browser.

Real-time road traffic reporting service giver et overblik af den nuværende trafiksituation mht. kødannelser og kan tilgås af de samme modtagerklienter som den foregående service.

Servicen med statistisk information er ligeledes tilgængelig via Internettet og kan give et overblik over, hvornår der typisk er kødannelser osv. Denne service kræver en større skærm end der er tilgængelig på en typisk smartphone og er derfor mest velegnet til at blive tilgået fra en standard pc med en Internet opkobling samt en webbrowser.

On Board Unit – OBU og internetserver

I forprojektet har M-tec udviklet den OBU der udgør rapportør klienterne til de enkelte taxier. Euman har stået for internetserveren.

For at minimere størrelsen af dataflow via GPRS fra rapportør klienterne har M-tec også udviklet *en modtagerserver*, der agerer bindeled mellem det kompakte UDP dataformat rapportør klienterne benytter og HTTP-GET requests til en *internetserver*.

Beskrivelse af klient hardware

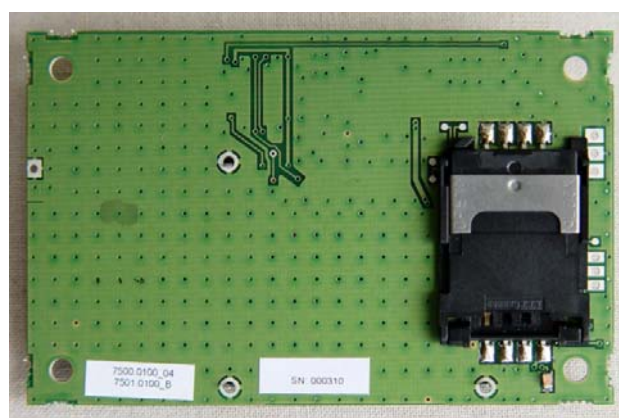
Klient hardwaren består af to hovedkomponenter:

- GSM/GPRS modul med tilhørende strømforsyning og antenne.
- GPS modtager med integreret antenne.

Normalt vil GSM/GPRS modulet have indbygget antenne, men da denne opgave kræver to store trykknapper med indbygget lys, er der af pladshensyn valgt at benytte en ekstern antenne. GPS modtageren er et færdigt produkt, der blot kræver en konfiguration for at være klar til opgaven.

Den nødvendige funktionalitet til at sikre dataflow mellem OBU og modtagerserver er implementeret i software indlejret i selve GSM/GPRS modulet. Der er altså ikke behov for en ekstra processor til at varetage samarbejdet mellem GSM og GPS. Dette muliggør trådløs opdatering af den indlejrede software. Det vil sige at en ny software kan sendes fra modtagerserver, uden behov for at indkalde klienten til service.

Det indvendige af klienten ses på de to nederste billeder. Når boksen åbnes er kun SIM kort læseren tilgængelig, som det ses nederst til højre.



Beskrivelse af kommunikationsomkostninger vha. special protokol

I projektet er driftomkostningerne ved de 10 rapportør klienter registreret. En måneds driftsomkostninger til GPRS trafik med et erhvervs abonnement hos Orange kostede således: Fast afgift 10 gange GSM abonnement på 56 kr. og 10 gange GPRS abonnement på 28 kr., i alt 840 kr. Trafikafgiften beløber sig til i alt 434,92 kr. Samlet giver det 1275 kr. (ekskl. Moms) for en måneds drift for drift at 10 rapportør klienter.

Samme trafik ville ved teleselskaberne CBB eller Telmore have beløbet sig til kun 195 kr. (ekskl. Moms). De to lavpris selskaber har ikke nogen fast abonnements afgift og desuden en lavere trafik afgift.

Hvis rapportør klienterne havde sendt direkte til internetsserveren ved HTTP-GET requests, ville den totale trafik have været ca. 700 Bytes pr. besked gange 525.000 beskeder (inkl. testbeskeder) hvilket giver 359.173 KBytes mod kun 24.629 KBytes med UDP protokol. Denne trafik ville hos Orange have kostet 6314 kr. + 840 kr. = 7154 kr. Eller hos CBB/Telmore : 2806 kr. Der er altså tale om en ganske betydelig besparelse ved dels at optimere data trafikken, og dels at vælge den rigtige tele operatør.

Erfaringer og delkonklusion

Et væsentligt mål med denne del af projektet har været at sikre datatransmission fra rapportør klienterne til lavest mulige driftsomkostning. Dette mål er til fulde nået, som det fremgår af forrige afsnit. Det er muligt at drive en klient med rapport hvert 5. sekund for ca. 240 kr. pr. år.

Dette beløb kan gøres endnu mindre, hvis det kan accepteres at data sendes i pakker med flere positioner pr. sending. Sendes f.eks. 6 positioner samlet hvert halve minut, sænkes den årlige drift til blot 100 kr. pr. klient.

Kødekeringsalgoritme

I projektet arbejdes der med to former for kødekering, dels en manuel, hvor rapportøren ved hjælp af et taktatur melder og afmelder kø, og dels en automatisk kødekering, hvor ideen er, at systemet automatisk melder kø hvis en rapportør klient bevæger sig langsommere end forventet på en given vejstrækning. Opgaven for kødekeringsalgoritmen er så at bearbejde disse kømeldinger og der ud fra at give en automatisk kømelding til trafikanterne.

Trængsel

Algoritmen vil i første omgang kunne melde om trængselsniveauer.

Trængsel er et udtryk for de gener, som trafikanterne påfører hinanden i form af nedsat bevægelsesfrihed, når de færdes i trafiksystemet.

Dvs. at trængsel ikke blot er en følge af lavere hastighed, men at den lavere hastighed skal være forårsaget af andre trafikanter.

Trængsel afhænger af de to faktorer **tæthed** og **hastighed**. I REMOTE projektet er det ikke muligt at måle tætheden på vejene. Derfor anvendes udelukkende hastigheden til at detektere trængselsniveauet.

Til REMOTE projektet bruges værdierne fra Tabel 1, hvor h er hastighedsgrænsen på strækningen,

Trængselsniveau	Tæthed	Rejsehastighed (v_{rejs})	LoS ¹	kommentar
Ubetydelig trængsel			A-B	Kan ikke rapporteres vha. REMOTE systemet
Begyndende trængsel				Kan ikke rapporteres vha. REMOTE systemet
Stor trængsel		$0,2 \times h < v_{rejs} < 0,6 \times h$	C-D	
Kritisk trængsel		$v_{rejs} \leq 0,2 \times h$	E-F	

Tabel 1 REMOTE trængselsniveauer

Trængselsmålinger og definition af strækninger

I dette afsnit beskrives, hvordan trængselsmålingerne i REMOTE projektet er foregået. Det beskrives herunder, hvordan vejnettet er udvalgt og opbygget med henblik på ud fra GPS registreringer at kunne vurdere, om der på en given strækning er kø.

Nedenstående kort viser det vejnet, der er udpeget til pilotforsøget. Dette vejnet dækker de veje, hvor der forventes at kunne opstå trængselsproblemer.

¹ Jævnfør definition af trængsel if “Highway Capacity Manual 2000”.



Figur 2 Forsøgsvejnettet i Aalborg anvendt i REMOTE

Trængsels målinger

Til brug ved den automatiske beregning af trængsel REMOTE defineres to begreber – se også Figur 3:

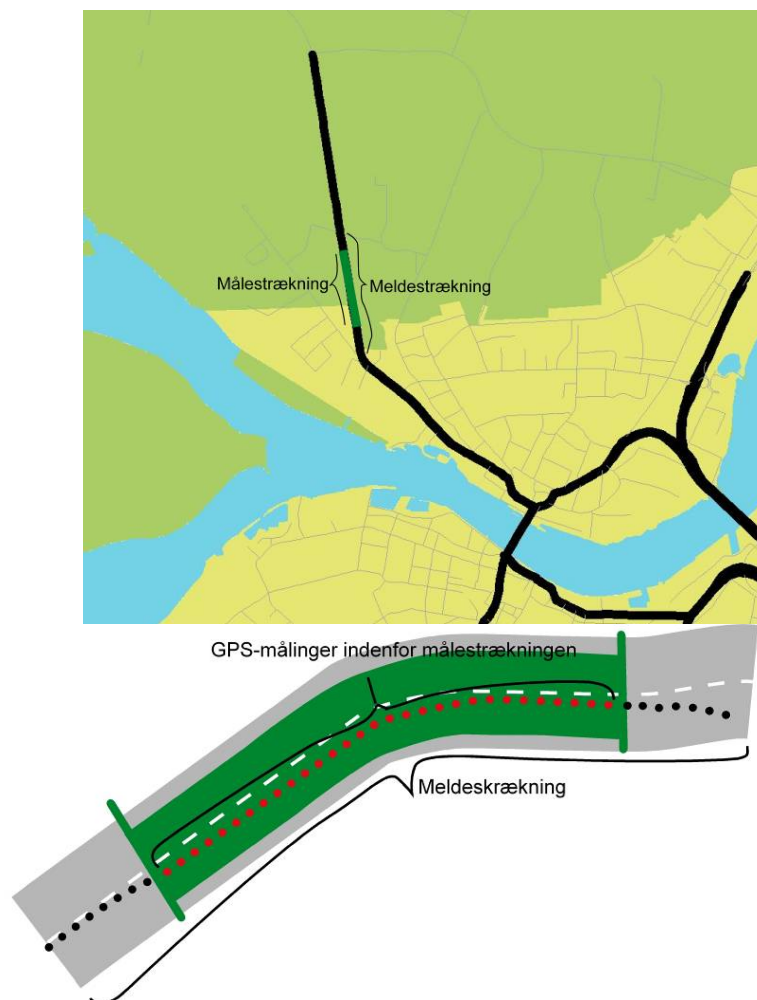
- Målestrækning

En målestrækning er en strækning på vejnettet, hvor det antages, at trafikanterne kører med den skilte hastighed medmindre hastigheden er påvirket af kødannelse.

Målestrækningerne placeres således på steder, hvor der er færrest mulige generende elementer som fx opstuvning ved kryds, parkering eller krydsende veje

- Meldestrækning

En meldestrækning er en strækning, der kan meldes ud til trafikanterne, typisk en strækning mellem to større kryds på den aktuelle vej. En meldestrækning kan omfatte mere en målestrækning.



Figur 3 Principillustration af melde- og målestrækning

Trængselsmeldingerne gælder for den vejstrækning/meldestrækning, den enkelte målestrækning er tilknyttet, det er dog muligt, at der senere skal indføres et ruteprincip således, at der meldes for en rute fx hele strækningen mellem Limfjordsbroen og Aalborg Lufthavnen. Hvordan en trængselsmelding detekteres bliver gennemgået i det næste afsnit.

Udvalgte meldestrækninger og målestrækninger

På følgende figur ses de udvalgte meldestrækninger, disse er nummererede og farvet i forskellige farver med tilhørende målestrækninger, der her er markeret som den "fede" del af hver meldestrækning. Der er i Aalborg og Nørresundby defineret i alt 42 meldestrækninger med tilhørende målestrækninger. Målestrækningerne er forsøgt valgt, så de så godt som muligt lever op til de krav, der er opstillet ovenfor mht. frit hastighedsvalg.



Figur 4 Udvalgte meldestrækninger. Strækningerne er nummererede og farvet i forskellige farver med tilhørende målstrækninger, der her er markeret som den "fede" del af hver meldestrækning

Trængseldetekteringsalgoritme

I det følgende, vil et forslag til implementering af en autodetekteringsalgoritme til trængsel blive beskrevet.

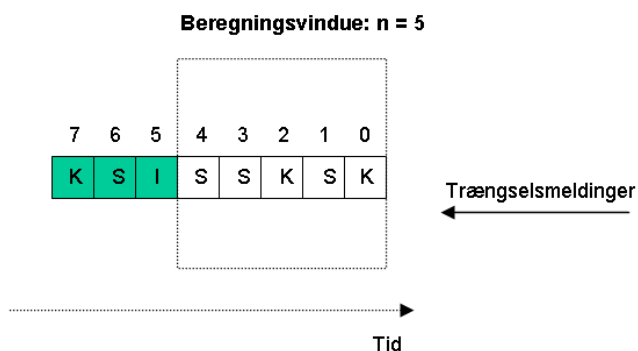
Algoritmen kan opdeles i 3 hoveddele.

1. Hvert 5 sekund eller ved manuel kømelding modtages en position fra rapportør klienterne som matches til vejnettet
2. Det tjekkes om positionen ligger indenfor en målstrækning, og i disse tilfælde sendes meldingen videre til trængseldetekteringsalgoritmen.
3. Trængseldetekteringsalgoritmen returnerer et trængselsniveau.

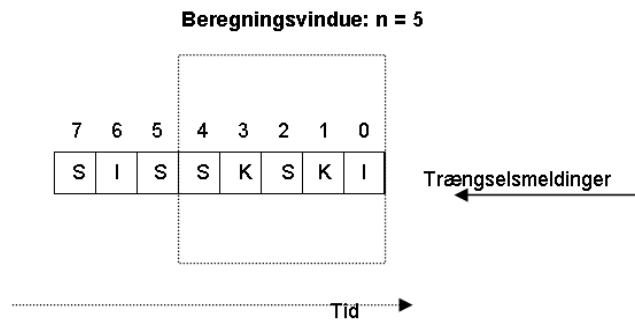
For at sikre at trængselsmeldingerne stemmer overens med virkeligheden bruges data fra tidligere meldinger til at vurdere om trængselssituationen er længerevarende eller kun midlertidig. Der kan forekomme situationer, hvor rapportør klienten kører under en given hastighedsgrænse ved første rapportering, men kører over den givne grænse ved næste rapportering. Det kunne for eksempel være, hvis rapportør klienten skal samle kunder eller venner og bekendte op eller hvis rapportør klienten skal stoppe op for at spørge om vej. Disse situationer kan ikke betragtes som trængsel og skal forsøges sorteret fra.

Trængseldetekteringsalgoritmen skal altså ikke melde trængsel hver gang en bilist stopper op. Algoritmen ville så blive for ”aggressiv”. For at bløde op på dette indlægges en slags ”træghed” i algoritmen. Derfor vil autodetekteringsalgoritmen, foruden at bedømme trængslen ud fra den nuværende hastighed, også bedømme trængslen ud fra de tidligere meldinger.

Udover dette tager algoritmen højde for at meldinger kan blive for gamle, således at for gamle trængselsmeldinger ikke kan påvirke det aktuelle trængselsniveau på en snitmåling. Dette gøres ved en tidparameter, som kan sættes til eksempelvis 5 minutter. Er meldingen på en meldestrækning over 5 minutter gammel tæller den ikke med i den samlede beregning af trængselsniveauet på den givne snitflade. I **Fejl! Henvisningskilde ikke fundet.** ses en illustration af trængselsestimeringsstrukturen, der anvendes til at estimere trængsel på baggrund af de tidligere meldinger. Strukturen består af to elementer. En kø, hvor trængselsmeldingerne bliver lagt ind, samt et beregningsvindue, som kan indeholde n sidste meldinger. I **Fejl! Henvisningskilde ikke fundet.** er $n = 5$, dvs. udregningerne er baseret på de 5 sidste meldinger. Hver gang en ny trængselsmelding beregnes lægges denne melding ind på plads 0 i køen og de øvrige meldinger flyttes en plads i køen (illustreret ved at flytte meldingerne en plads til venstre i **Fejl! Henvisningskilde ikke fundet.**). Når meldingerne overskrider plads 4 i dette eksempel bliver de ikke taget højde for i udregningerne. Trængselsniveauet udregnes som summen af værdierne for hver melding i beregningsvinduet og afgør hvilken trængselsmelding, som algoritmen returnerer. Meldingerne kan have forskellig værdi og er her alene benævnt ved et bogstav. I Figur 6 ses trængselsestimeringsstrukturen efter at en ny melding er indsat.



Figur 5 Trængselsestimeringsstruktur bestående af en kø samt beregningsvindue



Figur 6 Trængselsestimeringsstruktur efter en ny melding er blevet indsat (markeret i plads 0)

I det følgende beskrives beregningsvinduet og køstrukturen i detaljer. For at kunne udregne et trængselsniveau baseret på de seneste meldinger er det nødvendigt at definere nogle trængselsværdier for forskellige typer af trængselstyper.

Tabel 2 indeholder informationer over definerede trængselsværdier. *Tegn* er en ID for trængselstypen. *Værdi* er den værdi, trængselstypen bidrager i beregningen og *Trængselstype* indeholder beskrivelsen af trængselstypen. For at opnå størst mulige fleksibilitet, kan man definere værdierne for de forskellige trængselstyper. F.eks. ønsker man, at en trængsel hurtigt bliver afmeldt, kan man sætte værdien for *ingen trængsel* til en meget lav værdi. Dermed bidrager værdien for *ingen trængsel* melding meget til den samlede udregning af trængselsniveauet. Ønsker man eksempelvis at en *bruger tilmelding* trængselsmelding skal vægte højt i udregningen af trængselsniveauet, kan man sætte værdien for *bruger tilmelding* trængsel meget højt. Med denne fleksibilitet kan værdierne ændres og optimeres til at modellere den rigtige verden, hvilket kan tilnærmes ved hjælp af eksperimenter.

Tegn	Værdi	Trængselstype
A	-10	Bruger Afmelding
I	-2	Ingen trængsel
U	0	Ubetydelig trængsel
B	1	Begyndende trængsel
S	2	Stor trængsel
K	4	Kritisk trængsel
T	10	Bruger Tilmelding

Tabel 2 Trængselstype

Pladserne i beregningsvinduet dækker over flere meldinger. For at kunne differentiere mellem nye og ældre meldinger er det muligt at tildele vægte til pladserne i beregningsvinduet således at man eksempelvis kan vægte nye meldinger højere end ældre meldinger.

Tabel 3 beskriver attributterne i et beregningsvindue. *Indeks* er en ID for pladsen i beregningsvinduet. *Vægt* er den værdi, køpladsen bliver vægtet med. Vægtene afgør, hvor meget de enkelte køpladser skal vægtes i beregningsvinduet. Ønsker man eksempelvis, at de 3 seneste meldinger skal vægte mere i udregningen, kan man sætte værdien højere. I eksemplet i Tabel 3, vægter den nyeste melding (plads 0) med 50 %, den næst nyeste med 20 % og de resterende med hver 10 %. For at opnå størst fleksibilitet, kan man definere vægtene valgfrit. I

dette eksempel giver summen af vægtfordelingen 100 %. Vægtene optimeret til at modellere den rigtige verden, kan findes ved hjælp af eksperimenter.

Indeks	Vægt
0	0.5
1	0.2
2	0.1
3	0.1
4	0.1

Tabel 3 Beregningsvindue

I det efterfølgende gives to eksempler på udregningen af trængselsniveauet baseret på værdierne og vægtene fra henholdsvis Tabel 2 og Tabel 3. Tabel 4 beskriver køstrukturen som nævnt i **Fejl! Henvisningskilde ikke fundet.** Tabellen består af en *Indeks* attribut, som beskriver hvilken rækkefølge meldingerne er blevet rapporteret. Plads 0 er den nyeste rapportering og plads 7 er den ældste. Attributten *Melding* beskriver hvilken melding, som er blevet udregnet. Meldingerne svarer til attributten *Tegn* fra Trængselstype tabellen i Tabel 2. Attributten *BeregnetVærdi* beskriver den værdi som meldingen bidrager med i den samlede udregning af trængselsniveauet. Eftersom beregningsvinduet fra Tabel 3 kun er defineret for pladserne 0-4 bidrager kun meldingerne fra indeks 0-4 fra Kø tabellen til den samlede trængselsværdi. Derimod bidrager pladserne 5-7, som er udenfor beregningsvinduet ikke med nogen værdi.

Den beregnede Værdi for hver melding i Kø tabellen udregnes ved at tage vægten fra beregningsvinduet på den aktuelle plads ganget med værdien for den aktuelle trængsel, som findes i Tabel 2.

Indeks	Melding	BeregnetVærdi
0	K	Beregningsvindue.Vægt[0] x Trængselstype.Værdi[K]
1	S	Beregningsvindue.Vægt[1] x Trængselstype.Værdi[S]
2	S	Beregningsvindue.Vægt[2] x Trængselstype.Værdi[S]
3	I	Beregningsvindue.Vægt[3] x Trængselstype.Værdi[I]
4	I	Beregningsvindue.Vægt[4] x Trængselstype.Værdi[I]
5	K	0
6	S	0
7	I	0

Tabel 4 Køtabel for eksempel 1

For at udregne det samlede trængselsniveau summeres den beregnede værdi (*BeregnetVærdi*) fra Kø tabellen på pladserne defineret i Beregningsvindue tabellen. Dvs. i dette eksempel de beregnede værdier på pladserne 0-4.

$$\begin{aligned}
\text{Trængselsniveau} &= \sum_{i=0}^4 \text{Beregningsvindue.Vægt}[i] \times \text{Trængselstype.Værdi}[\text{Kø.Melding}[i]] \\
&= 0.5 \times 4 + 0.2 \times 2 + 0.1 \times 2 + 0.1 \times -2 + 0.1 \times -2 \\
&= 2 + 0.4 + 0.2 - 0.2 - 0.2 \\
&= 2.2
\end{aligned}$$

Værdien for *Trængselsniveau* afrundes til nærmeste trængselsværdi. Dvs. værdien 2.2 afrundes til 2, hvilket giver en *Stor Trængsel*.

Bemærk det er muligt at sætte en prædefineret tid, som beskriver, hvornår en melding bliver forældet. Dvs. at såfremt en melding er ældre en den definerede tid vil meldingen blive ændret til en "Ingen trængsel" melding i beregningsvinduet, og den vil derfor i dette tilfælde vægte med -2 i beregningen af trængselsniveauet. Defineres denne tid til 3 minutter, vil meldinger over 3 minutter gamle tælle som "Ingen trængsel" og derfor vil trængselsniveauet ende som "Ingen trængsel" efter 3 minutter, såfremt at nye meldinger ikke kommer ind.

I det følgende gives endnu et eksempel, eksempel 2, på udregning af trængselsniveau, Eksemplet beskriver hvordan en brugers tryk på afmeldeknappen influerer på det returnerede trængselsniveau. Tabel 5 viser trængselstyperne for eksemplet. Tabel 6 viser, at beregningsvinduet tager højde for de 5 sidste rapporteringer på snitmålingen, repræsenteret ved indeksene fra 0 til 4 og hvor højt de vægtes. 0 er den sidst ankomne melding beregnet fra SpeedCheck() metoden. Tabel 7 viser de 8 sidst ankomne meldinger på snitmålingen. Meldingerne beregnet indtil den nyeste melding har alle været *Kritisk trængsel* (K). Den nyeste melding viser, at brugeren har trykket på afmeldeknappen. Det betyder, at denne værdi, idet den står på plads 0, vægtes 50 % af trængselstypens værdi, dvs. i alt -5 til den samlede beregning.

Tegn	Værdi	Trængselstype
A	-10	Bruger Afmelding
I	-2	Ingen trængsel
U	0	Ubetydelig trængsel
B	1	Begyndende trængsel
S	2	Stor trængsel
K	4	Kritisk trængsel
T	10	Bruger Tilmelding

Tabel 5 Trængselstype for eksempel 2

Indeks	Vægt
0	0.5
1	0.2
2	0.1
3	0.1
4	0.1

Tabel 6 Beregningsvindue

Indeks	Melding	BeregnetVærdi
0	A	Beregningsvindue.Vægt[0] x Trængselstype.Værdi[A]
1	K	Beregningsvindue.Vægt[1] x Trængselstype.Værdi[K]
2	K	Beregningsvindue.Vægt[2] x Trængselstype.Værdi[K]

3	K	Beregningsvindue.Vægt[3] x Trængseltype.Værdi[K]
4	K	Beregningsvindue.Vægt[4] x Trængseltype.Værdi[K]
5	K	0
6	K	0
7	K	0

Tabel 7 Køtabel for eksempel 2

$$\begin{aligned}
\text{Trængselsniveau} &= \sum_{i=0}^4 \text{Beregningsvindue.Vægt}[i] \times \text{Trængseltype.Værdi}[\text{Kø.Melding}[i]] \\
&= 0.5 \times (-10) + 0.2 \times 4 + 0.1 \times 4 + 0.1 \times 4 \\
&= -5 + 0.8 + 0.4 + 0.4 \\
&= -3.4
\end{aligned}$$

Værdien for *Trængselsniveau* på snitfladen er -3.4, hvilket svarer til *Ingen Trængsel*. Dvs. brugeren ved at trykke på afmeldeknappen har fået trængselsniveauet fra 4 (Kritisk Trængsel) ned til -3.4 dvs. ingen trængsel.

Bemærk: En mulig optimering af algoritmen er at vægte troværdigheden af kømeldinger højere, såfremt der kommer den samme melding, det samme sted, fra flere uafhængige testbiler indenfor kort tid. Eksempelvis 3 biler holder i kø på Vesterbro. De trykker alle indenfor 1 minut på kø tilmeldingsknappen. Her er der stor evidens for at der virkelig er kø på Vesterbro. I algoritmens nuværende form er der følgende begrænsning. Antag at tiden defineret, hvorefter en melding bliver forældet er sat til 5 minutter. Her vil tre meldinger fra den samme bil indenfor 5 minutter give samme trængselsniveau, som situationen, hvor tre forskellige biler tilmelder kø, indenfor 1 minut. Disse situationer er dog mest realistiske, hvis systemet findes i en del biler, da det ellers vil være sjældent at to til tre biler kører det samme sted på samme tidspunkt. Et løsningsforslag er at implementere en monitor for hver snitmåling, som gemmer meldinger samt ID for den testbil, som indrapporterede indenfor de sidste x minutter, eksempel 1 minut. Ved hjælp af denne monitor er det muligt at se om flere forskellige biler har indrapporteret samme kømelding indenfor 1 minut. Hvis dette er tilfældet ganges eksempelvis 1.3 på vægten for hver af disse meldinger i den samlede udregning af trængselsniveauet på snitmålingen.

Implementering af trængseldetekteringsalgoritme i forprojektet

Trængseldetekteringsalgoritmen nåede ikke at blive implementeret på internetserveren i forprojektet og er dermed heller ikke testet i forprojektets feltforsøg, som beskrives i næste afsnit.

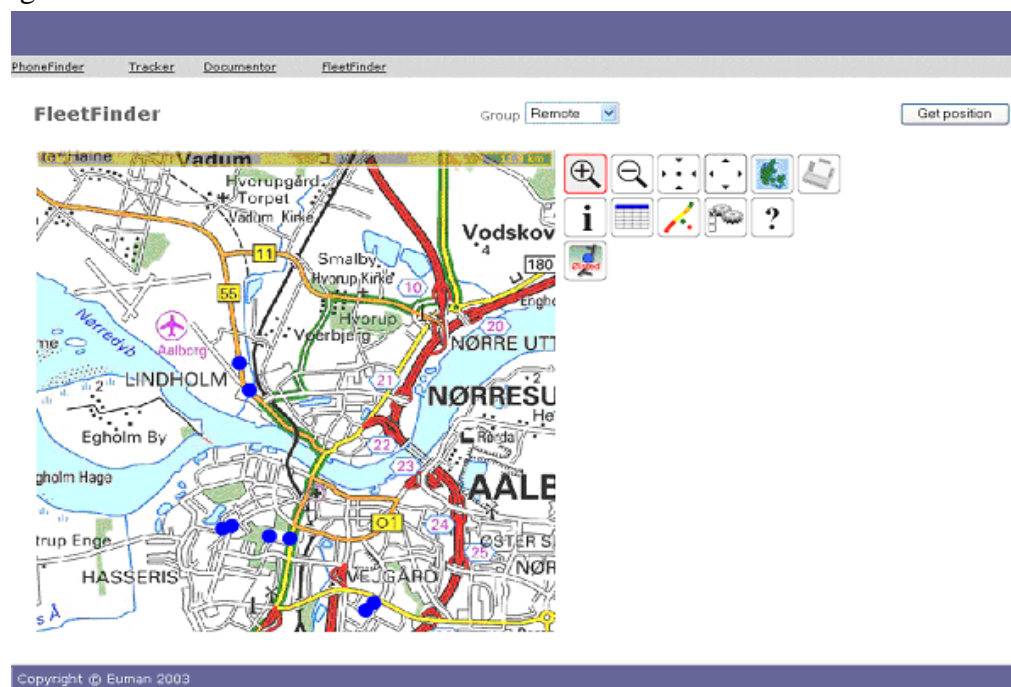
Feltforsøg

Rapporteringsklienten blev installeret i 10 taxier i Aalborg. Taxierne kørte med systemet i hele marts måned 2004. Formålet med feltforsøget var ikke at teste, om FCD kan erstatte infrastrukturbaserede systemer til en troværdig kødetektering, men derimod at teste det

udviklede udstyr og programmel, om der var hul igennem fra rapporteringsklienterne over modtagerserveren til internetserveren, ligesom troværdigheden af rapportørernes manuelle kømeldinger blev analyseret. Herudover er logdata fra de 10 taxaer analyseret med henblik på at se, hvor taxaer færdes og dermed vurdere, om de er velegnede til indsamling af FCD.

Test af om der var ”hul igennem”

Figur 7 herunder viser et skærmdump fra internetserveren hos EUMAN og viser et øjebliksbillede af rapportør klienternes position den 23. marts 2004 kl. 16.22, der er altså hul igennem.



Figur 7 Et øjebliksbillede af rapportør klienternes position den 23. marts 2004 kl. 16.22

Analyse af logdata fra testbiler

I dette afsnit analyseres logdata fra de ti taxier. Under forsøget blev alle data fra taxierne opsamlet og senere indlæst i en database. Formålet med logdataanalysen er primært at evaluere de manuelle kømeldinger og analysere, hvor de ti taxier kører i myldretiden med henblik på at vurdere om taxier er velegnede til trængselsdetektering. Herudover vil logdata blive brugt til at undersøge, om det på baggrund af disse historiske data er muligt at lokalisere steder på vejnettet, hvor der hyppigt opstår trængsel.

Klargøring af data

I feltforsøget er der i alt opsamlet 465.000 målinger for de ti taxier, hvor den taxi der er kørt mindst kun har sendt 4787 målinger, og den, der har sendt flest, har sendt 66.489 målinger.

Som det er beskrevet andetsteds i denne rapport, sender taxierne, når tændingen er slået til, følgende oplysninger til en server hvert 5 sekund:

- a. Position

- b. Hastighed i miles/hour
- c. Kompasretning i grader
- d. Antal satellitter
- e. Boks ID
- f. Dato og tid fra satellit

Dog sendes kun positioner hvert 5. minut, når taxien holdt stille. Denne funktionalitet blev lavet for at spare datatrafik, men viste sig under dataanalysen at være uheldig, fordi det så blev sværere at detektere køerne automatisk.

Foruden disse målinger sendes altid en måling, når der bliver trykket på en af de to kødetekteringsknapper.

Efter forsøgets afslutning er positionerne i disse målinger mapmatchet til et digitalt vejnet for Aalborg. Mapmatchingen er foretaget ved hjælp af et mapmatching program som tidligere er udviklet på Aalborg Universitet². Herefter er der dannet et datasæt med følgende oplysninger for hver måling:

- GPS-position
- Mapmatchet position
- Mapmatchet hastighed (dvs. den skilte hastighed på vejen positionen er matchet til)
- På eller udenfor valg meldevejnet
- Målestrækning (jf. tidligere definition)
- Værdi for manuel indtastning (0=ingen tastetryk, 1=rød tastetryk, 2=grøn tastetryk)
- Hastighed i km/t
- Kompasretning i grader
- Antal satellitter
- Boks ID
- Dato og tid fra satellit

Datasættet er efter mapmatchingen klar til analyser.

Hvor kører testbilerne?

En forudsætning for at kunne bruge FCD data til kødetektering er, at man får FCD data fra biler, der færdes på den del af vejnettet, hvor der er trængsel både i tid og rum.

I dette forsøg valgte vi at bruge taxier som FCD objekter. Vi ved at taxier kører mange km om året, men hvor og hvornår kører de? Og hvor meget kører de, hvor der er trængsel?

Som tidligere nævnt har vi 465.000 målinger på de ti taxier. Heraf lå 406.725 målinger på Aalborg kommunes veje, 131.501 på det udvalgte vejnet og 275.224 på den resterende del af Aalborgs vejnet.

Herefter er det for det udvalgte vejnet i databasen opgjort, hvor mange passager hver enkel taxa har haft af den enkelte vejstrækning. En sådan passage er herefter blevet optalt som en tur. På baggrund af dette er det muligt at tælle, hvor mange gange hver enkel vejstrækning er blevet passeret. Dvs. at en passage i det følgende betragtes som en tur.



Figur 8 Det samlede antal ture på målestrækningerne i morgenspids (hverdage mellem kl. 7.00 og kl. 9.59) i forsøgsperioden

Figur 8 viser antal ture på målestrækningerne i morgenmyldretiden. Det ses, at taxierne færdes mest på er ruten Lufthavnen – Vesterbro – Hobrovej og Motorvejen umiddelbart syd for fjorden (29-28-26-25-21-20)³. Tværforbindelserne i byen er dårligere dækket ind.

Tilsvarende kort er lavet for eftermiddagsmyldretiden, og også her er det de nord-sydgående ture, der dominerer, dog er der om eftermiddagen også en del ture på tværforbindelserne

Resultatet viser, at forsøgets 10 taxier dækker nogle af vejstrækningerne med trængsel i Aalborg godt og andre dårligt. Man skal derfor være omhyggelig ved udvælgelsen af biler til indsamling af FCD.

Evaluering af den manuelle kømelding

Et af hovedsigterne med dette pilotforsøg med FCD data til kødetektering var at vurdere kvaliteten af en manuel kødetektering udført af taxichauffører.

² Jens, Juhl, INFATI – Mapmatching, Notat 3, Institut for Samfundsudvikling og Planlægning, Aalborg Universitet, 2001

³ Disse numre refererer til kortet med meldestrækninger

Analyse af logdata

Chaufførerne i de ti taxier har i forsøgsperioden trykket på kønappen 176 gange og på køafmeldeknappen 151 gange på tidspunkter, hvor de har været på forsøgsvejnettet.

Af 176 kømeldinger ligger de 43 eller 25 % i myldretiden (hverdage mellem 7.00 – 9.59 og 15.00 – 17.59).

Boks ID	Køtilmeldinger	Køafmeldinger	Ingen/for sen køafmelding
264	22	19	16
265	65	67	9
266	66	63	30
267	43	363	9
268	14	11	3
269	6	6	1
270	39	54	14
271	13	16	3
272	23	22	5
273	13	13	4

Tabel 8 Kvaliteten af de manuelle kømeldinger

Tabel 8 er et forsøg på at vurdere kvaliteten på kømeldingerne. Den anden kolonne i Tabel 8 viser antal køtilmeldinger og tredje kolonne viser antal køafmeldinger. Vogn nr. 267 har 8 gange så mange afmeldinger som tilmeldinger, og også vogn 270 har alt for mange afmeldinger, men herudover stemmer antal køafmeldinger ganske pænt med antal køtilmeldinger. Kolonnen ingen/for sen afmelding dækker over det antal køtilmeldinger, som ikke er efterfulgt af en køafmelding inden den pågældende bils hastighed har passeret 80 % af den pågældende vejs skiltede hastighed. Tabel 8 viser, at chaufførerne generelt har forstået systemet med kø til og afmeldinger, men også, at mange for sent får afmeldt en kø.

Tilbagemeldinger fra taxivognmændene

Der er udført telefoninterview med 8 af de 10 taxivognmænd, som havde udstyret installeret, og interviewene gav følgende tilbagemeldinger:

- Vognene kørte alle i myldretiden på hverdage og fravalgte kun i begrænset omfang vejstrækninger med køproblemer, og taxier må således siges at være velegnede til kødektering.
- Interviewene viser, at et kort opstartsmøde ikke er tilstrækkelig, hvis man vil sikre en ensartet manuel kø rapportering, sandsynlig skal der en egentlig oplæring til.
- Knappanelet kan i de fleste vogne hensigtsmæssigt placeres til venstre for rattet. Det skal dog sikres, at panelet kan ses af føreren under kørsel
- Der var kun positive tilbagemeldinger omkring udstyrets driftspålidelighed.
- Det var let at huske at melde en kø, men svært at huske at afmelde den igen
- Meget positiv interesse for forsøget

Konklusion

I projektet er der udviklet en virkende prototype, hvor alle komponenterne i arkitekturen virker. Dette drejes sig om en speciel hardware trafikrapportør klient udviklet af M-tec og software implementeret og delvis integreret med Euman server platform. Dele af ideerne udviklet i projektet er allerede tager kommercielt i brug af Euman's produkter.

Rapportørerne (ti taxichauffører) udtrykte sig meget positivt overfor indrapportering. Data viser dog, at niveauet for rapportering ikke er ensartet og der synes at være et problem med afrapportering af kødannelse. Uddannelse af trafikrapportører kan antagelig give mere konsistente indrapporteringer. Det er dog tvivlsomt om uddannelse alene kan hindre at rapportørerne glemmer at afmelde køerne. Det er overvejende sandsynligt at man ved at arbejde med interaktionen mellem systemet og chaufføren kan optimere dette. Det er enkelt og muligt at indbygge en "promtning" for kørapportering, der er baseret på bilens hastighed og acceleration, og som når visse hastigheds- og accelerationskriterier overskrides "promter" chaufføren om at tilmelde eller afmelde kø. Specifikation og udviklingen af en sådan funktion har dog været udenfor rammerne af forprojektet".

REMOTE projekt har udover den manuelle indrapporting af køer også forsøgt at lave en automatisk kødetektering. Der er udviklet og implementeret algoritmer til dette. Det 30 dags felt forsøg med de ti taxier har vist, at vi har haft for få rapportører til at kunne teste den automatiske kødetektering.

Feltforsøget har vist, at prototypen er meget billig i drift. Dette skyldes at der er udviklet en speciel kompakt protokol til kommunikation. Vi kan ikke konkludere om prototypen er billig i vedligehold da feltforsøget har kørt for kort tid til at udtale sig om dette. Vi vil dog gerne anføre, at der kun har været ét problem med rapportør klienterne. Opstarten af større efterfølgende forsøg kan muligvis gøres billigere ved at bruge standard hardware i rapportør klienter. Dette vil sandsynligvis ske på bekostning af højere drift omkostninger, da den kompakte kommunikation protokol måske ikke kan bruges. Vi kan derfor ikke konkludere om det totalt set vil være billigere at brug standard hardware eller bruge speciel hardwaren udviklet i REMOTE projektet.

For analysen af log data fra de ti taxier kan det konkluderes, at de 10 taxier dækker nogle af Aalborgs veje med trængsel godt og andre dårligt. På baggrund af dette konkluderer vi, at en god stratificering af deltagerne i et sådan system er afgørende for anvendeligheden.

Ud fra de manuelle kømeldinger kan ses, at systemet generelt er forstået af trafikrapportørerne og brugt efter hensigten, men også, at der er svingende kvalitet i manuelle kømeldingerne mellem de ti vogne.

Der er gode muligheder for at måle trængsel vha. automatisk opsamlet log data, men et sikkert estimat for den enkelte stræknings fri operationshastighed er nødvendigt, hvis sådanne log data skal bruges til kødektering.

Referencer:

1. Torp, Kristian, (red) Lahrmann, Harry (red), Redskaber til monitorering af trafikken, REMOTE, Institut for Samfundsudvikling og Planlægning, Aalborg Universitet 2004.
2. Lahrmann, Harry, Madsen, Jesper Runge, Boroch, Teresa, Intelligent Farttilpasning i Dansk Vejtidskrift, nr. 1 2004.
3. Juhl, Jens, INFATI – Mapmatching, Notat 3, Institut for Samfundsudvikling og Planlægning, Aalborg Universitet, 2001